National Library
of Canada

Canadian Theses Service

Ottawa, Canada
K1A 0N4

Bibliothèque nationale
du Canada

Services des thèses canadiennes

CANADIAN THESES

THÈSES CANADIENNES

NOTICE

The quality of this microfiche is heavily dependent upon the quality of the original thesis submitted for microfilming. Every effort has been made to ensure the highest quality of reproduction possible.

If pages are missing, contact the university which granted the degree.

Some pages may have indistinct print especially if the original pages were typed with a poor typewriter ribbon or if the university sent us an inferior photocopy.

Previously copyrighted materials (journal articles, published tests, etc.) are not filmed.

Reproduction in full or in part of this film is governed by the Canadian Copyright Act, R.S.C. 1970, c. C-30. Please read the authorization forms which accompany this thesis.

THIS DISSERTATION
HAS BEEN MICROFILMED
EXACTLY AS RECEIVED

AVIS

La qualité de cette microfiche dépend grandement de la qualité de la thèse soumise au microfilmage. Nous avons tout fait pour assurer une qualité supérieure de reproduction.

S'il manque des pages, veuillez communiquer avec l'université qui a conféré le grade.

La qualité d'impression de certaines pages peut laisser à désirer, surtout si les pages originales ont été dactylographiées à l'aide d'un ruban usé ou si l'université nous a fait parvenir une photocopie de qualité inférieure.

Les documents qui font déjà l'objet d'un droit d'auteur (articles de revue, examens publiés, etc.) ne sont pas microfilmés.

La reproduction, même partielle, de ce microfilm est soumise à la Loi canadienne sur le droit d'auteur, SRC 1970, c. C-30. Veuillez prendre connaissance des formules d'autorisation qui accompagnent cette thèse.

LA THÈSE A ÉTÉ
MICROFILMÉE TELLE QUE
NOUS L'AVONS REÇUE

NL 339 (r. 86/01)

Canada

National Library of Canada
Bibliothèque nationale du Canada

Ottawa, Canada
K1A 0N4

TC -

ISE 0-315-23329-X

CANADIAN THESES ON MICROFICHE SERVICE – SERVICE DES THÈSES CANADIENNES SUR MICROFICHE

PERMISION TO MICROFILM – AUTORISATION DE MICROFILMER

- Please print or type – Écrire en lettres moulées ou dactylographier

AUTHOR – AUTEUR

Full Name of Author – Nom complet de l'auteur
PATRICK ST. AUBYN GILL

Date of Birth – Date de naissance
12th April, 1954

Canadian Citizen – Citoyen canadien
☐ Yes Oui ☒ No Non

Country of Birth – Lieu de naissance
BARBADOS

Permanent Address – Résidence fixe
GLEBE LAND
ST. GEORGE
BARBADOS

THESIS – THÈSE

Title of Thesis – Titre de la thèse
A Simulation of an Ethernet-based PACS

Degree for which thesis was presented
Grade pour lequel cette thèse fut présentée
M.Sc.

Year this degree conferred
Année d'obtention de ce grade
1985

University – Université
University of Alberta

Name of Supervisor – Nom du directeur de thèse
W. M. Maydell

AUTHORIZATION – AUTORISATION

Permission is hereby granted to the NATIONAL LIBRARY OF CANADA to **microfilm** this thesis and to **lend** or **sell copies of the film.**

The author reserves other publication rights, and neither the thesis nor extensive extracts from it may be printed or otherwise reproduced without the author's written permission.

L'autorisation est, par la présente, accordée à la BIBLIOTHÈQUE NATIONALE DU CANADA de **microfilmer** cette thèse et de **prêter** ou de **vendre des exemplaires du film.**

L'auteur se réserve les autres droits de publication; ni la thèse ni de longs extraits de celle-ci ne doivent être imprimés ou autrement reproduits sans l'autorisation écrite de l'auteur.

ATTACH FORM TO THESIS – VEUILLEZ JOINDRE CE FORMULAIRE À LA THÈSE

Signature
Patrick Gill

Date
Sept. 17, 1985

NL-91 (r. 84/03)

Canada

The University of Alberta

A SIMULATION OF AN ETHERNET-BASED PACS

by

Patrick St. A. Gill

A thesis
submitted to the Faculty of Graduate Studies and Research
in partial fulfillment of the requirements for the degree
of Master of Science

Department of Computing Science

Edmonton, Alberta
Fall, 1985

THE UNIVERSITY OF ALBERTA

RELEASE FORM

NAME OF AUTHOR: Patrick St. A. Gill

TITLE OF THESIS: A Simulation of an Ethernet-Based PACS

DEGREE FOR WHICH THIS THESIS WAS PRESENTED: Master of Science

YEAR THIS DEGREE GRANTED: 1985

Permission is hereby granted to The University of Alberta Library to reproduce single copies of this thesis and to lend or sell such copies for private, scholarly or scientific research purposes only.

The author reserves other publication rights, and neither the thesis nor extensive extracts from it may be printed or otherwise reproduced without the author's written permission.

(Signed) Patrick Gill

Permanent Address:

Glebe Land
St. George
Barbados
West Indies

Dated 28 August 1985

THE UNIVERSITY OF ALBERTA

FACULTY OF GRADUATE STUDIES AND RESEARCH

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies and Research, for acceptance, a thesis entitled **A Simulation of an Ethernet-Based PACS** submitted by **Patrick St. A. Gill** in partial fulfillment of the requirements for the degree of **Master of Science.**

W. A. Davis
Supervisor

Date 17 Sep 95

ABSTRACT

A Picture Archiving and Communication facility is generally acknowledged to be an integral part of every radiology department. A growing number of pictures in radiology are generated by digital imaging instrumentation. Such examinations employ computer-tomography scanners, ultrasonic scanners and magnetic resonance imaging scanners. The digital images produced by these devices must be processed, transmitted, stored, retrieved and displayed by radiologists on a continual basis.

In this thesis, the performance features for a proposed picture archiving and communication system usable within a radiology department is explored by the use of simulated model. The system connects a number of image display and image generation workstations with two image storage devices and a powerful processor on an Ethernet local communication network.

The workload is specified as a mix of user activities with functions such as single-image display, multi-image display, edit, image manipulation, and browse. Input to the simulation model are the system parameters such as transmission medium capacity, image storage capacity, screen resolution and the number of workstations connected to the system. By varying these parameters, suitable bounds are established for the system parameters that produces an Ethernet system with acceptable image response times.

Acknowledgements

I wish to thank Professor Ursula Maydell, my supervisor, for her advice, criticism and guidance throughout the research and preparation of this thesis. Her help has been invaluable towards the successful completion of this work.

I wish also to thank Professor Wayne Davis who has given many helpful suggestions throughout the design phase of this project.

~~The~~ author wishes to thank Mr. Walter Neilson who has read this thesis and made several helpful comments.

Many thanks are extended to the Association of Universities and Colleges of Canada which have awarded me a scholarship making my study abroad possible.

Table of Contents

Chapter Page

List of Tables

List of Figures

Chapter 1

Introduction

Every radiology department has some Picture Archiving and Communication System (PACS) for storing and and retrieving images. Usually these are manual systems consisting of a central film file library. The transmission and retrieval of film images is done by someone who physically takes them to where they are needed.

The concept of an *automated* PACS is to provide a facility for storing and retrieving images via a digital communication system. A new array of digital imaging devices and storage devices have made this possible. Duerinckx et al. [13] report that twenty-five percent of the examinations done in a radiology department are digitally formated and by the end of 1985 this will increase to fifty percent.

The basic components of a digital PACS are: *image acquisition devices* from which "external" images are acquired; a hierarchical *storage system* with at least an active storage and an archival storage device; a fast *image processor* for image manipulation; *workstations* which provide a viewing facility; and the underlying image *communication system* that is a local area network (LAN) used for transporting the images form one device to the next.

For a digital PACS, a LAN must be selected from among the currently available LAN technologies as the underlying communication system. In Chapter 2 an outline of the current LAN environments and attributes is given. This serves to bring present medical applications into perspective. The different aspects and properties of the three main categories of LANs are discussed.

The PACS model presented in this thesis uses an Ethernet LAN as its communication system consisting of a 500m Ethernet segment. This segment connects three image acquisition devices, a fast image processor, an active storage device, an archival storage device, and one or more display devices (workstations). The details of Ethernet itself are discussed in Chapter 3.

Ethernet was chosen for the following reasons. Firstly, its commercial availability and establishment in many communication systems have made it an attractive choice [34]. This facilitates future inter-networking with other existing communication systems. Secondly, the Ethernet protocol exhibits fairness to all users of the implemented communication system. There is no priority scheme, hence, all users have the same chance of gaining access to the communication facility. Thirdly, there are no explicit acknowledgement packets sent for data packets received at a destination device. Explicit acknowledgement of packets seem wasteful in other protocols which acknowledge each packet received. Instead Ethernet employs a timeout mechanism which causes only the outstanding packets to be retransmitted. This is especially advantageous for PACS which requires long sequences of packets to be transmitted for a single image. The PACS channel is otherwise utilized in sending data packets. In addition, errors occurring are handled at a higher layer of the protocol which requires only those packets in error to be retransmitted. Lastly, since the first three layers of the Ethernet protocol have been kept simple, much of the sophisticated functions usually found at these levels in other protocols have been excluded; only the necessary functions are implemented. Image data does not warrant an elaborate error checking facility, since a one bit error in a $512^2 \times 8$ image may have little consequence compared to a one bit error occurring in a bank balance. Furthermore, these layers of sophistication may be incorporated as needed at higher levels of the Ethernet protocol if the need arises.

A simulation model is developed to study the performance of Ethernet under a medical image workload as described in Section 3.3. The three layers as proposed by the IEEE-802 standard for CSMA/CD protocols [12] are implemented in the simulation model.

The simulation model is driven by a user workload consisting of a mix of user activities which are performed at a workstation. The basic functions provided by the workstation are single-image display, multi-image display, edit, manipulate, and browse.

In order for this system to be interactive, fast retrieval times for the activities are required. The system is acceptable if retrieval times of ten seconds or less per image are achieved as proposed by Cywinski in [11].

PACS differ from conventional LANs in the amount of data in an image to be transmitted. The Ethernet protocol specifies a maximum packet data field of 12,000 bits. This requires the basic $512^2 \times 8$ image to be fragmented into at least 175 packets before the image can be transmitted.

In this thesis the performance issues of the proposed PACS are addressed. Chapter 2 gives a general overview of LAN technology and brings into perspective the PACS application. Chapter 3 presents an architectural design description of the PACS which is used for digital image management in medical applications. Chapter 4 presents the details of the simulated PACS model which will be used to measure the performance of the proposed PACS. Chapter 5 gives the performance results obtained from the simulation and Chapter 6 gives a brief summary and conclusion for local area networks and in particular of the results established for PACS with Ethernet. Guide lines for future research, direction, use and application of PACS in the medical community are also given.

Chapter 2

An Overview of LAN Technology

In this chapter, an overview of local area networks is given. Other excellent reviews are found in the references [33,42,44].

2.1. Introduction

The 1980's marked the rapid evolution of local area networks (LANs) as evident by the large number that are commercially available and by the amount of research being devoted to them. Currently, there are some 250 vendors of LANs and network components [10] and as a result there are a variety of technical methods used to build them.

LANs and computer communication have become a part of many working environments. The basic building blocks of a LAN are:

(1) the physical and logical transport medium, and

(2) the computer-based workstation (WS).

The computer-based workstation may simply be a personal computer or a sophisticated powerful computer system such as an interactive graphic system used for computer aided design and manufacturing (CAD/CAM).

Today personal computers sell at a rate of 2 million a year in the USA alone [10] and this rate is rapidly rising. Personal computers, microprocessors, microfiles, databases, and interactive graphics when integrated into a LAN produce a unified system that can meet the growing needs of business and industry for generating, processing, transmitting and storing data of all kinds.

The following definition of a LAN is proposed to distinguish it from other types of local area data communication systems. A LAN may be defined as follows:

> A LAN is a general-purpose computer communication network that provides interconnections, in an efficient way, of various data communicating devices within a limited geographical area (maximum distance of 10 km) and at transmission speeds between 1 - 50 Megabits per second (Mbps). LANs provide access to the user from any terminal or workstation to any device on the network.

The LAN is a subclass of the general class of local network systems. The other two subclasses are the high-speed local network (HSLN) and the computerized branch exchange (CBX). The HSLN is designed to deliver end-to-end high throughput between high-speed devices with transmission speeds of 50 Mbps or more. The number of devices and the maximum distance is limited as in a computer room configuration of high speed devices. This is in contrast to the LAN that is distributed down a corridor, within a building, a group of adjacent buildings or campus-like environment. The CBX is an on-premise private branch exchange that uses circuit switching techniques in contrast to LANs and HSLNs that use packet switching techniques. The CBX is designed to handle both voice and data connections. Much lower data rates of 9.6 Kbps to 64 Kbps are realized. Note that there is no clear dividing line between these subclasses of local networks. There is some overlap and there are features that are common to two or more subclasses.

The LAN, and other subclasses of local networks, should be distinguished from long-haul networks or wide area networks (WAN). A long-haul network covers a wide geographical area spanning continents. Generally, these networks are not privately owned and must use the public and commercial utilities available. In particular, they use the public telephone trunks as their transmission medium. Telephone technology was designed for voice communication and as such presents many technical problems when used for data communication. Three problems that mainly distinguish long-haul networks from LANs are:

(1) high data transmission error rate experienced on a telephone line,

(2) low data transmission rate, and

(3) complex routing mechanisms.

The 1970's have seen the development and maturity of long-haul networks. They typically use transmission links of 9.6 to 56 Kbps rates. This low data transmission rate limits the types of applications that are possible with long-haul networks. Calling a bulletin board, reading electronic mail, looking up stock reports, or entering information in a database are possible examples. With these applications, a low transmission rate can be tolerated since the data is presented or accepted at a rate that is controlled by the user (e.g., reading, or typing), hence these minimum transmission speeds will suffice.

With LANs, new types of applications requiring fast and critical response times are now possible. They include such applications as large file transfers, filling a screen from a remote computer system, creating graphics, or intergrating voice and video images with digital information. Consider an interactive user sitting at a terminal and running a screen oriented program that requires painting an image on a low resolution screen of 256 x 256 pixels with each pixel represented by a minimum of 8 bits. This takes 54.61 seconds with a 9.6 Kbps line on a long-haul network whereas on a LAN with a modest 1 Mbps line it takes only 0.524 seconds. This simple example illustrates the new types of applications now possible with the use of LANs.

There are two main forces which are directly responsible for the growing use and popularity of LANs. The first is the favourable price/performance ratio that LANs offer over the centralized mainframes. The advances in VLSI technology gave birth to a set of powerful and inexpensive microprocessors with clock speeds well over 20 MHz.

Mainframes may be about 10 times faster than the best microcomputer, but they cost a thousand times more. With the use of a high-speed data transmission medium and a highly efficient medium access protocol, many of these microprocessor-based devices can be connected at affordable costs to give comparable performance to the centralized mainframe. The other force that is responsible for the popular use of LANs is the need for organizations to connect their existing systems into a local area network to take advantage of the benefits of resource sharing. Such sharing of resources, both software and hardware, increases productivity in a cost effective manner.

The LAN provides the user with many advantages over the central system. The first is the cost saving resulting from resource sharing. Instead of having separate modules of an expensive program at each user site, a user from any workstation may run a single copy of the program. This single program can be kept at a single location and be retrieved by using a communication facility. The LAN also provides a mechanism for sharing of scarce and expensive pieces of hardware. The typical example is the sharing of a graphic plotter, since a dedicated system attached to each user site cannot be economically justified. Such devices, though necessary, are not used extensively by each user, hence, it makes sense to connect a single device to the LAN where it may be shared by many users.

A second advantage is the greater reliability that these LANs provide. With unconnected systems, when one device goes down because of a hardware failure, the user loses all computing. With distributed computing, the user may still be able to access another section of the LAN for an alternate supply of computing power. For some applications this total loss of computing for even a short period of time cannot be tolerated and may even prove to be catastrophic. For example, a chemical or nuclear control process in an industrial application. There might be a slight degradation in the overall performance of the LAN with failure of one or more devices, but at least all users are still accommodated. In the general design of a LAN, the intention is to allow any access from any terminal to other devices.

Another advantage similar to the graceful degradation upon failure is the ability of the LAN to grow as more computer power is needed as the needs of the organization expands. A LAN allows this incremental growth of computing power by the addition of new devices without disrupting the operations or continuity of the organization. With a centralized system it may be possible to upgrade the hardware configuration in an attempt to improve the system performance, but eventually there comes a time when the system has to be replaced with a more powerful one. This usually results in some disruption of service to the user community for some period of time.

LANs may be designed both for office and factory environments. Information appears in a diversity of forms and presentations. These forms include audio, video, and digital, while the information may be presented as hard copy or soft copy (video reports and graphics). There is also the requirement for instantaneous queries, as well as access to text and data warehouses, and mail systems. These applications all separately require different protocols and transmission speeds, and may be integrated into a local area network. The new technologies available are capable of handling voice, video and digital information all on the same network. They will be employed increasingly by managers and administrators both in the office and in industry for reporting, teleconferencing, education, security and control purposes.

LANs are also being used in the medical community for storing, processing and managing digitally formated images generated by various imaging devices. These picture archiving and communication systems (PACS) allow physicians and radiologists to compare results from from various computer-based diagnostic imaging instruments before recommending a course of treatment. It is believed that these systems will help to minimize the risk of lost or misfiled patient images, and provide short retrieval times. This results in reduced turnaround times for managing these images, increasing productivity and reducing health care costs.

As offices and factories automate, the proliferation of LANs will continue. Since LANs are employed in diverse applications, it is possible to produce several incompatible LAN technologies. This is especially so since they are privately owned and LAN designers will be striving to produce the most efficient, reliable, and secure means of communication possible for the application at hand. To avoid these incompatibilities, there must be some standards to control the design of LANs among different manufactures. Once there is full standardization, system designers will be able to configure LANs that are compatible with computers, terminals, printers, and other equipment down to the cables and connectors from a variety of vendors. This, apart from being a goal in local area network design, will provide a bigger and more competitive market resulting in cheaper equipment. From a technical point of view, some of the standard functions of the LAN can be implemented in VLSI technology on a single chip. Formulating standards for local area networks is a difficult task because of the large amount of equipment already in the field and of the de facto standards set by each manufacturer. There are several organizations currently involved in establishing standards. Among such standards are the OSI (Open Systems Interconnection) model proposed by the ISO (International Standard Organization) [44], the ANS (American National Standards) for high performance networks [8], the ECMA (European Computer Manufacturers Association) proposed standards for local area networks [19], and the IEEE-802 standards for local area networks [12,19].

Cost and operational performance are two very important considerations in the design of a local area network. The current trend is for organizations to connect their existing independent computer systems into a local area network rather than install the entire network including the workstations. A goal in the design of local area networks is to design them at the lowest possible cost. The cabling and interfacing of the workstations to the cable should not cost no more than a single workstation. Chorafas

[10] estimates that the current cost of interfacing to a local area network is $500 per workstation. With the cost of VLSI technology decreasing, it should be about $100 by the end 1986. From a performance point of view, the design goal of local area networks is to maximize the message throughput between stations and minimize the message delay and response times and at the same time provide reliable and error-free communication between workstations.

Both the design cost and operational performance of a LAN are determined by the network technology which may be characterized in terms of its:

(1) network topology,

(2) transmission medium,

(3) transmission techniques, and

(4) medium access protocol.

There are multiple choices for each category and there are local area network systems implemented for various combinations of these. The network application would determine the appropriate combination of characteristics for the LAN design.

2.2. LAN Topology

LAN topology refers to the physical arrangement of the attached devices within the network. It is a factor which determines the network reliability and message delay. Along with the transmission medium, the topology determines the type of data that may be transmitted, the speed and efficiency of communication and even the type of applications that the network can support. The three most basic topologies used in LAN design are:

(1) the bus/tree,

(2) the ring, and

(3) the star.

The bus is a special case of the tree topology with a single trunk with no branches; or the tree may be considered as a generalization of the bus topology. To improve network reliability and performance, many network designers have adopted combinations of these basic topologies.

2.2.1. The Bus

The bus is characterized by a single open-ended, broadcast transmission medium. Devices are attached to the bus through passive taps. LAN standards [12] require the taps to be attached without breaking the bus medium. Clamp-on connectors are used that allow easy attachment and detachment of stations without the disruption of the signal on the bus.

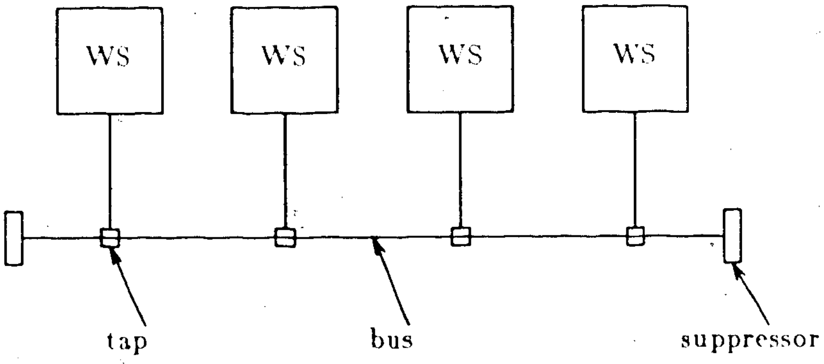


Figure 2.1 The Bidirectional Bus Toplolgy

Attached to each tap is the transceiver which contains the electronics for transmitting and receiving the signal between the bus medium and the station.

The broadcast nature of the bus permits many stations to receive the same transmission simultaneously. However, there must be some controlling mechanism to ensure that only one station be allowed to transmit at a time. Within the transmitted message there is a destination field that identifies the appropriate stations.

There are two distinct broadcast bus network configurations. The first is the *bidirectional bus system* (BBS) in which, as in Ethernet, the signal transmitted by one station propagates in both directions to reach all other stations on the bus. At both ends of the bus are signal suppressors that terminate or abort the signal thus preventing interference by the end-reflection of the signal (see Figure 2.1). The second is the *unidirectional bus system* (UBS) in which the transmitted signal propagates in only one direction. At the end, the cable is folded back onto itself to create two channels. There are an outgoing channel onto which the stations transmit messages and an inbound channel from which stations receive messages. All signals transmitted on the outbound channel are repeated on the inbound by the head end (see Figure 2.2).

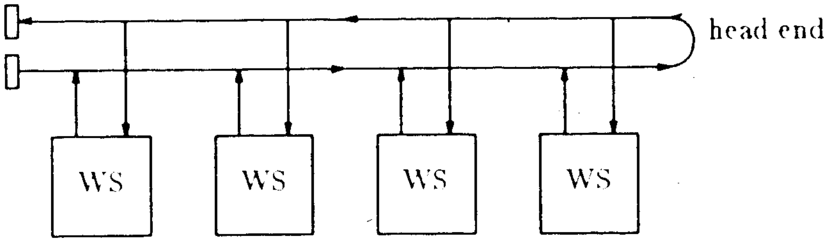


Figure 2.2 The Unidirectional Bus Topology

Another method of achieving unidirectional transmission is by using two distinct frequencies (frequency multiplexing) traveling on opposite directions in a single bus as in

Figure 2.1. All messages are transmitted by the stations on one frequency and travel in one direction. As the signal reaches the end of the bus it is converted to a second frequency and travels in the opposite direction where it may be received by a station. The transmission capacity of a unidirectional broadcast bus is half that of a bidirectional broadcast bus.

The bus structure may be extended into a generalized unrooted tree structure by connecting two or more buses through active repeaters. A requirement of this arrangement is that no two pairs of buses be connected by two or more repeaters. Multiple paths of possibly different lengths between pairs of stations will result in multipath interference. The repeaters are transparent to the rest of the network and are not part of the logical functioning of the network. Their function is to amplify and retransmit the signal from one bus to the other extending the transmitting distance of the network. This may result in a small delay of one or more bit times. No buffering, packet recognition or the like is done by the repeater.

2.2.2. The Ring

The ring network topology consists of a unidirectional transmission medium connected in a closed loop. The transmission medium is not continuous like the bus topology with its clamp-on connectors. Instead, active repeaters, one for each workstation connected to the ring, are inserted into the medium forming point-to-point connection between successive pairs of stations. Information travels sequentially, bit by bit, around the loop from repeater to repeater in fixed sized packets in one direction. The repeater is described as active since its function is to regenerate the message and retransmit it to the next repeater. An advantage of the ring topology over the bus topology is that the ring configuration eliminates attenuation of the signal over the ring and, hence, increases the signaling distance. In addition, repeaters provide the access point to insert and remove packets from the ring by the appropriate workstation (see Figure 2.3).

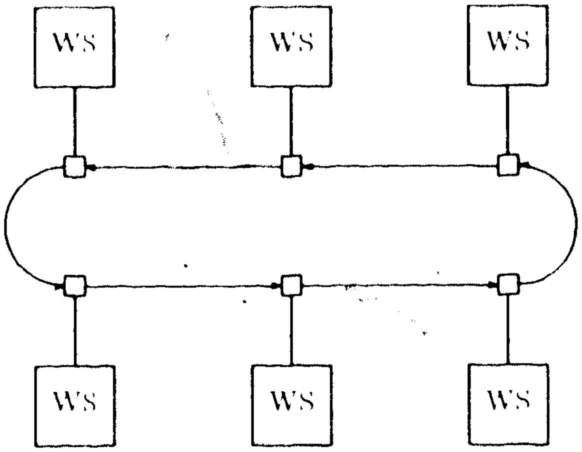


Figure 2.3 The Ring Topology

On the ring, usually only one station has the right to transmit data at any one time. There are many ways to gain access to the medium as will be discussed in Section 2.4.

This type of configuration, point-to-point physical links between pairs of repeaters, may be considered as a unidirectional broadcast system since multiple stations may in turn "listen" to the data on the ring. A potential problem in the ring is that the data packet may circulate indefinitely unless it is removed. The destination repeater copies the data and may remove the packet or leave it to be removed by the transmitting repeater after a complete trip around the ring. The later approach has the following advantages:

(1) it permits automatic acknowledgement of receipt for the packet, and

(2) it permits multicast addressing, i.e., the same packet sent simultaneously to multiple receiving stations.

The ring topology is popular in Europe due to an early establishment in industries and universities of the *Cambridge* ring developed at the University of Cambridge, England [6]. In the USA Ethernet and MITRENET bus structures are more prominent commercially.

Some of the potential problems associated with the ring topology are listed below:

(1) Cable vulnerability: In the basic ring topology, a break in any of the links between repeaters renders the entire ring inoperable.

(2) Repeater failure: The basic problem here is that a failure in any repeater also renders the entire ring inoperable. It is difficult to isolate the defective repeater; however, it could be shunted off if a double ring is used.

