PROCEEDINGS
SPIE—The International Society for Optical Engineering

High-Speed Fiber Networks and Channels

Kadiresan Annamalai
Chair/Editor

4–6 September 1991
Boston, Massachusetts

Sponsored and Published by
SPIE—The International Society for Optical Engineering

SPIE PROCEEDINGS SERIES

Volume 1577

SPIE (Society of Photo-Optical Instrumentation Engineers) is a nonprofit society dedicated to the advancement of optical and optoelectronic applied science and technology.

SESSION 1

Technology in Support of High-Speed Data Services

Chair

Andres Albanese

Bell Communications Research

#48

Multimedia Communications: Architectural Alternatives

Terrence P. McGarty, Ph.D.

MIT, Cambridge, MA
NYNEX, Orangeburg, NY

S.T. Treves, M.D.

Harvard Medical School, Boston, MA
Children's Hospital, Boston, MA

ABSTRACT

Multimedia Communications systems are a combination of human interfaces and end users interacting with multimedia data bases and highly disparate but interconnected communications networks. This paper discusses several architectural alternatives and system requirements that will assist in the design and development of MMCS in actual environments. The approaches taken in this paper are based upon the development of such systems in both medical and printing and publishing environments. This paper develops several key concepts as how best to define and structure data in a multimedia environment, how best to integrate the communications elements, and how best to permit the maximum flexibility to the end user to utilize the system's capabilities in the context of a fully-conversational environment.

1.0 Introduction

The development of both communications and computer systems have been driven by the need to increase the underlying performance of the technology. The pressure has been on producing more bits per second or more MIPS. Data base systems have also been directed towards a similar goal in that they are designed to store more data and to retrieve it faster. Only recently has there been increased interest in designing systems from the outside in rather than the inside out. That is, the old design methodology assumed that one designed from the technology out to the user, and that the user learned to adapt to the technology. UNIX (TM, AT&T) is an example of such a design. The new systems are more user driven, exemplified by the Apple MAC design.

In the last few years, the concept of multimedia systems has developed and is in its early evolutionary stages. Many authors have viewed multimedia as nothing more than a fancy display and several storage devices. In contrast, the seminal and brilliant work of Winograd and Flores has depicted the computer environment in a much broader context; one that recognizes the holistic interfaces necessary for the user, the designer, and the technology infrastructure. The authors develop a philosophy of design. It is this philosophy that we build upon in this paper with a view to developing an architecture.

2.0 Architectural Alternatives

To understand the overall concepts of multimedia communications, it is first necessary to place the total concept in the context of an architecture. We have developed such a construct as the basis of a philosophical approach to a system design. It has been argued persuasively in Winograd and Flores that having a philosophical set of underpinnings establishes a common set of beliefs that can be used for both building and expanding the body of knowledge as well as critiquing the concepts that evolve from it. All too often in the area of multimedia, there is the construct of the day in the absence of a base. The architecture developed in this paper is proposed as such a base.

An architecture can be defined as the conceptual embodiment of a world view, using a commonly-understood set of constructural elements, based upon the available set of technologies. To

0-8194-0708-9/92/$4.00

deconstruct this definition, we first indicate that our abilities to design and deliver functional systems are delimited by the set of operational technologies. These technologies may be hardware or software-related or a combination of the two.

The constructural elements of the architecture definition are the following:

(i) Human Interface: This is the end-user connection that allows the system to communicate directly with the human entities that are to be combined with the other architectural elements.

(ii) Interconnect: This element allows for the connection of all users, whether they be human, data bases, active data elements, or whatever, and support an underlying conversational element.

(iii) Data Bases: These are generalized storage devices for the retention, restructuring, and dissemination of the stored complex multimedia data structures. They may be embodied in the constructs of image storage devices or in voice storage devices.

(iv) Device Interface: These are the interfaces to non-human devices or users of the system. These elements are typically driven by the specific needs of the devices that are to be connected to the network.

(v) Transport: Transport is the element that allows for the data or other elements necessary for effective communications to move from one point to another.

(vi) Control: The control element is the element that allows for the underlying integrity of all of the other elements. It may be the network management control, the software processing control, the data element synchronization control, or an amalgam of all control infrastructures required to meet the anticipated level of network performance.

These six elements are integral in the design of any multimedia communications sytem. The interfaces (human and device) and the data bases represent the highest level of an architecture. They represent the visible elements that can be viewed as externalities of the system. In comparison, they may be viewed as the roof, walls, and floors of a building, all necessary elements, changeable by the technology and molded by the world view. The interconnect, transport, and control elements may be considered the wiring, heating, and foundation elements of the structure. They are not readily seen but are essential to the integrity of the architectural structure.

The architecture that we have developed for multimedia communications is a four-level structure. On the top level, we have called the elements the applications; namely, the externalities as viewed by the end user. These are the interfaces, both human and device, and the data base interfaces. The three applications of REVIEW, REPORT and CONSULTATION are described not only in terms of the applications software or the screen but in terms of the human interface externalities needed to ensure proper usage of the intended application.

The system services layer allows for the interconnect function. In this level, we place a set of clients that allow for the interaction of all elements at the top application layer. This architectural embodiment begs the question of centralized versus distributed. It clearly, in its very statement, indicates the need for a fully-distributed embodiment.

