Report Number: WUCS-TM-94-3

1994-01-01

Distributed Multimedia Systems Research Prospectus

Jonathan Turner

Distributed multimedia computing and communiation systems combine computer systems, networks and distributed software to facilitate applications that enable and enhance collaborative work, direct interpersonal communication, remote access to information and real-time presentation of information from a variety of sources. Over the next decade, we expect such systems to become central to the infrastructure of our increasingly information-driven society. This prospectus describes a program of research being pursued within the Computer and Communications Research Center of Washington University and the Departments of Computer Science and Electrical Engineering. This program seeks to promote the creation of effective distributed multimedia systems and... Read complete abstract on page 2.

Distributed Multimedia Systems Research Prospectus

Jonathan Turner

Complete Abstract:

Distributed multimedia computing and communiation systems combine computer systems, networks and distributed software to facilitate applications that enable and enhance collaborative work, direct interpersonal communication, remote access to information and real-time presentation of information from a variety of sources. Over the next decade, we expect such systems to become central to the infrastructure of our increasingly information-driven society. This prospectus describes a program of research being pursued within the Computer and Communications Research Center of Washington University and the Departments of Computer Science and Electrical Engineering. This program seeks to promote the creation of effective distributed multimedia systems and develop the fundamental understanding needed to determine how such systems can be best designed and used.

Distributed Multimedia Systems Research Prospectus

Jonathan Turner

WUCS-TM-94-03

July 23, 1994

Department of Computer Science
Campus Box 1045
Washington University
One Brookings Drive
St. Louis, MO 63130-4899

Abstract

Distributed multimedia computing and communication systems combine computer systems, networks and distributed software to facilitate applications that enable and enhance collaborative work, direct interpersonal communication, remote access to information and real-time presentation of information from a variety of sources. Over the next decade, we expect such systems to become central to the infrastructure of our increasingly information-driven society. This prospectus describes a program of research being pursued within the Computer and Communications Research Center of Washington University and the Departments of Computer Science and Electrical Engineering. This program seeks to promote the creation of effective distributed multimedia systems and develop the fundamental understanding needed to determine how such systems can be best designed and used.

Contents

Distributed Multimedia Systems Research Prospectus

Jonathan Turner

1. Executive Summary

In the last decade, computing and communication systems have become increasingly intertwined and interdependent. New high speed network technologies and the emerging, but still primitive, multimedia capabilities of personal computers and workstations, will vastly accelerate this trend. We define *distributed multimedia computing and communication systems* as systems which combine computers, networks, and distributed systems software to create applications which enable and enhance direct interpersonal communication and collaborative work; or provide access to and processing of information from a variety of sources for real-time presentation. Distributed multimedia systems can be a powerful tool for collaboration, for gathering and analyzing information and then using it to synthesize new knowledge. Because these new systems represent a major departure from traditional computing, developing useful, cost-effective distributed multimedia systems will require substantial rethinking of how we design and implement information systems at all levels. Designing systems that will stimulate the invention of creative and productive uses of distributed multimedia technology is a major intellectual and engineering challenge.

We divide distributed multimedia systems into four major research thrust areas:

- *Scalable Gigabit Networks.* Distributed multimedia systems require networks capable of providing a large population of users with gigabit information streams. Gigabit networks must be scalable and admit economical implementation. They must also be flexible enough to accommodate a variety of hardware platforms and application access methods, and they must be reliable enough to support operations that are central to the basic missions of organizations.
- *Multimedia Computing Platforms.* The distributed multimedia applications of the future require computing platforms that are closely coupled to high speed networks and capable of processing a potentially large number of continuous high bandwidth data streams. As applications make greater use of visual information, computing platforms will be expected to handle enormous computational demands and provide access to massive amounts of storage. These demands require new ideas in system architecture and a rethinking of how operating systems, computer hardware and networks interact.

- *Multimedia System Software.* Distributed multimedia systems raise challenges that require substantive changes to programming environments, operating system resource management and networking software. New programming tools are needed to facilitate the development of complex distributed applications that operate reliably in a heterogeneous, dynamic networking and computing environment.
- *Distributed Multimedia Applications.* Understanding application demands and matching those demands to technological capabilities is central to maintaining a research focus that meets actual needs and helps ensure that systems research has an impact on real users. For these reasons, our research program builds on collaborations with application groups across the university.

By the year 2000, distributed multimedia systems will play a central role in science, medicine, engineering and many businesses and will become a key factor in the competitive positions of all sorts of organizations. At the same time, this technology will become a major arena for international competition. Organizations seeking to establish and maintain leadership positions in the computing and communications fields, will need to continue to innovate in order to create the technologies that are so central to an information-driven society. This program seeks to play a role in these innovative processes by providing a proving ground and testbed for advanced multimedia applications and a laboratory for the creation of successive generations of technology.

Our research program seeks to maintain a strong systems focus combining both experimental and theoretical research. We are committed to the position that successful engineering research requires an intimate knowledge of the practical issues involved in building complex systems, as well as strong analytical capabilities. We find that theoretical research, if disconnected from practical concerns, can become so inwardly directed that it loses focus and drifts into arcane and ultimately sterile pursuits. On the other hand, experimental work that is uninformed by a deeper understanding of fundamental issues, can have only limited and short term value. The activities of our four research thrust areas, come together through various experimental systems projects. These projects provide a testbed for our research ideas and a departure point for theoretical studies aimed at broadening our understanding and preparing the ground for successive generations of distributed multimedia systems. They also provide a concrete demonstration of our research ideas, facilitating their effective transfer into commercial use.

Close interaction with industry is central to the research program. We seek to carry out research that will enable industry to create successful distributed multimedia systems products. To ensure that our research ideas are understood and can be properly assessed and evaluated by industry, we provide regular workshops to report our progress, develop concrete demonstrations of our research ideas in experimental systems and work with industrial partners to help them understand how research ideas can be adapted to meet their individual objectives. We also seek their advice and input concerning our research goals and our strategies for achieving them.

This program brings together a talented team of researchers with an established track record of research accomplishments and successful collaborations with industry. Our networking group has played a central role in the intellectual and practical development of

ATM network technology and is internationally known for its research in multipoint communication and applications using it. We have designed and built a complete demonstration network from integrated circuits through signaling software and applications, and our multicast switching technology is now being used in the flagship product of one of the leading commercial vendors of ATM LAN switches. Our Project Zeus is leading the way in the practical deployment of ATM technology in a university campus and we have initiated collaborations with groups in Biology, Earth and Planetary Sciences, Medicine, Architecture and the University Libraries.

2. Research Program

2.1. Introduction

Distributed multimedia systems can be used for a wide range of applications. To illustrate some of the key ideas in concrete terms, we introduce a particular application in Figure 1. It should be understood that this is merely an illustrative example, representative of the many applications for which distributed multimedia systems could be used.

The example was inspired by a distributed scientific collaboration application in which *computational optical sectioning microscope* images are viewed by three scientists at different locations who are simultaneously linked by a collaborative work environment including video conferencing and a shared multimedia document editor, which they are using to prepare a paper for publication in a multimedia electronic journal. The scientists are each working at an advanced multimedia workstation that includes two large flat panel displays, including one which is both a display and an electronic tablet on which the user can directly draw pictures or write notes, which can be either captured directly or converted to digitally-manipulable text. Each workstation includes a pair of cameras, as well as microphones and speakers to permit the participants to converse in a natural and direct fashion.

The optical sectioning microscope is an instrument that produces three dimensional data sets of a translucent specimen by imaging the light from a series of closely spaced focal planes using one or more CCD cameras. Because the light imaged at each focal plane is contaminated by out-of-focus light from adjacent focal planes, massive image processing is needed to produce acceptable quality images. This instrument requires much lower light levels than other three dimensional imaging methods, allowing it to be used to create time-lapse movies of living specimens as they grow and develop. Advanced multimedia systems like the one depicted would permit such movies to be included directly in a scientific publication, tremendously enhancing the effectiveness of communication within the scientific community. Scientists at Washington University are currently using this instrument and other advanced microscopy techniques to study a variety of problems in biology and neuroscience [7, 119, 120, 128, 134, 135, 136, 137].

The CCD cameras can generate data at up to 200 Mb/s each and configurations with multiple cameras sensitive to various fluorescent dyes can be used to observe structures with different chemical compositions and relate them to one another. The figure shows a specimen being viewed by two cameras with the output data being processed separately by

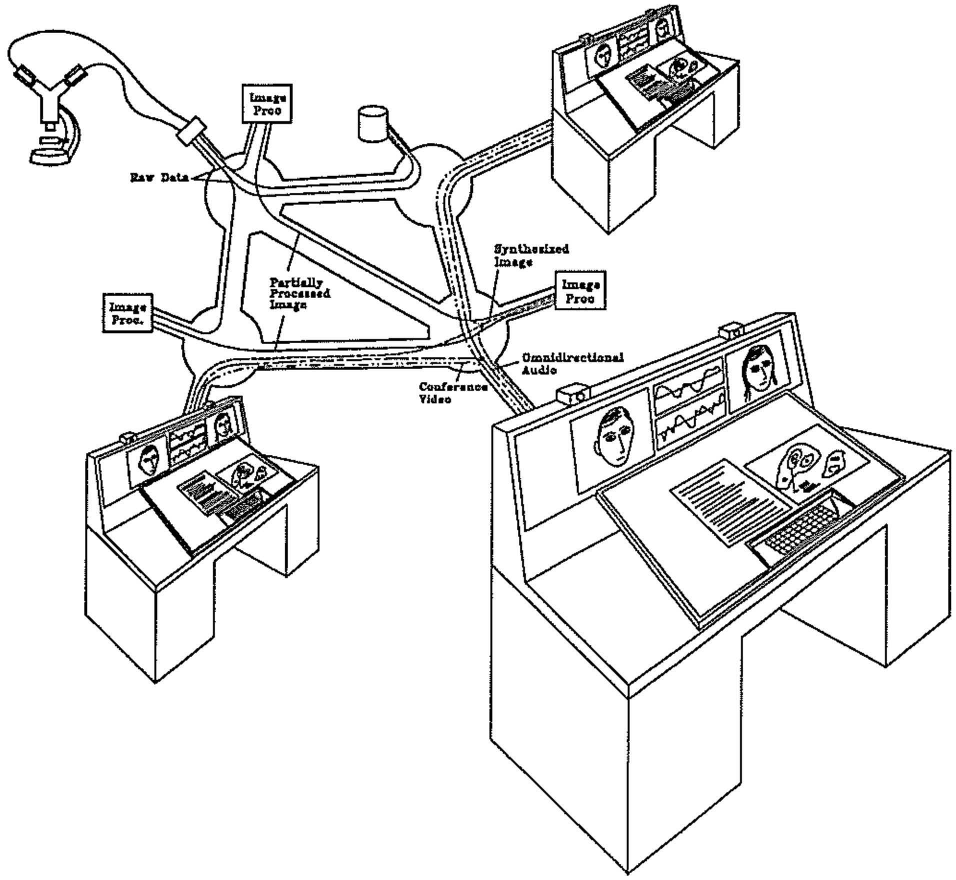

Figure 1: Scientific Collaboration using Distributed Multimedia Systems

two image processing computers accessed over the network and the partial results forwarded to a third computer which combines the data streams and produces the required synthesized images and sends them to each of the three users. The data is also stored on a networked mass storage system for archival purposes and for subsequent analysis. The audio channel is shown as an omnidirectional multicast virtual circuit on which audio is transmitted by each of the endpoints with audio bridging done at the receivers to combine the individual streams. In addition to observing the images, the users can execute local programs that analyze the data represented by the images and present the results graphically on their local displays.