(3) New installation: Installing a new repeater to support new workstations requires identification of the two topologically adjacent stations. If the total cable length of the ring changes, the ring may have to be retuned.

(4) Distributed initialization and recovery: If a transient error occurs, some strategy is needed where all active repeaters can quickly and simply agree on the need for initialization and recovery. The responsibility of initialization and recovery should not be assigned to a single station.

The IBM Zurich Ring solves most of these problems by using a star-ring hybrid wiring structure with partially centralized concentrators [24,25]. The basic idea is that repeaters or links may be attached or detached from the ring by using bypass relays in the concentrators.

Apart from the removal of the ring problems, the ring technology will become increasingly popular for another reason. With the ring, every station is guaranteed a transmission within some maximum time limit. This makes them extremely valuable in time constraint applications if the maximum time can be reduced below the threshold required by the application. An example is audio communication by packetized voice, where the packet delay must be maintained below a maximum value for practical communication.

2.2.3. The Star

The star topology connects end stations point-to-point through a centralized master. Communication between end stations is one-to-one in contrast to the broadcast nature of the bus and the ring topologies. The centralized station acts as a switching station between pairs of end stations.

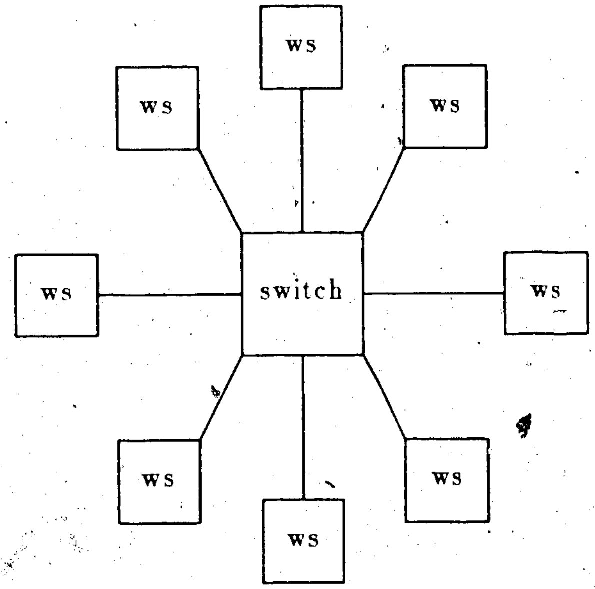


Figure 2.4 The Star Topology

The reliability of a star network largely depends on the central station. Failure of this station brings down the entire network since monitor and switching functions cannot be transferred to another station. Usually there is a fault tolerant system of parallel processors at this central site. To improve reliability and performance, the computing power at the central station is greater than any of the end stations to handle the combined load from the end stations.

The star topology is the oldest and least reliable type of topology and is used in computer branch exchange networks. This topology currently simplifies the design of networks using fiber optic links. By using a single central star repeater, many problems associated with optical connections are eliminated.. By having a common connection point the number of physical connections is reduced, hence, reducing the design cost. The *Fibernet II* designed by Xerox is an example of an active star network [38].

2.3. Transmission Techniques

There are two main transmission techniques used in LANs for the propagation of a signal over the transmission medium:

(1). baseband, and

(2) broadband.

The main differences are in the maximum transmission speed possible with each technique and the fact that a single broadband carrier supports multiple channels. In LAN design, a key selection decision is whether to used baseband or broadband. The selection must be based on the relative cost and benefits of each since they offer advantages and disadvantages over each other.

2.3.1. Baseband

Baseband uses digital signaling methods with speed between 1 and 10 Mbps and may be used on twisted pair and coaxial cables. The entire bandwidth of the medium is used to form the digital signal and is represented as constant voltage pulses. The signal is bidirectional and, hence, is supported by a bus topology. Over long distances, the signal rapidly attenuates, especially at higher frequencies. The signal loses its shape making communication impractical because of the transmission errors introduced. This transmission technique is good for short distances of 500m or less, while greater distances will require signal amplification.

2.3.2. Broadband

Broadband transmission uses analog signaling in the radio frequency (RF) range 5 to 400 MHz and is used on twisted pairs, coaxial cables or optical fibers. Frequency division multiplexing (FDM) is employed, where the carrier bandwidth is divided into several sections of bandwidths. Each channel may separately support either digital voice, data, video or audio information or a combination of these. Both digital and analog devices may be attached to the same local area network with the necessary modem interface.

In a special case of broadband transmission, known as single-channel broadband, the entire bandwidth of the medium is used as a single channel. The characteristics of broadband communication are still maintained.

Broadband signaling is unidirectional and covers a greater distance than the baseband technique. Two channels are needed, one for each direction, to establish full communication. The propagation delay is thus twice that of a baseband channel. RF modems are also required to transmit and recover the information that is placed on the channel. With broadband there are three methods of establishing communication between work stations:

(1) dedicated,

(2) switched, and

(3) multiple access.

In dedicated service, network devices communicate at a fixed frequency. Their RF mo[illegible] are tuned to that frequency, thus requiring no special protocols. It is possible to partition the set of network devices into groups with all devices in one group tuned to a common frequency.

Devices using switched transfer techniques require modems that are capable of changing their frequency by electronic commands. To establish a connection between two devices, a central controller, at the request of the user, assigns a common frequency to their modems. The devices then communicate at that frequency locking out all other devices. This is analogous to the switching technique employed by the telephone network.

The multiple access service, which is primarily used by all baseband systems, allows several devices to communicate on the same frequency, but this requires medium access protocols to control transmission.

2.4. LAN Transmission Media

There are two categories of transmission media used to implement local area networks, physical lines and radio waves. For the former, the most common ones used are twisted pairs, coaxial cables, and optical fibers while the later includes microwaves and HF (High Frequency) radio waves. Each medium offers a different level of performance. A choice in the medium with a given performance level is made to match the application at hand.

2.4.1. Twisted Pairs

A twisted pair is the low-cost transmission medium most commonly used in many analog communication systems. It is also used in local area networks with low transmission speeds (1 Kbps to 2 Mbps). Its limitation is its susceptibility to interference and noise especially at higher speeds. The interference effect is minimized by proper shielding. The *Corvus Omninet* is a baseband local area network that uses twisted pairs at a 1 Mbps transmission speed [10]. The IBM PCnet also uses twisted pairs at a 2 MHz transmission speed [10].

2.4.2. Coaxial Cables

For higher performance requirements, coaxial cables provide higher throughput, support many more devices and cover a greater transmitting distance at higher frequencies than twisted pairs.

There are two types of coaxial cables, one is a 50 ohm impedance cable that supports baseband transmission and is used in the Ethernet network. The other is a 75 ohm impedance cable that is used in broadband communication and is used extensively in CATV technology. The coaxial cable supports most medium access protocols and network topologies except possibly the star topology. The central station of a star topology may not be fast enough to fully utilize the speed of the coaxial cable.

2.4.3. Optical Fibers

Fiber optic cable is the prime candidate for future local area network installations. It provides the highest potential capacity as well as other advantages over other types of transmission media.

In an optical fiber, light goes in at one end and comes out of the other irrespective of the number of twists and turns on its length. Light travels down the fiber by a phenomenon known as *total internal reflection*. There are two types of devices used to input light to the optical-fiber waveguide: injection laser diodes for high speeds, and light emitting diodes for low speeds.

Optical fiber's greatest attraction is the high bandwidth of light transmission. Signal loss is minimal compared to metallic media. Optical fibers are immune to electrical and magnetic interference that other media experience, especially in a factory environment. Optical fibers provide greater reliability and data security since it is very easy to detect any taps illegally installed on fiber cables carrying confidential data.

There are currently many technical difficulties that impede the progress and feasibility of this new technology. An obvious drawback is that new techniques must be developed in the physical handling and installation of the cable. In addition, expensive and complex interface equipment is needed to make use of the high bandwidth.

The connection of optical fibers to the remainder of the network which is electronic in nature presents some serious problems to designers. Special two-way optoelectronic devices are needed. These devices accept light signals from one side and convert them to the corresponding electronic signals on the other side and vice versa.

The inherent characteristics of fiber-optic technology currently favour the implementation of the star topology. With the star topology the number of active connectors can be reduced. This technology is still new, but expects to dominate the design of all kinds of local area networks. Hence, an increased utilization of computer networks will be seen as current problems are solved and design costs pushed down.

2.5. Medium Access Protocols

The collection of devices on the local area network must share a common communication medium. The properties of the medium are such that only a single message can be successfully transmitted over a reasonably short channel at any one time. Two or more messages on the channel will interfere with one another and none will be correctly received. There must be some mechanism for controlling access to the transmission medium to produce correct exchange of information between devices. The set of mechanisms which permit the medium to be shared among several stations are referred to as *multiaccess protocols*.

An efficient multiaccess protocol would have the following desirable characteristics:

(1) permits high effective bandwidth utilization,

(2) produces low message delays,

(3) has the ability to support traffic of different types, different priorities, with variable lengths, and different delay constraints, and

(4) detects and recovers from errors that may result.

Multiaccess protocols differ by the centralized or distributed nature of the decision-making process, the static or dynamic nature of the channel bandwidth allocation algorithm, and the degree of adaptivity of the algorithm to the changing traffic of the stations. The multiaccess method selected is determined by the network topology, cost in designing the access method, its performance, its complexity and, the intended application.

A key issue in implementing any multiaccess protocol for the effective utilization of the channel is whether or not it should be *centralized control* or *distributed control*. A centralized medium access control mechanism offers the following advantages over the distributed scheme:

(1) There is greater control and monitoring functions are simplified.

(2) The electronics at each station become simpler and are not duplicated.

(3) Problems of synchronization, initialization, recovery and coordination are easy to solve.

The obvious disadvantages of centralized control over decentralized control are:

(1) There is a single point of failure, i.e, if the central controller fails the network becomes inoperative.

(2) There is the possibility of creating a bottleneck at the central controller under heavy load, thus, degrading the performance of the network.

Multiaccess protocols are broadly divided into four groups:

(1) controlled access or fixed assignment,

(2) random access or contention-based,

(3) demand assignment, and

(4) adaptive strategies.

Figure 2.5 shows a taxonomy of the medium access protocols discussed here.

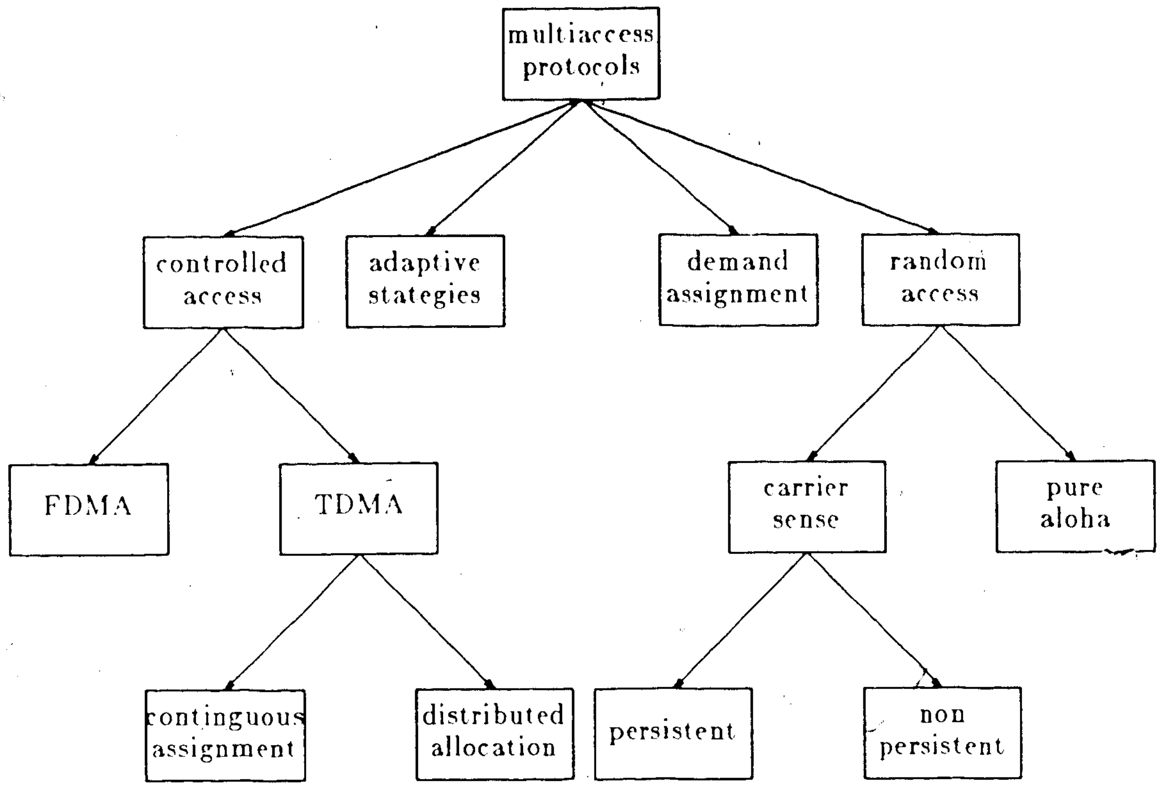


Figure 2.5 LAN Multiaccess Control Techniques

Controlled-access protocols offer collision-free access of the channel to a station. Transmissions are coordinated in such a way that no two stations will attempt to transmit messages simultaneously. The channel bandwidth is allocated to the stations, independent of their activity, by partitioning the time bandwidth space into slots in a predetermined fashion. These techniques take two forms; frequency division multiple access (FDMA) and synchronous time division multiple access (TDMA). FDMA consists of assigning a fraction of the bandwidth to each station and confining its access to that allocated sub-band. This is employed in broadband local area networks.

TDMA consists of assigning fixed predetermined channel time slots to each station in turn. The entire bandwidth is assigned to a station, but only during its time slot. This fixed allocation of time slots does not have to be equal for all users, but may be changed from time to time to reflect a station's needs. There are two variations to adjusting the time a stations gets. For the first, called *contiguous assignment*, the users are cyclically ordered in a time sequence in which they access the channel with their time periodically adjusted. For the second, called *distributed allocation*, all access periods are equal for all users, but the frequency of access may be different from one user to the next.

Fixed subchannel allocations are inefficient when communication is bursty as is typical in interactive terminal traffic. Much bandwidth is wasted granting access rights to stations with little or nothing to send.

2.5.1. Random Access Techniques

Random access techniques address the randomness of the traffic pattern and attempt to overcome the inefficient use of the bandwidth by offering access rights to a number of stations with the hope that exactly one station will have a message to send. This, however, may lead to the multiaccess problem which may result in two or more stations attempting to use the channel simultaneously. This scenario, in which a *collision* is said to have occurred, must be detected and resolved by the protocol.

Collisions degrade the performance of the channel utilization. To reduce the number of collisions two techniques are commonly used. One is to synchronize stations so that transmissions coincide with the boundaries of time slots (slotted protocols). The other is for the stations to sense the busy or idled state of the carrier prior to transmission (Carrier Sensed Multiple Access - CSMA protocols).

The functionality of the CSMA protocols depends on the fact that the propagational delay between end stations is small compared to the packet transmission time. In this environment, collision is avoided by having a station "listen" to the carrier before transmitting. If the carrier is busy the station does not transmit. CSMA protocols are further subdivided depending on the action taken after sensing a busy channel. In all cases if a station senses that a collision has occurred, it reschedules the transmission according to some randomly distributed delay. There are two main CSMA protocols known as *nonpersistent* and *p-persistent* CSMA depending on whether a retransmission occurs later or immediately following the current one with a probability p respectively.

With nonpersistent CSMA, retransmission is attempted after some random time such as the *truncated binary exponential back-off* strategy used with Ethernet (see Section 3.5.2). With p-persistent CSMA, the parameter p specifies the probability of delaying the retransmission by the propagational delay. There are other variations to nonpersistent and p-persistent CSMA protocols [45].

2.5.2. Demand Assignment Techniques

These access techniques require that explicit information expressing the station's need for use of the communication channel be exchanged. The information may be exchanged using either a centralized or distributed algorithm. These protocols include polling systems in which there may be a central controller whose function is to assign access rights to the stations based on the information received. The centralized strategies are unreliable since there is a single point of failure and the performance is degraded for systems with long propagational delays. With distributed algorithms all stations receive the same information and independently execute the same algorithm. There are several distributed demand assignment protocols used for the bus and ring topologies using an implicit or explicit token strategy [42].

2.5.3. Adaptive Strategies

Each of the protocols discussed previously has its advantages and limitations. No one scheme performs better than all other schemes over the entire range of system traffic. At low traffic a TDMA protocol is inefficient, but gradually provides higher utilization as the traffic increases for all stations. The converse is true for CSMA protocols, the channel is better utilized at low traffic, but quickly decreases as the station traffic intensity increases since collisions are more frequent. A truly adaptive strategy is one which offers high utilization of the channel independent of the varying load placed on the network.

There are few stand-alone protocols which have this adaptive charteristic. Adaptivity is usually achieved by a combination of two or more control strategies [45]. This requires that a certain amount of information be collected about the stations activities through some distributed method. The protocol may then be switched between control modes based on the information received in an effort to maintain a near-optimum performance at all times [45].

The Ethernet CSMA/CD protocol is used for the simulated PACS. It is an adaptive protocol which maintains high channel utilization by adjusting to the total load offered to the channel. The details of the Ethernet protocol are discussed in Section 3.4 of the next chapter.

Chapter 3

The Proposed PACS

Picture Archiving and Communication Systems (PACS) are becoming increasingly important in the medical field for storing, processing, and managing digitally formated radiographic images generated by various medical imaging devices. These systems have received much interest and attention in the medical community for the diagnosis and treatment of illness [16].

Two differences between PACS and the conventional local area network are:

(1) the large amount of data to be exchanged between devices on the PACS, and

(2) the vast amount of storage needed in PACS.

A typical picture with 512 x 512 pixels and an 8-bit grey scale corresponds to 2.1 Megabits of information. This magnitude of data will require high-volume storage units for storing digital images. In contrast, an I/O record size of tens to thousands of bytes are common for most computers. A new array of storage devices and technologies have recently become available, allowing digital (or electronic) archiving of medical images. The most popular is the optical disk which allows in the neighbourhood of 1000M bytes of data to be recorded on a single side of an optical disk.

One immediate concern is to determine whether or not the available local area network technologies as discussed in Chapter 2 can be combined with these storage devices to provide the desired response times for the application at hand. *Response time* is defined as the time interval between the instant the user types the last character of the input request and the instant the last pixel of the output image is presented at the display. A PACS is workable if it allows the radiologist and physician easy and fast access to radiological images of patients, past and present, at any time from any location. It is, therefore, important that all performance aspects of these rapid-response, high-volume systems be understood in order that they may be used as valuable tools.

This chapter describes a proposed architecture for a PACS. A design study with the aid of a simulated model is presented in order to establish meaningful system parameters and performance parameters which may be used in the design of a PACS. Such parameters include the expected response time to be experienced in retrieving or processing an image and the communication channel capacity needed to provide the required response times.

3.1. The PACS Architecture

The proposed system (see Figure 3.1) consists of a single 500m bus communication link and attached at points along the bus are several imaging devices which may be divided into four classes. The first class includes all digital image acquisition devices from which the still frame form of a digital image can be acquired. Picture storage devices constitute the second class with at least two levels of storage, an active storage and an archival storage. The third class consists of an image processor that is connected to the bus. The fourth class of devices is the user *workstations* which provide the picture viewing facility.

Common acquisition devices are ultra sound, nuclear medicine systems, X-ray computed tomography scanners and magnetic resonance imaging systems. These devices may be installed at any point on the bus. They all produce digital images that allow the radiologist to perform more significant image transformations than the conventional analog counterpart. Digital images can be manipulated to construct different views of the subject, to measure different densities of the tissue and other enhancements only possible with digital processing.

For image storage, an erasable medium such as a magnetic disk may be used to provide active (on-line) storage serving as an intermediary storage device.

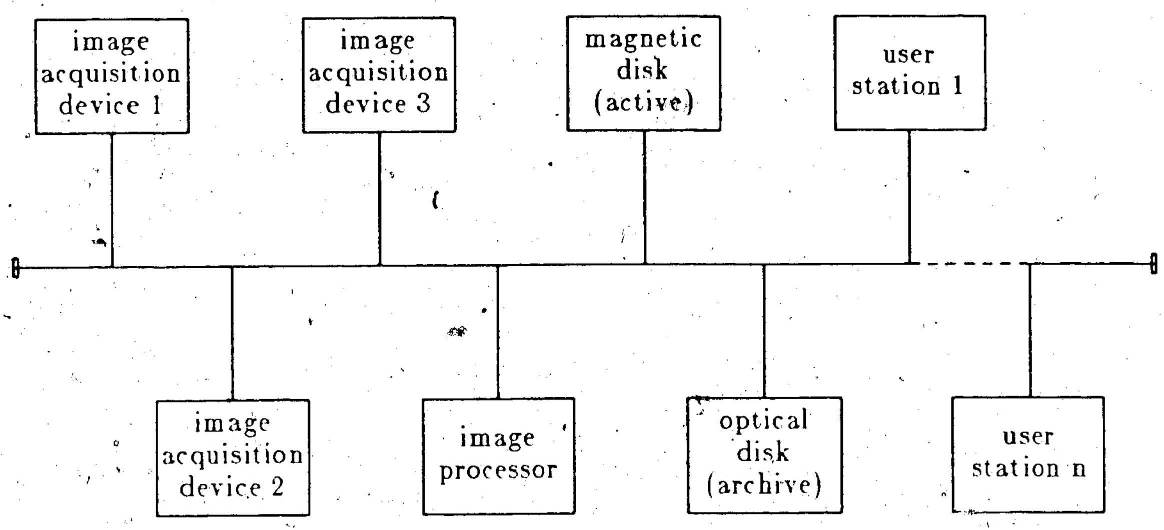


Figure 3.1 Bus Configuration for the Simulated PACS

The most recently referenced images which are acquired, inspected for completeness, processed or collated are kept in active storage. For functional reasons, the active store should not exceed the equivalent of one week's work (the number of images processed in one week) [11]. Archive storage is done on a read-after-write-once medium such as optical laser disk. For this application, it is desirable to keep all images acquired for an indefinite period of time. In addition, these disks have significantly higher capacity than the magnetic disk and are thus suited for archival storage. With either medium, the retrieval time is minimized by computer controlled search and access. On-line retrieval (retrieval from active storage) time should be less than two seconds per image while that for archival retrieval should be less than ten seconds per image [11].

The image processor is a computer system which provides enormous processing power to the workstations. The processor will be shared by the workstations eliminating expensive duplication of processor power at the workstations.

The workstations will display images stored or processed appropriate to the radiologist's needs. The workstation includes memory for local image storage. In the model presented it is assumed that enough RAM memory is provided to allow a maximum of four images to be stored locally. Hence, 512^2 x 8 images require 262 Kbytes of memory, 1024^2 x 8 images require 1.049 Mbytes and 2048^2 x 8 require 4.1943 Mbytes. Each workstation is provided with processing power; however, this is limited and is restricted to performing local functions such as editing the image on the screen. Other graphic functions and local image processing are performed by the local processor. It is expensive and, hence, not economically feasible to provide all the processing power at each of the workstations. An array of display devices are available on the market which offer different levels of sophistication regarding resolution, colour, and intelligence. In this system the smallest image to be displayed has resolution of 512 x 512 picture elements with 8-bit intensity levels. To explore the system activity for larger images, display devices with higher resolution are considered.

These different devices all communicate with each other via an Ethernet protocol on a single 500m bus. The bus may be extended to 1500m through the use of repeaters. Any bus transmission medium may be used that will provide the high capacity needed. In this system, the communication bus is assumed to be error-free.

3.2. Image Acquisition

An image acquired by an input device is said to be stored when it is transferred from the source or acquisition device to active storage. This input activity as depicted in Figure 3.2 and called *generate* is performed by a technician at the image acquisition device. It is assumed that: input devices produce a single frame of a digital image to be stored in the system; the image acquisition times are random, i.e. image arrival behaviour is Poisson. At the input device the technician decides when an image is to be stored and initiates the corresponding request.

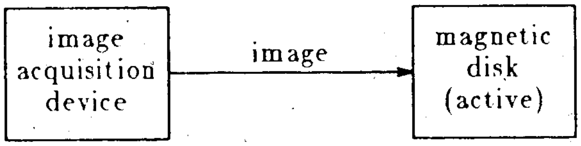


Figure 3.2 Image Acquisition Communication Path

The image is then stored in a buffer at the input device until it is transmitted to active storage device via the communication link.

3.3. User Activities and Workload Characteristics

The PACS provides the radiologist with an interactive interface to picture information. The radiologist sits at a workstation and conducts an interactive session with the information on the screen by the use of a keyboard, mouse or other interactive device. The activities defined for the simulated system, and performed by the user at a workstation, are divided into five groups: single-image display, multi-image display, edit, manipulate and browse. This mix of activities performed by a user is given by a probability distribution function that states the percentage with which each activity is performed. For example, for all activities performed, 10% may be single-image displays, 30% multi-image displays, 30% edits, 20% manipulations, and 10% browses. The overall workload on the PACS consists not only of the above activities from all user workstations, but also includes external input from the image acquisition devices and from the images transferred from the archival storage to the active storage. The transmission between the archive and active storage devices is transparent to the user, but of course impact on the traffic on the channel. The initial research on the workload activities is reported in [29].

A user session consists of two hours of continuous user activity at a workstation. A user session includes a short break of five to ten minutes at the end allowing a change of user at the station. It is assumed that the first activity of a user session is a single-image display with probability one. This system allows for multiple workstations, hence, several user sessions will be conducted simultaneously. The details of the five user activities will now be described.

3.3.1. The Display Function

The most important function of a workstation is concerned with displaying images which are previously stored on the PACS. The two display modes possible are *single-image display* mode and ***multi-image display*** mode. In single-image display mode, a single image is displayed at the workstation at a time. Multi-image display mode supports workstations with multiple screens. It is assumed that up to four images will be returned to complete each multi-image display request.

Figure 3.3 shows the communication path for the image retrieval protocol. The radiologist at the workstation makes a request for a specific image which is delivered via the communication net to the workstation. In this system only the most recently referenced images are kept in active storage and "old" images are sent to archive storage (e.g., optical disk). Furthermore the current location of a requested image is unknown to the radiologist.

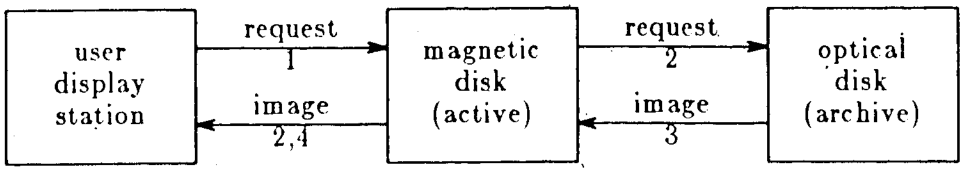


Figure 3.3 Single-Image Display Communication Path

Consequently two communication paths are possible for displaying an image depending on whether the image currently resides on active storage or on archival storage. If the image is on active storage it is immediately posted to the requesting workstation. Images not on active storage must first be brought from archival storage to active storage where a copy of the image is kept before it is sent to the requesting workstation.