The network services layer is the control layer of the architectural elements. It is at this level that control over the integrity of all communications and data structures is maintained. It also is structured in a distributed fashion, recognizing furthermore that the system must function in a highly nonhomogeneous fashion.

The bottom layer is the transport layer. This layer also goes directly to the heart of the world view taken in the architecture. It is a world view that states that there will be multiple overlay networks, with capabilities for high-speed transport, but operating at different speeds and controlled by varying protocols. It also assumes, contrary to a hierarchical telephone world view, that the users will want to control their destiny, not a centralized telephone authority. Once liberated, the end users will want only more freedom, not less.

3.0 Human Interfaces

The importance of interfaces between human beings and these systems cannot be emphasized enough. Successful interfaces permit the use of computer systems to their fullest potential. Conversely, poor human interfaces can prevent the optimal utilization of a good computer system and can even completely discourage potential users to use such systems. Those users will find ways to avoid the system and work "around" it. Human interface designs must be viewed as solutions to existing problems, rather than as partial technical solutions looking for problems which they may solve. A successful interface is simple, intuitive, easy to learn and to remember, and it provides advantages over manual systems.

Putative advantages of good human interfaces include:

a) time and cost savings
b) improved data processing capability
c) completeness of transactions
d) improved access to same or even greater amounts of pertinent information, normally not easily available
e) improved quality of the end product
f) elimination of redundancies
g) improved human communications.

Depending on the user, many or most of these features are needed in order to make a human interface successful.

The design process of a system provides a good opportunity to re-evaluate established work habits and workflow, of which the human interface is an essential component. Human interface design should not necessarily aim at just replacing the older, manual system, but to improve on it. Human interface designs should be the result of a combined effort between ultimate users and designers of the interface. During the process of studying the design of a user interface, active interaction of an informed user and the architect will likely unveil newer approaches to methods and procedures that may be an improvement over older or purely manual procedures.

Development and introduction of new systems human interfaces must take into consideration present workflow and work habits of the local culture, must identify and anticipate perceived and real needs, and aim to avoid adding a new process into an already busy environment. The application must represent a replacement of current activities and routines and prove to be an improvement over old methods. (See McGarty and Sununu)

During the early process of human interface design, it is important to consider the most generic aspects of applications, the kind of people involved, and the environment in which these will exist so that these are not forgotten while designing the very specific user needs. Modular designs with a very broad range of options are much better than rigid designs. Special subsets of such modules can be then concatenated for specific users. It is important to consider the actual means of interaction such as keyboard, joystick, mouse, touch screen, voice, etc.

deconstruct this definition, we first indicate that our abilities to design and deliver functional systems are delimited by the set of operational technologies. These technologies may be hardware or software-related or a combination of the two.

The constructural elements of the architecture definition are the following:

(i) Human Interface: This is the end-user connection that allows the system to communicate directly with the human entities that are to be combined with the other architectural elements.

(ii) Interconnect: This element allows for the connection of all users, whether they be human, data bases, active data elements, or whatever, and support an underlying conversational element.

(iii) Data Bases: These are generalized storage devices for the retention, restructuring, and dissemination of the stored complex multimedia data structures. They may be embodied in the constructs of image storage devices or in voice storage devices.

(iv) Device Interface: These are the interfaces to non-human devices or users of the system. These elements are typically driven by the specific needs of the devices that are to be connected to the network.

(v) Transport: Transport is the element that allows for the data or other elements necessary for effective communications to move from one point to another.

(vi) Control: The control element is the element that allows for the underlying integrity of all of the other elements. It may be the network management control, the software processing control, the data element synchronization control, or an amalgam of all control infrastructures required to meet the anticipated level of network performance.

These six elements are integral in the design of any multimedia communications sytem. The interfaces (human and device) and the data bases represent the highest level of an architecture. They represent the visible elements that can be viewed as externalities of the system. In comparison, they may be viewed as the roof, walls, and floors of a building, all necessary elements, changeable by the technology and molded by the world view. The interconnect, transport, and control elements may be considered the wiring, heating, and foundation elements of the structure. They are not readily seen but are essential to the integrity of the architectural structure.

The architecture that we have developed for multimedia communications is a four-level structure. On the top level, we have called the elements the applications; namely, the externalities as viewed by the end user. These are the interfaces, both human and device, and the data base interfaces. The three applications of REVIEW, REPORT and CONSULTATION are described not only in terms of the applications software or the screen but in terms of the human interface externalities needed to ensure proper usage of the intended application.

The system services layer allows for the interconnect function. In this level, we place a set of clients that allow for the interaction of all elements at the top application layer. This architectural embodiment begs the question of centralized versus distributed. It clearly, in its very statement, indicates the need for a fully-distributed embodiment.

The network services layer is the control layer of the architectural elements. It is at this level that control over the integrity of all communications and data structures is maintained. It also is structured in a distributed fashion, recognizing furthermore that the system must function in a highly nonhomogeneous fashion.

The bottom layer is the transport layer. This layer also goes directly to the heart of the world view taken in the architecture. It is a world view that states that there will be multiple overlay networks, with capabilities for high-speed transport, but operating at different speeds and controlled by varying protocols. It also assumes, contrary to a hierarchical telephone world view, that the users will want to control their destiny, not a centralized telephone authority. Once liberated, the end users will want only more freedom, not less.