The figure does not show the virtual circuit used to communicate the changes to the multimedia document, nor does it show the flows of control information that would allow changes to be made to the application as it runs. These changes might include changing control variables in the microscope cameras or the image processors, or even interposing a new processing step, which might require the services of another computer.

The communication and processing resources required for such an application extend well beyond current capabilities. Today's CCD cameras have resolutions of 1024×1024 pixels and a typical data set might involve 64 "optical slices" with hundreds of such data sets needed to construct a one minute time lapse movie. While continuing improvements in basic electronics will help in bringing these developments about, innovations in system architecture and design at all levels will be required to accelerate their arrival and provide the best possible cost-performance to serve user needs.

It should be clearly understood that this particular example is merely an illustration of the kind of capabilities that distributed multimedia systems can provide in the future. Any endeavor in which people need to communicate, share ideas and work together to analyze information or create some kind of intellectual product can benefit by these systems' ability to provide high bandwidth communication and direct access to computational and information resources. Over time, we expect computing and communication capabilities to become increasingly ubiquitous wherever people live and work. In addition to individual workstations, like the one described above, the multimedia environment of the future will include large wall-mounted displays with tens of millions of pixels that will be used for entertainment, conferencing and presentations, acting as both a shared white-board and display screen. Cameras, microphones and speakers embedded in the environment will allow people at remote sites to converse naturally with perhaps real-time video processing used to alter the images to establish and maintain eye contact. Display screens will be placed in public areas to create windows between remote locations and facilitate the informal interactions that can create a sense of community among different parts of large organizations. Computing will become an invisible resource that is available wherever one happens to be. The environment will adapt to each individual's preferences transparently and will make his or her private workspace directly accessible upon demand. The only visible part of the system will be the interfaces through which users interact with one another and with computational processes. The computer and networking hardware that makes it all possible will recede into the background. This is the long term vision that will inform our programs as we move forward into the next century.

2.2. Technical Challenges

In order to realize the vision of distributed multimedia systems, technical advances are needed at several different levels.

- *Scalable Gigabit Networks.* To fully realize our vision of distributed multimedia systems, communication technology must be capable of delivering information at rates in excess of a gigabit per second to thousands of individual users in a single organization, and at a cost per user that makes this not only technically feasible but economically realistic. Innovative architectural ideas are needed to continually expand the boundaries of what can be accomplished and bring the benefits of this technology to a widening user population. Equally important are innovations in network control and management to make the network's tremendous information carrying capacity readily available and provide the flexibility and reliability necessary in a rapidly expanding information society.

- *Multimedia Computing Platforms.* The distributed multimedia applications of the future require computing platforms that are closely coupled to high speed networks and capable of processing a potentially large number of continuous high bandwidth data streams. They will be expected to handle huge computational demands for everything from handwriting and speech recognition to real-time image processing. They will need to support massive amounts of storage as applications make more and more use of visual information. These demands require new ideas in system architecture and a rethinking of how operating systems, computer hardware and networks interact.

- *System Software.* The growing role of the network in providing access to remote resources and the introduction of continuous media such as audio and video requires some fundamental changes in the way operating systems interact with system hardware and how they schedule the use of processing resources. The development of distributed multimedia applications requires novel programming concepts to make it possible to construct complex software systems in which different media streams interact with one another and with the user in a natural and flexible way. Creating tools to facilitate the development of complex distributed applications that operate reliably in a heterogeneous, dynamic networking and computing environment is a central challenge.

- *Matching Technological Capabilities with Application Needs.* The best technology is of little value if it has no connection to real applications. We maintain strong connections to application groups across the University and are committed to working with these groups to create real applications that will advance their research and teaching missions. The feedback from those interactions helps us maintain our focus on the issues of greatest importance to the ultimate users.

There are several broader technical challenges that appear repeatedly in different contexts throughout distributed multimedia systems. One of these is *performance-oriented design.* Much of the recent progress in high speed networks has come from a recognition that there is an important qualitative distinction between *continuous media applications*, such as audio and video and *discrete media applications* such as electronic mail or file transfers. While it's difficult to give a precise characterization, continuous media applications involve the nonstop delivery of information, and even when they involve substantial processing (as in compressed video, for example), the processing has a predictability and regularity not found in more general applications. They are also very performance sensitive, since failure to keep up with the ongoing stream of information leads to jerkiness or gaps in the delivery of information to the end user, making for a distracting and ultimately unusable system. Emerging high speed networks provide different handling of these two classes of applications to ensure that continuous stream applications obtain the network bandwidth needed for acceptable performance while allowing the discrete stream traffic to dynamically share the bandwidth not required by continuous streams. To effectively deliver multimedia applications to users will require that similar ideas be applied within the operating system and application software. The key challenge is to find ways of supporting the allocation of the computer system's processing resources to multimedia streams while maintaining the ability to manipulate and process these streams in a flexible fashion.

A common argument is made that in creating multimedia applications, one should concentrate strictly on flexibility and the creation of novel application features, leaving performance as a secondary concern. Continuing improvements in processor technology, it is argued, will eventually eliminate the performance gap, so there is no need for software designers to address it directly. We believe this argument is flawed and note that state-of-the-art stand-alone multimedia workstations today support single video streams with about 77K pixels per frame and 10 frames per second. The applications we envision require multiple video streams with from 300K to over 1M pixels per frame and frame rates of 30–60 frames per second, together with multiple audio streams that must be processed for the purposes of echo elimination, noise reduction and conference bridging. This represents performance increases of at least two orders of magnitude, requiring twenty years to achieve if one relies only on the historical doubling in the performance of digital technology every three years. Accelerating the development of multimedia technology is going to require architectural and design innovations that consider performance as a central design issue at all levels, and which couple hardware and software in novel ways to achieve dramatic performance gains.

Another recurring element in distributed multimedia systems is the need for *scalable multiway communication*. Collaborative work and multiparty conferencing require flexible and extensible ways to allow users to share information and interact with one another at a distance. This requires that we move beyond traditional models of point-to-point communication and interaction and that we provide users with natural models of multiway communication that are easily understood and used, but also are flexible enough to give the user control over complex, highly structured applications. Multiway communication demands more general mechanisms at the programming and operating system levels; mechanisms that go beyond traditional interprocess communication mechanisms, such as message passing or remote procedure call. At the network level, multiway communication must be supported by facilities that can replicate user information streams and deliver them to different destinations. Novel transport protocols are needed to ensure reliable multipoint information delivery, where called for, in a way that allows applications to scale up without deterioration in performance. Computing platforms may require similar replication capabilities to deliver data streams to different devices.

The research program is organized into four major research thrust areas: scalable gigabit networks, multimedia computing platforms, multimedia system software and distributed multimedia applications. Our vision and near-term plans for these areas are detailed below.

2.3. Scalable Gigabit Networks for Distributed Multimedia Systems

In the last few years, it has become clear that *Asynchronous Transfer Mode* or ATM technology will set the direction for the next generation of both public and private networks. Applications, such as the distributed scientific collaboration example shown in Figure 1 require the bandwidths and scalability that only ATM can realistically promise to deliver. While advances in ATM technology are very encouraging for distributed multimedia systems, there remain substantial challenges to be met before ATM can realize its full potential. In brief, we've addressed the basic problem of how to provide dramatically higher network bandwidths, on both a per port and aggregate basis, but we have at best a partial understanding of the more difficult problems of network control, reliability and management.

Moreover, we estimate that price-performance must improve by a factor of 100 to 150 before multimedia networks can be successful in the broader population. Reaching this level by the end of the decade will require a continuing stream of architectural and design improvements in networking technology.

2.3.1. Multimedia Switching Systems. Switching systems are the key hardware components of multimedia networks. In order to support multimedia applications such as video distribution and conferencing, switching systems must be able to support multipoint communication, including *one-to-many* and *many-to-many*, and must ultimately be scalable to port speeds up to ten gigabits per second and aggregate throughputs of perhaps 100 terabits per second. Unfortunately, most ATM switch architectures are poorly suited to multipoint communication, when one attempts to scale them up to large sizes. In general, there are three cost and performance parameters that can be adversely affected as the system scales. The first is the complexity of the switching network itself. The second is the complexity of the routing tables required by multipoint communication. The third is the time required to add or remove an endpoint in a multipoint connection.

A system with n inputs and outputs, m virtual circuits per port and which does not block any virtual circuit request that can be supported by the external links has optimal complexity and call setup performance if the network complexity grows no faster than $n \log n$, the size of the routing tables grows no faster than mn and the time to setup or modify a virtual circuit does not increase with the size of the system. Unfortunately, all multipoint switching systems developed up to this point fall far short of these ideals. The most popular approach, based on fixed path Beneš networks has network complexity which grows as $n^2 \log n$, translation tables which grow as $mn^2 \log n$ and worst-case call setup complexity of $\sqrt{n} \log n$. A new architecture based on cell recycling [179] provides the only known method for achieving optimal performance simultaneously for all three parameters. For $n = 2^{16}$, a reasonable size for a public network switch, this cuts the complexity of the network by a factor of 2^{16}, the size of the translation tables by a factor of one million, and the call setup complexity by a factor of four thousand.[1] This architecture promises, for the first time, to make it feasible to implement multipoint communication on a large scale at a reasonable cost.

Another important issue in the design of gigabit switching systems is the mismatch between electronic circuit speeds and optical fiber bandwidths. *Multichannel switching* addresses this mismatch by distributing the traffic from one fiber across multiple channels in the switch. Novel switching system architectures which exploit the multichannel context to reduce implementation complexity have been demonstrated in [139]. These approaches promise to facilitate the practical implementation of cost-effective gigabit switching systems.

2.3.2. Control and Operation of Multimedia Networks. The success of large scale multimedia networks will depend critically on the ability to provide a seamless, uniform interface to the applications that use the networks. This must be accomplished in the

[1]This statement assumes that the constant factors in the two cases are equal, which is approximately correct if they are implemented using the same technology, clock rates, etc.

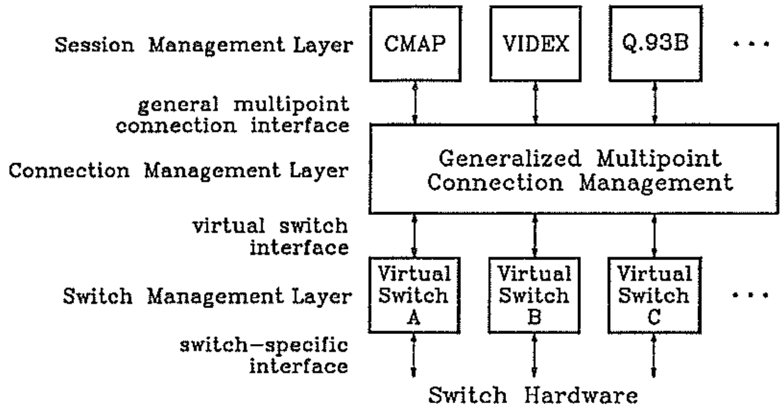

Figure 2: Multipoint Connection Management Software Organization

context of a network comprising equipment owned and operated by different organizations and produced by vendors. Currently, interoperability among different vendors has been achieved through the use of standard protocols, but with each vendor implementing those protocols and the algorithms that use them in a proprietary fashion. This approach has become increasingly unworkable as the control algorithms have increased in sophistication and subtle differences in vendors' implementations have led to either outright incompatibilities or reliability problems that often become apparent only in an operational context.