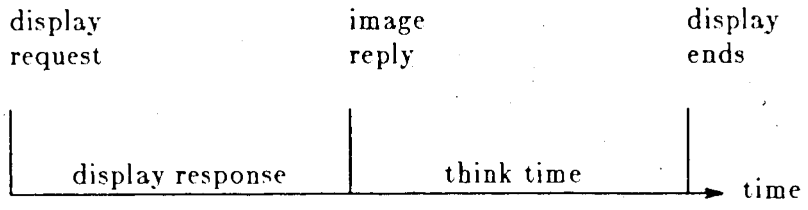


Figure 3.4 Time Sequence for Single-Image Display Function

Both paths will produce different retrieval response times. After the image is delivered to the workstation, the user needs a think period for examining the image and deciding on the next activity to be performed. Figure 3.4 shows the time sequence of events for the display activity.

3.3.2. The Edit Function

The *edit* activity is used to edit images which are currently being displayed at the workstation. This image would have been accessed by a previous activity such as display. Modifications performed locally at the workstation include the addition of a brief note or icon for future reference or flagging the image for deletion from the system.

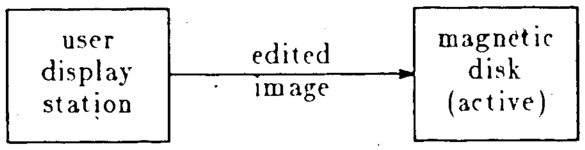


Figure 3.5 Edit Communication Path

The edited image is then transmitted to active storage where it replaces the previous unedited version or causes it to be deleted from the system in the case of a deleted image as depicted in Figure 3.5. Figure 3.6 shows the time sequence for the edit activity.

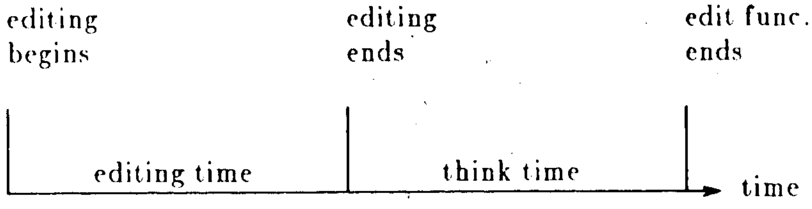


Figure 3.6 Time Sequence for Edit Function

A time interval is required for editing the image and this editing process is then followed by a think time during which verification of the desired modification is made. All editing functions are done by the local processor at the workstation.

3.3.3. The Manipulate Function

Extensive modification to an image such as image enhancement or image smoothing requires more processor power than is available at the workstation. Hence, manipulation activities require the image processor of the PACS. Using the communication facility, the image is transmitted to the processor which performs the required enhancement and retransmits it to the workstation (Figure 3.7). After the enhanced image is received at the workstation, the image is redisplayed for final inspection. The radiologist has the option of storing the enhanced image or rejecting it. At this stage, the manipulation function is complete and the next activity may be initiated at that station.

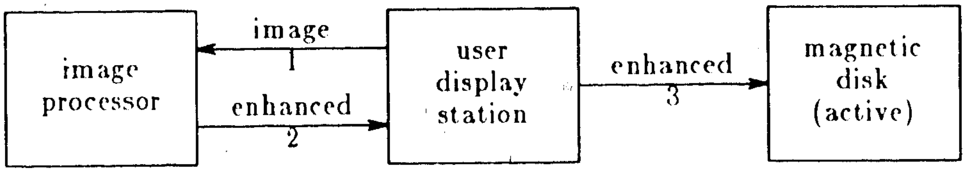


Figure 3.7 Manipulate Communication Path

All enhanced images kept are now considered new images. The new image is stored in buffers at the workstation to be transmitted to active storage.

The total time required to manipulate an image is the sum of the time to transmit the image to the processor after the request to manipulate is made, a time period needed to actually process the image, the time to transmit the processed image back to the workstation and the final think time after the image is received. Figure 3.8 depicts the time sequence of events associated with the manipulate activity.

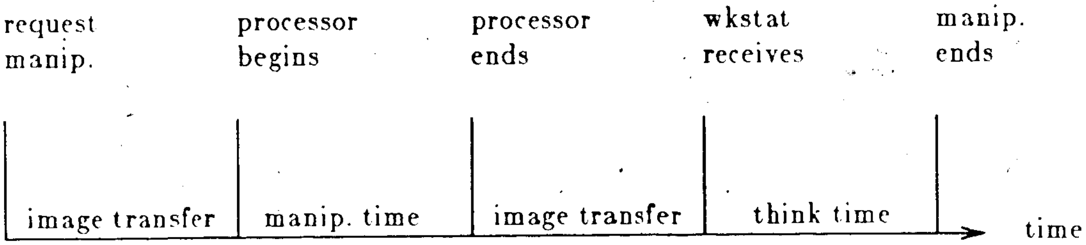


Figure 3.8 Time Sequence for Manipulate Function

During the entire interval since no other activity can be performed at the display station, the radiologist waits for the response.

3.3.4. The Browse Function

The browse activity provides the radiologist with the facility for searching the image database for a particular image with some specified characteristic. For functional and performance reasons, the complete image from a browse activity is not initially returned to the workstation. Instead a low resolution image of 4^n pixels is returned where n successively takes on the values 2,3,4,... to a maximum value N. If the size of an image (horizontal x vertical) is I pixels then $N = \log_4 I$, so that for the value $n = N$ the complete image will be returned. However, the radiologist may be provided with enough image detail at an earlier stage in the partial search to make a decision on whether to abandon the transmission of that image and continue the search with a next image or to request the complete image from active storage. This progressive procedure reduces the total data transmitted in a browse activity, since, for each of the images browsed less than the full image is returned except for the last image.

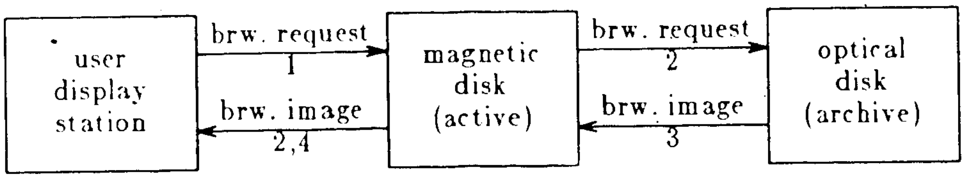


Figure 3.9 Browse Communication Path

Figure 3.9 shows the communication path for a browsed image which may reside either on active storage or on archival storage.

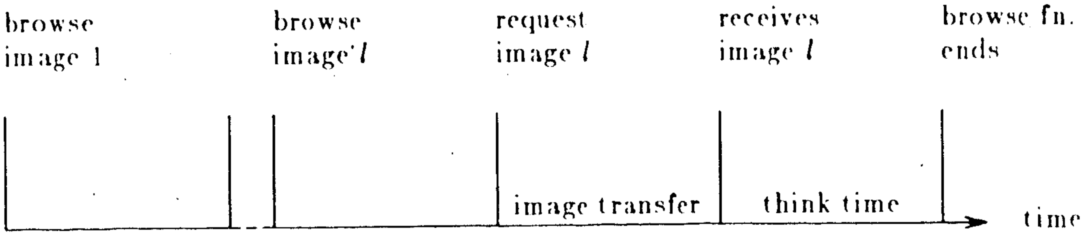


Figure 3.10 Time Sequence for Browse Function

Figure 3.10 shows the time sequence of events associated with the browse activity. After l images are browsed according to the progressive scheme described above, the final image is requested from storage as the required one. This is then followed by a think period during which the user visually analyzes the image.

3.4. The Ethernet Network

Ethernet is a local area network developed jointly by Xerox Corporation, Digital Equipment Corporation, and Intel Corporation [47]. The Ethernet LAN provides a communication facility for high speed data exchange among computers and digital devices located within a moderate geographical area.

Ethernet is intended primarily for use in areas such as office automation, distributed data processing, and other situations requiring economical connection for local communication carrying bursty traffic at high data rates. It should be noted that Ethernet is not intented for real time applications or applications requiring a guaranteed response time [47].

The primary physical specifications of Ethernet include:

(1) a maximum data rate of 10 Million bits per seconds (10 Mbps).

(2) a maximum station separation of 2.5 Kilometers (5 x 500m segments).

(3) a maximum number of 100 interfaces per 500m segment.

(4) The medium is a shielded coaxial cable that uses baseband signaling.

The Ethernet protocol is fully distributed and employs a statistical algorithm for contention resolution (CSMA/CD, see Chapter 2). The truncated binary exponential backoff algorithm (see Section 4.3) is used to stabilize the network under all load conditions, in the sense that the delivered traffic does not decrease as the total load offered increases. The message protocol permits variable size packets to be transmitted. The Ethernet packet accommodates a data field of 368 to 12,000 bits (18 to 1500 bytes) of information.

The chief goal in the Ethernet design is fairness and simplicity, only necessary functions are incorporated, thus minimizing the cost of connection. All stations have equal access to the network with no priority given to any station. Error control is limited at the lower layers to the detection of bit errors in the physical channel, and the detection and recovery from collision. Complete data error control is provided at the higher layers of the network architecture which is outside the scope of the Ethernet specifications.

3.4.1. The PACS Network Model

The physical communication system and the protocols interfacing with workstations are two essential parts of any PACS. The PACS model presented uses Ethernet on a 500m baseband coaxial cable system for image transmission. The cable connects several imaging devices. The quantity of data in an image that will be exchanged among the devices of the PACS and the transmission speeds are limiting factors to the maximum number of devices to be connected.

Bridge Communication Inc. provides communication servers (called CS/1) which can connect up to 32 standard imaging devices to an Ethernet [15].

Since the Ethernet protocol specifies a maximum data field of 1500 bytes per frame transmitted over the network channel, images must be fragmented into packets of appropriate sizes before they can be transmitted. The original image is reconstructed at the destination from the sequence of packets received.

The PACS communication model presented here adopts the layered approach to computer network architecture, based on the first three layers of the reference model of the Open Systems Interconnection proposed by the International Standards Organization [12]. The corresponding three layers for this PACS network model are as follows:

(1) the Physical Layer,

(2) the Data Link Layer, and

(3) the Data Network Layer.

In the model simulated, the communication channel is assumed to be error free so that data is neither lost or mangled. Only the above three network layers are addressed in this design. These protocols are fully distributed, hence, each devices executes the protocol independently.

3.4.2. The Physical Layer

The main design issue concerned with the physical layer is the transmission of the information bits over the channel. This layer provides a channel with a capacity of C bits per second. The channel can therefore transmit D bits of raw data from source to destination in t seconds, t is given by:

$$t = \frac{D}{C} + d,$$

where d is the propagation delay between source and destination. The channel capacity is one of the system variables considered in evaluating the performance of the system designed.

3.4.3. Data Link Layer

The second layer takes the raw transmission facility provided by the physical layer and transmits packets from source to destination, hence providing an interface for the next higher layer (the Data Network Layer). The three main functions of the Data Link Layer are:

(1) data encapsulation,

(2) data decapsulation, and

(3) link control management

In *data encapsulation*, performed just before a packet is transmitted, the packet is framed by prefixing and postfixing a number of information fields to the data transmitted (see Figure 3.11). Two address fields are prefixed to identify the destination and source stations for this frame. A third prefixed field indicates the type of frame to be encapsulated, this field is used by the higher layer protocols. The data field contains the user data to be sent on the channel. Postfixed to the data field is the frame check sequence used by the data link layer to detect collisions or any errors that occur during the frame transmission.

Data decapsulation, the inverse of data encapsulation, occurs just after a frame has been received by the data link layer. Frame boundaries are recognized, the frame check sequence is generated and validated with the received one. Accepted frames are disassembled, fields are striped away and the resulting packet is passed to the next higher level protocol (picture network layer) at this receiving station.

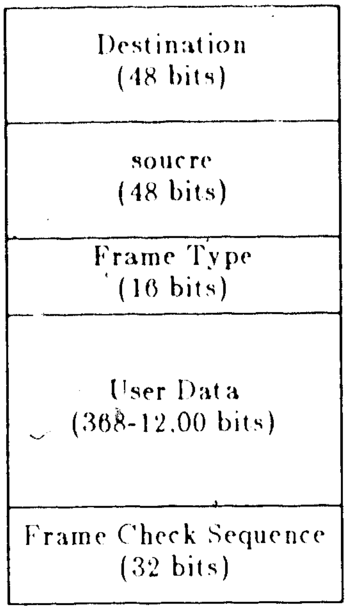


Figure 3.11 The Ethernet Packet

Link control management is concerned with channel allocation and contention resolution. It executes the protocol for gaining access to the communication channel and for removing a frame from the channel by the appropriate station.

To transmit a constructed frame (a packet to which the subfunctions of framing, addressing and the adding of error detection information has been performed), the data link layer protocol first senses whether or not the channel is busy to avoid collision with other traffic on the channel. If the channel is free then the station wishing to transmit puts its information on the channel. Whenever a station senses a busy channel it defers its transmission to the passing traffic. The station continues to sense the channel until the channel becomes free (the last bit has passed) and then after a brief interframe delay its transmission is initiated. The interframe delay facilitates the recovery of good frames from damaged ones and allows all transmitting devices to compete faily for the channel.

When multiple stations attempt to transmit at the same time, it is possible for their transmission to interfere with each other despite their sensing an idle channel. The collision of transmissions occurs only during the initial part of a transmission, called the collision window, which is the time the signal needs to propagate to all parts of the Ethernet channel, referred to as the *propagation delay* of the link. The length of the collision window is equal to the round-trip propagation time of the physical channel which in this case is 6 microseconds.

In the event of a collision, the data link layer executes a collision detection protocol. First the collision is enforced by transmitting a special bit sequence called the jam signal (48 bits long). The jam signal ensures that the duration of the collision is sufficient to be detected by all transmitting stations. After the jam signal is sent, transmissions are terminated for all stations and are rescheduled to occur after a random time period.

The random time period selected, before retransmission is attempted, is a function of the number of previous retransmission attempts made for the same packet at a specific workstation. The controlled randomized process, called *truncated binary exponential backoff*, is adopted in an attempt to resolve collisions as well as to adjust channel load. The length r of the random interval, before the j^{th} retransmission attempt, is a multiple m of a slot time (a slot time is the sum of the propagation delay and the time for the jam signal). The multiple m is a uniform random number in the range $0 \leq m \leq 2^k = min(j,10)$, where $k = min(j,10)$.

Frame reception includes both data decapsulation and some data link management aspects. Data decapsulation is comprised of the functions of address recognition, frame check sequence validation and frame disassembly producing the original data packet. The data link layer main management function is to filter out collision fragments from the completed incoming frames received. The packets are then passed to the next higher layer (the data network layer).

3.4.4. The Data Network Layer

The third layer provides the interface between the computer at the workstation and the network. The fragmentation of an image is done by this layer. At the source the image is partitioned into a set of packets of maximum size, each containing appropriate header information. These packets are passed down to the data link layer for transmission to the appropriate destination. Also at the data network layer,, house keeping functions such the collection of network traffic statistics may be performed; however, this level of sophistication is not added. At the destination, the data network layer receives the packets from the data link layer and reassembles the set packets received into the complete image before passing the image to the workstation.

A simulated model is used to represent the proposed PACS. Its implementation details are presented in the next chapter.

Chapter 4

The Simulation Model

The simulated model for the PACS and the workload discussed in the previous chapter is an event driven model [18]. The input parameters, the functions to be simulated and the output variables are discussed in detail in the following sections. An overall view of the main modules of the simulation model is shown in Figure 4.1.

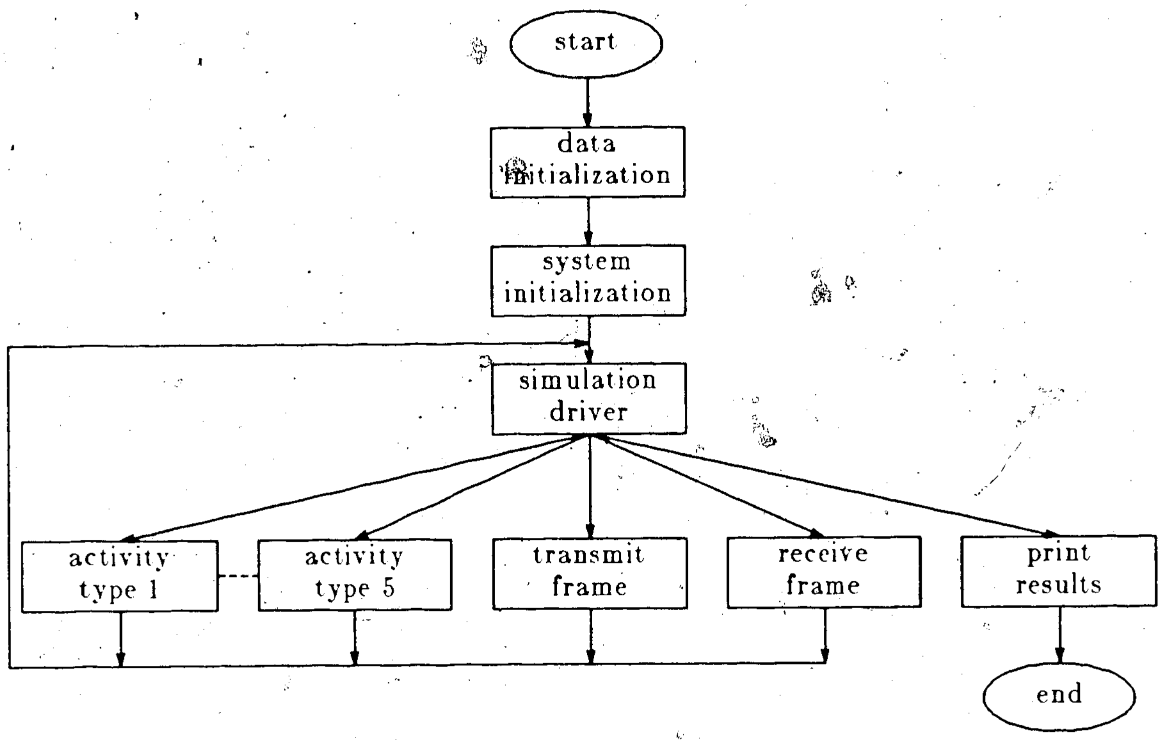


Figure 4.1 The Simulation Model

The main modules of the simulation model are:

(1) a data initialization module,

(2) a system initialization module,

(3) a simulation driver module, and

(4) a print result module.

The *data initialization* module initializes all the output variables used for the collection of performance information as the simulation proceeds. Details of the output variables are discussed in Section 4.3

The *system initialization* module builds the simulated PACS by setting up the necessary queues, and reading in all system input parameter values and user workload characteristics. The system parameters define the system configuration to be used for a simulation run. These parameters include the length of the simulation period, the capacity of the channel, the image size processed by the PACS, the number of input devices, the number of user workstations and the memory capacity of the active storage. Details are given in Section 4.1. The archival storage is assumed to be as large as necessary.

For each simulation run the length of the observation period is specified in days. Each day is assumed to have ten hours of consistent PACS usage which are divided into two-hour sessions. At the end of the day a system house-keeping function is performed. During this period, the house-keeping function reduces the number of images stored in active storage to thirty percent of its capacity by transferring all excess images to archival storage.

Initially, active storage provides one tenth of the total image storage of the PACS and is loaded to thirty percent capacity, while the rest of the images are considered to be stored in archival storage. Images processed during the course of a day are temporarily stored at the active storage. Should the active storage capacity be exceeded, the house-keeping function, which causes the least frequently referenced images to be transferred to archival storage is activated, thus freeing active disk space.

The system initialization module also specifies the random time before the first image arrives at each input device. The interarrival times for images acquired are taken from a truncated exponential distribution with a specified arrival rate.

The first user activity for each workstation starts during a small random time of about 30 seconds at the beginning of a session. The first activity function that a user performs at the workstation is assumed to be a single-image display. Subsequent activities (single-mode display, multi-mode display, edit, manipulate and browse) are determined from specified probability distributions as discussed in Chapter 3.

The *simulation driver* determines the "next event" occurring in the system, the time of that event and the associated device. The next event defines the next state of the system. This event is found by maintaining a queue of events in chronological order. The lowest time event is removed from the event queue and processed. A subroutine corresponding to that event is triggered which performs the appropriate actions causing the transition to the next system state. In processing an event a projected event may be generated and added to the event queue. For example, if the system is currently initiating the transmission of a packet then the next receive time of that packet at its destination is determined and the corresponding event is inserted into the event queue. The time for the *receive* event is calculated from the channel capacity, packet size, and propagation delay of the channel and will be processed at the correct time.

There are five user initiated events corresponding to the five user functions (single-image display, multi-image display, edit, manipulate and browse) and two system initiated events (transmit and receive). In addition, there is an event associated with image input which causes newly acquired images to be stored in active storage.

The simulation driver takes the system through the eight events until the specified simulation time has expired.

The *print result* segment is entered after the simulation is completed. The results of the specified performance measures are calculated from the output variables and are presented in tables. Note that after each user section the measures for the mix of activities are calculated and averaged over all users, then at the end of the simulation run these values are averaged over all sessions.

At the center of the model is a pseudo-random number generator that uses the *mixed congruential* method to generate values from a uniform distribution between 0 and 1 [17,26,36]. This is by far the most popular pseudo-random number generator used for producing a uniformly distributed sequence, X_0, X_1, X_2,... The general formula is given by:

$$X_{n+1} = (aX_n + c) \bmod m,$$

where:

m is the modulus, $m > 0$,

a is the multiplier, $0 \leq a < m$,

c is the increment, X_0 is the starting value (seed), $0 \leq X_0 < m$.

By choosing the values for m, a, c, and X_0 carefully, the sequence of numbers produced can be shown to be completely random with a period of maximum length m [26]. The resulting uniform sequence of random numbers in the range (0,1) can be used to generate values of specified statistical distributions.

For example, if images arrive at the input devices with rate r and have a Poisson behaviour then the interarrival time t between images is generated from an exponential distribution where t is given by:

$$t = -\frac{1}{r} ln(1 - p),$$

and p is the probability that an image arrives at an input device in the next time interval t.

The performance of the PACS depends on the system parameters and on the input workload placed on it by the users. The input workload is generated from two sources (see Figure 4.2). The first source is from the external image input devices, and the other is from the activities created by the user.

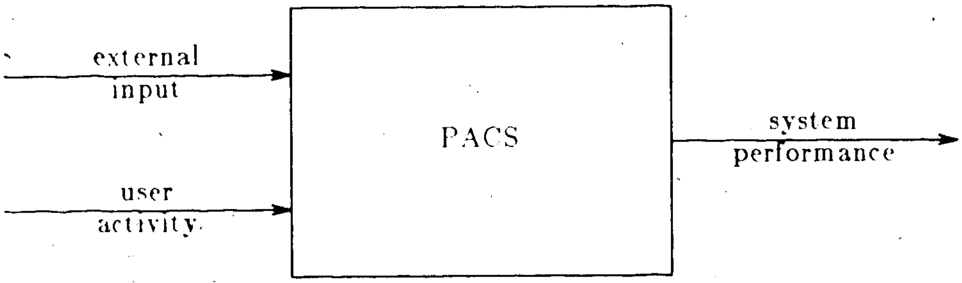


Figure 4.2 The PACS Workload Model

The measured system output performance indicates how well the system is performing its function. The output performance will, in turn, influence the number of activities performed by the users. The following three sections describe the input to the system, the system functions and the measured output variables. The default values and assumptions made for the simulated PACS is given in Section 4.4. Some highlights of the simulation implementation are briefly discussed in Section 4.5.

4.1. The Input System Parameters

The input system parameters to the simulated model include the following:

(1) the bus transmission speed (capacity),

(2) the number of image acquisition devices (input devices),

(3) the number of user workstations connected to the bus,

(4) the size of the image to be processed,

(5) the capacity of the magnetic disk used for active storage, and

(6) the user workload characteristics.

The channel transmission speed is one factor which limits the minimum achievable image response time. For example, to transmit a digital image of standard size (512 x 512 x 8 = 2.1M bits) on a channel with 1 Mbps capacity requires a theoretical transmission time of 2.1 seconds.

	Ideal Transmission Times (secs)			
Image Size (bits)	19.2 Kbps	1 Mbps	10 Mbps	100 Mbps
256^2 x 8 = 524,288	27.3067	0.5243	0.0524	0.0052
512^2 x 8 = 2,097,152	209.2267	2.0972	0.2097	0.0210
1024^2 x 8 = 8,388,608	436.9067	8.3886	0.8388	0.0839
2048^2 x 8 = 32,554,432	1747.6267	32.5544	3.2544	0.3254
4096^2 x 8 = 134,217,728	6990.5067	134.2177	13.4218	1.3422

Table 4.1 Image Ideal Transmission Times

Table 4.1 gives a number of image sizes and their corresponding theoretical transmission times for various channel capacities, where the theoretical transmission time of an image is the ratio of the number of bits in the image and the channel capacity in Mbps.

Since the image sizes exceed the maximum packet size specified by the Ethernet protocol, images must be fragmented into a sequence of packets before transmission on the bus. Header information is added to each packet which effectively increases the total number of bits sent per image. For example a standard image of 512 x 512 x 8 bits consists of 175 packets which adds 25,200 (175 x 144) bits of overhead. In addition, packets may collide, further increasing the image transfer time. The actual transmission time including, for example, protocol overhead, contention period, and propagation delay, is called the *image transfer time* here. The image transfer time is thus defined to be the time interval between the instant the first bit of the first packet for the whole image is placed on the bus until the last bit of the last packet arrives at its destination. The values presented in Table 4.1 are ideal lower limits for the image transmission time. The simulated PACS include all the factors affecting transmission, hence, longer image transfer times are realized.

The simulated system may be configured with one to five image acquisition devices connected to the bus. Images are acquired from these devices according to a Poisson arrival process with different mean arrival rates $r_1, r_2, \ldots, r_5$. The results presented in Chapter 5 were produced with three input devices each with a specified average interarrival time of 5 minutes or an arrival rate of 12 images per hour. An average size radiology department deals with about 90,000 new images per year [27]. With 250 working days a year, this results in 360 images per day or 36 per hour (0.6 images per minute) over all three input devices. Hence, each one of these input device should generate 12 images per hour which gives an average interarrival time of 5 minutes.

The number of workstations connected to the bus can be one or more, the results presented in Chapter 5 show the potential maximum number for given image sizes, channel capacities, and workloads.