3.0 Human Interfaces

The importance of interfaces between human beings and these systems cannot be emphasized enough. Successful interfaces permit the use of computer systems to their fullest potential. Conversely, poor human interfaces can prevent the optimal utilization of a good computer system and can even completely discourage potential users to use such systems. Those users will find ways to avoid the system and work "around" it. Human interface designs must be viewed as solutions to existing problems, rather than as partial technical solutions looking for problems which they may solve. A successful interface is simple, intuitive, easy to learn and to remember, and it provides advantages over manual systems.

Putative advantages of good human interfaces include:

- a) time and cost savings
- b) improved data processing capability
- c) completeness of transactions
- d) improved access to same or even greater amounts of pertinent information, normally not easily available
- e) improved quality of the end product
- f) elimination of redundancies
- g) improved human communications.

Depending on the user, many or most of these features are needed in order to make a human interface successful.

The design process of a system provides a good opportunity to re-evaluate established work habits and workflow, of which the human interface is an essential component. Human interface design should not necessarily aim at just replacing the older, manual system, but to improve on it. Human interface designs should be the result of a combined effort between ultimate users and designers of the interface. During the process of studying the design of a user interface, active interaction of an informed user and the architect will likely unveil newer approaches to methods and procedures that may be an improvement over older or purely manual procedures.

Development and introduction of new systems human interfaces must take into consideration present workflow and work habits of the local culture, must identify and anticipate perceived and real needs, and aim to avoid adding a new process into an already busy environment. The application must represent a replacement of current activities and routines and prove to be an improvement over old methods. (See McGarty and Sununu)

During the early process of human interface design, it is important to consider the most generic aspects of applications, the kind of people involved, and the environment in which these will exist so that these are not forgotten while designing the very specific user needs. Modular designs with a very broad range of options are much better than rigid designs. Special subsets of such modules can be then concatenated for specific users. It is important to consider the actual means of interaction such as keyboard, joystick, mouse, touch screen, voice, etc.

The authors have observed that implementation of certain commercial products in the areas of radiology information systems and image storage and transmission systems which were expected to improve work, reduce redundant tasks, increase efficiency, or reduce the number of employees have, in fact, produced the opposite effect. That is, these new systems have failed to replace older systems, have added new processes, more paper work, and increased the level of complexity of work. In addition, implementation of these commercial systems have required additional people in order to support their operations and to maintain them. These factors resulted in more confusion, lengthening of the overall process, and increased expenditures.

User interfaces should adapt to the user's functions and needs and not the other way around. Interface designs must consider the user's needs first and the technical idiosyncrasies second. Although design of user interfaces must take into consideration the perceived current technical limitations of existing software and hardware, it is important to keep in mind the rapid evolution of systems that is occurring at present, and that in, fact, additional solutions and tools may be available. Meticulous attention to user needs is paramount. The design should come from careful examination of the system and process that the interface aims to address. In concentrating on the specific needs of a user in an organization, one must evaluate the effect and interrelations of such application in the global functions of the organization of which the user is a part.

There are independent, highly-specialized, human interfaces that serve a rather homogeneous and specialized group of users within a highly-sophisticated local culture. These are relatively easier to design. There are also human interfaces that must address the needs of larger and heterogeneous populations with varying levels of education, cultures and habits. These interfaces, which should be simpler, are, in fact, much more difficult to design.

The most common user interface is a workstation, the simplest of which consists of a video display monitor and a keyboard. The interactions are limited to typing commands on the keyboard and reading the text responses on the monitor. This type of human interface may suffice in a number of applications, such as word processing and electronic mail.

Other applications require display of images as well as text. Some applications may only require image representation of a pixel resolution in the range of 256 x 256. There are applications that require the representation of images of high spatial resolution in the range of 1,000 x 1,000 or even 4,000 x 4,000 pixels. In addition to display of text and images, other applications require voice recording and playback, and the representation of moving images, video, and three-dimensional images. Diagnostic medical imaging is an example of an area where these features are needed.

Some user interfaces require highly-sophisticated access to data, including high-resolution images. The requirements for these kinds of human interfaces are highly complex and sophisticated. Image enhancements, display of several images, image processing, display of several imaging modalities simultaneously, image quantification, cinematographic, and 3-D displays are but a few of the features required. The issues of display resolution, number of displays, voice annotation, text annotation, integration with other processes relating to the workflow, must all be addressed in order to achieve true integrations and, therefore, the two goals stated above.

Other human user interfaces require "view only" features; that is, these interfaces need to allow the user to recall in a rapid, easy, and straightforward way the data, including images, voice, text, etc. These user interfaces should be as easy to operate as the automatic teller machines (ATMs) widely installed in banking institutions.

Other user interfaces will require the additional complexity of providing real-time consultation among two or more persons who could be viewing the same data (images, etc.) on-line, in order to add their individual knowledge to these data (Session). Synchronized pointers should be available so that several users could concentrate on specific features of the data. During these sessions, data exchange may take place where several dispersed pieces of information could be shared by the group of consultants. This interaction could occur, as I mentioned, on-line, or even on a time delay, such as electronic mail is used commonly in large corporations today.