Our group at Washington University has created the first complete protocol suite for multipoint connection management and has developed a first implementation in our campus ATM network [29, 48, 70, 89]. The software has been structured in three layers, as illustrated in Figure 2. The lowest layer isolates all the hardware-dependent code and provides a *virtual switch* interface to the upper layers. The top layer provides the access protocols used by application programmers and can support multiple access protocols. The middle layer responds to commands from the top layer and provides network-wide coordination of the various switching and transmission resources. This separation greatly simplifies the task of adding a new platform to the network and allows new application interfaces to be added without directly affecting the middle layer. The middle connection management layer is crucial to the reliable operation of the overall system and includes the most subtle and critical algorithms from a reliability standpoint. Separating this from the hardware and the applications allows it to provide an abstract and general, but conceptually simple communication service which is largely unaffected by evolving hardware and applications. Moreover, if this software layer is implemented in an open, non-proprietary subsystem and used by different vendors, the chances of achieving full inter-operability are tremendously enhanced.

Connection management software in ATM networks must provide mechanisms to create and modify multipoint connections in support of multimedia applications. An essential part of this is managing the switching and transmission resources and selecting the resources to serve a given request as effectively as possible. This includes mechanisms for selecting routes and managing the bandwidth on links. Managing dynamic multipoint connections while ensuring efficient use of the overall network resources remains a difficult and challenging problem, although we have made substantial progress in recent years [90, 91, 185, 187]. An important next step is to understand how distributed routing protocols for gathering and updating the local routing database [132] affect the quality of multipoint connections. Effective bandwidth management in the context of guaranteed quality of service requires algorithms for determining if an arbitrary set of connections with known traffic statistics can safely share a link [2, 8, 22, 26, 27, 115, 121]. References [174, 176] offer one approach to this problem which is both general and efficient enough to make it a candidate for practical use.

2.3.3. Reliability and Management of Multimedia Networks. As society becomes more dependent on networks, the reliability of the distributed algorithms that manage networks becomes increasingly critical. The dependence of ATM networks on extensive and dynamic connection state information makes them more vulnerable to reliability problems than networks based on datagram protocols. While the public telephone networks have successfully dealt with these challenges, they have done so in an environment which is more homogeneous, more tightly controlled, less dynamic and less complex than will be the case for emerging multimedia networks. This will create major new challenges that will have to be met through innovative systems-oriented engineering research.

The notion of *self-stabilizing* algorithms has proven to be an extremely useful tool in making distributed algorithms robust in the face of a wide variety of failures [10, 11, 12, 132, 133, 183]. A protocol is self-stabilizing if when started from an arbitrary state it exhibits "correct" behavior after finite time. While typical protocols cope with a specified set of failure modes like message loss and link failures, a self-stabilizing protocol copes with a set of failures that subsumes these categories and is also robust against transient errors (e.g., memory corruption and incorrectly sent messages). There is evidence from real networks that transient errors occur and cause networks to fail unpredictably. Thus stabilizing protocols are attractive because they are *more robust* as well as *potentially simpler*, since they avoid the use of a large number of independent mechanisms to deal with a catalog of anticipated faults.

We have identified general techniques such as *local checking and correction* and *counter and timer flushing* for making protocols fault-tolerant. We are applying these techniques to many real protocols (e.g., the IBM token ring, the 802.1 spanning tree protocol, to produce what we expect will be simpler and more robust protocols. For instance, we have found that by adding a counter to the token in a token ring protocol we can provide a very robust scheme for dealing with lost or duplicate tokens. In the context of ATM signalling, we feel that techniques for self-stabilization will prove to be invaluable in the design of reliable multipoint connection management protocols.

While the design and analysis of switching systems for ATM networks has received considerable attention, there has been relatively little thought given to the issues of reliability as they pertain to ATM switching systems. The design of cost-effective but highly reliable switching systems requires a careful balancing act. Since the resources that can be spent to achieve high reliability are finite, it's important to apply those resources where they can have the greatest effect. This requires an understanding of how systems are most likely to fail and the ability to find the simplest possible mechanisms for coping with those failures. Successfully attacking the reliability problem requires a combination of software and hardware mechanisms that can detect failures when they occur, allow the system to be reconfigured to operate in the presence of failures if necessary and assist in the process of localizing and repairing the fault.

To ensure that networks operate smoothly with high reliability, network managers must have monitoring and control mechanisms that allow them to observe the network's behavior and detect potential problems as quickly as possible. Control mechanisms are needed to allow problems to be isolated so that they affect as few users as possible. Ideally, network operational problems should be invisible to the user. Where possible, automated tools should detect problems and reconfigure around them before the users notice that anything is wrong. Since network problems can arise in subtle and unpredictable ways, there will be a continuing need for manual tools as well that make problems visible to network managers and give them the mechanisms needed to correct them. The tremendous amount of connection state information present in ATM networks complicates this task significantly. To assist network planners in deploying equipment in the most effective way possible, performance monitoring and analysis tools should be integrated with network design tools that can configure networks to avoid unnecessary performance constraints and can make them robust against link and switch failures. We have initiated a study of the network design problem in the context of ATM networks and have devised a novel and very promising approach for solving it [51, 53, 53].

2.4. Multimedia Computing Platforms

The design of flexible and high performance multimedia computing platforms (MCP) is critical to the successful realization of advanced distributed multimedia applications, such as the scientific collaboration application shown in Figure 1. While a variety of computing platforms are commercially available, industry is only beginning to come to grips with the requirements which a truly versatile and effective MCP must satisfy. These requirements stem from three basic sources which together, differentiate an MCP from typical computing platforms:

- *Multiple High-Bandwidth Continuous Data Streams.* The typical workstation has one or perhaps a few input and output streams. An MCP must deal with a much larger number of such streams. In addition, the streams will have a mixture of characteristics at both the application and display levels (e.g., video, speech, pen, etc.) and at the underlying interconnection network level.

- *High Computing Requirements.* MCPs will often be used as shared computation servers, processing data streams from multiple users simultaneously. This will lead to extremely demanding computational requirements. Applications involving simulation and real-time visualization or image processing are among those that will place extreme demands on networked computation servers.

- *Massive Storage Requirements.* Multimedia applications require fast and continuous access to massive amounts of storage. Meeting these requirements is likely to require large hierarchical mass storage systems with inexpensive storage devices, such as optical disk jukeboxes at the bottom of the hierarchy, with fast magnetic disk arrays and semiconductor file caches at the top.

There is a clear need for new architectural approaches which are oriented toward the evolving requirements of distributed multimedia computing. They must meet not only existing needs, but have a built-in flexibility which will permit the addition of bandwidth, processing power, and real-time performance guarantees in response to application demands. One possible approach is introduced below.

2.4.1. System Architecture. The real-time processing of high-bandwidth continuous media streams is a fundamental feature of multimedia systems. More broadly, systems must be designed so that guaranteed performance can be supplied in a dynamic fashion. Visual communication and user interaction dominates decisions concerning the MCP design space. Keeping up with the needs of visualization (e.g., image processing, 2D and 3D graphics, etc.), taking into account the requirements of the human perceptual system and including demands for smooth and natural interactivity over a wide range of applications, requires that future MCP designs provide non-stop delivery of continuous stream data and careful balancing of processing resources.

To meet the demands of advanced multimedia applications, we need to balance the requirement for non-stop processing of continuous media streams with the need to provide flexible, responsive user interaction and efficient processing of computationally demanding discrete media streams. On the one hand, the high bandwidth and processing requirements of video and certain audio applications to necessitate special-purpose hardware support. *Continuous media processors* (CMP) (e.g., devices that perform compression, decompression and filtering) should be shareable so that they can be allocated to different streams as necessary. In addition, a smooth path must be present so that the output of such devices can be accessed by more general purpose computers for further processing activities (e.g., feature extraction). On the other hand, general multimedia applications and operating system support requires high powered general-purpose instruction-processing engines for managing user interaction and executing demanding computations. Current and emerging computing platforms are well-suited to this side of the multimedia equation. The challenge now is to bring the ability to handle discrete and continuous media applications together in an effective way.

We expect emerging high speed networks to accelerate the trend to network-centered computing that has been taking place over the last ten years, to the point where many peripherals now directly associated with a single processor will be accessed across the network. This is illustrated in Figure 3 which shows a conventional workstation on the left, with a single CPU, memory and peripheral subsystem consisting of a system I/O bus connected to the memory bus through a system bus interface (SBI), with disks, network interface (NI) and user interaction devices such as display, mouse, keyboard and audio (some user devices connect through the system bus with others on the memory bus). The middle part of the

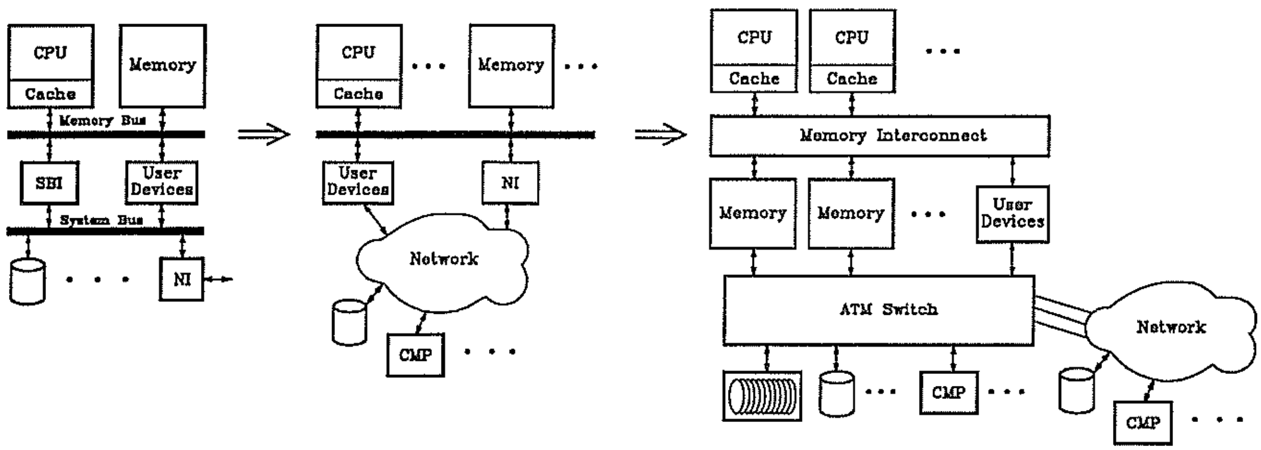

Figure 3: Architectural Trends for Multimedia Computing Platforms

figure shows a system in which the network interface has moved up to the memory bus and the network has taken over the role of the system bus, facilitating shared access to files and continuous media processors. Our current research activities on multimedia over ATM have included adaptations to conventional workstations that can be viewed as examples of this evolutionary trend. Our current activities include the design and implementation of an experimental 620 Mb/s network interface for this type of architecture.