The image size is described by its horizontal and vertical dimensions giving the number of pixels in each row and column, respectively, and by the intensity level for each pixel. In the simulation runs the image size selected is common for all devices on the PACS. Note, however, that this may not be the case for a real PACS. Each type of imaging device may have different image resolution and format requirements, however, for a clear interpretation of the results the image size used is the same for all devices. The standard image formats of 512^2 x 8, 1024^2 x 8, and 2048^2 x 8 were selected for the results discussed in Chapter 5.

The active storage provides one or more 400M byte magnetic disks. The number of disk units selected depends on the average image space consumed daily during active storage. The simulated PACS is reconfigured by adding one or more 400M bytes disk as needed. 1600M bytes of active storage were used for the results in Chapter 5.

4.2. Simulated Functions

There are five user initiated events corresponding to the five user activity functions that may be executed at the workstation. The user makes a request to the system which then responds appropriately.

For the *single-mode display*, the request for a specific image causes a request packet to be assembled and transmitted to the active storage. If no other packet is awaiting transmission at that device and the bus is free the request packet is transmitted to the active storage device immediately after the packet is assembled. When the request packet is received at the active storage unit the corresponding image is placed in the transmit buffer as a series of packets to be sequentially transmitted to the requesting station. After the entire image is received and displayed at the workstation, a think time of random length is added, i.e. the image remains on the screen during that time period. The think time interval has a truncated exponential distribution (see Table 4.2) with an average time of about half a minute. After the completion of this single display activity, the next activity is specified. The total time for the display activity consists of the image response time plus the think time. For this system workload, it is assumed that ninety percent of the images displayed by the single-image display function will be accessed from active storage and the other ten percent from archival storage.

	Distribution Function Parameters			
Function think time (secs)	a	$\frac{1}{\mu}$	b	$E[x]$
Single-image display think time	2	30	120	29.64
Multi-image display think time	2	30	120	29.64
Edit think time	30	60	300	86.97
Manipulation think time	60	120	600	173.93
Browse think time	30	60	300	86.97
Image processing time (secs)	10	60	120	49.07
Image editing time (secs)	2	5	10	4.98
Image interarrival time (secs)	10	300	600	305.40

Table 4.2 Truncated Exponential Distribution Function Parameters

The *multi-image display* consists of two, three, or four simultaneous single-image display functions. This may be viewed as displaying two to four images on a multi-screen display unit. The number of images, 2 or 3 or 4, is selected from a uniform distribution with a probability of 1/3 for each. As for the single-image display function, ninety percent of all images displayed using the multi-image display function come from active storage with the other ten percent from archival storage.

The *Edit* function consists of an editing period (local processing time) followed by a think period. Both of these have truncated exponential distributions (see Table 4.2).

The *Manipulate* function causes the image to be transmitted from the user workstation to the processor. The amount of processor time needed at the processor ranges from 2 seconds to 10 seconds and has a truncated exponential distribution with an average time of 5 seconds. After the enhanced image is returned from the processor it is displayed on the screen for a random think time of about 120 seconds. The think time is taken from a truncated exponential distribution (see Table 4.2).

The *Browse* function. first requires the number n of low resolution images to be returned before a decision is made about the the acceptability of the complete image. The value n is randomly chosen from the range $(2,N)$ (n and N are defined in Section 3.3.4). Twenty-five percent of the time the full image is acceptable and, hence, the complete image is transferred from the storage. Seventy-five percent of the time the image is not desired and the browse procedure is repeated for one image after another until an acceptable image is found. The display of the accepted image is followed by a random think period. The browse function is considered complete only when a desired image is found, its transfer is complete and the user think period expires. During the display of the low resolution data no think times are included on the the assumption that the user views the low resolution images as they build up to form higher resolution images. The random think time after the final image is accepted is taken from a truncated exponential distribution with an average think time of 60 seconds.

The truncated exponential distribution function $f(t)$ is used in generating all think times and other time intervals used in the simulated model and is given by:

$$f(t) = \frac{\mu e^{-\mu(t - a)}}{1 - e^{-\mu(b - a)}},$$

where a and b are the lower and upper limits respectively of an interval t from a distribution with specified average value $1/\mu$ and $a \leq t \leq b$.

In addition to the user functions, there are system functions *transmit* and *receive* that include the communication protocols for transmitting and receiving images sent on the PACS (see Section 3.4.3).

The other modules which are executed by the simulation are the *system's house-keeping* module and the *clean up* module. The system house-keeping module transmits images from active disk storage to archival disk storage. This transfer of images, to provide space for the start of the next day, occurs at the end of each day. The number of images transferred is the minimum number that will free-up seventy percent of the total active disk capacity.

The *clean up* module is used to purge the system of all incomplete user activities when the two-hour user session terminates. Purging is done at each device with the exception of the input devices which are allowed to continue their input functions.

4.3. The Output Variables

During the simulation, values for several performance measures are collected and the minimum, average, maximum and standard deviation of these measures are calculated. These values are calculated for each user for each session, and then over all users for each user session. At the end of the simulation run the average values produced in each session are then averaged over all sessions. The values are presented in tables for each device type. The device types are the input device, the user workstation, the communication channel, the processor, and the active and archival storage devices.

The performance values for the measures on input devices are collected and calculated for the whole simulation run. For this system, it is assumed that these devices will be constantly available for the entire day. At each of the image acquisition devices the performance measures of interest are:

(1) the number of images acquired per user session,

(2) the number of images transmitted to active storage,

(3) the observed interarrival time of images to the acquisition devices,

(4) the system response time for storing an image on active storage, and

(5) the image queue length as seen by the arriving image before it can be transmitted along the bus.

Statistics for each user workstation are collected per session and for each of the defined user functions. Measures are separately determined for those functions which could access either the active storage or the archival storage. At the end of the simulation run, the average values for each session are averaged over all users and over all sessions. The performance measures include the following variables:

(1) the numbers of each activity initiated by the user,

(2) the numbers of each activity completed and, hence, the numbers of incomplete activities,

(3) the data transfer rate for each activity

(4) the system response time for each activity

(5) the observed think time for each activity, and

(6) the function time (function response time plus user think time) for each activity.

The performance measures collected for the bus (channel) includes the following:

(1) the number of images transmitted on the bus,

(2) the number of packets transmitted on the bus,

(3) the number of packet collisions experienced by the channel,

(4) the channel throughput,

(5) the image input rate to the bus,

(6) the image transfer time (averaged over full images only),

(7) the percentage of time the bus actually spends transmitting packets,

(8) the percentage of time lost due to packet collision,

(9) the percentage of time the bus is idle, and

(10) the length of the queue of incomplete images to the channel at the end of a user session.

Performance data are collected for the image processor every session and are then averaged over all sessions. The values collected for the image processor are for the following measures:

(1) the number of images manipulated per session,

(2) the number of incomplete images queued to be processed by the processor at the end of the session, and

(3) the number of processed images queued from the processor to be sent back to the workstations.

Similarly, performance data for the storage devices are collected for every session and are averaged over all sessions. The measures for the active storage device consist of:

(1) the number of images that are brought to active storage, and

(2) the number of images queued for the channel from active storage at the end of the session.

The measures for the archival storage device consist of:

(1) the number of images archived during a session due to a full active storage, and

(2) the number of images queued for the channel from archival storage at the end of the session.

Appendix A1 gives examples of tables of values for performance measures produced from a simulation run. The PACS is configured with four workstations connected to a 1 Mbps capacity channel and processes 512^2 x 8 images.

4.4. Assumptions and PACS Default Values

The data access to the storage devices is assumed to be fast and that data can be accessed at a rate which matches the channel capacity. If more than one image request is received at a storage device, a first-come-first-serve discipline is used to satisfy the requests.

It is also assumed than there is enough memory at the image processor to temporary store all images to be manipulated. The images are processed one at a time in order of thier arrival.

The communication channel is assumed to provide error-free communication, hence, packets are neither lost or mangled. A propagation delay of 3 microseconds for a 500m segment cable was selected. This corresponds to the average grade cable operating under moderate conditions. The collision window, the time during which collisions can occur, is taken to be twice the propagation delay. An inter-frame delay of 9.6 microseconds, as stipulated by the Ethrenet [illegible]ation, is selected.

The minimum packet size is taken to be 512 bits which consist of:

- a 48 bit destination field,

- a 48 bit source field,
- a 16 bit packet type,
- a 368 bit minimum data field (12,000 bits maximum), and
- a 32 bit frame check sequence.

Other Ethernet implementations may specify different minimum packet lengths. The maximum packet length is the length of a packet which contains a maximum data field of 12,000 bits, therefore, the maximum packet length is 12,144 bits.

4.5. Implementation Aspects

The simulation model was implemented using the Pascal programming language which provides a dynamic memory storage facility, i.e. memory space is provided as needed. Much of the parallelism and facilities to represent continuous process, features of many simulation languages, were not needed. The program, named *pacsim,* was executed on the University of Alberta's Amdahl 5680 mainframe under the Michigan time sharing (MTS) operating system.

Pascal offers only single precision for real numbers which for this machine corresponds to 15 decimal digits. To maintain the accuracy of the values produced, the time on the simulated clock for the model was expressed in microseconds. At the end of every user session the clock was reinitialized to zero to prevent round off errors which occur when the numbers expressed become large.

Chapter 5

Analysis of Results

In this chapter the simulation results and analysis of the data generated from a number of simulation runs are presented. The three main factors which influence the values obtained are:

(1) the number of workstations attached to the channel,

(2) the channel capacity, and

(3) the size of the image to be processed.

Each of these factors is varied and its effect on PACS performance is observed. The values for the specified performance measures are first obtained by averaging over a session, then the average values produced are averaged over all sessions.

There are three aspects to the performance analysis to be considered. The first consideration is from the user's point of view, for example, the effective transfer rate, or the response time for an activity. The second performance consideration is the workload statistics which give the number of each activity performed per session. The third consideration is the performance as far as the system is concerned, for example, the channel utilization, and the number of packet collisions. Consequently, the performance measures for the simulated system are respectively collected in three main tables.

Two performance measures which need to be defined are the *throughput* and *channel utilization* since there is no general agreement on their definition in the literature. *Throughput* is the rate of the actual amount of user data which can be sent over the channel summed over all devices during a specified time. The *user data* is the actual image data transmitted on the channel and excludes protocol overhead such as header information, the capacity lost due to collision, and request and control packets. The channel capacity or bandwidth is the maximum possible data rate that is provided by the physical channel (e.g., a 1 Mbps channel).

The *channel utilization*, which is also referred to as effective capacity, is defined as the ratio of throughput to channel capacity, i.e. the amount of user data that can be transmitted within a given channel capacity [41]. It is expected that as the total offered load increases, the channel utilization would also increase until it reaches the maximum effective capacity. Some shared systems become congested at this point and the throughput degrades rapidly as the maximum effective capacity is approached. The dynamic control algorithm employed in Ethernet ensures stability under high load and high channel utilization is maintained. The channel utilization must be less than one since there is protocol overhead and propagation delay, as mention above.

A good estimate of an upper bound for the maximum channel utilization achievable for the implemented Ethernet is obtained by considering the dynamics of transmitting a packet. Each packet transmission includes an interframe delay of 9.6 microseconds which is followed by the actual transmission of the packet. For a channel capacity of C Mbps, the number of bit times lost during the interframe delay is $9.6C$. The maximum length packet contains 12,000 bits of user information and 144 bits of header information. The bound on the maximum channel utilization, $U_{\max}$, is therefore given as:

$$U_{\max} = \frac{D_{\max}}{P + 9.6C},$$

where $D_{\max}$ is the length of maximum data field and P the length of a packet. Table 5.1 shows the maximum channel utilization achievable for channel capacities of 1, 10, and 100 Mbps. These values are calculated on the assumption that there is only one device transmitting at all times, that no packets collided and that 9.6 microseconds is greater than the propagation delay.

Channel Capacity	Max. Utilization
1 Mbps	0.9874
10 Mbps	0.9803
100 Mbps	0.9158

Table 5.1 Maximum Channel Utilization

With the simulated PACS it is difficult to achieve the maximum channel utilization for a few workstations and small image sizes since the user functions, as described in Section 3.2, are dominated by a relatively large think-time compared to the image transfer time. Higher utilizations are obtained when several workstations are connected to the channel or the image size increases. On the other hand, as more devices are added to the channel, the number of packet collisions increases since the images transmitted are large and consist of long sequences of packets. The packet collisions increase the image transfer time and, hence, the response time, and decrease the channel utilization.

5.1. Experiment 1

In this experiment, the image size is kept constant at 512^2 x 8 and the effect of the number of workstations and channel capacity on performance is observed. Tables 5.2, 5.3 and 5.4 give values for several performance measures as the number of workstations increases from 1 to 20 in increments of 4 for a channel capacity of 1 Mbps.

performance measure	1 wkstat	4 wkstat.	8 wkstat.	12 wkstat.	16 wkstat	20 wkstat.
display(1) d.t.r. [Mbps] (act)	0.9717	0.9044	0.8151	0.6789	0.5073	0.3423
display(1) response time (act)	2.1759	2.4650	2.9651	4.0282	6.6413	11.1036
display(1) think time (act)	28.9817	28.6047	29.7385	29.5300	29.4143	29.9413
display(1) function time (act)	31.1576	31.0698	32.7036	33.5582	36.0556	41.0449
display(1) d.t.r. [Mbps] (arc)	0.4829	0.4459	0.3940	0.2840	0.2057	0.1414
display(1) response time (arc)	4.3841	4.9512	6.1139	9.9832	15.1869	22.3017
display(1) think time (arc)	30.8121	30.8500	29.5805	30.1438	29.9791	30.0383
display(1) function time (arc)	35.1963	35.8013	35.6944	40.1270	45.1660	52.3401
no. images in display(n)	3.0088	2.9917	3.0120	3.0114	2.9947	2.9957
display(n) d.t.r. [Mbps] (act)	0.9748	0.9133	0.8207	0.6836	0.5228	0.3538
display(n) response time (act)	2.1667	2.4273	2.9253	4.0552	6.2003	10.4443
display(n) d.t.r. [Mbps] (arc)	0.4851	0.4454	0.3888	0.2900	0.2156	0.1498
display(n) response time (arc)	4.3515	4.9920	6.2586	9.6155	14.1268	21.3071
display(n) think time [secs]	29.2214	29.4067	29.9754	29.6594	29.6675	29.7699
display(n) function time [sec]	94.9601	95.6024	99.6479	102.6941	109.4264	123.5182
image editing time [secs]	49.2082	49.4872	48.9978	49.0476	48.9801	48.9570
edit think time [secs]	86.0842	86.4884	87.4299	87.4239	86.7328	87.0616
edit function time [secs]	135.2924	135.9756	136.4276	136.4715	135.7129	136.0186
image manip. time [secs]	4.8762	4.9512	4.9858	4.9169	4.9886	4.9466
mnaipulation d.t.r. [Mbps]	0.2395	0.2222	0.2011	0.1750	0.1497	0.1304
manip. response time [secs]	9.1973	10.0402	11.2230	13.0015	15.3643	17.4473
manip. think time [secs]	174.2366	174.0638	174.7581	174.3256	174.1414	174.0965
manip. function time [secs]	183.4339	184.1040	185.9812	187.6271	189.5057	191.5438
no. of low res. searches/image	5.4176	5.4861	5.4737	5.5372	5.4660	5.4885
no. of images inspected/browse	3.9474	3.8840	3.9089	4.0249	4.0022	3.9805
browse response time	14.4318	16.3709	20.9314	32.6857	56.8145	122.2541
browse think time [secs]	86.2173	85.0889	86.6868	86.2301	87.4258	87.7703
browse function time	100.6491	101.4598	107.6182	118.9157	144.2404	210.0244

Table 5.2 User performance measures per session as the number of workstation increases. The PACS image size is 512 x 512 x 8 and channel capacity is 1 Mbps.

performance measure	1 wkstat.	4 wkstat.	8 wkstat	12 wkstat.	16 wkstat.	20 wkstat.
no. of displays(1) requested	5.8700	24.4200	47.1700	71.0600	90.4500	105.0500
no. of displays(1) completed	5.8400	24.3000	46.9900	70.8000	90.1300	104.5000
no. of displays(1) aborted	0.0300	0.1200	0.1800	0.2600	0.3200	0.5500
no. of displays(1) (active)	5.3400	21.8600	42.3800	64.1600	81.2400	94.8900
no. of displays(1) (archive).	0.5300	2.5600	4.7900	6.9000	9.2100	10.1600
no. of displays(n) requested	16.2200	67.4700	131.5600	192.3200	246.7200	281.8200
no. of displays(n) completed	15.9800	66.4000	129.7500	189.3200	242.5500	276.7300
no. of displays(n) incompl.	0.2400	1.0700	1.8100	3.0000	4.1700	5.0900
no. of displays(n) (active)	44.1800	181.6900	357.7900	521.5400	664.8800	759.4600
no. of displays(n) (archive)	4.8200	20.1100	38.1100	57.2800	73.8000	84.7200
no. of image edits requested	16.6800	66.0300	129.2100	190.8600	246.7900	283.5900
no. of image edits completed	16.3400	64.6800	126.4800	186.9400	241.6400	277.9700
no. of image edits incompl.	0.3400	1.3500	2.7300	3.9200	5.1500	5.6200
no. of image manip. requested	11.3800	44.3300	87.9000	128.9600	163.9200	188.5600
no. of image manip. completed	11.0700	43.1700	85.3800	125.2000	159.2500	182.6800
no. of image manip. incompl.	0.3100	1.1600	2.5200	3.7600	4.6700	5.8800
no. of image browse [illegible]quested	5.5900	21.7300	43.0600	63.3100	81.4600	95.0500
no. of image browse completed	5.5100	21.4300	42.3000	62.2500	79.7700	92.1900
no. of image browses incompl.	0.0800	0.3000	0.7600	1.0600	1.6900	2.8600
no. of browses from (active)	0.5200	2.3600	4.4300	6.6300	8.0100	9.1700
no. of browses from (archive)	4.9900	19.0700	37.8700	55.6200	71.7600	83.0200
end of session idled time	512.9000	2048.7672	4115.7605	6134.2551	8121.2625	10185.0174

Table 5.3 Workload statistics per session as the number of workstation increases. The PACS image size is 512 x 512 x 8 and channel capacity is 1 Mbps.

performance measure	1 wkstat.	4 wkstat.	8 wkstat.	12 wkstat.	16 wkstat.	20 wkstat.
no. of images transmitted	426.5900	1454.5700	2798.4100	4282.6900	5551.7300	6422.4900
no. of full images transmitted	182.4900	518.7800	958.9700	1499.7300	1965.2200	2279.2300
no. of packets transmitted	39087.8600	117577.5700	221194.4200	342972.7800	447128.5800	518491.8600
no. of packet collision	1852.8800	20872.5700	79415.0700	197448.4800	352020.9100	502994.2700
image input rate (images/sec)	0.0592	0.2020	0.3887	0.5948	0.7711	0.8920
total user data sent (Mb)	463.1248	1388.5052	2610.4863	4049.4280	5280.1695	6123.3943
image transfer time (secs)	2.1999	2.4208	2.7872	3.3100	3.9016	4.4638
percentage busy (successful)	6.5126	19.5276	36.7142	56.9507	74.2592	86.1180
percentage busy (collision)	0.0010	0.0114	0.0422	0.1012	0.1746	0.2416
percentage idled	93.4864	80.4610	63.2436	42.9481	25.5662	13.6404
incomp. images at channel	0.0200	0.0100	0.0200	0.0300	0.0900	0.0500
no. of images manipulated	11.0700	43.1700	85.3800	125.2000	159.2500	182.6800
incompl. images at processor	0.0	0.0	0.0	0.0	0.0	0.0
incompl. images from processor	0.3100	1.1600	2.5200	3.7600	4.6700	5.8800
no. of images stored at active	109.9100	223.2500	368.5800	512.5600	637.4300	722.1500
incompl. images at active str.	0.0	0.0	0.0	0.0200	0.0400	0.0400
no. of images archived	0.0	0.0	10.0700	135.9600	244.1200	316.4300
incompl. images at archives	0.0	0.0	0.0	0.0	0.0	0.0

Table 5.4 System performance measures per session as the number of workstation increases. The PACS image size is 512 x 512 x 8 and channel capacity is 1 Mbps.

Table 5.2 gives those performance measures which concern the user, and measure the user satisfaction, e.g., response time or the effective data rate. Table 5.3 gives the workload statistics for the number of activities performed by the users at workstations per session, and summed over all devices, e.g., the number of edit activities. Table 5.4 on the other hand gives those performance measures which concern the system performance analyst and manager, e.g. the data throughput or number of collisions the channel experienced.

5.1.1. User Measures

In Table 5.2 the response time and user think-time are given for images displayed from both active storage and archival storage for the single-image display activity. The total display function time is the sum of the response time and user think-time. The *data transfer rate* (d.t.r) is defined as the ratio of the amount of user data which is returned for the activity to the response time of the activity.

Table 5.2 shows for 1 workstation that there is an average of 3.0088 images per multi-image display activity. Ninety percent of the images displayed are accessed from active storage with an average image response time of 2.1667 seconds while ten percent are displayed from archival storage with an average image response time of 4.3515 seconds. The expected response time for displaying an image is therefore (0.9 x 2.1667 + 0.1 x 4.3515) seconds, i.e. 2.3852 seconds. The average think time per single display is 29.2214 seconds. Hence, the expected time for a multi-image display activity is 3.0088 x (2.3852 + 29.2214) = 95.0979 seconds. This expected result compares favourably with an average value of 94.9601 seconds obtained from the simulation.

The editing activity consists of the actual time to perform the required editing (local processing) and the final think period. Both these times are independent of the number of workstations connected to the channel and as a result the total edit function time is approximately the same for PACS with different numbers of workstations as shown in the respective columns of Table 5.2. Storing of the edited image on active storage impacts the traffic on the channel, but is transparent to the user.

The manipulate activity uses the channel to transmit the image to and from the processor. The total manipulate function time increases as the number of workstations increases, as a result of the increase in image response time. The manipulate response time consists of the time to transmit the image to the processor, the time needed to process the image, and the time to return the image to the workstation. From Table 5.2, the PACS with 1 workstation has an average manipulate response time is 9.1973 seconds which is then followed by an average think-time of 174.0638 seconds giving an average time of 183.4339 seconds for the manipulate activity.

The response time of the browse activity consists of the total time to find an acceptable image. For example, in Table 5.2, 1 workstation gives a response time of 14.4318 seconds during which an average of 3.9474 images were inspected before one was accepted. For each image inspected an average of 5.4176 levels of increased resolution were necessary to determine whether or not the image is acceptable. An average think time of 86.2173 seconds follows the transfer of the final image giving a total average browse function time of 100.6491 seconds.

5.1.2. Workload Statistics

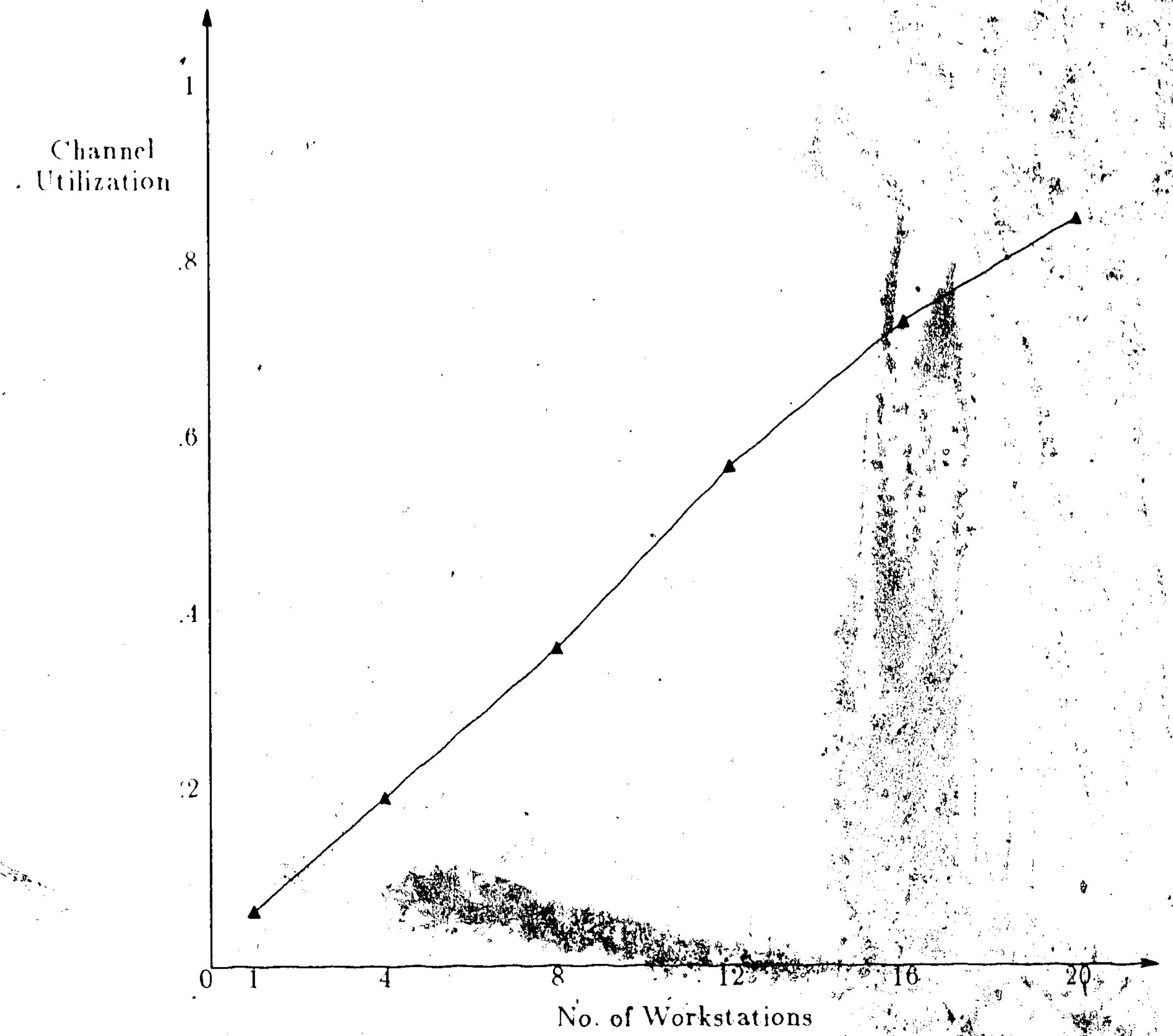
Considering the case of 1 workstation in Table 5.3, the total average time a user spends at the workstation per session is determined. Table 5.5 give the total time for each type of activities performed. Including an average idle time of 512.90 seconds, the user activities accounts for 7011.2667 seconds of a user session which is 2 hours or 7200 seconds. The values are averaged over the sessions, hence, an exact sum is not produced.