Issues of security, access and confidentiality remain important considerations in the technology, design, and development of human interfaces and exchange of data of a sensitive nature.

Example: Diagnostic medical imaging. Diagnostic medical imaging encompasses several kinds of modalities. These include radiography, nuclear medicine, computed tomography (CT), ultrasonography, magnetic resonance imaging (MRI), digital angiography, etc.

During the past two decades, impressive developments in several areas of medical imaging technology have taken place. The amount of information about the human body that is being produced is enormous. Unfortunately, this fantastic progress in the technology of medical imaging has not been paralleled with a similar improvement in our ability to manage the very vast amount of information produced by this advanced and multifaceted technology. Information overload has taken place. Unfortunately, it is difficult to convert this massive amount of information into knowledge. It is only with the intelligent introduction and implentation of excellent computer systems that this information overload can be tackled.

Except for conventional radiography, some of these areas of imaging are highly specialized and require complex computer processing of images in order to extract useful information. Images are stored on film and on magnetic or optical storage in local applications. Each of these imaging modalities have their specific acquisition, storage, retrieval, and image format. Image formats and communications standards are not yet fully realized. Manufacturers typically retain their proprietary stance, making it difficult or nearly impossible to represent, manipulate, store, transmit, or retrieve images from these modalities in the same format. Even conventional radiography is expected to become computerized during the next decade (computed radiography). Often, several kinds of images of the same patient need to be evaluated in conjunction in order to arrive at better diagnoses, rather than by looking at one of these images only. At present, these consultations occur while physicians are looking at films. Often, films are not available or have been lost. In addition, films can only be at one place at a time.

As the specialist in medical imaging evaluates images from a particular patient, he will need to have available other pertinent data such as demographics, referring physician's name, address and telephone number, the reason why the imaging examination has been requested, and also images and interpretations of previous imaging procedures performed on the patient. In addition, the imaging specialist may require results of non-imaging tests that the patient had such as blood tests, urine tests, etc. Having access to pertinent additional information helps the imaging specialist to produce a more accurate and specific diagnosis or recommendation. In the hospital of today, typically this additional information is not easily available and it may be dispersed over several departments in the hospital. The process of producing a diagnostic report usually consists of a dictation on magnetic media, and this dictation is then transcribed by a transcriptionist. Such transcribed voice is reviewed, edited and approved by the imaging specialist, either on paper or on a video display terminal. Such review and approval typically takes several hours or several days and is usually done without the images, so that the physician has to rely on his memory of the case. Because it takes so long for the printed, approved report to be produced, the information in such documents may be irrelevant to the immediate care of a patient. By the time the paper report is mailed and received by the referring physician, the report may be perfunctory. In many instances, the results of imaging examinations are reviewed by the referring physician and the imaging specialist, either in person or by phone. Many medical decisions take place based on this immediate interaction/consultation between the referring physician and the imaging specialist.

For example, in diagnostic medical imaging, a design could serve a specialist in computed tomography quite well; however, the functionalities developed may not be adequate for a specialist in ultrasonography, magnetic resonance imaging, nuclear medicine, etc.

In medical imaging there are some common functional features to the work in all imaging modalities. That is, there are common transactions that have to do with admitting the patient into the imaging department, specific clinical and examination features, resource utilization capture, image processing, image viewing, image recall, interpretation, voice and/or text (knowledge) annotations, billing, transmission of information to referring physicians. Therefore, in the human interface design, one should think of the totality of the process of patient care, department and hospital functions, as well as the individual imaging specialist's needs. Major and necessary goals in the design and implementation of human interfaces and of their underlying systems are that they produce one or both of the following results: (1) economies and (2) improvements in patient care.

Functional specification and early prototypes of three applications have been developed for medical imaging. These are REPORT, REVIEW and CONSULTATION.

REPORT: This application provides a single environment for the imaging specialist to evaluate images and pertinent non-imaging information in the process of producing a multimedia report. REPORT provides a convenient environment where the knowledge of the imaging specialist is attached to the image. The following is available to the reporting physician:

1) Patient name and demographic data
2) Referring physician's name, address and telephone contact
3) Reason(s) for the patient's referral
4) Working diagnosis
5) Images and pertinent quantitative data obtained during the current encounter and from previous encounters
6) Other imaging modalities.

The following processes are available to the REPORT application:

1) Image display, enhancements, and other manipulations
2) Image fusion (current and other modalities)
3) Single and multiple serial images
4) Cinedynamic display
5) 3-D imaging
6) Freeze image/publish image
7) Add brief voice annotation
8) Pointers and text annotation on the image
9) Final voice report, review, edit, and approve
10) View, edit and approve final transcription text
11) Multimodality display and review
12) Ability to enter text report directly
13) Create final multimedia report (image, voice and/or text).

Once the multimedia report is completed, reviewed, and approved, the report is "published." This means that the multimedia report now becomes available for review by other members of the patient-care team. The multimedia report so created is part of the patient's electronic medical

Second, the data environment in a multimedia communications world must have the capability to allow for interactive, conversational, and transactional communications between all of the users on the system. It must ensure the reconstruction of the reality of shared presence between the users in order to preserve the basic nature of the original multimedia communications message.