In the future, we see a need to extend this trend one step further, as shown in the right hand side of Figure 3. In this design, the primary network connection is through the main memory modules, with the functions of the memory controller integrated with the network interface controller, eliminating much of the network traffic from the memory interconnect. This architecture allows the server's access to network bandwidth to be scaled with memory capacity by adding network ports along with memory modules. In the figure, a dedicated ATM switch is shown, to provide low latency access to disk and continuous media processors, ensuring high performance for streams which may require pipelining through multiple CMPs or through CMPs and memory modules to allow for complex processing by the server's general purpose CPUs. The diagram shows a mass storage subsystem such as an optical disk juke box for archival storage, along with magnetic disks for rapid retrieval. To support many of the advanced multimedia applications of interest, these different storage elements may be linked into a hierarchical mass storage system in which the disk and the server's memory module are used as staging areas for data requested from remote workstations.

2.4.2. Workload Characterization and Data Collection. Driving the design of the MCP are multimedia application requirements. Currently, there are no accepted multimedia workload benchmarks, analogous to the SPEC benchmark for workstations, on which to test design ideas and determine performance associated with alternative approaches. Thus, one of the initial research tasks will be to explore multimedia application requirements and develop a multimedia benchmark set.

Initial investigations will center on the gathering of data to characterize our experience with our 155 Mb/s multimedia ATM network, described in section ??. Experiments will be carried out, for example, to determine how often users elect studio-quality (full-resolution) video as compared to one-quarter sized images (NTSC vs CIF). In addition, we will examine the effects of multiple video windows on user interaction and system performance. User preferences regarding the size, number and update rate of windows for video will be an important factor in characterizing the anticipated workload of the MCP. The MMX is instrumented to allow for the collection of workload data.

Concurrently with the activities described above, we will develop analytical models of the anticipated workload. These models will focus on video, images, multirate data streams, and instruction and memory access traces. Video connections will be modeled by a fluctuating data rate with a low peak-to-average ratio typical of JPEG video. The average data rate depends upon Q-factor, image size and frame rate. Image connections will be modeled by file transfers whose statistical characteristics will be determined in conjunction with electronic radiology collaborators at the Mallinkrodt Institute of Radiology. One modeling approach for multirate data is to use ergodic Markov chains with states representing different data rates. The instrumented MMXs on the Multimedia ATM network will be used for estimating model parameters. These models, along with appropriate instruction traces, will in turn be used to drive general simulation models of the MCP, and for the development of synthetic workloads which can be used to test MCP prototypes under more controlled conditions.

2.4.3. MISC (Multimedia Instruction Set Computer) Design. An MCP with the requisite processing power and the flexibility to adapt to new requirements needs a combination of special and general purpose processing capability. As discussed in Section 2.2 general purpose processors, and even a moderate size parallel processor system, may not have the required capability for many years. Due to this, our architectural approach provides for special purpose hardware for many of the stream oriented operations such as compression/decompression and image transformation. Figure 3 gives one view of MCP partitioning between memory, general purpose processors, special purpose processors, and interconnections. However, much additional work remains to be done before we can determine if it represents the best partitioning. Thus, a key research component of the Center will be to use the results and models developed to determine the processing requirements, and to study the way in which the various functions required can be partitioned into operations with large computation requirements while still allowing new functions to be realized by composition.

Another aspect of this research concerns how the instruction and memory access associated with multimedia applications differs from general purpose scientific and commercial applications. If such differences are significant, then architecture modifications at the basic instruction set and memory organization levels may be appropriate. These results would be useful in guiding the design of future processors which may contain architectural elements important to distributed multimedia applications.

We also plan to study and parameterize the extent to which the necessary multimedia operations are stream oriented and to which they are random-access oriented in order to refine the MCP architecture. Key issues are maintaining processing bandwidth, minimizing latency, flexibility to adapt to new requirements (both speed and function), and affordability.

For example, given the organization shown in Figure 3, can the multimedia computing requirements be adequately dealt with, or is it necessary to move at least some of the special purpose processing to the memory side of the ATM switch in order to reduce latency? Note that a 155 Mb/s connection requires about 40 ms to move a moderate resolution 24-bit-per-pixel image. For operations that are truly stream oriented and can be pipelined, this does not represent a problem. However, for operations that require the entire image before processing on the first part of the image can be completed (e.g., histogram equalization) this quickly becomes an intolerable delay because of the latency that several such operations would introduce. Similar questions will be addressed concerning the design of the shared memory. For example, it is not clear at this point whether full shared memory is necessary, or whether it is equally effective to have just a portion of memory shared.

2.5. Multimedia System Software

System software provides the linkage among user and application needs and the system's computational and communication resources. Distributed multimedia systems raise new challenges that will require substantive changes to programming environments, resource management and networking software. System software will need to provide mechanisms to efficiently and effectively support continuous media streams and balance their resource demands with those of discrete media streams. Mechanisms will be needed for audio and video processing and providing tight linkages between software and hardware processing levels. There will be a growing need for programming environments that make the design of complex distributed applications involving multiple media streams and a heterogeneous computational and communication environment easier and more certain to result in systems that operate correctly and reliably. Mechanisms for high bandwidth communication among different processing elements (both local and remote) will have to be integrated into the operating system fabric to a greater level than ever before to ensure timely delivery of continuous media data.

2.5.1. Research Issues. Distributed multimedia systems raise a variety of challenges for system software. These can be divided into three principal categories, which are discussed in the following paragraphs.

Distributed application development. Distributed multimedia systems will make far more extensive use of distributed computing than is now common. This is driven both by application demands (e.g., collaborative work) and the ability of emerging high speed networks to support extensive resource sharing. However, the design and implementation of reliable distributed applications remains difficult, due to our incomplete formal understanding of distributed systems and the limitations of the programming tools currently at our disposal. The design of effective multimedia applications will depend critically on advances in our basic understanding of distributed systems and the creation of programming tools that address practical programming requirements and rest on a strong formal foundation, allowing precise mathematical reasoning to be applied to help ensure the reliability of the resulting systems.

One of the factors driving the increased distribution of processing is the desire to share resources across the network. Distributed access to remote storage systems is perhaps the most basic example and while such applications are already commonplace, distributed multimedia systems will greatly expand the set of resources that are accessible remotely and it will become increasingly desirable to access them in a completely transparent fashion. Distributed multimedia systems will also employ more sophisticated resource sharing, including shared access to networked compute servers and access to specialized processors that perform specific types of multimedia functions. Examples include digital signal processors capable of performing complex audio functions, including echo cancellation, conference bridging, mixing, filtering, gain control and so forth. One can also imagine video processors that perform compression and decompression, image enhancement, feature recognition, mixing of multiple video streams, combining video and computer-generated graphics. To allow such resources to be effectively shared among a community of users requires that system software be capable of allocating these resources fairly among the different requesting users and doing so in a distributed fashion to allow easy scaling to large configurations. Ideally, the system software should also be capable of mapping computational tasks onto the available processing resources, allowing it to make use of the available resources in the most effective way possible (so for example, if an audio mixing task could be done either on a specialized processor or within the local workstation, the system software should be able to select between these options without any explicit instruction from the user).

Distributed multimedia systems will be characterized by heterogeneity and continual change. This creates one of the biggest challenges for system software, which must accommodate these differences and insulate the application programmer from their effects. Heterogeneity exists in the computing platforms which use different data representations and have different performance characteristics and sometimes fundamentally different computational models. It exists in the networks which provide widely differing basic capabilities and provide both different performance levels and different ways of obtaining the requisite performance on behalf of applications. It exists in different data representations used for higher level application data (alternative video coding representations, for example). It is the responsibility of the system software to cope with these variations in a transparent fashion while simultaneously maintaining the high level of performance that will be demanded by many of the emerging applications.

Multimedia Processing Features. Multimedia system software must provide a number of specific mechanisms to support multimedia applications. These should include mechanisms that allow application programmers to declare performance requirements for modules that perform continuous stream processing. These can be in the form of *operable range declarations* that give minimum and maximum performance requirements, making it possible for an application to adjust its processing to the resources available. Given such information, it becomes possible for the system software to allocate processing bandwidth and possibly reallocate it as needed, so long as it keeps each processing element operating within its acceptable range.

The system software must also provide basic mechanisms for multiway communication in order to support collaborative work and other applications which require more than simple point-to-point communication channels. These mechanisms should offer a range of features and performance options. Some applications need only best-effort mechanisms which can be accommodated fairly directly by underlying network mechanisms, while others may require reliable multiway communication or causal ordering.

To make it easier for application programmers to construct multimedia applications, commonly used processing components must be made available in a library of multimedia processing components. The library must support *intermedia coordination*, that is the creation of linkages between different media streams for synchronization (e.g. synchronizing audio and video streams to provide lip-sync) and control interactions (e.g. using speech detection on an audio stream to trigger camera switching in a video conference).

Multimedia Run Time Environment. The run time environment required for distributed multimedia systems has some distinct differences from the current general purpose computing environment. Many of these differences are driven by performance demands. Continuous media streams used for real time audio and video presentation, require access to processing resources in frequent time slices in order to provide a smooth, high quality presentation that is free of jerkiness or gaps in the information flow. Typical general purpose operating systems allocate time slices in relatively large chunks (10-20 milliseconds) and cannot switch contexts quickly enough to support multimedia applications with tens of simple processing elements. Moreover, context switching times have not improved greatly in the last decade, despite the dramatic improvements in processor performance, since much of the computational gains have been due to architectural changes that have greatly increased the amount of context that must be saved and restored during a context switch (we include both explicit context such as registers and memory management data structures and implicit context such as cache contents). While real-time operating systems are capable of meeting these demands, they generally do so only by sacrificing the flexibility and protection mechanisms that have become a central element of modern computing systems.

One step toward achieving the necessary performance is the introduction of scheduling mechanisms that would allow continuous stream processing modules to be executed on a regular, timed basis, to prevent the accumulation of backlogs that can detract from the subjective quality of the multimedia data streams delivered to the end user. These scheduling mechanisms can be made visible to the application programmer through the performance range declarations discussed above and the system software can then use this information to allocate processing bandwidth accordingly.

Another essential component is providing a closer coupling between application software and system hardware. Current systems insert one or more software layers between the application process and the hardware to protect the hardware and multiplex its use among different applications. Emerging high speed networks have demonstrated that these intervening software layers can be greatly reduced if the hardware directly supports concurrent access by multiple application programs

A crucial element of the multimedia run time environment is the network protocols that provide application programmers with an appropriate abstraction of the communication capabilities of the underlying network or networks. Transport protocols must mask the peculiarities of particular networks from the applications above, while simultaneously allowing applications to obtain full use of the performance inherent in the available networks. They must provide the required low level support for multiway communication, including mechanisms for flow control and error recovery. Internetwork protocols must support delivery of data across diverse subnetworks, including conventional LANs, such as Ethernet, and emerging high speed networks, such as ATM.

2.5.2. Solution Approaches. To meet the technical challenges raised above, it is necessary to formulate novel solutions in the areas of programming systems, resource management and communication software. The following paragraphs outline our approaches in each area.

Programming systems. We have adopted a programming model for distributed multimedia systems based on the concept of *I/O abstraction* [77, 167]. In I/O abstraction, modules communicate implicitly through *published* data structures. Explicit linkages are established between data structures in different modules at run-time, through a separate module configuration process. These linkages are strongly typed, allowing the system to use the type information to facilitate efficient execution and can be one-to-one, one-to-many or many-to-many. The I/O abstraction model was designed to make it easier to reason systematically about the correctness of individual modules and their composition into larger systems. It is closely related to the I/O automaton model of Lynch and Tuttle [75, 76, 114] and has a strong theoretical foundation, facilitating the expression and verification of correctness properties.