User Activity	Ave. Completed	Ave. Func. Time	Total Time
Single Display (act)	5.34	31.1578	166.3827
Single Display (arc)	0.53	35.1963	18.6540
Multi Display	15.98	94.9601	1517.4624
Edit	16.34	135.2924	2210.6778
Manipulate	11.07	183.4339	2030.6133
Browse	5.51	100.6491	554.5765
		User Idle Time (secs)	512.9000
		Total User Time (secs)	7011.2667

Table 5.5 Total User Time per Session

5.1.3. System Measures

Figure 5.1 shows the channel utilization as a function of the number of workstations connected to the channel. The channel utilization is a useful measure in that it reflects the PACS behaviour under heavy load. As the number of workstations increases, and hence, the total load offered to the channel, utilization increases linearly as predicted. With 20 workstations the average channel utilization per session is 0.8505 while the maximum utilization is 0.9305 which approaches the maximum value of 0.9874 that can be obtained. The protocol overhead and the channel bandwidth lost due to collisions places this limit on the maximum achievable utilization.



512^2 x 8 image, 1 Mbps capacity.

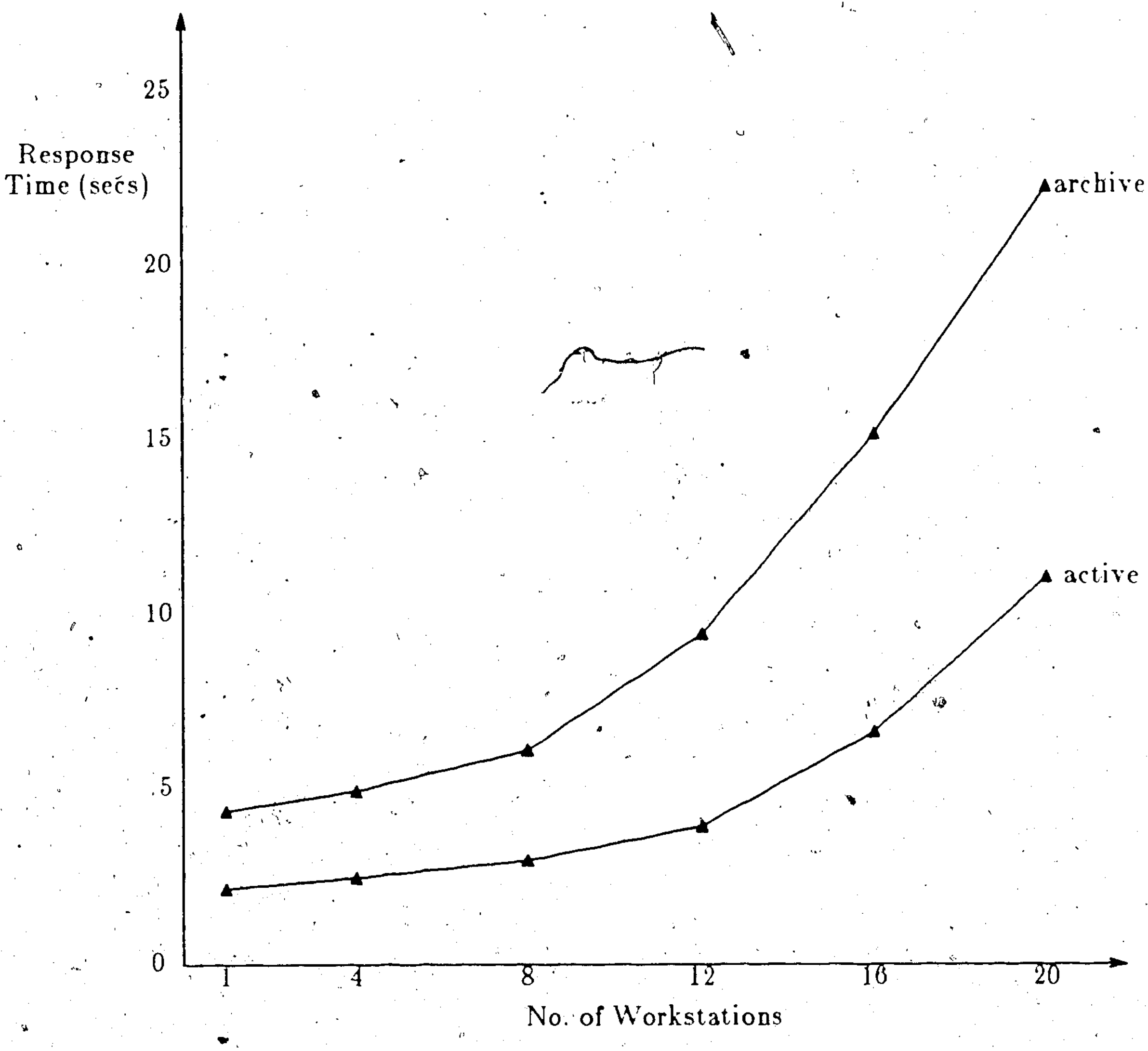
Figure 5.1 Channel Utilization vs. No. of Workstations

Note that the channel utilizations obtained are in fact higher if the average end-of-session idle periods, when there is no user activity at the workstation, are excluded from the calculations. Then this gives a value of 0.8735 for the average utilization and 0.9412 for the maximum utilization obtained for 20 workstations.

Shoch [39] observed a similar result of 0.95 for the value of the channel utilization for a measured Ethernet system when the channel was heavily loaded. Note however, that the traffic in Shoch's case was "bursty" with short packet lengths. The packet length distribution was bimodal (modes 32 and 550 bytes) with an average packet length of 122 bytes. The mean inter-packet arrival time was 39.5 microseconds with a standard deviation of 55 microseconds. This is typical of terminal communication interaction consisting of request/response transactions. For the system presented here, the packets are primarily 12,000 bits in length except for request packets which are 512 bits. With respect to inter-packet arrival time, a sequence of packets arrives at a device whenever there is an image to be transmitted, so that the inter-packet arrival time is zero for each sequence of packets.

The results obtained for the simulated PACS, also indicate that Ethernet under high load shows no instability since the channel utilization curve does not decline or become unstable as the number of workstations increases, i.e. as the total offered load increases.

Figure 5.2 shows the response time for the single-image display user activity as a function of the number of workstations. The graph shows display response times for images accessed from both active storage and archival storage. The access from archival storage is about double that from active storage. This [illegible] explained by the fact that referenced images which are not in active storage must first be brought from archive storage to active storage, where a copy of the image is kept, before the image is transmitted to the requesting workstation, thus creating a double path.



512^2 x 8 image, 1 Mbps capacity.

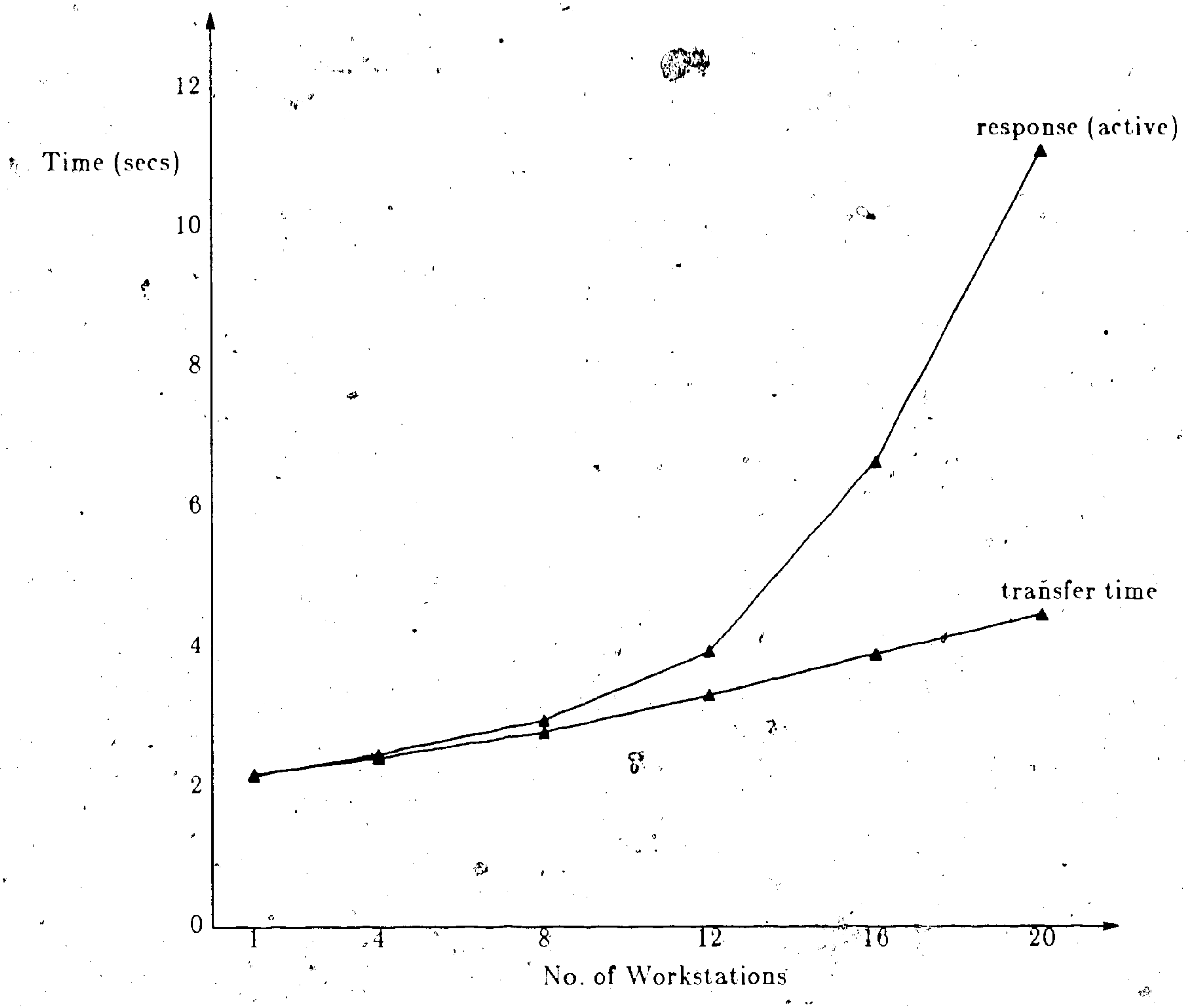
Figure 5.2 Image Reponse Time vs. No. of Workstations

The philosophy for having this data path is based on the assumption that the image which is currently being referenced from archival storage has a high probability of being referenced in the near future, hence, it will be found in active storage where it may be accessed quickly.

With 20 stations connected to the channel, the response time for accessing an image from active storage is about 11 seconds, which is just over the ten seconds stipulated in Chapter 1 as being acceptable. The response time from archival storage is 22 seconds, on the average. As more workstations are added to the channel the response time will rapidly increase as indicated by the graph. It is therefore concluded that twenty workstations is the potential maximum number which may be connected to the channel with a 1 Mbps bandwidth for the 512^2 x 8 image size and the specified workload. Note that even though the channel utilization increased linearly for 1 to 20 workstations, the image response time does not.

No actual workload was readily available, but the one selected for this system specifies a lower bound on the various times. The think-time per activity, the time required at the processor, and the interarrival time of images to the channel from the input devices were lower limits on realistic values. The access time of an image at a storage device was assumed to be infinitely small. This resulted in a heavily loaded channel. It may be conjectured that if the system with the same configuration is driven with an average case workload, then the response time will be reduced allowing additional workstations to be added to the system.

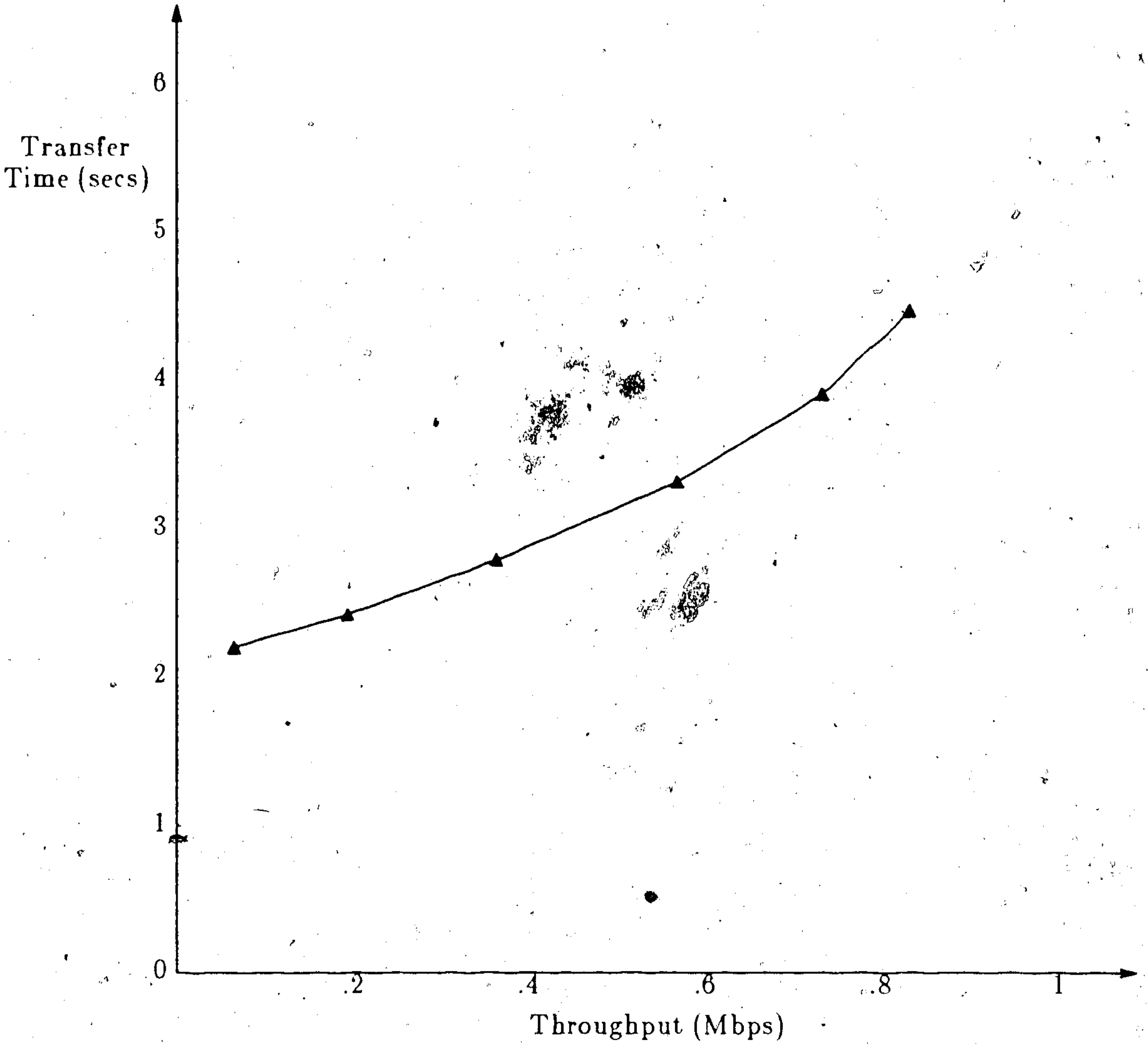
Figure 5.3 shows the response time for accessing an image from active storage as a function of the number of workstations. The image transfer time, a component of response time, is shown on the same graph. The other component of response time is the *wait time* which is the interval between the instant the image request is made until the first bit of the first packet of the requested image is transmitted. As the number of workstations increases, the image transfer time increases linearly with a slope of 0.15.



512^2 x 8 image, 1 Mbps capacity.

Figure 5.3 Image Transfer Time vs. No. of Workstations.

The rapid increase in response time, as the number of workstations increases, is due to the increased wait time of the request packet, as well as the image.



512^2 x 8 image, 1 Mbps capacity.

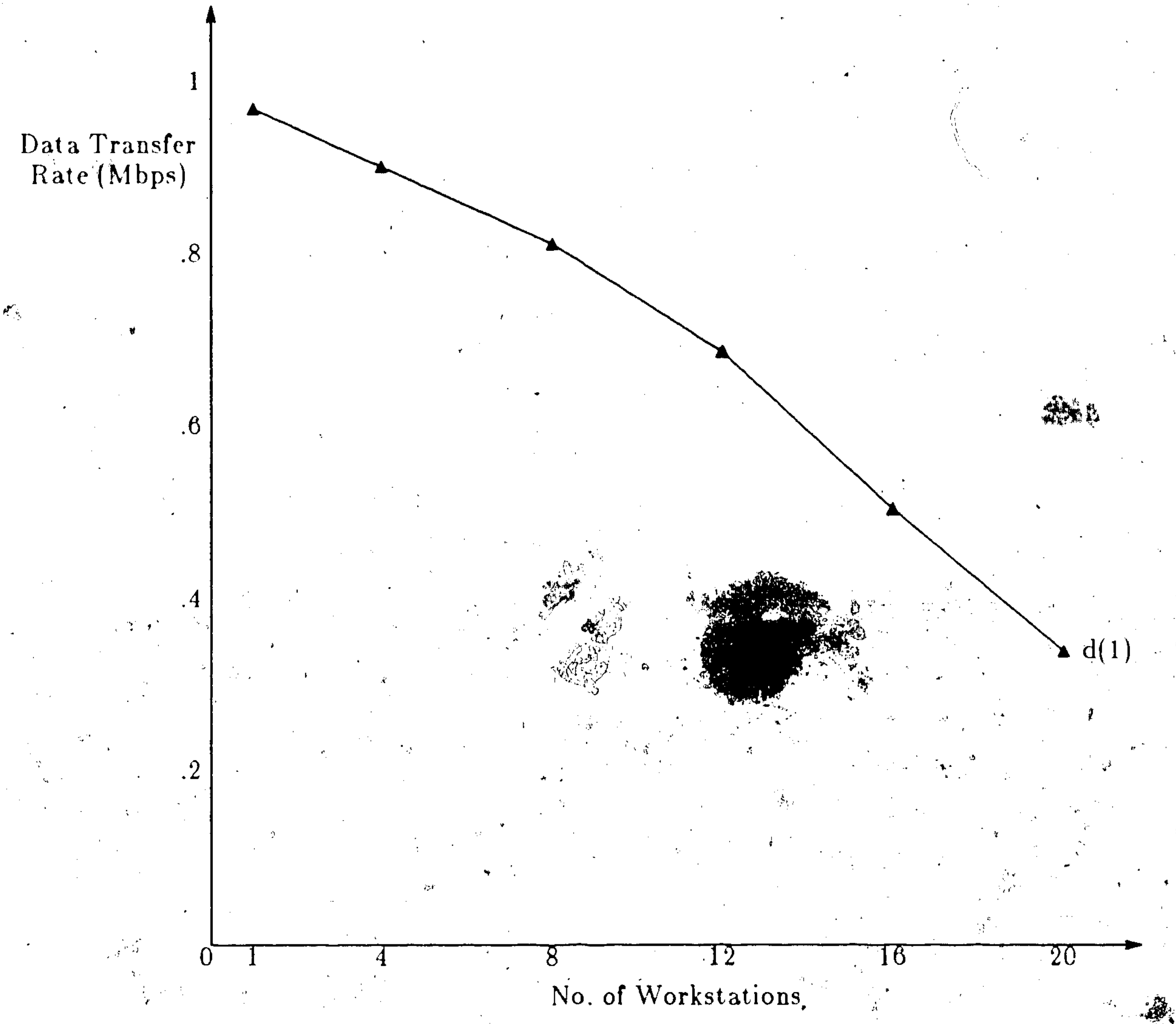
Figure 5.4 Image Transfer Time vs. Channel Throughput

This increase in response time is also manifested in the increased queue length of images at the devices which are waiting for the use of the channel.

Figure 5.4 shows the Image transfer time as a function of the throughput. The image transfer time was calculated only for full images transmitted on the channel, the partial images returned for the browse activity are not included. The transfer time has an almost linear increase with a slope of about 0.3, however, above a throughput of 0.6 Mbps the image transfer increases much faster.

Figure 5.5 shows the data transfer rate for the single image display activity as a function of the number of workstations. The graph shows that the data transfer rate decreases as the number of workstations increases, i.e. as the total load offered to the channel increases. The response time has increased due to packet collisions and increased image wait times, thus, effectively decreasing the data transfer rate. With 1 and 20 workstations respectively connected to the 1 Mbps channel the data transfer rate has been reduced from 0.9044 Mbps to 0.3423 Mbps for the single-image disply activity. Similar reductions in the data transfer rate for other activities were experienced.

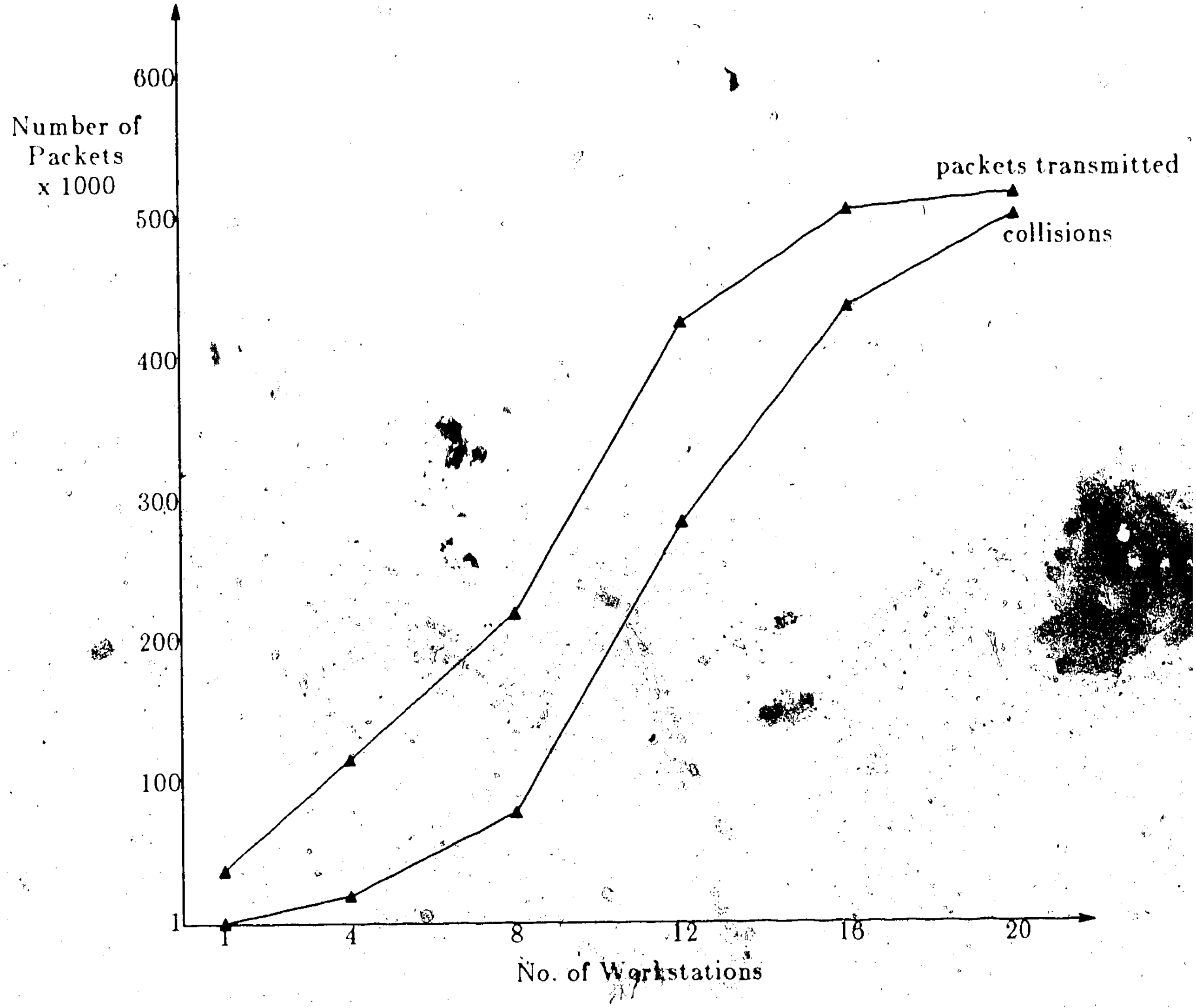
Figure 5.6 shows the number of packets transmitted on the channel as a function of the number of workstations. On the same graph the number of collisions is also plotted. This graph shows that the curve for the number of packets transmitted begins to flatten at about 16 workstations with an average of 447,000 packets successfully transmitted and 352,020 packets resulting in collision i.e. 0.7874 collisions per packet. Figure 5.7 shows the average number of collisions per packet transmitted as the number of workstations increases. An increase in the number of workstations to 20 did not result in a proportionate increase in the number of packets transmitted. The number of collisions did increase dramatically. With 20 workstations connected to the bus the average number of collisions was 0.9701 for every packet transmitted. In fact the maximum number of collisions occurring was 554,116 for 567,651 packets successfully transmitted, i.e., a rate of 0.9761 collisions per packet.



512^2 x 8 image, 1 Mbps capacity.