Third, the data structures must be both flexible enough to allow for significant further development but common enough to allow for the interfacing of systems in a large-scale fashion.

To develop a better understanding of how one can implement a system that meets the constraints, consider the following example of a system that has been implemented by the authors. The system is used in the context of a medical imaging environment in a major U.S. hospital. The system is used in the radiology department and initially stores images retrieved from the radiological imaging systems. These systems are varying in nature and may produce a film copy of a direct digital copy. The physician then takes the images, which are all scanned into the system, and reviews them for a specific set of findings or diagnosis. Once complete, the radiologist then records a voice description of the findings into the system.

At this point, this system is like so many others. It begins to diverge, by now having the capability of having the voice information transcribed into a text format to be used in subsequent analyses. The system also supports such functions as a CONSULT function that permits several physicians to view the current record, past records of the same patient, records of patients having similar disorders, and correlating these findings with other diagnostic findings. The CONSULT can occur both in a simultaneous fashion amongst several physicians or in a time-shifted or displaced fashion amongst a set of consulting physicians.

This simple example shows that to achieve all of the structure required in the applications, it is necessary to meet the three requirements that we presented above. Specifically, we must define structures for each of the data elements, inherent to their own specific characteristic. We cannot store just bits for latter reconstruction of separate parts of the overall image.

We define two overall data constructs that allow for the development of a robust multimedia communications system. They are the Simple Multimedia Object (SMO) and the Compound Multimedia Object (CMO).

The SMO is composed of three elements. At the tail end is the Basic Multimedia Object (BMO). This is the actual digitized or digitized representation of a data object. The BMO may be segmented or streamed. A segmented BMO (SG:BMO), is a bit limited, a priori, data element that may be the representation of a single image, a prior recorded voice segment, or any other data element that has a known prior length to its bit sequence. The second BMO is the streamed BMO (ST:BMO) because, ab initio, we do not know how long it will be. For example a voice sequence from a single user is a ST:BMO.

The SMO has two other fields. They are the SYNCH and the DECOMP fields. The SYNCH field allows for the determination of the temporal characteristics of the data element. Thus the SYNCH field for a voice segment details the sample rate, the bit structure, and the sample density. It may also contain any other temporal information necessary for the reconstruction of the temporal sequence by any arbitrary but intelligent processor. The DECOMP field contains all of the information relating to the logical and spatial structure of the data element. Thus if it is an image, and the image is in Postscript, then this and the Postscript settings are set in this field. This is an example of logical settings.

A CMO is a concatenation, in a logical sense, of several SMOs. For example, in the radiological case discussed above, we used the construct of a consultation wherein several physicians are using a combined set of voice, image, video, and text information on a specific patient. This specific collection may be a compound object. This object must itself have not only integrity within its constituents but integrity between them as well. It is this integrity between constituents that defines a compound object and it is in this combining in a spatially or temporally disperse environment that establishes the boundaries of multimedia communications.

The CMO is composed of a generalized data object which is not in more than the set of SMOs. The header of the CMO, however, is two separate and generalized fields. They are the ORCH, or orchestration, field and the CONCAT, or concatenation, field. The ORCH field allows for temporal timing relationships between the separate SMOs. Thus it allows, for example, the synching of the voice with the moving image. This is done through the out-of-band fields. It is important to note that we have, in this architecture, performed this task out of band. This is critical for several reasons. First, it allows the communications and storage media to understand the relationships amongst the bits and to provide for integrity amongst them as stored and transported. Second, this choice allows the applications programmer to deconstruct the images and to reprocess them in further detail. Third, it allows for the incremental building of new compound images in a simple fashion.

The typical CMO dynamics are those that may be observed in any conversation, as shown in Figure 1. We can show three separate sets of events. At the top, we display the CMO headers as they are set out across the network. It is these headers that provide the network and data storage elements with the information necessary for proper communications. The second set is the change of events. These represent the type of loads perceived by various processors in the network. The third set is the actual video, voice, image, pointer, and text durations. This latter set is, in essense, the multimedia session.

In the classical data base system design, the focus is on access time and storage density effectiveness. All too frequently the design of all of the other system elements are set secondary to the large, centralized data base. Large-scale billing and transaction processing systems are built around large data bases and the processing is done on those data bases to provide minimum cost per bill or minimum time per transaction. The issues of synchronization or orchestration are not even considered. In a multimedia environment, the load on the system may be significantly greater. It now must address the issues of temporal, spatial, and logical integrity of data both within and between elements as they are moving in space and time.

To address the above concerns, three architecturally different data base access alternatives, or data base management systems (DBMS), have been considered. Recall that in a multimedia environment there are multiple data base, storing images, voice, video, text, and other elements of the overall image. We show in Figure 2 three of these architectural alternatives.

(i) Centralized Single DBMS: This is the simplest construct and is what is currently in use by most "Multimedia" data base providers. It assumes that a single DBMS processor can be provided under the management and control of a single user or user community. It further assumes local, temporal, and logical proximity of the data storage devices that store the different multimedia elements.

(ii) Centralized Multiple DBMS: This system assumes that there are several or multiple independent data base elements that store and retrieve the data in different media. This system has multiple DBMS, some of which have disparate interfaces and may not necessarily be consistent. These are all then front-ended by a DB Router that, in effect, takes the responsibility of placing headers on the data and removing the headers when a CMO is received.