The I/O abstraction model offers several other benefits for distributed multimedia computing, because of its declarative approach to communication. First, it promotes a programming style that actively facilitates the construction of complex multimedia applications from a set of simple components and the basic paradigm of interconnected modules maps directly to a natural graphical representation, making it easier for application programmers to understand the interactions among modules and facilitating the creation of programming tools using visualization. Because connections between modules are explicitly declared, it is natural to associate resource declarations with both processing modules and the communication linkages between them. This facilitates the allocation of processing modules to available resources by the operating system and makes the application's resource requirements available to the operating system for use in configuring the computing and communications bandwidth appropriately. Moreover, the quasi-static nature of the interconnections makes it possible to optimize these linkages in ways that are not readily possible in conventional programming environments. For example, if at run time, two communicating modules of a distributed system are allocated to different processors, data format transformations need be inserted between the two modules only if required by architectural differences in the two machines. It is also possible in this environment to perform more complex *inter-module optimizations* which combine different modules together to avoid communications overhead and reduce the need for context switching. Of course, to ensure memory protection, such radical optimizations will require extensive compile time analysis and may be applicable only to a restricted class of programs.

Resource Management. In order to allocate a collection of distributed resources among a large user population in a distributed multimedia system, the operating system must have knowledge of the available resources, including information about how each particular resource can be used and its current availability. This information will be maintained by the operating system in a distributed database, which can be accessed and modified from any location in the system. Using this information and knowledge of the communications structure and resource requirements of an application, the system will be able to allocate the system's resources suitably to meet each application's requirements.

Another aspect of resource management concerns the scheduling of the processing resources within a single computer system. This involves both the allocation of processing tasks to multiple processors and the sharing of a single processor by multiple tasks. As pointed out above, effective sharing of a single processor for multimedia applications, requires the ability to allocate processing to a given application module in small time slices. The basic scheduling mechanisms must be modified to support this style of scheduling in addition to the classical priority based scheduling for less time-sensitive applications. Making such fine-grained scheduling practical may require the use of inter-module optimization techniques discussed above, to eliminate or reduce the usual context-switching overhead.

Communication Software. The central role of communications in distributed multimedia systems make communication software a particularly crucial part of the system software complex. The role of communication software is to make the facilities of the underlying network available to the applications in the most efficient way possible, while simultaneously masking network differences and limitations from the upper layers. It should provide appropriate communication abstractions such as *segment streaming* [78, 80, 81, 157], allowing applications to operate without excessive concern for the underlying network idiosyncrasies. It should support operation across an internet of separately administered and/or technologically different networks [46, 116, 130, 131]. For connection-oriented networks, it must provide signaling mechanisms to set up and modify connections either in response to explicit application requests or in response to implicit requests (for example, in order to deliver a datagram over a connection-oriented network). The communication software must provide mechanisms for coping with network imperfections, including data loss and variable delay. It may provide different mechanisms for different classes of applications reflecting different error handling and flow control needs [83, 84, 131, 163].

2.6. Distributed Multimedia Applications

Distributed multimedia computing and communication can have a major impact on a wide range of different application areas. While there is a rapidly developing commercial interest in multimedia applications, ready-to-use multimedia products are unlikely to satisfy all the needs of applications that experience frequent requirement changes, demand high quality audio and video coupled with extensive computation, and which function in a distributed environment. Effective construction of such applications requires flexible tools that facilitate careful crafting of systems which harmonize evolving user needs with emerging technologies. The study of these kinds of applications will provide valuable insights into fundamental issues of distributed multimedia computing and will facilitate full integration of multimedia into scientific, engineering, and educational computing as compared to mere casual use of multimedia capabilities.

2.6.1. Multimedia Application Components. There are a number of general capabilities that we expect to be common to a broad class of multimedia applications. We plan to create these components and make them available in a form that will allow them to be easily re-packaged into custom applications for specific purposes.

- *Multimedia conferencing*, which combines n-way audio and video conferencing with collaborative work software providing shared access to text and graphics [40, 141, 142]. We will examine alternative approaches, including architectures which shift the balance between software and hardware and provide flexible handling of video.
- *Multimedia production system*, which allows interactive construction of hypermedia documents combining audio and video elements with text and graphics. This must include the ability to script a coordinated multimedia presentation, sophisticated mechanisms for audio and video processing, and mechanisms for specifying relationships among different media streams. We plan to extend our prior work on hypertext into the distributed multimedia environment [36, 37, 57, 58, 59, 60, 61, 62, 111, 112, 113, 153, 154, 155, 156].
- *Automated visualization*, which allows users to create visual representations of computational processes and natural phenomena using the concept of *declarative visualization*, introduced in the Pavane program visualization system [43, 44, 45, 143, 149]. We seek to extend these concepts to shared, interactive visualizations admitting multiple views.
- *Multimedia archive access*, which is the process of navigating through large repositories of multimedia information in order to locate particular items of interest. Mechanisms to support effective access include browsing, search and directory mechanisms, as well as anticipatory caching mechanisms that analyze user access patterns in order to prefetch multimedia objects.

2.6.2. Matching Multimedia Capabilities to Specific Applications. In order to test the utility of multimedia application concepts and determine what approaches are most effective, it is necessary to apply research ideas in specific application settings. The Center works in partnership with a variety of application groups at Washington University to create distributed multimedia applications tailored to each group's specific needs and evaluate those applications to determine how well they work in their target settings. Five specific projects have been identified as the principal testbed applications. The testbed applications serve as demonstration vehicles for Center technical developments, provide focus for the research activities, ensure thorough evaluation of the research results on very challenging problem areas of national importance, and facilitate rapid technology transfer across interdisciplinary boundaries and into the commercial arena.

In the remainder of this section, we present several specific applications that we plan to implement in our multimedia testbed. It should be noted that while the Center works with the application groups to help them implement their applications on the testbed, the application groups all have independent funding to pursue their primary scientific goals.

This creates a synergistic relationship that provide maximum leverage for the Center's research sponsors while providing a valuable service to the application groups.

Advanced cooperative software engineering. The application of multimedia distributed computing to software development environments provides an ideal testbed for the Center research activities. Software development involves all forms of multimedia communication from informal interpersonal communication to sophisticated collaborative work on shared documents and challenges us to optimize both individual performance within the confines of a personal workspace as well as team performance within the context of a project.

Our approach to cooperative software development involves treating software as a shared multimedia document consisting of formal requirements and design specifications, code, annotations, sample test cases, visualizations, tutorial explanations, documentation, etc. Its development requires specialized forms of cooperative work, the integration of activities initiated by people and tools, task management and monitoring, and informal communication among the members of the design team. The final product will be a sophisticated multimedia publication which serves as an environment for subsequent evolution of the product, as a presentation vehicle for potential users and, ideally, also as a component of systems in which the software may be used. This research will seek to integrate formal design methods [47, 73, 144, 145, 148] and novel program derivation techniques [147, 150] with conventional programming tools and powerful communication tools, including program visualization [43, 44, 45, 143, 146, 149].

Biomedical imaging and telemedicine. Biological systems are intrinsically complex; their behavior and properties can be understood only in context and biological responses are innately multi-faceted. In this setting, multimedia interfaces that maximize the rate of information/idea transfer to users and among users are essential. For the research scientist or physician to effectively probe biomedical phenomena, information from a variety of sources must be gathered and integrated. The volumes of data being generated and the requirement for data integration make distributed multimedia capabilities particularly useful. An application of distributed multimedia capabilities is 3-D microscopy where biologists, engineers, and scientists are working together to provide real-time 3-D visualization of microscopic living specimens, as illustrated in Figure 1. Expertise in optics, image processing, biology, display and visualization are all required to produce effective systems that advance our understanding of biological science [7, 50, 54, 72, 118, 119, 134, 135, 136, 137, 182]. Support for distant collaboration is needed to bring together individuals from geographically remote sites and departments. The microscopy technology presents another particularly exciting application of multimedia in its application to neuroscience. Current optical microscopy technology can easily image the nervous system at a cellular resolution, and computational image registration techniques allow slices to be assembled into 3-D structures. The tools are therefore in place to create a cellular resolution map of the entire nervous system [3, 87, 88, 125]. The resulting data set would, however, be massive (many terabytes). Making effective use of such a data resource will depend critically on integrated multimedia browsing and analysis tools, high bandwidth networking, and very high performance computing.

High-speed, public network telecommunications services are critical to current efforts in the healthcare sector to improve access to medical care, to improve the quality of care and to support cost reduction programs. Research to develop new technologies that enable the delivery of care outside the major medical center environment is needed. ATM-based transmission of digital medical video to support the remote real-time visualization of multiple-frame diagnostic studies such as fluoroscopy and ultrasound, can reduce the need for a physician be present at the examination [20, 40, 41, 42, 63, 93, 141, 140, 142]. We propose to show that remote, real-time, multimedia-based visualization can reduce personnel (physician) costs while increasing access to expert physician practitioners. A related effort is being planned in Radiation Oncology which is used clinically and is continuing development of a 3-D treatment planning system [138, 151, 152] that is strongly image-oriented and supports treatment planning for improved dose distribution.

Education. The application of computing technology to improve elementary education has been an ongoing theme of two major projects at Washington University. The Show-and-Tell [35, 97] and Kumon machine [66, 102] projects have investigated young children's ability to learn programming in the context of a dataflow paradigm and mathematics using the notebook metaphor. Building directly on these successful experiments, we propose to study the impact of multimedia capabilities on the communication process among the four segments of a typical educational community (students, parents, teachers, and administrators) during a basic educational cycle involving the following four steps: diagnostic testing, identification of individual educational needs, customized curriculum design and production, and teaching.

The remote delivery of classroom instruction appears to be ideally suited to advanced multimedia applications. We expect that the availability of high quality, two way audio and video will dramatically improve the effectiveness of remote instruction systems. We plan to explore the application of these techniques to deliver university-based instruction to local industry and to provide links between area high schools in order to allow students to receive specialized instruction (such as advanced placement science classes, for example) from a remote location. Such experiments could have important implications nationally, possibly reducing the cost of education, equalizing availability of educational resources among school districts, and improving the quality of the workforce.

Image and data access in the planetary sciences. Washington University's Department of Earth and Planetary Sciences has been deeply involved in various planetary exploration missions, such as the Magellan Project and has been responsible for planetary data archives [1, 9, 126, 127]. The purpose of this research effort is to create an understanding of the planets and their environments at a level of detail never possible before. To succeed in this investigation it is necessary for planetary scientists to collaborate with their peers on an unprecedented scale. With the help of gigabit networks, high-performance computers, state-of-the-art archives, and visualization engines, we expect to be able to construct an environment that allows coordinated data analysis in almost real time.

Distributed access to digital libraries. Washington University has been in the forefront of efforts to exploit hypertext technology [36, 37, 57, 58, 59, 60, 61, 62, 111, 112, 113, 153, 154, 155, 156] and is actively seeking to extend this work to the domain of distributed multimedia systems. The StudySpace project is a new initiative aimed at building an environment for highly focused, task-oriented (individual and group) learning using digital library information. Emphasis is placed on the integration of emerging technologies and their deployment in the workaday world of medical students, faculty, and researchers at the Washington University School of Medicine. The overarching question that StudySpace seeks to address is: "How can advanced information systems technologies be used to facilitate learning that takes place over large quantities of rapidly changing multimedia information acquired from widely distributed sources?" The project provides a testbed for the development of systems and theory to help answer this question. We anticipate that the StudySpace framework will enhance the teaching faculty's effectiveness by enriching the dialog between students and teachers; it will improve teaching efficiency by providing a comprehensive framework for the creation, annotation, and delivery of digital coursework and library materials.