Figure 5.5 Data Transfer Rate vs. No. of Workstations

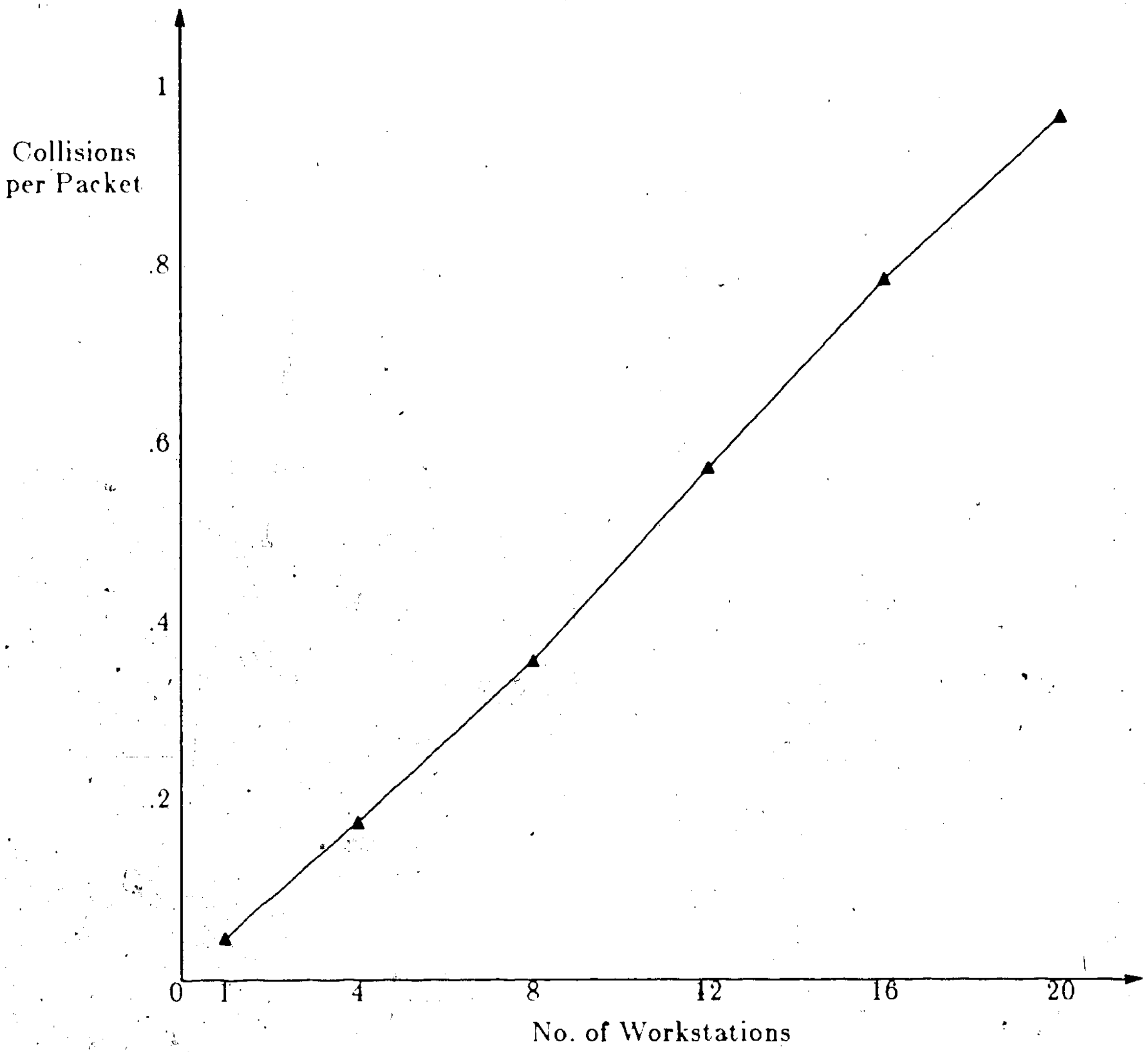
The number of collisions occurring in this system is high compared to a system carrying bursty traffic with short packet lengths.



512^2 x 8 image, 1 Mbps capacity.

Figure 5.6 No. of Collisions vs. No. of Workstations

With the maximum data field of 12,000 bits, a device will have a sequence of 175 packets for each 512^2 x 8 image transmitted. A 1024^2 x 8 image produces 700 packets,

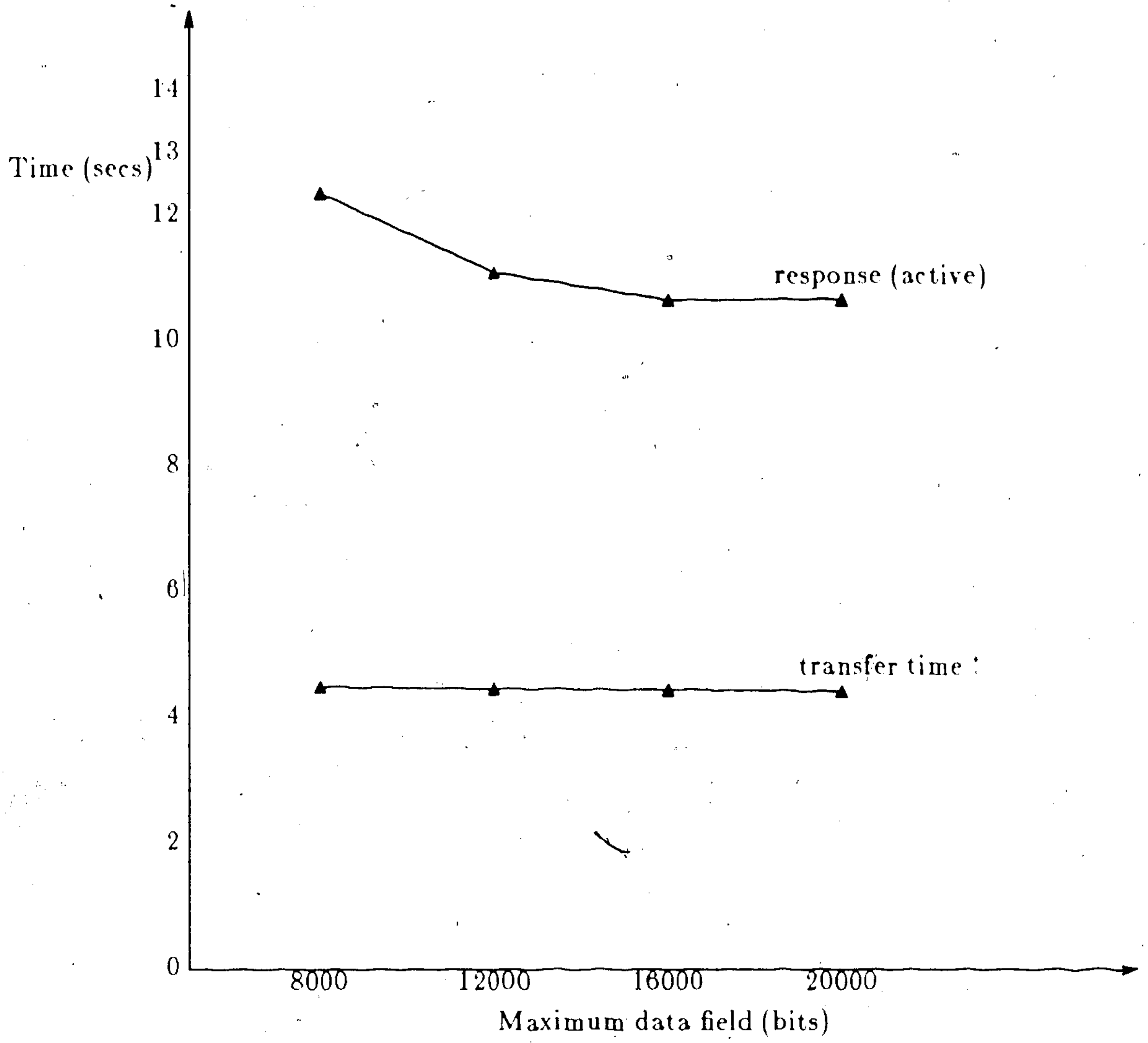


512^2 x 8 image, 1 Mbps capacity.

Figure 5.7 Collisions per Packet vs. No. of Workstations.

and a 2048^2 x 8 image requires 2,797 packets. If during the transmission of an image, a second device wishes to transmit, collisions will result for at least all the remaining packets to be transmitted for the first device. This is so, since, the first device initiates the transmission of the next packet in the sequence just after the previous packet is transmitted. This collision is largely responsible for the increase in response time since following each collision the colliding packets are rescheduled for some random time in the future.

Figure 5.8 shows the image response times and transfer times for different values of the maximum data field per packet. Maximum data fields of 8,000, 12,000, 16,000 and 20,000 bits were selected. The last two selections are outside the Ethernet specifications. The response time was greatest when the PACS was configured with 20 workstations. The results show that there is a 10 percent decrease in response time when the maximum data field was 16,000 bits instead of 12,000. With a further increase to 20,000 bits for the maximum data field, there was little decrease in response time. An equilibrium point seems to have been reached after which a larger maximum packet size will cause the response time to increase. The image transfer time has shown a small decrease with the bigger maximum data field. The same amount of user data is still transmitted per image except that it is transmitted with fewer packets. The number of collisions were reduced, but the time spent on collisions were very small compared to the time required to transmit a complete packet. The decrease in image response time is largely due to the decrease in wait time. The image queue length at the devices is reduced since a bigger packet size results in fewer packets per image. The maximum data field of 16,000 bits was favoured since this resulted in a reduced response time and fewer collisions were experienced.



512^2 x 8 image, 1 Mbps capacity, 20 workstations.

Figure 5.8 Image Response Time vs. Maximum Data Field

5.1.4. Storage Requirement

In this section, the active storage requirement is investigated. Images stored at active storage are produced from two sources, the external input, and from user activities. Hence, the total active storage required depends on the number of input devices and on the number of workstations.

With three input devices and image input rate as specified in Section 4.1, the expected number of external images is 72 per session or 360 per day. Appendix A1 gives, for each workstation as well as for all workstations, the average number of images stored in active storage per activity. For example, Table C for workstation number 2 shows that on average per session, the single-image display resulted in 0.49 images transferred to active storage from archival storage; multi-image display resulted in 4.74 images transferred to active storage from archival storage; edits required no additional active storage; manipulation resulted in 11.14 new images stored in active storage; browses resulted in 4.64 images stored in active storage from archival storage. Hence, for each workstation an average of 21.01 images per session or 105.05 images per day are stored in active storage. The daily storage requirement of S_{act} Megabytes of active storage is therefore given by the formula:

$$S_{act} = I(360 + 106w),$$

where I is the image size in Megabytes and w the number of workstations. This equation gives a conservative estimate of the actual storage needed since the number of activities per workstation decreases slightly with an increasing number of workstations. For example, Table 5.2 shows that, for 1 workstation, 11.07 manipulations were performed whereas for 20 workstations there were 9.13 manipulations per workstation. This occurs because the response time and hence the total activity time increases as the number of workstations increases.

Using this formula, a PACS with four workstations and 512^2 x 8 images would require

784 images a day which corresponds to 205 Megabytes of storage space. The results in Table D of Appendix A1 show an average of 157.4 images were stored in active storage per session or 787 images per day which corresponds to 207 Megabytes a day. This compares quite favourably with the calculated value of 205 Megabytes.

New storage space is needed for the arriving "external" images as well as for 56 images per day produced from the *manipulate* activity; *displays*, *edits* or *browses* do not produce new images. The daily growth in new storage space, S_{new}, of this system is given by the formula:

$$S_{new} = I(360 + 56w),$$

where I and w are defined as before. Four workstations and 512^2 x 8 images gives a daily new storage requirement of 564 images or 148 Megabytes. The value obtained from the simulation (Appendix A1 Table D) shows a memory requirement of 151 Megabytes for new images which compares reasonably well with the calculated value.

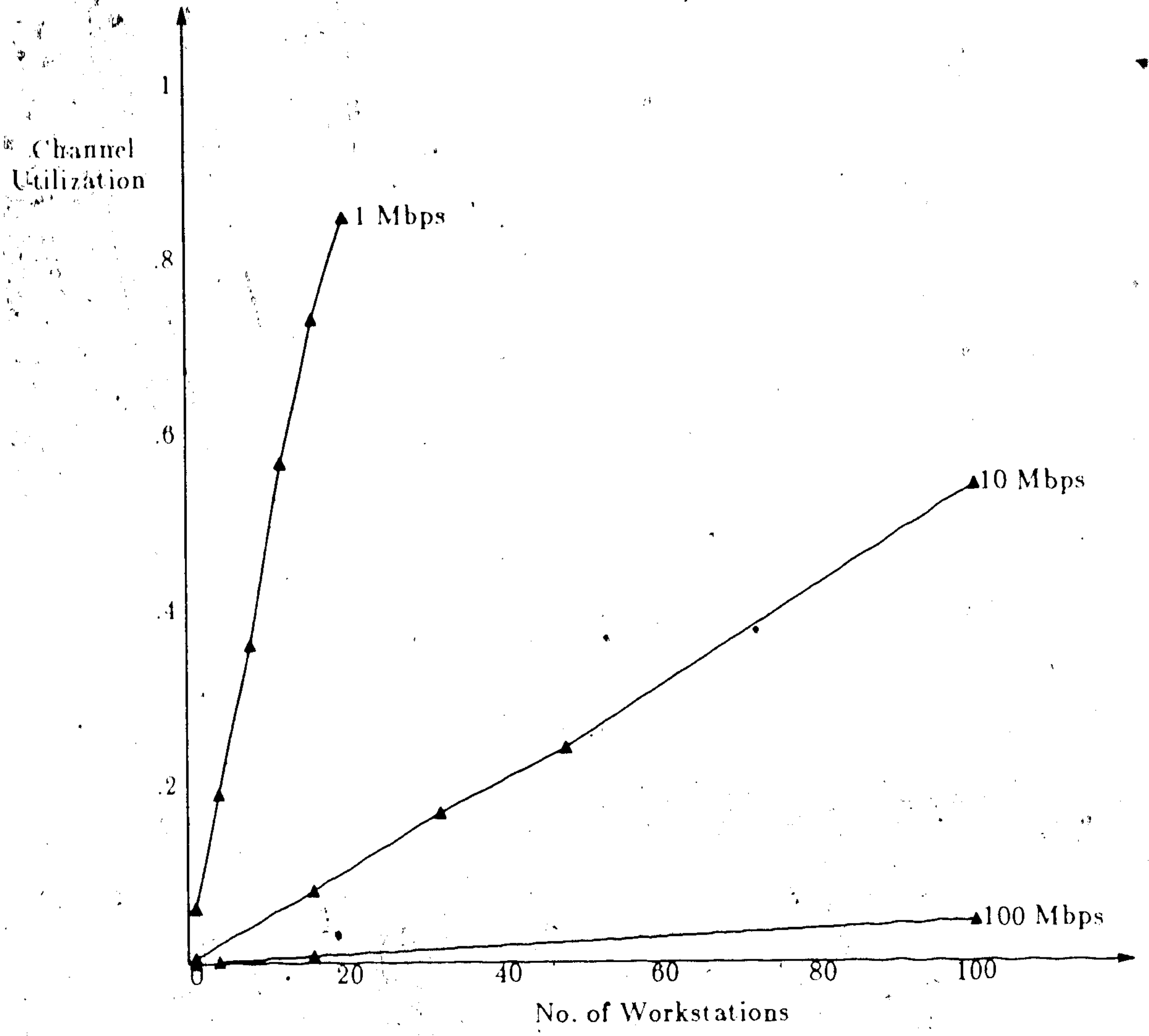
5.1.5. Fairness

The Ethernet control algorithm has shown remarkable fairness in allocating the channel to individual stations. Table C of Appendix A1 shows that the average number of activities completed per session is approximately equal for each of the four workstations. This was also true for other simulation runs. For example, the average number of edit activities completed per session for each of the four stations are 15.14, 16.20, 16.18 and 16.89 respectively (Appendix A1).

5.1.6. Higher Channel Capacities

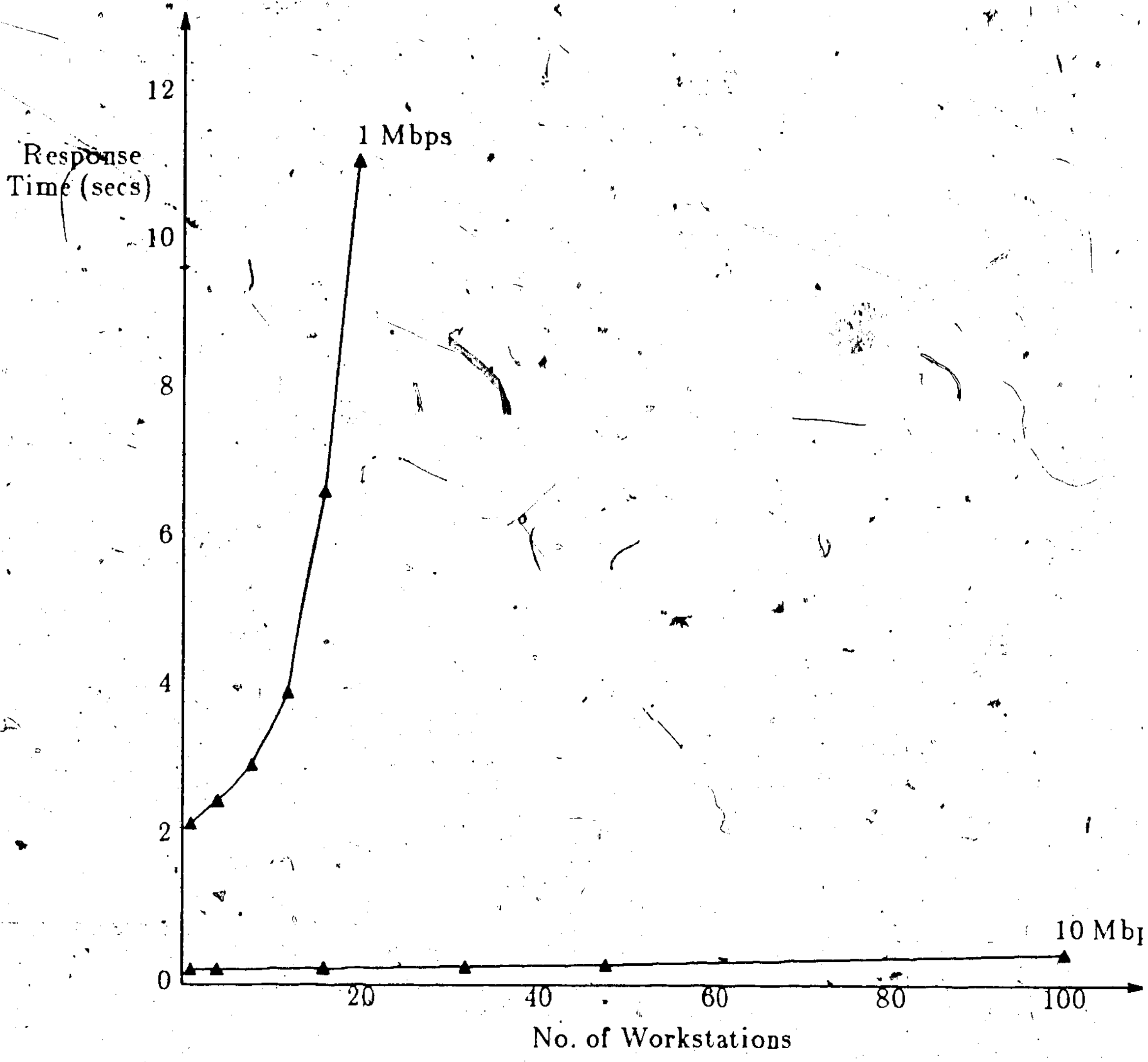
The results presented in Figures 5.1 and 5.2 suggest that for more than 20 workstations or an image size higher than 512^2 x 8, PACS would require a channel capacity greater than 1 Mbps.

Simulation runs were partially repeated for PACS with the 1 Mbps capacity channel respectively replaced by a 10 and 100 Mbps channels. Figure 5.9 shows the channel utilization plotted as a function of the number of workstations for 1 to 100 workstations and for channel capacities 1, 10, and 100 Mbps, while Figure 5.10 shows the corresponding image response times for images accessed from active storage for 1 and 10 Mbps capacity channels. At 100 workstations the average channel utilization for the 10 Mbps capacity channel is about 0.5 with an average image response time of 0.4285 seconds from active storage and an average image transfer time of 0.3100 seconds. The channel utilization for the 100 Mbps capacity channel was 0.5971 seconds for a PACS with 100 workstations. The image response time for the 100 Mbps capacity channel, not shown on the graph, is 0.0223 seconds and is approximate constant for 1 to 100 workstations. [illegible]s also been established in Figure 5.1 that the channel utilization increases linearly as the number of workstations increases. Using this linear relationship, a conservative projection of the graphs in Figure 5.9 estimates that about 170 workstations may be connected to the 10 Mbps channel before a utilization of 0.9 is achieved. However, Standards given in [19] stipulate that a maximum number of 100 interfaces should be connected to a single 500m Ethernet segment.



512² x 8 image; 1, 10, and 100 Mbps capacity.

Figure 5.9 Channel Utilization vs. No. of Workstations



512^2 x 8 image, 1 and 10 Mbps capacity.

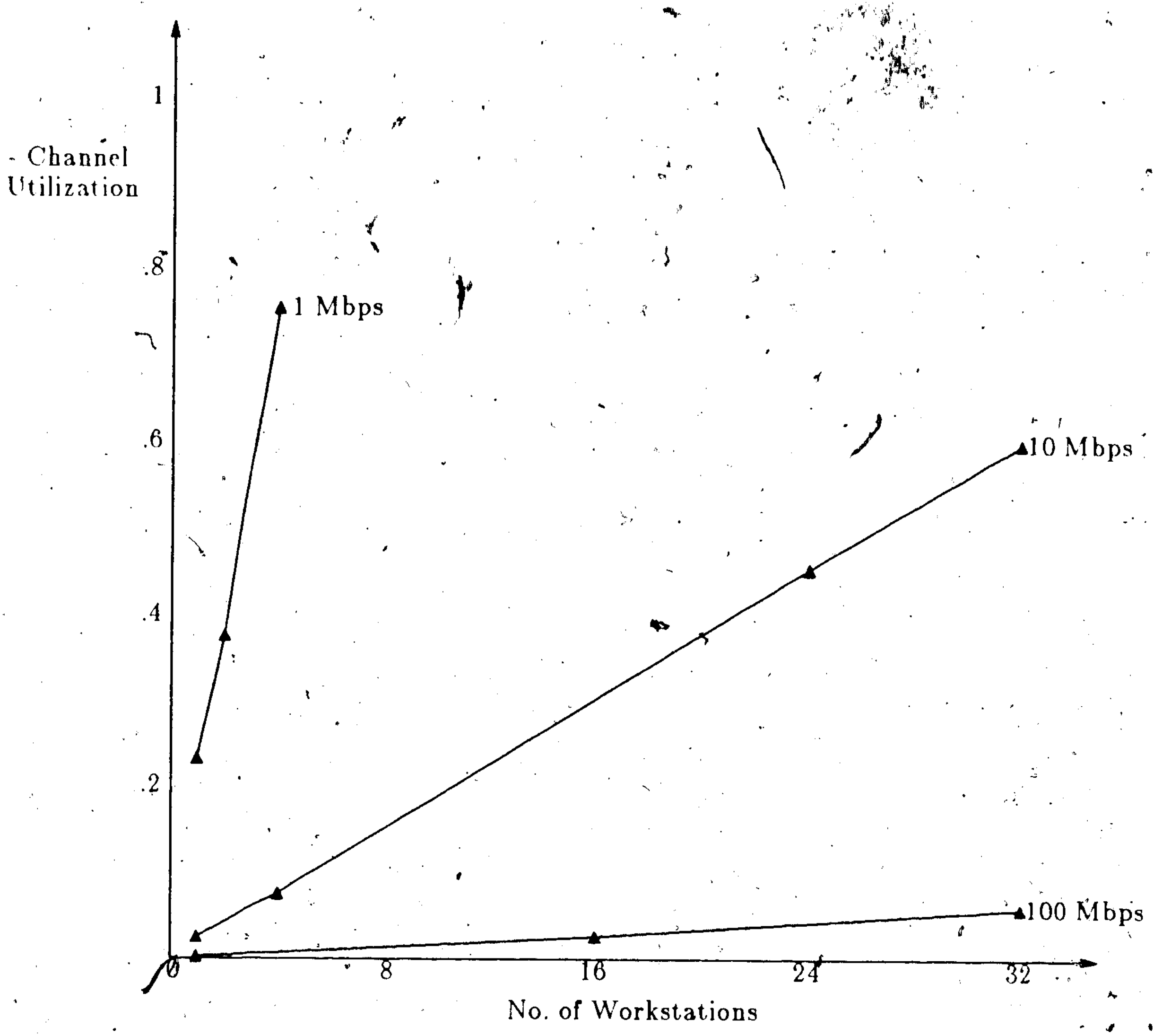
Figure 5.10 Image Response Time vs. No. of Workstations

5.2. Experiment 2

In this experiment the performance of a PACS which utilizes $1024^2 \times 8$ images is investigated for a varying number of workstations and different channel capacities. The size of a $1024^2 \times 8$ image is four times that of a $512^2 \times 8$ image. A maximum of 20 workstations with the $512^2 \times 8$ image was established for a PACS with a 1 Mbps channel. It is therefore expected that at most 5 workstations may be connected to a 1 Mbps PACS which processes $1024^2 \times 8$ images and produces acceptable response times. Figure 5.11 shows that an average channel utilization of 0.75 is obtained for a PACS with 4 workstations where as Figure 5.12 shows an average image response time form active storage is 11.53 seconds. Such a system severely limits the number of users, hence, for a practical system higher capacities are needed for larger size images.

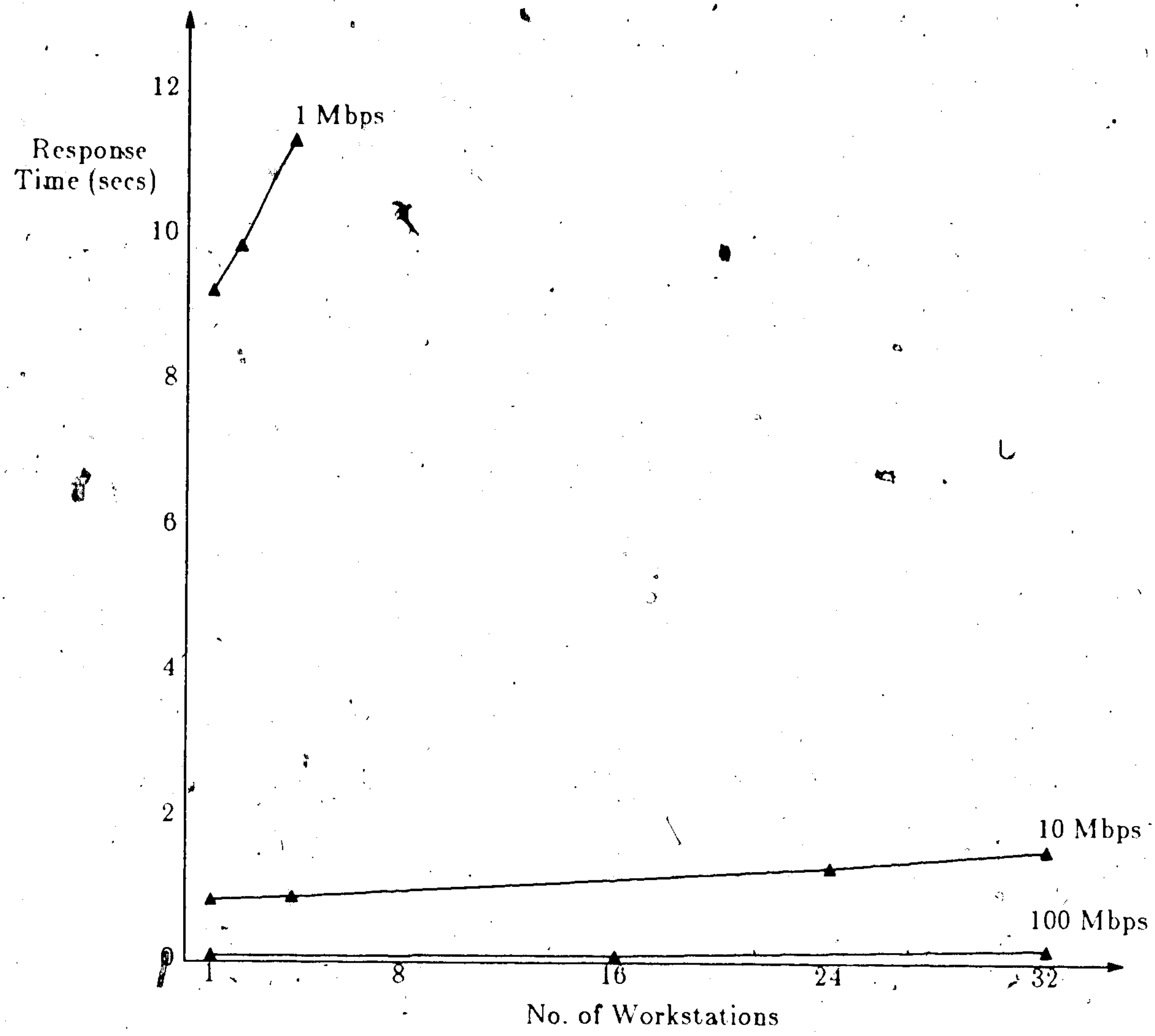
A number of simulations were run for PACS with a 10 Mbps channel which use $1024^2 \times 8$ images. At 32 workstations an average channel utilization of 0.6013 was obtained and the response time for images accessed from active storage was 1.5659 seconds while from archive storage was 3.2391 seconds. Using the linear relationship established for channel utilization, it is predicted that a PACS with about 50 workstation may be configured with the 10 Mbps capacity channel and $1024^2 \times 8$ images before the utilization reaches 0.9. It is difficult to predict the image response time in this case since its increase, as the number of workstations increases, is not linear. A configuration which requires more than 50 workstations and uses image size higher than $1024^2 \times 8$ would require a channel capacity higher than 10 Mbps.