(iii) Distributed: A fully-distributed system requires that CMOs be generated at remote locations and that the data base processors (DBP) located at each of the distributed nodes has the capability of reading the CMO headers and passing out the SMOs that may be stored in its local storage. The communications network will provide for the retrieval processing necessary for the SMO and CMO.

5.0 Communications Alternatives

The communications aspects of the area of Multimedia Communications is more complex than the data base elements. In the last section, we described how to define multimedia objects and discussed ways in which we can store and retrieve them. In this section, we are concerned with the underlying mechanisms necessary for the interconnection, transport and control of the multimedia communications service.

The raw transport function, that of transporting bits from one place to another, is provided in a variety of fashions. It is this variety that is both a strength and a weakness. It is a strength from the perspective that the user has the flexibility to choose the best transport for the purpose at hand at the most cost-effective level. It is a disadvantage from the perspective that there is no underlying cohesiveness from the perspective of the multimedia objects defined above. The problem with providing this cohesiveness at the lowest level of the architecture is that the ultimate beauty of the building may be governed by the sewer-system design, rather than our ability to deliver walls, ceilings, and floors, the externalities viewable by the user.

Thus we shall assume that transport is a given, it will be disparate, and cacophonous. It will be chosen by the system designer or user on a cost-effective basis, subject to performance constraints. If we consider the existing hierarchies for system development and those that have standards associated to them, the ISO/OSI seven-layer system provides a basis around which to expand our communications elements. The first four layers of the OSI architecture, i.e., Physical, Data, Network, and Transport (see Stallings), are all related to bit transport in an errorless fashion across error-prone communications systems. The top, or seventh, layer relates to the applications programs that we have already described in the past sections. The presentation layer relates to how best to "present" a data stream, not the information. The session, or fifth, layer is the one that has the most robustness for allowing effective multimedia communications. It will be with the session layer that we will build our communications elements to the architecture.

The session layer has four main functions as shown in Figure 3. These functions are as follows:

(i) Dialog Management: This function allows for the control over who speaks when and how the control of this dialog is handled from a local cultural perspective.

(ii) Synchronization: This is the most critical function. It controls the relationship of all of the CMOs to the end users.

(iii) Activity Management: Activities are concatenations of users' interactions that need to be completed before the entire task is complete. For example, if a woman goes to have a mammography, the activity is defined as the entry of the woman to see her physician, up to and including the notification of negative findings or the resolution of any positive findings. In the event of a positive finding, the activity may last for several years as various protocols for treatment and resolution are taken. Failure to take all steps is life threatening. We shall not discuss activity management in this paper.

(iv) Event Management: This is the underlying control function that we have discussed before.

Figure 4 describes the synchronization problem. Synchronization in the session service ensures the overall temporal, spatial, and logical consistency of the multimedia objects as they are shared in a conversational mode. Consider a simple example of the nature of the problem. Consider three individuals who are in a multimedia conversation at three different locations. Consider, in addition, three different data bases, video, text, and voice, that are also part of the session. A typical session conversational element may, at any one time, include a video segment from the stored data base, a real-time voice segment from one of the users, and a text element created as an annotation to the stored video segment. As the conversation continues, each user may interject new information or select, process, and store data from one of the three data bases. The synchronization process is to ensure the integrity of the overall conversation.

Recall also that the session being displaced in location may have access to different communications systems that may introduce errors, delays, and other variants in the quality of service. In Figure 4, we depict the session process as having three major roles. First, it must collect the SMOs from the user and logically bind them into a CMO. The reverse process on reception of a CMO is also necessary. Second, the session server must interface to the lower protocol layers in the transport schema, such as the transport and network layers. Third, the session servers are distributed from session cluster to session cluster. It is necessary to have some form of communication between them to enable the management of the synchronization functions. This is accomplished by means of an interprocess control (IPC) function.

The IPC level provides for total session control. The details of this have been discussed in McGarty (1991, [3]). Simply, there are source session servers (SSS) and receiver session servers (RSS) that communicate in an out-of-band channel. The communications uses the CMO header information and the servers and then manages the disparate communications assets via the transport layers available to them. If there are several transport options, then the SSS and RSS must allocate and manage these resources. Management of the data packets by the SSS and RSS is done through the insertion of different but interpretable SYNCH pulses that are set in the data streams. The SYNCH pulses may take one of three forms: SYNCH_T (time); SYNCH_S (space); and SYNCH_L (logical). Thus the designer has full flexibility with the out-of-band channel for pre and post-processing.

Dialog management is simply the control of who talks and when. It permits the flow of control from one user to another. We have developed several dialog schemes that range from: Hierarchical (one leader on the conversation); Round Robin (next in line talks); Priority (one with the agreed-to greatest need talks); Random Access (next one to get the token talks).

Further detail in the implementation of the system architecture is depicted in Figure 5. Here we have shown a fully-distributed client server approach to the system. Each workstation has a set of software that supports it; specifically, the applications as we have discussed and the clients supporting such things as mail, file, and directory, as well as the session client. Recall that the session services are at the heart of the multimedia communications system. As such, clusters of session servers are used to support local session clients. These servers are supported by network servers. The network servers are controlled by a network client. It is the network client that is at the heart of the transport network.