3. Industrial Partnership Program

Washington University has a history of successful collaborative research with industry. These collaborations have given us a deep appreciation of the R&D priorities of industrial organizations and have allowed us to help companies to achieve their objectives, while simultaneously allowing us to explore a variety of fundamental engineering research questions.

To help support the research program described here and to enable close collaboration with participating companies, Washington University has created an Industrial Partnership Program. Companies have the option to participate either as program *Supporters* or *Sponsors* and can enroll in the program on an ongoing basis, renewing their commitment each year. Companies that commit for two years are given a 20% reduction in the annual membership fees. This inducement is designed to encourage longer term participation in order to facilitate financial planning.

The annual fee for program Supporters is $75,000 ($60,000 with a two year commitment), which entitles the company to the following benefits.

- Timely access to all technical developments including copies of all publications, technical reports, theses and progress reports mailed directly to a designated list of three representatives in the company, plus email announcements of electronically retrievable publications, sent to a Supporter-maintained email distribution list.
- Unlimited free attendance at semi-annual workshops reporting progress on Center research and relating that research to work taking place in other academic and industrial research laboratories and seeking insight into the important directions for future efforts.

- Access to experimental systems created in the program. This can include access at the University, software distributions and at-cost purchase of experimental hardware.
- The right to send short term ($\leq$ 1 week) visitors to the Center at no cost, and long term visitors at an added cost of $30,000 annually per visitor for basic facilities. Visitors may enroll in courses offered by program faculty at no cost.
- Participation in the Center's summer internship program which matches undergraduate and graduate students with internship opportunities within companies.
- One week of consulting by program faculty and staff to assist Supporter in taking advantage of Center research products.
- A royalty-free license for internal use of all patents, software and hardware designs created by faculty, staff and graduate students associated with the research program during the period of Supporter's participation, plus the right to license patents and technical developments for commercial use (non-exclusive, terms to be negotiated).
- A credit against future royalty payments on licensed technical developments equal to the total research investment.

The annual fee for program Sponsors is $250,000 ($200,000 with a two year commitment), which entitles the company to the following benefits.

- Timely access to all technical developments including copies of all publications, technical reports, theses and progress reports mailed directly to a designated list of ten representatives in the company, plus email announcements of electronically retrievable publications, sent to a Sponsor-maintained email distribution list.
- Unlimited free attendance at semi-annual workshops reporting progress on program research and relating that research to work taking place in other academic and industrial research laboratories and seeking insight into the important directions for future efforts.
- A seat on the Industrial Partnership Program's Steering Committee which meets at least once a year and advises the program director on general management issues, selection of future research directions, budget priorities, dissemination of research results and recruitment of additional sponsors.
- Access to experimental systems created in the program. This can include access at the University, software distributions and at-cost purchase of experimental hardware.
- The right to send both short and long term visitors to the Center at no added cost. Visitors may enroll in courses offered by program faculty at no cost.
- Participation in the Center's summer internship program which matches undergraduate and graduate students with internship opportunities within companies. For one student per year, the program will pay the student's stipend during the internship.

- Five weeks of consulting by program faculty and staff on subjects of interest to Sponsor. Results of the consulting efforts are retained by the Sponsor.
- A royalty-free license for internal use of all patents, software and hardware designs created by faculty, staff and graduate students affiliated with the program during the period of Sponsor's participation in the program, plus the right to license patents and technical developments for commercial use (non-exclusive, terms to be negotiated).
- The right to license specified background patents and other technical developments.
- A credit against future royalty payments on licensed technical developments (including background technical developments) equal to twice the total research investment.

The Industrial Partnership Program includes several mechanisms designed to maximize communication between the program's faculty, staff and students and its industrial partners and to facilitate collaboration. First, we seek to maximize the flow of technical information within our industrial partners' organizations by creating direct lines of communication to as many of the partners' technical staff as possible. In our experience, this is beneficial in allowing the unplanned, serendipitous connections that frequently initiate productive collaborations. Second, industrial partners are encouraged to send long term visitors to be active participants in the program's research activities, participate fully in the intellectual life of the program and report on all program activities to their companies. Third, undergraduate and graduate students are urged to spend one or more summers as interns with our industrial partners, giving students a first-hand opportunity to learn about the companies, give the companies a chance to better understand a portion of the research program and become better acquainted with an individual who they might wish to recruit upon graduation. Finally, the consulting opportunities with faculty and research staff can give industrial partners mechanisms through which to confidentially explore possible applications of the program's research results while receiving expert advice from those who best understand the work.

These mechanisms will help ensure that our industrial partners have effective input into the research processes and are well-prepared to use its results. It strikes an appropriate balance between the University's need to engage in unfettered, open collaborative research and industry's need to use research results in competitive products.

4. Personnel

Program Director
– Jonathan Turner (design/analysis of switching systems)

Affiliated Faculty
– G. James Blaine (electronic radiology and telemedicine)
– Jerome Cox, Jr. (multimedia and medical imaging)
– Kenneth J. Goldman (programming environments for multimedia systems)
– Paul Min (telecom network management and control)

- Guru Parulkar (protocols and host interfacing)
- Dave Richard (multimedia workstation design)
- George Varghese (distributed network algorithms)

Collaborators

- Ron Cytron (Department of Computer Science)
- Martin Dubetz (Academic Computing and Networking)
- Mark Franklin (Computer and Communications Research Center)
- Mark Frisse (Medical School Library)
- Dan Kimura (Department of Computer Science)
- James McNally (Department of Biology)
- Michael Miller (Department of Electrical Engineering)
- Catalin Roman (Department of Computer Science)
- Fred Rosenberger (Institute for Biomedical Computing)
- John Schnase (Medical School Library)
- Douglas Schmidt (Department of Computer Science)
- David States (Institute for Biomedical Computing)

Research Staff

- Tom Chaney (hardware design)
- Pierre Costa (hardware design & supervision)
- Ken Cox (software design)
- John DeHart (software design & supervision)
- Maynard Engebretson (hardware design)
- Andy Fingerhut (system architecture and testing)
- Margaret Flucke (hardware design)
- Brian Gottlieb (software design)
- Craig Horn (hardware design)
- Mike Richards (hardware design)
- Randy Richards (hardware construction and testing)
- Hossein Saidi (multi-channel switching)

Visitors

- Peter Chung (Goldstar)
- Taeck Geun Kwon (Goldstar)

DSc students

- Ehab Al-Shaer (networking software)
- Milind Buddhikot (high speed host interface design)
- Chuck Cranor (IPC for multimedia)
- Zubin Dittia (high performance transport protocols)
- Raman Gopalakrishnan (network-wide traffic control)
- Adiseshu Hari (networking software)
- Saied Hosseini-Khayat (multimedia workstation design)
- Seyed Mahdavian (optimal data transfer in ATM nets)
- Nader Mirfakhraei (switching system design)
- Christos Papadopoulos (multimedia transport protocols)

- Einir Valdimarrson (performance of switching systems)
- Dakang Wu (ATM signaling)
- Peter Yan (asynchronous high speed switching)

MS students

- Bob Akl (multichannel switching)
- Jim Barta (networking hardware design)
- Andres Blazquez
- Alex Chandra (analysis of copy networks)
- Joy Chatterjee (multichannel switching)
- Adam Costello (high performance network protocols)
- Rob Jackson (network design/configuration)
- Anurag Maunder (routing in multirate networks)
- Jyoti Parwatikar (signaling software)
- Chan Park (multichannel switching)
- Yang Qian (multimedia application software)
- M. Shreedhar (high performance network protocols)
- Chris White (multichannel switching)

5. Facilities and Research Equipment

The Computer and Communications Research Center and the Departments of Computer Science and Electrical Engineering have extensive facilities to support the research program.

> *Computing Equipment.* Approximately 50 workstations and file servers, primarily Suns, but also NeXTs, DEC, Silicon Graphics, Macintosh and PC. Sun SparcCenter 2000 multiprocessor computer server with 20 processors.
>
> *Network Equipment.* Ethernet connectivity to every office with ATM installation to all faculty offices plus selected other locations now underway. Eight SynOptics ATM switches, 20 MultiMedia Explorers (MMX) for full-rate bidirectional multimedia support. Thirty ATM adapter cards for Sun Sparc Stations. In addition, we have a prototype ATM network comprising four ATM switches plus related multimedia interfaces. While most of the prototype equipment will be decommissioned shortly, parts are being adapted to operate on the new ATM network.
>
> *Special Facilities.* Our Computer Visualization Laboratory includes a complete video editing facility to support the creation of high quality computer visualizations. The department has outstanding facilities for the design, fabrication and testing of electronic systems, including a substantial suite of high speed electronic test equipment, leading commercial tools for integrated circuit and PC board design and the experienced staff needed to maintain and properly use these facilities.

In addition, our key application collaborators in the Electronic Radiology Laboratory, the Institute for Biomedical Computing, the Office of the Network Coordinator and the Electronic Signals and Systems Laboratory have additional facilities that can be used to further the program's activities. These include:

Computing Equipment. 100 workstations and servers (Sun, DEC, Silicon Graphics), DECmpp 12000/Sx model 200 with 4096 MP-2 processors; one AMT-DAP-510 with 1024 processors, one AMT DAP-610, six node DEC Alpha cluster integrated with DEC Gigaswitch, dual processor Stardent Titan 3000, three Image Recognition Concepts Processors, 10 medical display units (Imlogix), one high resolution display unit (AVP-Megascan, 5 Megapixel display), Image Archival and Retrieval Systems (ATM image server, 6.5 Gbytes; DEC 7000, 12 Gbytes; Kodak IARS with 5 1/4 inch WORM Jukebox and 14 inch WORM optical).

Network Equipment. Three FDDI subnetworks; 45 Cisco routers; 20 DEC bridges; two full-duplex microwave radio links (MA-COM 23VXLAN, 10 Mb/s each); two broadband CATV backbone networks; extensive campus fiber plant; ten T1 circuits for connection to Internet and metropolitan area sites.

References

[1] Albee, A., R. Arvidson, F. Palluconi. "Mars Observer Mission," *J. Geophys. Res.*, 97, 7665-7680, 1992.

[2] Akhtar, Shahid. "Congestion Control in Fast Packet Networks," Washington University Electrical Engineering Department, MS thesis, 11/87.

[3] Amit, Y., U. Grenander and M. Piccioni. "Structural Image Restoration through Deformable Templates," *Journal of the American Statistical Association*, 6/91.

[4] Anderson, James, Zubin Dittia and Guru Parulkar. "Persistent Connections in High Speed Internets," Proceedings of the IEEE Globecom 91, 12/91.

[5] Apte, A. and T. D. Kimura, T.D. "A Comparison Study of the Pen and the Mouse in Editing Graphic Diagrams," *Proceedings of 1993 IEEE Symposium on Visual Languages (VL'93)*, Bergen, Norway, 8/93, pp. 352–357.