With a 100 Mbps capacity channel and 32 workstations, the $1024^2 \times 8$ images gave a response time from active storage of 0.1892 seconds while channel utilization was 0.5971.



1024^2 x 8 image; 1, 10 and 100 Mbps capacities.

Figure 5.11 Channel Utilization vs. No. of Workstations



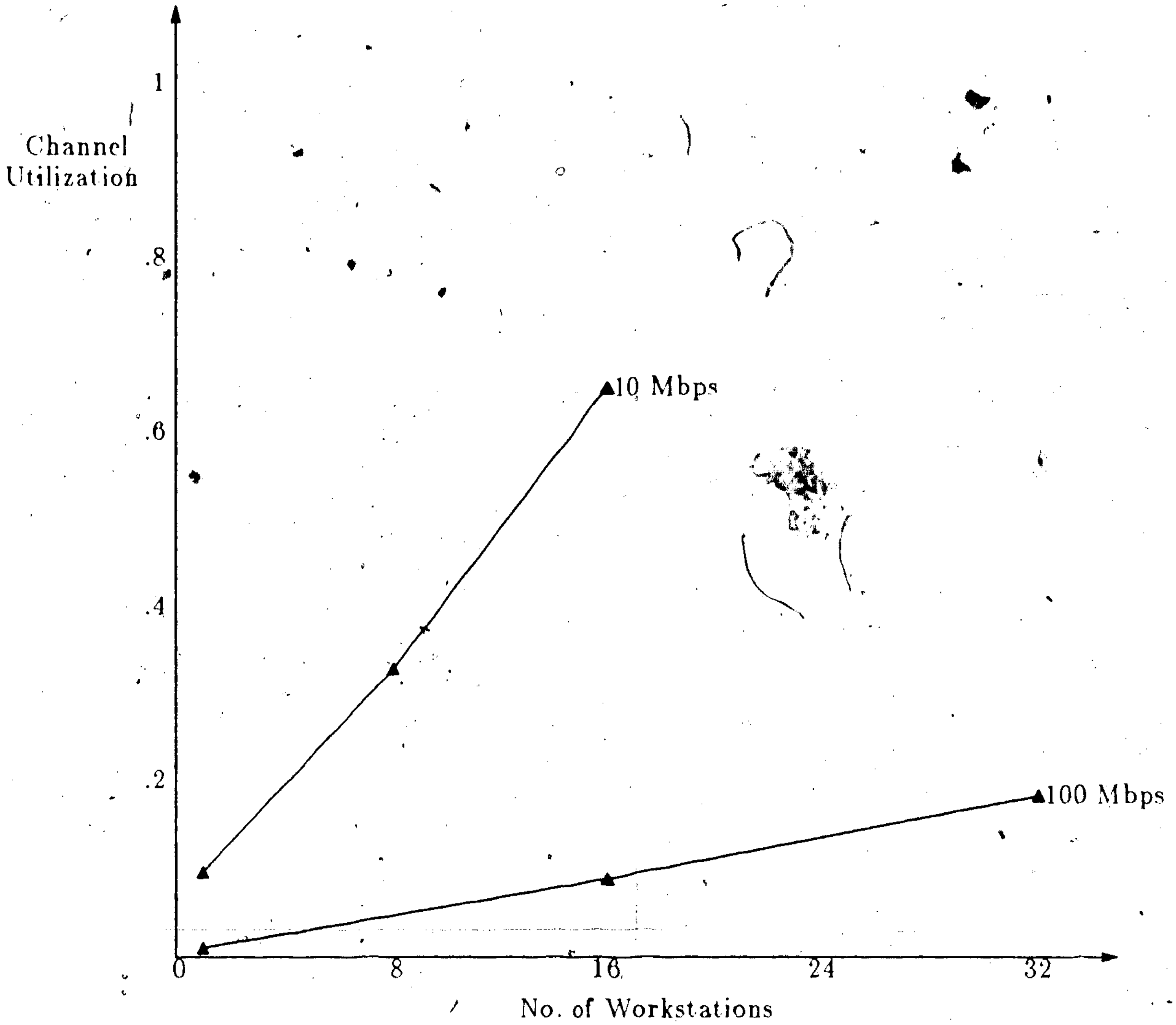
1024^2 x 8 image; 1, 10, 100 capacities.

Figure 5.12 Image Response Time vs. No. of Workstations

5.3. Experiment 3

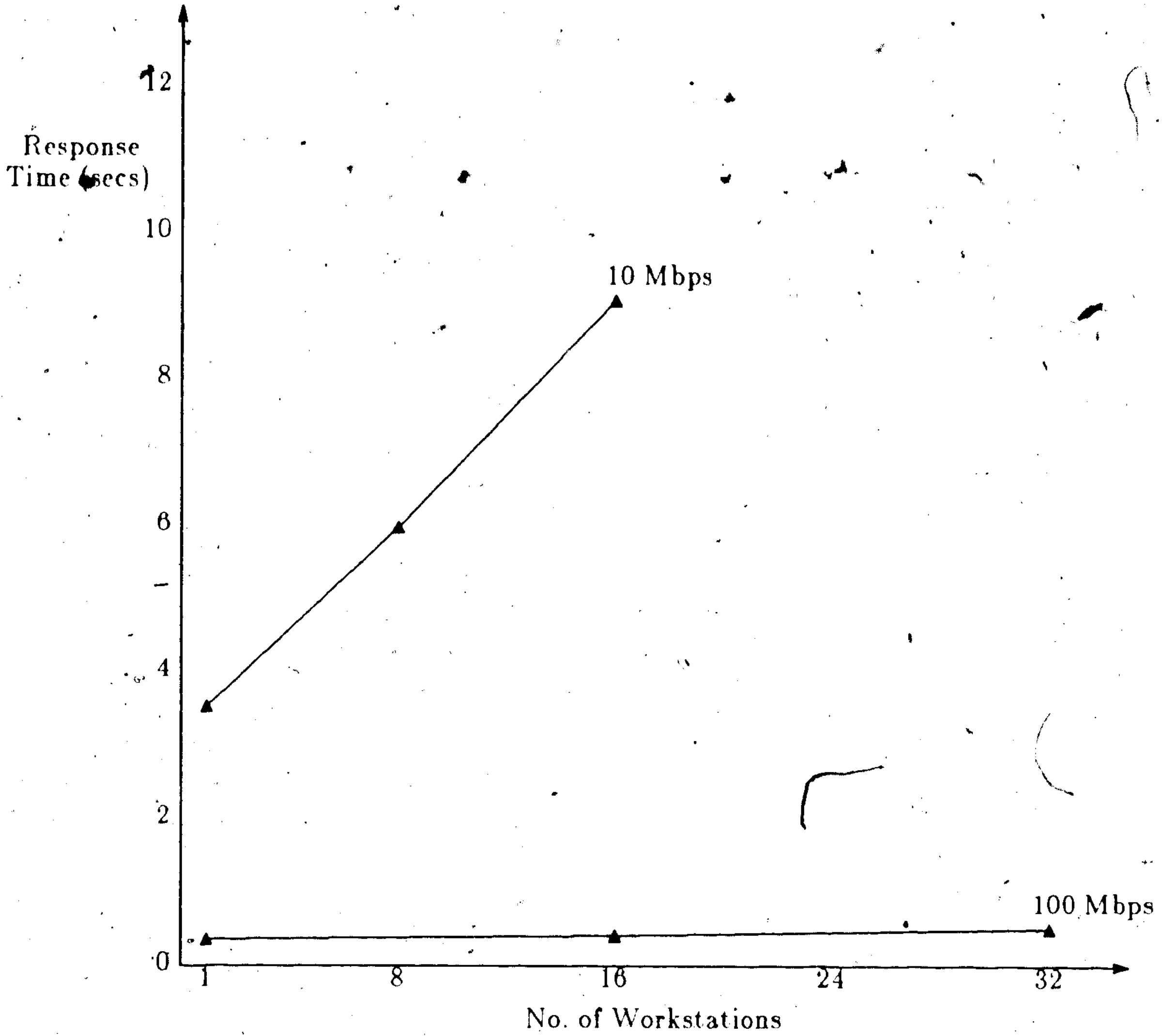
In this experiment a few simulations were run for PACS with 2048^2 x 8 image and for 10 Mbps and 100 Mbps capacity channels. The 1 Mbps channel was ruled out since the ideal transmission time for this size image as shown in Table 4.1, is 32.5544 seconds which is unacceptable to the interactive user.

Figure 5.13 shows the channel utilization as a function of the number of workstations while Figure 5.14 shows the corresponding image response time for images accessed from active storage. At 16 workstations, the utilization for the 10 Mbps capacity channel was 0.6537 while while the image response time was 9.0741 seconds. A projection of the graph in Figure 5.13 for the 10 Mbps capacity channel suggests that at most 24 workstations may be attached before a channel utilization of 0.9 is obtained. At 32 workstations the utilization of the 100 Mbps capacity channel was 0.1854 and the image response time was 0.5149 seconds.



2048^2 images; 10 and 100 Mbps capacities.

Figure 5.13 Channel Utilization vs. No. of Workstations



2048^2 x 8 image; 10, 100 Mbps capacities.

Figure 5.14 Image Response Time vs. No. of Workstations

Chapter 6

Summary, Conclusion and Future Research

6.1. Overview

A PACS differs from a conventional local area network in that the amount of data which is to be transmitted in an image is several orders of magnitude greater. In contrast, an I/O record size of tens to thousands of bytes is typical to most computer network applications. In addition, there is the requirement for fast response times in radiology applications. This suggests architectures which are different from those popular for distributed data processing and office automation.

The Ethernet protocol requires an image to be packetized into a sequence of packets, thus, the traffic pattern contrasts to the "bursty" traffic pattern characteristic of terminal interaction in a conventional local area networks. With a maximum data field of 12,000 bits as specified by the Ethernet protocol, the standard image size of 512^2 x 8 must be fragmented into a sequence 175 of packets. Similarly a 1024^2 x 8 image produces a sequence of 700 packets and a 2048^2 x 8 size image produces a sequence of 2,797 packets per image. It is this long sequence of packets to be transmitted which distinguishes the PACS application from other LAN applications.

6.2. Conclusion

In the experiments, simulation is used to investigate the performance of Ethernet as the underlying communication architecture of a PACS used for the management of digital formated images.

Under a high medical imaging workload, packets suffered a very high collision rate, however, the collision detection and resolution mechanism worked quite well. High channel utilization was maintained approaching the 0.98 theoretical limit.

The PACS remained stable as the total offered load were increased. The increase in channel utilization was linear with no signs of sudden decrease or becoming instable.

The Ethernet protocol was shown to exhibit good fairness to all devices of the PACS. The image response time and image transfer time for the respective PACS were approximately the same for all devices. Also the amount of work done at each workstation by the users on the PACS was about the same.

A key observation is that Ethernet at 1 Mbps provided acceptable response times for the given workload when fewer than 20 workstations processing 512^2 x 8 images were connected to the PACS. With more than 20 stations the response time rapidly increased, even though the channel utilization increases linearly, and almost every packet transmitted suffered a collision. This suggests that channel capacities higher than 1 Mbps are needed for a PACS using Ethernet connecting more than 20 workstations. The PACS with the same capacity supported at most 5 workstations with a 1024^2 x 8 image size.

With a 10 Mbps channel the potential maximum number of workstations supporting 512^2 x 8 images was projected to extend to 160; for 1024^2 x 8 images 50 workstations were suggested and for 2048^2 x 8 images it was suggested that about 24 workstations were the potential maximum.

A channel capacity of 100 Mbps provided excellent response times for all images considered with up the 100 imaging devices connected to the channel. The 100 Mbps does not currently exist but indicates that Ethernet will produce a viable imaging system should this capacity become available.

6.3. Future Research

Although Ethernet performance results look promising for the given system parameters and workload, other LAN topologies and medium access protocols need to be explored, in particular with the view towards real-time medical activities where digital voice transmission and colour images may be essential in the future.

The performance of the simulated PACS was explored with a single 500m segment. A interesting extension would be to analyze an Ethernet system of several segments connected by repeaters and also several Ethernets connected by bridges. Bridges support independent and simultaneous communication on individual Ethernet system and well as communication between Ethernet systems.

The simulated system explored in this thesis was designed primarily for image management. It assumes that there exist other data management systems which may be run in parallel. The image management system must be combined with other management systems such as a Hospital Information System and other administrative systems to run as a unified system.

Bibliography

[1] R. Allan, "Local Networks: Fiber Optics Gains Momemtum", *Electronic Design*, Vol. 31, No. 13, June 23 1983, pp. 97-106.

[2] R. A. Bauman and G. S. Lodwick, "The Future of Digital Computers in Medical Imaging", *Pictorial Information Systems in Medicine*, Vol. NATO-ASI, Sep. 1984, .

[3] B. S. Baxter and M. P. Zelennik, "Communication and Storage Protocols for PACS", *IEEE Computer*, Vol. 16, No. 8, Aug. 1983, pp. 31-36.

[4] G. J. Blaine, R. L. Hill, J. R. Cox and R. G. Jost, "PACS workbench at Mallinckrodt Institute of Radiology (MIR)", *SPIE Proc. on Picture Archiving and Communication Systems (PACS II) for Medical Applications*, Vol. 418, May 1983, pp. 80-86.

[5] G. S. Blair and D. Shepard, "A Performance of Ethernet and the Cambridge Digital Communication Ring", *Computer Networks*, Vol. 6, No. 2, July 1982, pp. 105-113.

[6] G. S. Blair, "A performance Study of the Cambridge Ring", *Computer Networks*, Vol. 6, No. 2, July 1982, pp. 13-20.

[7] D. E. Bulatek and S. B. O'Connell, "Working PACS prototype", *SPIE Proc. on Picture Archiving and Communication System (PACS II) for Medical Applications*, Vol. 418, May 1983, pp. 104-111.

[8] W. E. Burr, "An overveiw of the American National standards for local distributed data interfaces", *Communications of ACM*, Vol. 26, No. 8, Aug. 1983, pp. 553-561.

[9] W. Bux, "Performance Issues in Local-Area Networks", *IBM System Journal*, Vol. 23, No. 4, June 1984, pp. 351-373.

[10] D. N. Chorafas, *Designing and Implementating Local Area Networks*, MCGRAW, 1984.

[11] J. K. Cywinski, L. M. Cywinski and L. Lee, "Medical Image Distribution, Storage, and Retrieval Network: the M/NET", *SPIE Proc. on Picture Archiving and Communication Systems (PACS II) for Medical Applications*, Vol. 418, May 1983, pp. 74-79.

[12] DATAPRC "IEEE 802: A Summary of the Proposed Standards for Local Area Networking", *Datapro Research Corporation*, Oct. 1983.

[13] A. J. Duerinckx, S. J. Dwyer III and J. M. Prewitt, "Digital Picture Archiving and Communication Systems in Medicine", *IEEE Computer*, Vol. 16, No. 8, Aug. 1983, pp. 14-16.

[14] A. J. Duerinckx, "Digital Picture Archiving and Communication Systems in Diagnostic Radiology: A Review", *IEEE Computer*, Vol. 15, No. 12, Dec. 1982, pp. 477-478.

[15] A. J. Duerinckx, "PACS Status Review: 1983", *SPIE Proc. on Picture Archiving and Communication System (PAC II) for Medical Applications*, Vol. 418, May 22-25 1983, pp. 68-73.

[16] W. A. Erdman, T. J. Stahl, R. J. Tokarz, G. Q. Maquire Jr. and M E Noz, "Development of a Digital Nuclear Medicine System", *SPIE Proc. on Picture Archiving and Communication Systems (PACS II) for Medical Applications*, Vol. 418, May 1983, pp. 100-102.

[17] G. Fisher, *Principles of Discrete Event Simulation*, John Wiley & Sons, 1978.

[18] P. S. Gill and U. M. Maydell, "Simulation of an Ethernet-Based PACS", Technical Report, University of Alberta, Aug. 1985.

[19] M. Graube, " Local Area Nets: a Pair of Standards", *IEEE Spectrum*, June 1982, pp. 60-61

[20] B. E. Henderson and May 1983, "Prototype for an Electronic Document Storage and Retrieval Program", *SPIE Proc. on Picture Archiving and Communication Systems (PACS II) for Medical Applications*, Vol. 418, pp. 112-115.

[21] S. H. Hughes, "Picture Archiving and Communications systems (PACS): Introductory Systems Analysis Considerations", *SPIE Proc. on Picture Archiving and Communication Systems (PACS II) for Medical Applications*, Vol. 418, May 1983, pp. 122-126.

[22] J. R. Cox, Jr., G. J. Blaine, T. L. Hill, R. G. Jost and C. D. Shum, "Some Design Considerations for Picture Archiving and Communication Systems", *IEEE Computer*, Vol. 16, No. 8, Aug. 1983, pp. 39-48.

[23] F. Kalivoda, D. R. Spiegelhoff, J. A. Steinman, J. A. Steinman Jr. and D. L. Yuille, "Clinical Experience with an Operating Prototype Picture Archiving and Communication System (PACS)", *SPIE Proc. on Picture Archiving and Communication Systems (PACS II) for Medical Applications*, Vol. 418, May 1983, pp. 92-99.

[24] H. Keller and H. R. Mueller, "IBM Zurich - Local Area Network (LAN) Prototype", *IBM Zurich Research Highlights*, 1983.

[25] H. J. Keller, H. Meyer and H. R. Mueller, "Transmission Design Criteria for a Synchronous Token Ring", *IEEE Journal on Selected Areas in Communications*, Vol. SAC-1, No. 5, Nov. 1983, pp. 373-382.

[26] D. K. Knuth, *The Art of Computer Programming*, Addison-Wesley Pub. Co., 1981.

[27] J. Lehr, "Impact of Manual and Computer-assisted PACS on Automated PACS", *SPIE Proc. on Picture Archiving and Communication Systems (PACS II) for Medical Applications*, Vol. 418, May 1983, pp. 6-13.

[28] K. Maxwell, "High-Speed Dial-Up Modems", *BYTE*, Vol. 9, No. 13, Dec. 1984, pp. 179-182.

[29] U. M. Maydell and W. A. Davis, "Simulation of an Interactive Workstation for Image Display", *Proc. of 3rd International Conference on Picture Archiving and communication Systems (PAC III) for Medical Applications*, Feb. 3-8 1985, pp. 110-116.

[30] B. Meister, P. Janson and L. Svobodva, "File Transfer in Local-area Networks: A Performance Study", *IBM Zurich Research Laboratory*, May 1985, pp. 13-17.

[31] R. Metcafe and D. Boggs, "Ethernet: Distributed Packet Switching for Local Computer Network", *Communications of ACM*, Vol. 19, No. 7, July 1976, pp. 395-404.

[32] D. Meyer-Ebrecht and T. Wendler, "An Architectural Route Through PACS", *IEEE Computer*, Vol. 16, No. 8, Aug. 1983, pp. 19-28.

[33] W. J. Neilson and U. M. Maydell, "A Survey of Current LAN Technology [illegible] Performance", Technical Report, Jan. 1985.

[34] S. Orr, "Special Report: LAN Chips", *Electronic Design*, Vol. 31, No. 13, June 1983, pp. 81-90.

[35] J. R. Perry, B. G. Thompson, E. V. Staab, S. M. Pizer and R. E. Johnston, "Performance Features for a PACS Display Console", *IEEE Computer,* Vol. 16, No. 8, Aug. 1983, pp. 51-56.

[36] L. A. Ramshaw and P. D. Amer, "Generating Artificial Traffic Over Local Area Network using Random Number Generators", *Computer Networks,* Vol. 7, Aug. 1983, pp. 233-251.

[37] J. H. Saltzer, K. T. Pogran and D. D. Clark, "Why a Ring?", *Computer Networks,* Vol. 7, Aug. 1983, pp. 223-231.

[38] R. V. Schmidt, E. G. Rawson, R. N. Norton, S. B. Jackson and M. D. Bailey, "Fibernet II: A fiber optic Ethernet", *IEEE Journal on Selected Areas in Communications,* Vol. SAC-1, No. 5, Nov. 1983, .

[39] J. Shoch and J. Hupp, "Measured Performance of an Ethernet Local Network", *Communications of ACM,* Vol. 23, No. 12, Dec. 1980, pp. 711-721.

[40] C. D. Shum, J. R. Cox and G. J. Blaine, " Design Analysis of a Wide-band Picture Communication System", *Dept of Computing Science, Washington Univ.,* July 1984.

[41] W. Stallings, "Local Network Perforance", *IEEE Communiations Magazine,* Vol. 22, No. 2, Feb. 1984, pp. 27-36.

[42] W. Stallings, "Local Networks", *Computing Survey,* Vol. 16, No. 1, Mar. 1984, pp. 2-41.

[43] B. W. Stuck, "Calculating the Maximum Mean Data Rate in Local Area Networks", *IEEE Computer,* Vol. 16, No. 5, Jan. 1983, pp. 72-76.

[44] A. Tanenbaum, *Computer Networks,* Prentice Hall, Englewood Cliffs, NJ,

[45] F. A. Tobagi, "Multiaccess Protocols in Packet Communication Systems", *IEEE Transactions on Communications,* Vol. Com-28, No. 4, Apr. 1980, pp. 468-488.

[46] F. A. Tobagi and B. V. Hunt, "Performace Analysis of Carrier Sense Multiple Access with Collision Detection", *Computer Network,* Vol. 4, Jan. 1980, pp. 245-249.

[47] XEROX, "*The Ethernet: A Local Network Data Link Layer and Physical Layer Physical Layer Specifications. Version 1.0*", Xerox Digital Intel, Sep. 1980.

[48] J. S. Zielonka and R. S. Hellman, "Distributed Architecture for Image Aquisition, Analysis and Archiving: Application and Local PACS", *SPIE Proc. on Picture Archiving and Communication System (PACS II) for Medical Applications,* Vol. 418, May 1983, pp. 135-137.