In this architectural embodiment, we show that the data bases, as well as devices and human users, are clustered in session server clusters. The servers must communicate through the SMO/CMO protocols to ensure that the underlying session services are provided.

6.0 Conclusions

Multimedia Communications has taken on many forms over the past several years, all being disparate in form and nature. In this paper, we have gathered together a set of underlying and unifying constructs based upon our experiences in implementing several such systems. These common threads have been developed in the context of an architecture. This paper develops such concepts, detailing the elements of interfaces, data bases, transport, interconnect, and control.

We have demonstrated that a cohesive theory can be developed to ensure that all of the elements are integrated. We have also argued that one's world view can strongly influence an architecture. Thus, given a structure, it is often necessary to deconstruct the developer's world view to see if it is consistent with that of the end users who ultimately must use the system for beneficial result.

7.0 References

[1] Adiha, M., N.B. Quang, Historical Multimedia Databases, Conf. on VLDB, Kyoto, 1986.

[2] Christodoulakis, S., et al., The Multimedia Object Presentation Manager of MINO, ACM., 1986, pp. 295-310.

[3] Coulouris, George F., Jean Dollimore, Distributed Systems, Addison Wesley (Reading, MA), 1988.

[4] Elmarsi, Ramez, Shamkant Navathe, Fundamentals of Database Systems, Benjamin (Redwood City, CA), 1989.

[5] Fortier, Paul J., Design of Distributed Operating Systems, McGraw Hill (New York, NY), 1986.

[6] Leffler, Samuel, et al., 4.3 Unix Operating System, Addison Wesley (Reading, MA), 1989.

[7] Little, T.D.C., A. Ghafoor, Synchronization and Storage Models for Multimedia Objects, *IEEE Journal on Selected Areas in Communications*, April 1990, pp. 413-427.

[8] McGarty, T.P., Multimedia Communications, Wiley (New York, NY), to be published.

[9] McGarty, T.P., Multimedia Data Base Systems, Presented at Syracuse University, April 30, 1990.

[10] McGarty, T.P., Session Management in Multimedia Communications, Presented at MIT, May 2, 1990.

[11] McGarty, T.P., Stochastic Systems and State Estimation, Wiley (New York, NY), 1974.

[12] McGarty, T.P., Understanding Multimedia Communications, Presented at MIT, February 1990.

[13] McGarty, T.P., Multimedia Communications Technology in Diagnostic Imaging, *Investigative Radiology*, Vol. 26, No. 4, pp. 377-381, April, 1991.

[14] McGarty, T.P., Image Processing in Full Multimedia Communications, *Advanced Imaging*, Vol. 5, No. 11, pp. 28-32, November 1990.

[15] McGarty, T.P., M. Sununu, Applications of Multimedia Communications Systems to Health Care Transaction Management, Presented at Healthcare Information and Management Systems Society, San Francisco, CA, pp. 71-89, February 1991.

[16] Mullander, Sape, Distributed Systems, ACM Press (New York, NY), 1989.

[17] Nicolau, Cosmos, An Architecture for Real Time Multimedia Communications Systems,