[6] Apte, A., V. Vo, V. and T. D. Kimura. "Recognizing Multistroke Geometric Shapes: An Experimental Evaluation," *Proceedings of UIST 93*, Atlanta, 11/93.

[7] Armbrust, E. V., P. J. Ferris and U. W. Goodenough. "A Mating Type-Linked Gene Cluster Expressed in Chlamydomonas Zygotes Participates in the Uniparental Inheritance of the Chloroplast Genome," *Cell*, 1993.

[8] Arutaki, Akira. "Trajectory Method for Evaluating Periodic Bursty Traffic," Washington University Computer Science Department, WUCS-89-53, 1989.

[9] Arvidson, R.E., V.R. Baker, C. Elachi, R.S. Saunders, J.A. Wood, "Magellan: Initial Analysis of Venus Surface Modification," *Science*, 1991.

[10] Awerbuch, Baruch and George Varghese. "Distributed Program Checking: a Paradigm for Building Self-Stabilizing Distributed Protocols," *Proceedings of the IEEE Symposium on the Foundations of Computer Science*, 10/91.

[11] Awerbuch, Baruch, Boaz Patt and George Varghese. "Self-Stabilization by Local Checking and Correction," *Proceedings of the IEEE Symposium on the Foundations of Computer Science*, 10/91.

[12] Awerbuch, Baruch, Shay Kutten, Boaz Patt, Yishay Mansour, and George Varghese, "Time Optimal Self-Stabilizing Synchronization," *Proceedings of the ACM Symp. on Theory of Computing*, 1993.

[13] Balentine, Jennifer, Umesh Berry and Anne Johnstone. "The DIM System: WOZ Simulation Results — Phase I," Washington University, Computer Science Department, WUCS-92-20, 1992.

[14] Balentine, Jennifer, Anne Johnstone and David Mathias. "The DIM System: An Empirically Derived NLP System," Washington University, Computer Science Department, WUCS-92-21, 1992.

[15] Berry, Umesh and Michael Groner. "The DIM System: Plan Recognition in Spoken Dialogues," Washington University, Computer Science Department, WUCS-92-22, 1992.

[16] Berry, Umesh and Anne Johnstone. "The DIM System: Turn-Taking in Dyadic Telephone Dialogues," Washington University Computer Science Department, WUCS-93-33, 1993.

[17] Bhargava, Deepak, "Synchronization and Collaboration in Multimedia Applications," MS Thesis, Department of Computer Science, Washington University. 12/92.

[18] Bianchi, Giuseppe and Jonathan Turner. "Improved Queueing Analysis of Shared Buffer Switching Networks," *Proceedings of Infocom '93*, 3/93, pp. 1392-1399.

[19] Bi, Haifeng. "Queueing Analysis of Buffered Switching Networks," Washington University Computer Science Department, MS thesis, 8/90.

[20] Blaine, G.J., Moore, S.M., Cox, J.R.,Whitman, R.A., Teleradiology via Narrow-band ISDN and JPEG Image Compression, Journal of Digital Imaging Vol 5, No 3, pp. 156-160, 8/92.

[21] Bojanczyk, A.W. and T. D. Kimura, T.D.. "A Systolic Parsing Algorithm for a Visual Programming Language," *Proceedings of FJCC'86*, Dallas, 11/86, pp. 48-55.

[22] Bovopoulos, A. D., "Resource Allocation Algorithms for Packet Switched Networks," *First ORSA Telecommunications SIG Conference, "Operations Research in Telecommunications,"* 3/90.

[23] Bovopoulos, Andreas and Aurel Lazar. "Optimal Resource Allocation for Markovian Queueing Networks: the Complete Information Case," *Stochastic Models*, 1991.

[24] Bovopoulos, A. D., R. Gopalakrishnan and S. Hosseini, "SYMPHONY: A Hardware, Operating System and Protocol Processing Architecture For Distributed Multimedia Applications," Washington University Computer Science Department Technical Report WUCS-93-06, 1993.

[25] Bovopoulos, A. D. and Lazar, A. A., "The Effect of Delayed Feedback Information on Network Performance," *Annals of Operations Research,* Vol. 36, 1992, pp. 101-124. 1992,

[26] Bovopoulos, A. D., "Performance Evaluation of a Traffic Control Module for ATM Networks," *Proceedings of INFOCOM,* 5/92, pp. 469-478.

[27] Bovopoulos, A. D., "Optimal Burst Level Admission Control in a Broadband Network" *STOCHASTIC MODELS,* Vol. 8, No 2, 1992, pp. 289- 305, and presented at *The First International Conference of Computer Communications,* San Diego, California, 6/92.

[28] Bubenik, Richard and Jonathan S. Turner. "Performance of a Broadcast Packet Switch." *IEEE Transactions on Computers*, 1/89.

[29] Bubenik, Richard, John DeHart and Mike Gaddis, "Multipoint Connection Management in High Speed Networks," *Proceedings of Infocom*, 4/91.

[30] Buddhikot, M.M., Kapoor, S., Parulkar, G.M., "Design of an ATM-FDDI Gateway," *Journal of Internetworking: Research and Experience*, 3/93.

[31] Cheriton, D., "VMTP: A Transport Protocol for the NeXT Generation of Computer Systems", *SIGCOMM '86 Symposium: Communications Architectures and Protocols Computer Communication Review*, Vol. 16, No. 3, ACM, New York, pp. 406-415, 1986.

[32] Cheriton, D., "The V Distributed Operating System," *Communications of ACM,* Vol. 31, pp. 314-333, 3/88.

[33] Chesson, G., "Protocol Engine Design," *Proceedings of the Usenix Conference*, 1986.

[34] Chesson, G., et. al., "XTP Protocol Definition," Revision 3.1, Protocol Engines, Inc., PEI 88-13, Santa Barbara, California, 1988.

[35] Choi, J.W. and T. D. Kimura. "A Compiler for a Two-dimensional Programming Language," *Proceedings of ACM Conference on Personal and Small Computers*, San Francisco, 12/86, pp. 72-80.

[36] Cousins SB, Silverstein JC, Frisse ME. "Query Networks for Medical Information Retrieval - Assigning Probabilistic Relationships." In R. Miller, ed., *Proceedings of the Fourteenth Annual Symposium for Computer Applications in Medical Care*, Washington, D.C., IEEE Computer Society Press 1990; 14:800–804.

[37] Cousins, S.B., W. B. Chen, M. E. Frisse. "Stochastic simulation algorithms for belief networks: A tutorial." *Artificial Intelligence in Medicine* 1993 (to appear).

[38] Cox, Jerome R. and Jonathan Turner. "Project Zeus: Design of a Broadband Network and its Application on a University Campus," Washington University Computer Science Department, WUCS-91-45, 1991.

[39] Cox, Jerome R., Jr., Mark Franklin, Guru Parulkar and Jonathan Turner. "A Gigabit Local ATM Testbed for Multimedia Applications," ARPA contract, 9/93-8/96.

[40] Cox, Jerome R. Jr., Mike Gaddis and Jonathan Turner. "Project Zeus: Design of a Broadband Network and its Application on a University Campus," *IEEE Network*, March 1993, pp. 20-30.

[41] Cox, J.R., Moore, S.M., Whitman, R.A., Blaine, J.G., Jost, R.G., Karlsson, L.M., Monsees, T.L., Hassen, G.L., and David, T.C., "Rapid Display of Radiographic Images", *SPIE Proceedings, Vol. 1446, Medical Imaging V: PACS Design and Evaluation*, pp. 40-51, 2/91.

[42] Cox, J.R., Muka, E., Blaine, G.J., Moore, S.M., Jost, R.G., "Considerations in Moving Electronic Radiography into Routine Use," *IEEE Journal on Selected Areas in Communications*, Vol. 10, No. 7, pp. 1108-1120, 9/92.

[43] Cox, K. C. and Roman, G.-C., "Visualizing Concurrent Computations," *Proceedings of the IEEE Workshop on Visual Languages*, 10/91, pp. 18-24.

[44] Cox, K. C., and Roman, G.-C., "Abstraction in Algorithm Animation," *Proceedings of the IEEE Workshop on Visual Languages*, 9/92, pp. 18-24.

[45] Cox, K. C., and Roman, G.-C., "A Characterization of the Computational Power of Rule-Based Visualization," Washington University, Department of Computer Science WUCS-93-15. (To appear in *Journal of Visual Languages and Computing*.)

[46] Cranor, C., Parulkar, G.M., "An Implementation Model for Connection-oriented Internet Protocols," *Proceedings of INFOCOM*, 3/93.

[47] Cunningham, H. C. and Roman, G.-C., "A UNITY-Style Programming Logic for a Shared Dataspace Language," *IEEE Transactions on Parallel and Distributed Systems* 1, No. 3, 7/90, pp. 365-376.

[48] DeHart, John. "CMAP/CMIP Scenarios: A Tutorial," Washington University Computer Science Department, ARL-90-03, 1990.

[49] Dehart, John. "BPN Connection Management Access Protocol Specification," Washington University Computer Science Department, ARL-89-06, 1989.

[50] Felleman, D.J. and Van Essen, D.C., "Distributed Hierarchical Processing in the Primate Cerebral Cortex," *Cerebral Cortex*, Vol. 1, pp. 1-47, 1991.

[51] Fingerhut, Andy. "Designing Communication Networks with Fixed or Nonblocking Traffic Requirements," Washington University Computer Science Department, WUCS-91-55, 1991.

[52] Fingerhut, A., "Algorithms for Designing Nonblocking Communication Networks with General Topologies," Washington University Computer Science Department, WUCS-TM-92-05, 1992.

[53] Fingerhut, A., "Approximation Algorithms for Configuring Hierarchical Nonblocking Communication Networks," Washington University Computer Science Department, WUCS-93-19, 1993.

[54] Fox, P.T., Miezin, F.M., Allman, J.M., Van Essen, D.C., and Raichle, M.E., "Retinotopic Organization of Human Visual Cortex Mapped with Positron-Emission Tomography," *Journal of Neuroscience* Vol. 7, pp. 913-922, 1987.

[55] J. Frailong et al. SPARCcenter 2000: Multiprocessing for the 90's. *Proc. COMPCON*, 1993.

[56] J. Frailong et al. The Next-Generation SPARC Multiprocessing System Architecture . *Proc. COMPCON*, 1993.

[57] Frisse, M. E. "Searching for Information in a Hypertext Medical Handbook," *Communications of the Association for Computing Machinery* 1988; 31(7): 880–886.

[58] Frisse M. E., S. B. Cousins. "Information Retrieval from Hypertext: Update on the Dynamic Medical Handbook Project," *Proceedings of Hypertext 89*, ACM Press 1989, 199–212.

[59] Frisse, M. E., S. B Cousins. "Guides for Hypertext: An Overview," *Artificial Intelligence in Medicine* 1990; 2:303–314.

[60] Frisse M. E., S. B. Cousins and S. W. Hassan SW. "WALT: A Research Environment for Medical Hypertext," *Proceedings of the Third ACM Conference on Hypertext*, San Antonio, ACM Press 1991; 389–394.

[61] Frisse, M. E., S. B. Cousins. "Models for Hypertext," *Journal of the American Society for Information Science*, 1992, 43(2):183–191.

[62] Frisse, M. E., K. Marrs, P. A. Schoening. "A Method for Publishing Genomic Maps," *Proceedings of the Sixteenth Annual Symposium for Computer Applications in Medical Care*, Washington, D.C., McGraw Hill, 1992; 16: 376-382.