Appendix A1

Sample Output for a Simulation Run

Appendix A1 gives the sample output of the performance measures for a simulation run. The system includes four workstations connected to a 1 Mbps capacity channel and the standard image format of 512^2 x 8.

```
simulation no.  1          I = 512   C =  1  W =  4   D =  20

The instalation parameters of the given PACS
--------------------------------------------

 1. PACS system parameters
 -------------------------

   observation period [days]          20
   transmission speed [Mbps]           1
   number of image input devices       3
   number of user display devices      4
   vertical image size [pixels]      512
   horizontal image size [pixels]    512
   number of intensity levels          8
   active storage cap. [MBytes]     1600

 2. PACS workload characteristics
 --------------------------------

   image average interarrival time [secs]

     input device    1            300.00
     input device    2            300.00
     input device    3            300.00

   user activity distribution
```

wkstat	% display_1	% display_n	% edit	% manip.	% browse
all	10.00	30.00	30.00	20.00	10.00

```
   average think time for activity [secs]
```

wkstat	display_1	display_n	edit	manip.	browse
all	30.00	30.00	60.00	120.00	60.00

```
   average time for activity [secs]
```

wkstat	edit	manipul
all	60.00	5.00

```
                                      simulation no.  1        I =  512   C =   1   W =   4   D =  20

values of the system output parameters
-----------------------------------------

  stats for image acquisition device no  1

    no. of images acquired/session     23.9600
    no. of images stored/session       23.9600

     performance measure         minimum     average     maximum    std. dev.

     image interarrival time     10.0269    300.3032   1781 8769    283.6771
     response storage time        2.1240      2.5764      8.0050      0.8759
     queue length at input        0.0         0.0         0.0         0.0


  stats for image acquisition device no  2

    no. of images acquired/session     24.0000
    no. of images stored/session       24.0000

     performance measure         minimum     average     maximum    std. dev.

     image interarrival time     10.1766    299.9378   1788.5454    282.1139
     response storage time        2.1240      2.5554      7.7780      0.8484
     queue length at input        0.0         0.0         0.0         0.0


  stats for image acquisition device no  3

    no. of images acquired/session     23.6200
    no. of images stored/session       23.6200

     performance measure         minimum     average     maximum    std. dev.

     image interarrival time     10.1197    304.7914   1690 6489    294.1919
     response storage time        2.1240      2.5788      8.3686      0.8728
     queue length at input        0.0         0.0         0.0         0 0
```

table A - stats for workstation no 1 /session simulation no. 1 I = 512 C = 1 W = 4 D = 20

performance measure	ave (ave)	sdv (ave)	ave (min)	sdv (min)	ave (max)	sdv (max)
display(1) d.t.r. [Mbps] (act)	0.9011	0.0742	0.6813	0.2179	0.9863	0.0077
display(1) response time (act)	2.4751	0.3882	2.1264	0.0179	3.5087	1.6273
display(1) think time (act)	30.4851	13.4252	9.0494	7.1666	62.7369	27.2641
display(1) function time (act)	32.9603	13.3855	11.3786	7.1463	65.4622	27.0799
display(1) d.t.r. [Mbps] (arc)	0.4559	0.0723	0.4488	0.0803	0.4629	0.0713
display(1) response time (arc)	4.7931	1.1498	4.6991	1.1398	4.8872	1.2457
display(1) think time (arc)	29.9867	21.9159	26.6782	22.6231	33.2953	24.1944
display(1) function time (arc)	34.7799	22.0687	31.5041	22.7763	38.0556	24.2974
no. images in display(n)	3.0140	0.1775	2.0000	0.0	3.9800	0.1407
display(n) d.t.r. [Mbps] (act)	0.9108	0.0273	0.4187	0.1035	0.9871	0.0
display(n) response time (act)	2.4323	0.1269	2.1246	0.0	5.4158	1.7744
display(n) d.t.r. [Mbps] (arc)	0.4305	0.0511	0.3257	0.1084	0.4877	0.0342
display(n) response time (arc)	5.2975	1.0646	4.3372	0.5364	7.4539	3.5605
display(n) think time [secs]	29.8115	3.7620	2.6852	0.7479	103.7611	12.8682
display(n) function time [sec]	97.5280	13.4883	29.5877	11.6525	194.7680	34.7712
image editing time [secs]	48.8925	6.2623	12.6289	2.7219	105.2544	12.0209
edit think time [secs]	86.9906	13.3793	33.3245	3.2747	207.1843	45.5627
edit function time [secs]	135.8832	15.2886	56.8356	9.2988	262.5477	45.8482
image manip. time [secs]	4.9522	0.6396	2.3463	0.3387	8.7079	0.9788
mnaipulation d.t.r. [Mbps]	0.2248	0.0164	0.1497	0.0190	0.3072	0.0238
manip. response time [secs]	9.9216	0.7934	6.8702	0.5810	14.2456	1.9542
manip. think time [secs]	178.8329	29.4053	72.6099	11.3734	389.7599	100.7469
manip. function time [secs]	188.7544	29.4727	82.2576	12.2771	399.7235	100.7169
no. of low res. searches/image	5.5041	0.9153	3.0600	1.2212	7.8300	1.1896
no. of images inspected/browse	3.6939	1.4520	1.1700	0.4726	8.2600	4.0142
browse response time	16.2031	7.1138	4.6424	1.8463	38.1617	19.8614
browse think time [secs]	87.1481	21.7728	40.3125	10.8618	164.9071	55.6883
browse function time	103.3512	23.0117	51.0079	13.9823	184.2048	56.3073

table A - stats for workstation no 2 /session simulation no. 1 I = 512 C = 1 W = 4 D = 20

performance measure	ave (ave)	sdv (ave)	ave (min)	sdv (min)	ave (max)	sdv (max)
display(1) d.t.r. [Mbps] (act)	0.9043	0.0832	0.6863	0.2391	0.9860	0.0111
display(1) response time (act)	2.4767	0.4088	2.1273	0.0269	3.5110	1.4495
display(1) think time (act)	29.3922	12.7308	7.9057	8.0324	61.9718	26.7545
display(1) function time (act)	31.8688	12.8036	10.2702	8.0302	64.5594	26.9204
display(1) d.t.r. [Mbps] (arc)	0.4556	0.0686	0.4486	0.0797	0.4626	0.0663
display(1) response time (arc)	4.7784	1.0755	4.6740	1.0302	4.8985	1.3246
display(1) think time (arc)	28.0294	23.8933	25.6936	23.1853	30.1566	25.1946
display(1) function time (arc)	32.8078	24.0113	30.3779	23.2053	34.9501	25.3467
no. images in display(n)	3.0004	0.1897	2.0200	0.1407	3.9700	0.1714
display(n) d.t.r. [Mbps] (act)	0.9170	0.0267	0.4356	0.1136	0.9871	0.0000
display(n) response time (act)	2.4047	0.1225	2.1246	0.0000	5.1811	1.5417
display(n) d.t.r. [Mbps] (arc)	0.4358	0.0549	0.3406	0.1051	0.4849	0.0332
display(n) response time (arc)	5.1568	1.0838	4.3520	0.4020	6.9349	2.9041
display(n) think time [secs]	30.0421	4.0783	2.6308	0.5724	101.4625	14.3429
display(n) function time [sec]	97.4943	13.3149	30.5304	11.6177	198.5565	40.0948
image editing time [secs]	48.1682	6.9233	12.8617	2.8473	104.0856	12.3821
edit think time [secs]	86.9381	12.8568	33.9191	3.2776	200.8254	48.8532
edit function time [secs]	135.1063	14.4714	58.6103	11.4717	257.9258	49.5778
image manip. time [secs]	4.9345	0.6343	2.3319	0.2977	8.6912	0.9493
mnaipulation d.t.r. [Mbps]	0.2226	0.0178	0.1490	0.0172	0.3101	0.0228
manip. response time [secs]	10.0096	0.8404	6.8058	0.5808	14.2616	1.6681
manip. think time [secs]	176.1700	32.2435	71.6005	12.1388	395.4701	108.7804
manip. function time [secs]	186.1796	32.2415	81.6253	12.1237	405.4854	108.4919
no. of low res. searches/image	5.3945	1.0145	3.0500	1.1135	7.6900	1.3610
no. of images inspected/browse	4.1054	1.9674	1.3900	0.7900	8.2600	4.9044
browse response time	17.1919	9.7091	5.3581	4.0892	37.8215	23.1087
browse think time [secs]	86.4833	25.5208	44.1479	16.3018	148.9256	53.2647
browse function time	103.6752	28.8816	55.7665	21.1792	168.2059	56.8532

table A - stats for workstation no 3 /session simulation no. 1 I = 512 C = 1 W = 4 D = 20

performance measure	ave (ave)	sdv (ave)	ave (min)	sdv (min)	ave (max)	sdv (max)
display(1) d.t.r. [Mbps] (act)	0.8988	0.0745	0.6383	0.2272	0.9851	0.0201
display(1) response time (act)	2.5255	0.4558	2.1300	0.0544	3.8647	1.9252
display(1) think time (act)	27.9187	9.6619	8.0313	5.7889	60.2788	25.5864
display(1) function time (act)	30.4443	9.6699	10.5305	6.1206	62.8186	25.7330
display(1) d.t.r. [Mbps] (arc)	0.4389	0.0778	0.4331	0.0829	0.4446	0.0781
display(1) response time (arc)	4.9895	1.1518	4.9114	1.1501	5.0681	1.2322
display(1) think time (arc)	28.8271	23.2429	25.0208	22.9732	33.3187	27.2640
display(1) function time (arc)	33.8166	23.4048	29.9647	23.1964	38.3531	27.4042
no. images in display(n)	3.0189	0.1800	2.0200	0.1407	3.9500	0.2190
display(n) d.t.r. [Mbps] (act)	0.9127	0.0256	0.4087	0.0997	0.9871	0.0
display(n) response time (act)	2.4254	0.1239	2.1246	0.0	5.5320	1.7507
display(n) d.t.r. [Mbps] (arc)	0.4428	0.0449	0.3513	0.0993	0.4884	0.0263
display(n) response time (arc)	4.9853	0.7629	4.3111	0.3262	6.5652	2.3715
display(n) think time [secs]	29.9545	3.8050	2.6109	0.6588	102.9496	11.9375
display(n) function time [sec]	98.4432	13.4159	30.9496	11.2317	198.5521	36.3606
image editing time [secs]	49.2004	7.2620	12.9690	3.5536	105.4898	11.7211
edit think time [secs]	85.6049	11.5441	33.7836	3.3582	209.6742	47.5819
edit function time [secs]	134.8053	13.2339	58.3514	11.3897	267.5273	49.8753
image manip. time [secs]	4.9188	0.7384	2.3766	0.4136	8.5896	1.1771
mnaipulation d.t.r. [Mbps]	0.2213	0.0194	0.1480	0.0189	0.3042	0.0276
manip. response time [secs]	10.0749	0.8974	6.9585	0.7253	14.3984	1.8176
manip. think time [secs]	169.3212	31.3774	71.1109	11.0569	369.6508	111.5584
manip. function time [secs]	179.3961	31.3333	80.8950	11.5874	379.6496	111.4677
no. of low res. searches/image	5.4666	1.0157	3.2323	1.3080	7.6465	1.3576
no. of images inspected/browse	4.0009	1.7187	1.5051	1.2320	8.0707	4.0964
browse response time	16.5756	7.8552	5.3503	4.5968	37.7830	21.6827
browse think time [secs]	91.0098	28.3753	43.1906	20.3087	164.7176	56.6800
browse function time	107.5855	29.6063	54.0076	21.4402	186.7124	59.4149

table A - stats for workstation no 4 /session simulation no 1 I = 512 C = 1 W = 4 D = 20

performance measure	ave (ave)	sdv (ave)	ave (min)	sdv (min)	ave (max)	sdv (max)
display(1) d.t.r. [Mbps] (act)	0.9104	0.0743	0.6932	0.2219	0.9871	0.0000
display(1) response time (act)	2.4467	0.4048	2.1246	0.0001	3.3818	1.1980
display(1) think time (act)	31.2337	11.2252	8.1302	6.6657	66.2982	25.5726
display(1) function time (act)	33.6804	11.2787	10.6603	6.7125	68.7460	25.6413
display(1) d.t.r. [Mbps] (arc)	0.4426	0.0800	0.4341	0.0915	0.4518	0.0775
display(1) response time (arc)	5.0005	1.3656	4.8512	1.3112	5.1461	1.5889
display(1) think time (arc)	27.1931	23.0557	24.6711	23.0063	29.7236	24.5902
display(1) function time (arc)	32.1936	22.8452	29.6887	22.7829	34.7033	24.4101
no. images in display(n)	3.0001	0.1640	2.0000	0.0	3.9900	0.1000
display(n) d.t.r. [Mbps] (act)	0.9124	0.0283	0.4004	0.1186	0.9871	0.0
display(n) response time (act)	2.4403	0.1398	2.1246	0.0	5.8195	2.1616
display(n) d.t.r. [Mbps] (arc)	0.4394	0.0476	0.3341	0.1050	0.4878	0.0284
display(n) response time (arc)	5.1128	1.0451	4.3190	0.3516	7.0929	3.0332
display(n) think time [secs]	29.6512	2.9046	2.6042	0.6185	103.2558	12.3478
display(n) function time [sec]	96.8862	10.8341	28.0187	11.9255	195.1572	31.8573
image editing time [secs]	49.2083	8.0055	13.0428	2.8315	104.4394	12.3429
edit think time [secs]	86.4911	11.6011	33.3609	3.7871	208.4121	47.9513
edit function time [secs]	135.6994	14.8446	59.1083	9.9619	269.8098	46.8826
image manip. time [secs]	4.9952	0.7132	2.3194	0.3383	8.5854	1.1743
manipulation d.t.r. [Mbps]	0.2231	0.0162	0.1504	0.0173	0.3101	0.0210
manip. response time [secs]	9.9797	0.7834	6.7972	0.5084	14.1230	1.6290
manip. think time [secs]	175.5941	36.8366	72.4310	14.7308	377.8009	101.5441
manip. function time [secs]	185.5738	36.8846	82.3360	15.3289	388.2983	101.3923
no. of low res. searches/image	5.3723	0.8453	3.1100	1.1450	7.6700	1.0546
no. of images inspected/browse	3.9561	1.5010	1.2900	0.7148	8.6000	4.4721
browse response time	16.0963	8.1367	4.7064	2.0794	40.9198	27.0353
browse think time [secs]	91.2883	29.3707	42.7981	16.4054	166.7767	64.9999
browse function time	107.3847	29.8322	53.6796	20.5572	187.4271	65.0990

table A - stats for all workstations/session   simulation no 1   I = 512   C = 1   W = 4   D = 20

performance measure	ave (ave)	sdv (ave)	ave (min)	sdv (min)	ave (max)	sdv (max)
display(1) d.t.r. [Mbps] (act)	0.9036	0.0765	0.6748	0.2269	0.9861	0.0121
display(1) response time (act)	2.4810	0.4146	2.1270	0.0316	3.5665	1.5769
display(1) think time (act)	29.7574	11.8710	8.2791	6.9495	62.8214	26.2977
display(1) function time (act)	32.2384	11.8907	10.7099	7.0224	65.3965	26.3412
display(1) d.t.r. [Mbps] (arc)	0.4480	0.0749	0.4409	0.0836	0.4553	0.0735
display(1) response time (arc)	4.8951	1.1928	4.7875	1.1641	5.0051	1.3548
display(1) think time (arc)	28.5105	22.8195	25.5066	22.7520	31.6399	25.1404
display(1) function time (arc)	33.4056	22.8643	30.3834	22.7873	36.5349	25.1877
no. images in display(n)	3.0083	0.1776	2.0100	0.0996	3.9725	0.1637
display(n) d.t.r. [Mbps] (act)	0.9132	0.0270	0.4159	0.1095	0.9871	0.0000
display(n) response time (act)	2.4257	0.1287	2.1246	0.0000	5.4871	1.8286
display(n) d.t.r. [Mbps] (arc)	0.4371	0.0498	0.3379	0.1045	0.4872	0.0306
display(n) response time (arc)	5.1380	1.0004	4.3298	0.4107	7.0119	3.0030
display(n) think time [secs]	29.8648	3.6532	2.6328	0.6509	102.8572	12.8861
display(n) function time [sec]	97.5879	12.7758	29.7716	11.6206	196.7584	35.8046
image editing time [secs]	48.8674	7.1267	12.8756	2.9995	104.8173	12.0878
edit think time [secs]	86.5062	12.3368	33.5970	3.4279	206.5240	47.4463
edit function time [secs]	135.3735	14.4319	58.2264	10.5659	264.4526	48.1161
image manip. time [secs]	4.9502	0.6809	2.3435	0.3489	8.6435	1.0726
mnaipulation d.t.r. [Mbps]	0.2230	0.0175	0.1493	0.0181	0.3079	0.0240
manip. response time [secs]	9.9965	0.8286	6.8579	0.6052	14.2571	1.7679
manip. think time [secs]	174.9795	32.6445	71.9381	12.3776	383.1704	105.8425
manip. function time [secs]	184.9760	32.6565	81.7785	12.8772	393.2892	105.6958
no. of low res. searches/image	5.4343	0.9482	3.1128	1.1967	7.7093	1.2444
no. of images inspected/browse	3.9389	1.6727	1.3383	0.8526	8.2982	4.3743
browse response time	16.5166	8.2391	5.0135	3.3767	38.6737	23.0269
browse think time [secs]	88.9773	26.4108	42.6108	16.3085	161.3233	58.0628
browse function	105.4939	27.9353	53.6144	19.5301	181.6248	59.8151

table B - stats for all workstations/session simulation no 1 I = 512 C = 1 W = 4 D = 20

performance measure	mim (min)	max (min)	min (max)	max (max)
display(1) d.t.r. [Mbps] (act)	0.1570	0.9871	0.7858	0.9871
display(1) response time (act)	2.1246	2.6689	2.1246	13.3536
display(1) think time (act)	2.0200	52.3443	5.9772	119.4640
display(1) function time (act)	4.1446	54.4688	8.1017	125.7715
display(1) d.t.r. [Mbps] (arc)	0.1820	0.4936	0.1820	0.4936
display(1) response time (arc)	4.2491	11.5220	4.2491	11.5220
display(1) think time (arc)	2.0459	115.8665	2.3827	115.8665
display(1) function time (arc)	6.2950	120.6222	6.7928	120.6222
no. images in display(n)	2.0000	3.0000	3.0000	4.0000
display(n) d.t.r. [Mbps] (act)	0.1405	0.8796	0.9871	0.9871
display(n) response time (act)	2.1246	2.1246	2.3841	14.9264
display(n) d.t.r. [Mbps] (arc)	0.1820	0.4936	0.1820	0.4936
display(n) response time (arc)	4.2491	8.7059	4.2491	24.1568
display(n) think time [secs]	2.0006	5.6668	57.4884	119.8651
display(n) function time [sec]	9.0511	75.3346	116.3558	329.8333
image editing time [secs]	10.0005	31.3997	54.8364	119.9622
edit think time [secs]	30.0386	50.5688	105.3093	299.2569
edit function time [secs]	40.6232	108.2760	156.1636	398.9021
image manip. time [secs]	2.0018	4.0104	4.4240	9.9909
manipulation d.t.r. [Mbps]	0.0875	0.2222	0.2132	0.3356
manip. response time [secs]	6.2498	9.8370	9.4363	23.9696
manip. think time [secs]	60.0131	156.3472	131.0552	599.3608
manip. function time [secs]	66.7287	165.2411	140.8918	614.7289
no. of low res. searches/image	2.0000	8.0000	2.0000	9.0000
no. of images inspected/browse	1.0000	10.0000	1.0000	26.0000
browse response time	2.1268	44.1833	4.2590	186.7350
browse think time [secs]	30.0047	193.9258	32.4080	299.7335
browse function time	32.6231	200.3762	40.1430	360.0362

table C - stats for workstation no 1 /session simulation no. 1 I = 512 C = 1 W = 4 D = 20

performance measure	minimum	average	maximum	std. dev.
no. of displays(1) requested	2.0000	5.9100	12.0000	2.2297
no. of displays(1) completed	2.0000	5.8900	12.0000	2.2197
no. of displays(1) aborted	0.0	0.0200	1.0000	0.1407
no. of displays(1) (active)	2.0000	5.3300	11.0000	2.1930
no. of displays(1) (archive)	0.0	0.5800	2.0000	0.6989
no. of displays(n) requested	9.0000	16.1800	25.0000	3.3586
no. of displays(n) completed	8.0000	15.9200	25.0000	3.3565
no. of displays(n) incompl.	0.0	0.2600	1.0000	0.4408
no. of displays(n) (active)	25.0000	44.1800	75.0000	9.4167
no. of displays(n) (archive)	0.0	4.4900	9.0000	2.0425
no. of image edits requested	10.0000	16.6200	27.0000	3.5584
no. of image edits completed	9.0000	16.2800	26.0000	3.4731
no. of image edits incompl.	0.0	0.3400	1.0000	0.4761
no. of image manip. requested	5.0000	10.5700	17.0000	2.6676
no. of image manip. completed	5.0000	10.3000	17.0000	2.6150
no. of image manip. incompl.	0.0	0.2700	1.0000	0.4462
no. of image browse requested	1.0000	6.0800	16.0000	2.3428
no. of image browse completed	1.0000	5.9700	16.0000	2.3114
no. of image browses incompl.	0.0	0.1100	1.0000	0.3145
no. of browses from (active)	0.0	0.6700	3.0000	0.8294
no. of browses from (archive)	1.0000	5.3000	15.0000	2.1672
end of session idled time	287.5368	519.4191	598.6077	65.9954

table C - stats for workstation no 2 /session simulation no. 1 I = 512 C = 1 W = 4 D = 20

performance measure	minimum	average	maximum	std. dev.
no. of displays(1) requested	2.0000	5.7100	12.0000	2.2124
no. of displays(1) completed	2.0000	5.6900	12.0000	2.2279
no. of displays(1) aborted	0.0	0.0200	1.0000	0.1407
no. of displays(1) (active)	2.0000	5.2200	10.0000	2.0577
no. of displays(1) (archive)	0.0	0.4900	3.0000	0.7035
no. of displays(n) requested	9.0000	16.0600	29.0000	4.1312
no. of displays(n) completed	9.0000	15.8000	29.0000	3.9975
no. of displays(n) incompl.	0.0	0.2600	1.0000	0.4408
no. of displays(n) (active)	2[illegible].0000	43.4200	73.0000	11.7492
no. of displays(n) (archive)	0.0	4.7400	15.0000	2.4562
no. of image edits requested	9.0000	16.4000	26.0000	3.3212
no. of image edits completed	8.0000	16.0600	25.0000	3.2993
no. of image edits incompl.	0.0	0.3400	1.0000	0.4761
no. of image manip. requested	6.0000	11.4400	18.0000	2.5319
no. of image manip. completed	5.0000	11.1400	18.0000	2.5742
no. of image manip. incompl.	0.0	0.3000	1.0000	0.4606
no. of image browse requested	1.0000	5.2800	12.0000	2.2432
no. of image browse completed	1.0000	5.2000	12.0000	2.2019
no. of image browses incompl.	0.0	0.0800	1.0000	0.2727
no. of browses from (active)	0.0	0.5600	4.0000	0.7292
no. of browses from (archive)	1.0000	4.6400	11.0000	2.0914
end of session idled time	205.7790	514.5287	596.3114	73.0571

table C - stats for workstation no 3 /session - simulation no 1 I = 512 C = 1 W = 4 D = 20

performance measure	minimum	average	maximum	std. dev.
no. of displays(1) requested	2.0000	6.3100	12.0000	2.1867
no. of displays(1) completed	2.0000	6.2800	12.0000	2.1792
no. of displays(1) aborted	0.0	0.0300	1.0000	0.1714
no. of displays(1) (active)	1.0000	5.7700	12.0000	2.2195
no. of displays(1) (archive)	0.0	0.5400	3.0000	0.7166
no. of displays(n) requested	7.0000	16.6800	31.0000	4.6010
no. of displays(n) completed	7.0000	16.5000	30.0000	4.5870
no. of displays(n) incompl.	0.0	0.1800	1.0000	0.3861
no. of displays(n) (active)	19.0000	44.9700	82.0000	12.0080
no. of displays(n) (archive)	0.0	5.2200	14.0000	2.6384
no. of image edits requested	9.0000	16.5200	30.0000	3.5319
no. of image edits completed	8.0000	16.1100	29.0000	3.4928
no. of image edits incompl.	0.0	0.4100	1.0000	0.4943
no. of image manip. requested	4.0000	11.0600	16.0000	2.4363
no. of image manip. completed	4.0000	10.7600	16.0000	2.4746
no. of image manip. incompl.	0.0	0.3000	1.0000	0.4606
no. of image browse requested	0.0	5.4300	12.0000	2.4670
no. of image browse completed	0.0	5.3500	12.0000	2.4797
no. of image browses incompl.	0.0	0.0800	1.0000	0.2727
no. of browses from (active)	0.0	0.5400	3.0000	0.7577
no. of browses from (archive)	0.0	4.8100	11.0000	2.2460
end of session idled time	185.2737	513.8685	597.4406	73.4710

table C - stats for workstation no 4 /session simulation no. 1 I = 512 C = 1 W = 4 D = 20

performance measure	minimum	average	maximum	std. dev.
no. of displays(1) requested	2.0000	5.9600	13.0000	2.1878
no. of displays(1) completed	2.0000	5.9300	13.0000	2.1754
no. of displays(1) aborted	0.0	0.0300	1.0000	0.1714
no. of displays(1) (active)	2.0000	5.3600	12.0000	2.0327
no. of displays(1) (archive)	0.0	0.6000	3.0000	0.7107
no. of displays(n) requested	10.0000	16.3200	26.0000	3.3781
no. of displays(n) completed	9.0000	16.0800	25.0000	3.2588
no. of displays(n) incompl.	0.0	0.2400	1.0000	0.4292
no. of displays(n) (active)	27.0000	43.9900	76.0000	9.3803
no. of displays(n) (archive)	1.0000	4.8900	11.0000	2.1268
no. of image edits requested	8.0000	16.5000	27.0000	3.5746
no. of image edits completed	8.0000	16.1700	26.0000	3.4467
no. of image edits incompl.	0.0	0.3300	1.0000	0.4726
no. of image manip. requested	6.0000	10.8700	17.0000	2.7621
no. of image manip. completed	5.0000	10.5800	17.0000	2.7384
no. of image manip.. incompl.	0.0	0.2900	1.0000	0.4560
no. of image browse requested	2.0000	5.9100	13.0000	2.2790
no. of image browse completed	2.0000	5.8000	13.0000	2.2874
no. of image browses incompl.	0.0	0.1100	1.0000	0.3145
no. of browses from (active)	0.0	0.5700	3.0000	0.7143
no. of browses from (archive)	1.0000	5.2300	13.0000	2.1875
end of session idled time	296.9467	514.6224	599.2791	64.4676

table C - stats for all workstations/session simulation no. 1 I = 512 C = 1 W = 4 D = 20

performance measure	minimum	average	maximum	std. dev.
no. of displays(1) requested	13.0000	23.8900	35.0000	4.4538
no. of displays(1) completed	13.0000	23.7900	35.0000	4.4953
no. of displays(1) aborted	0.0	0.1000	1.0000	0.3015
no. of displays(1) (active)	11.0000	21.6800	33.0000	4.2850
no. of displays(1) (archive)	0.0	2.2100	6.0000	1.2894
no. of displays(n) requested	50.0000	65.2400	84.0000	6.9792
no. of displays(n) completed	49.0000	64.3000	84.0000	6.8865
no. of displays(n) incompl.	0.0	0.9400	4.0000	0.8741
no. of displays(n) (active)	128.0000	176.5600	233.0000	19.1287
no. of displays(n) (archive)	11.0000	19.3400	35.0000	4.4568
no. of image edits requested	50.0000	66.0400	80.0000	6.4148
no. of image edits completed	49.0000	64.6200	79.0000	6.2584
no. of image edits incompl.	0.0	1.4200	3.0000	0.7545
no. of image manip. requested	31.0000	43.9400	57.0000	4.8280
no. of image manip. completed	29.0000	42.7800	56.0000	4.8107
no. of image manip. incompl.	0.0	1.1600	4.0000	0.9070
no. of image browse requested	12.0000	22.7000	32.0000	4.5137
no. of image browse completed	12.0000	22.3200	32.0000	4.5011
no. of image browses incompl.	0.0	0.3800	2.0000	0.5993
no. of browses from (active)	0.0	2.3400	7.0000	1.4990
no. of browses from (archive)	11.0000	19.9800	31.0000	4.2331
end of session idled time	1693.8277	2062.4388	2343.0943	139.4072

table D - stats for bus, processor, storage/session   simulation no. 1   I = 512   C = 1   W = 4   D = 20

performance measure	minimum	average	maximum	std. dev.
no. of images transmitted	973.0000	1487.2800	2091.0000	267.5456
no. of full images transmitted	458.0000	516.0000	581.0000	25.9771
no. of packets transmitted	97786.0000	118454.6600	138401.0000	9022.0921
no. of packet collision	12891.0000	21109.2000	30494.0000	3986.0330
image input rate (images/sec)	0.1351	0.2066	0.2904	0.0372
total user data sent (Mb)	1155.8219	1398.4468	1627.3227	104.0375
image transfer time (secs)	2.1240	2.4234	8.5248	0.7042
percentage busy (successful)	16.2547	19.6676	22.8894	1.4642
percentage busy (collision)	0.0069	0.0115	0.0167	0.0021
percentage idled	77.0952	80.3209	83.7369	1.4656
incomp. images at channel	0.0	0.0200	1.0000	0.1407
no. of images manipulated	29.0000	42.7800	56.0000	4.8107
incompl. images at processor	0.0	0.0	0.0	0.0
incompl. images from processor	0.0	1.1600	4.0000	0.9070
no. of images stored at active	132.0000	157.4000	187.0000	10.8330
incompl. images at active str.	0.0	0.0	0.0	0.0
no. of images archived	0.0	5.2200	61.0000	13.3042
incompl. images at archives	0.0	0.0	0.0	0.0

total new images acquired is 11552
i.e a storage rate of 151.41 MB/day