FIGURE 2: MULTIMEDIA DATA BASE ACCESS ALTERNATIVES

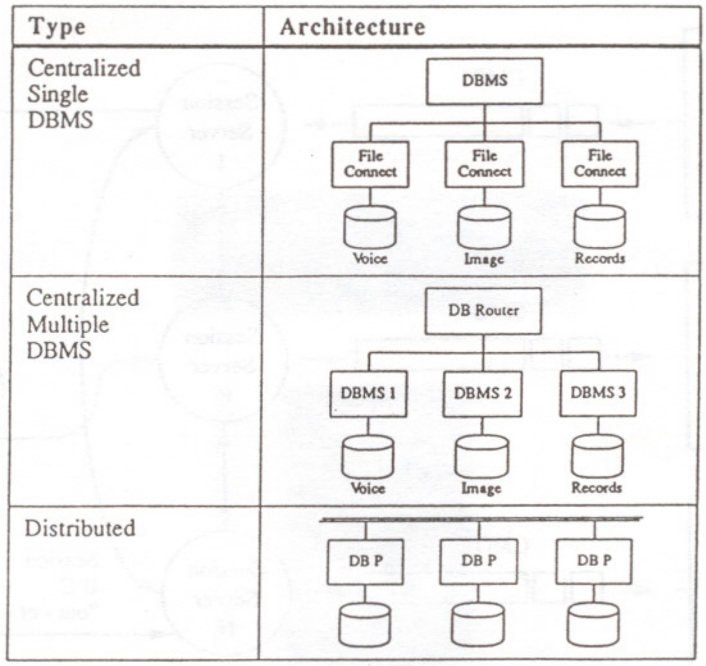

FIGURE 3: SESSION FUNCTIONS

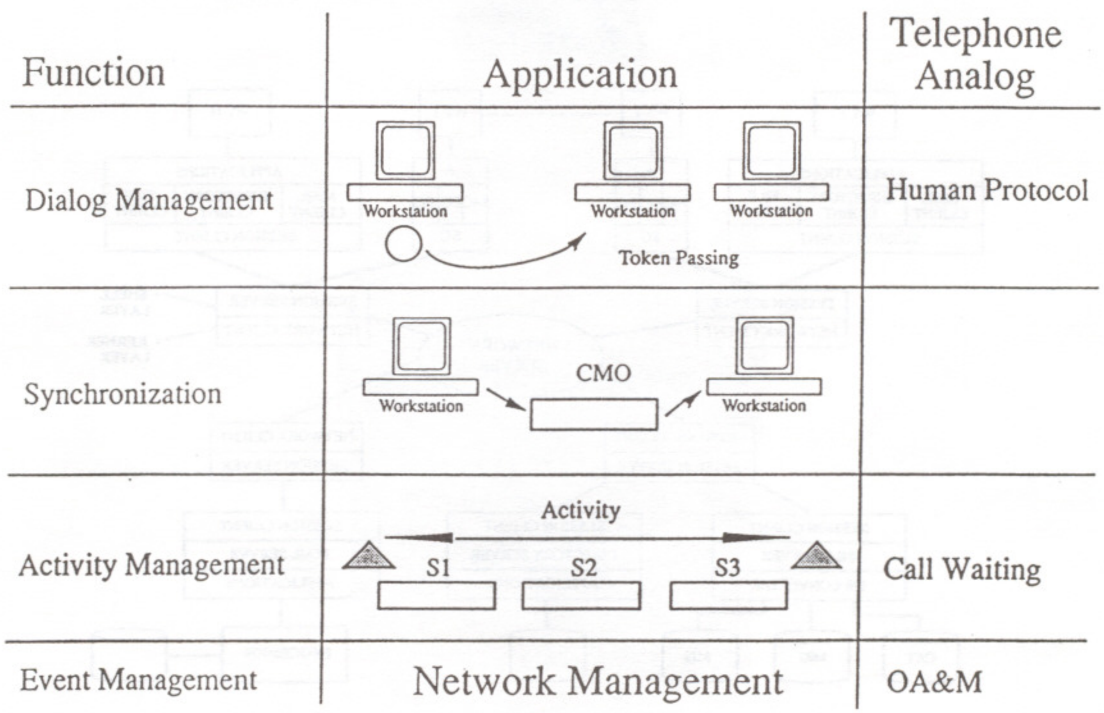

FIGURE 4: SYNCHRONIZATION

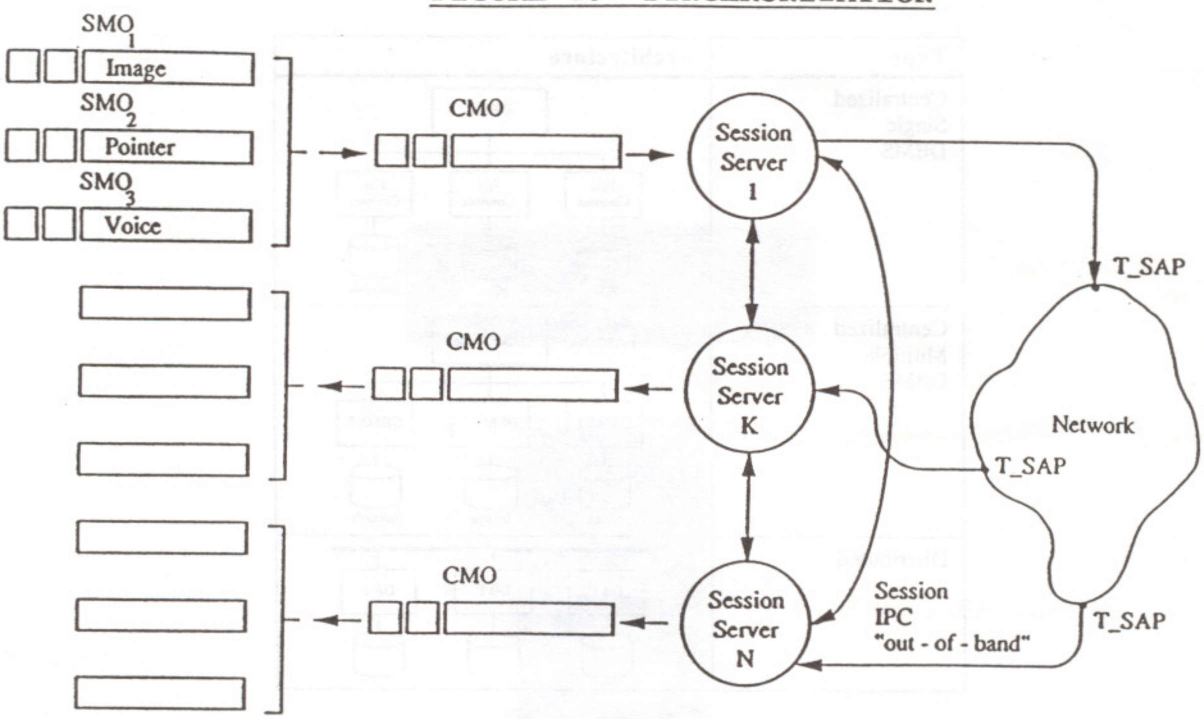

FIGURE 5: DIALOG MANAGEMENT