[63] Fuhrmann, D., Baro, J., and Cox Jr. J.R., "Experimental Evaluation of Psychophysical Distortion Metrics for JPEG-Encoded Images" *Human Vision, Visual Processing, and Digital Display IV*, Bernice E. Rogowitz and Jan P. Allebach, Editors, Proc. SPIE 1913, 2/93.

[64] Fukunaga, A., W. Pree and T. D. Kimura. "Functions as Data Objects in a Data Flow Based Visual Language," *Proceedings of 21st ACM Annual Computer Science Conference (CSC'93)*, Indianapolis, 2/93, pp 215–220.

[65] Fukunaga, A., T. D. Kimura, and W. Pree. "The Object-Oriented Design and Implementation of a Data Flow Visual Programming Language Interpreter," *Proceedings of 1993 IEEE Symposium on Visual Languages (VL'93)*, Bergen, Norway, 8/93, pp. 134 - 141.

[66] Fuller, Jr., T.H. "The Kumon Approach to Learning Mathematics: An Educator's Perspective," Washington University Computer Science Department, WUCS-91-49, 1991.

[67] Fuller, Jr., T.H. and T. D. Kimura. "Supervised Competitive Learning, Part I: SCL with Backpropagation," in Dagli, C, et al. (ed.) *Intelligent Engineering through Artificial Neural Networks*, Volume 2. New York: ASME Press, 11/92, pp. 185-190.

[68] Fuller, Jr., T.H., Ce Wang and T. D. Kimura. "Supervised Competitive Learning, Part II: SCL with Fuzzy Logic," in Dagli, C, et al. (ed.) *Intelligent Engineering through Artificial Neural Networks*, Volume 2. New York: ASME Press, 11/92, pp. 351-356.

[69] Fuller, Jr., T.H. and T. D. Kimura. "Supervised Competitive Learning," To appear in *Journal of Intelligent Material Systems and Structures and Mathematical and Computer Modelling.*

[70] Gaddis, Michael E. "Prototype Connection Management: a Progress Report," Washington University Applied Research Laboratory, ARL-89-01, 1989.

[71] Gaddis, Mike. "FPP Packet Formats and Link Level Protocol Descriptions," Washington University Computer Science Department, ARL-89-07, 1989.

[72] Gallant, J.L., Braun, J., and Van Essen, D.C., "Selectivity for Polar, Hyperbolic, and Cartesian Gratings in the Macaque Visual Cortex," *Science*, Vol. 259, pp. 1000-1003, 1993

[73] Gamble, R. F., Roman, G.-C., Ball, W. E., and Cunningham, H. C., "Applying Formal Verification Methods to Rule-based Programs," Washington University, Department of Computer Science WUCS-91-1. (To appear in *International Journal of Expert Systems Research and Applications.*)

[74] Gillett, W.D. and T. D. Kimura. "Parsing Two-dimensional Languages," *Proceedings of Tenth International COMPSAC'86*, Chicago, 10/86, pp. 472-477.

[75] Goldman, Kenneth and Nancy Lynch. "Modelling Shared State in a Shared Action Model," *Proceedings of the IEEE Symposium on Logic in Computer Science*, 6/90.

[76] Goldman, Kenneth. "A Compositional Model for Layered Distributed Systems," *Proceedings of the International Conference on Concurrency Theory*, 8/91.

[77] Goldman, Kenneth, Michael Anderson and Bala Swaminathan. "The Programmers' Playground: I/O Abstraction for Heterogeneous Distributed Systems," To appear in *Proceedings of the 27th ACM-IEEE Hawaii International Conference on Systems Sciences*, 1/93.

[78] Gong, Fengmin. "Segment Streaming for Efficient Pipelined Televisualization," Washington University Computer Science Department, WUCS-90-39, 1990.

[79] Gong, Fengmin, Zubin Dittia and Guru Parulkar. "A Recurrence Model for Asynchronous Pipeline Analysis," Washington University Computer Science Department, WUCS-91-14, 1991.

[80] Gong, Fengmin, Zubin Dittia and Guru Parulkar. "A Novel Interprocess Communication Paradigm for Pipelined Televisualization," *Proceedings of Workshop on Computer Graphics in a Network Environment*, 7/91.

[81] Gong, F. "A Transport Solution for Pipelined Network Computing," Doctoral Dissertation, Department of Computer Science, Washington University, 12/92.

[82] Gong, F., and Parulkar, G., "Segment Streaming for Efficient Pipelined Televisualization," *Proceedings of IEEE Military Communications Conference,* 1992.

[83] Gong, F. and Parulkar, G. "An Application-Oriented Error Control Scheme For High Speed Networks," To appear in *ACM/IEEE Transactions on Networking,* 1994.

[84] Gong, F., and Parulkar, G., "A Two Level Flow Control Scheme for a High Speed Transport Protocol," to appear in *Journal of High Speed Networks*, 1994.

[85] R. Gopalakrishnan and A.D. Bovopoulos, "Design of a Multimedia Applications Development System," Washington University Department of Computer Science, WUCS-92-27, 1992.

[86] R. Gopalakrishnan, and Andreas D. Bovopoulos "A Protocol Processing Architecture for Networked Multimedia Computers," Washington University Department of Computer Science, WUCS-93-07, 1993.

[87] Grenander, U., Y. Chow and D. Keenan. *Hands: A Pattern Theoretic Study of Biological Shapes*, Springer-Verlag, New York, 1990.

[88] Grenander, U. and M. I. Miller. "Jump-diffusion Processes for Abduction and Recognition of Biological Shapes," *Monograph of the Electronic Signals and Systems Research Laboratory*, Washington University, 1991.

[89] Haserodt, Kurt and Jonathan Turner. "An Architecture for Connection Management in a Broadcast Packet Network," Washington University Computer Science Department, WUCS-87-3, 1987.

[90] Imase, Makoto and Bernard Waxman. "Worst-case Performance of Rayward-Smith's Steiner Tree Heuristic," *Information Processing Letters*, 12/8/88.

[91] Imase, Makoto and Bernard Waxman. "The Dynamic Steiner Tree Problem," Washington University Computer Science Department, WUCS-89-11, 1989.

[92] Johnstone, Anne, Umesh Berry and Tina Nguyen. "The DIM System: WOZ Simulation Results — Phase II," Washington University Computer Science Department, WUCS-93-34, 1993.

[93] Jost, R.G., Blaine, G.J., Moore, S.M., Whitman,, R.A., Leith,, T.R., Wimmer, M.S., Muka, E., "Primary Interpretation of ICU Radiographs via Softcopy Display," *Medical Imaging VI: PACS Design and Evaluation,* R. Gilbert Jost, Editor, Proceedings of SPIE 1654, pp. 416-423, 1992.

[94] Kapoor, Sanjay and Guru Parulkar. "Design of an ATM-FDDI Gateway," *Proceedings of SIGComm*, 9/91.

[95] Khakoo, Shabbir. "Improved Search Algorithms for Video Codecs," Washington University Electrical Engineering Department, MS thesis, 6/88.

[96] Kimura, T.D. "A Neuron Model with Inhibitory Field," Invited talk, OSU/OCATE Workshop on Parallel Processing, Portland, Oregon, 3/88.

[97] Kimura, T.D., Choi, J.W. and Mack, J.M. "Show and Tell: A Visual Programming Language," Invited paper in Visual Computing Environments, E.P. Glinert (ed.), IEEE Computer Society Press Tutorial , Washington, D.C., 1990, pp 397-404.

[98] Kimura, T.D., "Visual Programming by Transaction Network," Visual Computing Environments, E.P. Glinert (ed.), IEEE Computer Society Press Tutorial , Washington, D.C., 1990, pp 124-130.

[99] Kimura, T.D. "An Algebraic Proof for Backpropagation in Acyclic Neural Networks," *Proceedings of International Joint Conference on Neural Networks*, San Diego, CA, 6/90, pp 554-557.

[100] Kimura, T.D. "Silicon Paper and A Visual Interface for Neural Networks," *Proceedings of 1990 IEEE Workshop on Visual Languages*, Chicago, IL, 10/90, pp. 241-246.

[101] Kimura, T.D. "Pen-based User Interface," Panel session chairman's position paper, *Proceedings of 1991 IEEE Workshop on Visual Languages*, Kobe, Japan, 10/91, pp. 168-173.

[102] Kimura, T.D. "Learning Math with Silicon Paper," Washington University Computer Science Department, WUCS-92-11, 1992.

[103] Kimura, T.D. "Hyperflow: A Uniform Visual Language for Different Levels of Programming," *Proceedings of 21st ACM Annual Computer Science Conference (CSC'93)*, Indianapolis, 2/93, pp. 209 - 214.

[104] Kimura, T.D. "Potentials and Limitations of Pen-Based Computers," Panel session chairman's position paper, *Proceedings of 21st ACM Annual Computer Science Conference (CSC'93)*, Indianapolis, 2/93, pp. 536 - 537.

[105] Kimura, T.D. and Brown, T.B. "BoxGraph: A Two-Dimensional Visual Computation Model," Submitted to *ETH Conference on Programming Languages and System Architectures*, Zurich, 3/94.

[106] Kumar, A., Fuhrmann, D., Frazier, M., and Jawerth, B., "A New Transform for Time-Frequency Analysis", IEEE Transactions on Signal Processing, Vol. 40, No. 7, pp. 1697-1707, 6/92.

[107] Kumar, A., and Fuhrmann, D., "The Frazier-Jawerth Transform", Proceedings of ICASSP 90 (Albuquerque, NM), pp. 2483-2486, 4/90.

[108] Kumar, Lakshmana. "A 3-Tier Fairness Protocol: A Proposed Solution for the Unfairness Problem in DQDB MANs," Washington University Computer Science Department, MS thesis, 8/91.

[109] Kumar, L. and Bovopoulos, A. D., "An Access Protection Scheme for Heavy Load Unfairness in DQDB," *Proceedings of INFOCOM*, 5/92, pp. 190-199.

[110] Kumar, L. and Bovopoulos, A. D., "An Access Protection Scheme for Heavy Load Unfairness in DQDB," *Proceedings of INFOCOM*, 5/92, pp. 190-199.

[111] Leggett, J.L. and J. L. Schnase. "Viewing Dexter with open eyes," *Communications of the ACM*, Vol. 37, No. 2. (to appear 2/94)

[112] Leggett, J.J., Schnase,J.L., Smith, J.B., and Fox, E.A. *Final Report of the NSF Workshop on Hyperbase Systems.* NSF, Gaithersburg, MD, 1993.

[113] Leggett, J. J., Schnase, J. L., and Kacmar, C. J. "Hypertext for learning," in D. Jonassen and H. Mandl (Editors), *Designing Hypertext for Learning*, NATO ASI Series, Vol. F 67, Springer-Verlag, New York, pp. 27-38, 1990.

[114] Lynch, Nancy and Mark Tuttle. "An Introduction to Input/Output Automata," *CWI-Quarterly*, 1989.

[115] Mahdavian, Seyyed and Andreas D. Bovopoulos, "Virtual Loss of Multiplexed ATM Cell Streams," Washington University Department of Computer Science, WUCS-93-08, 1993.

[116] Mazraani, Tony and Guru Parulkar. "Specification of a Multipoint Congram-Oriented High Performance Internet Protocol," *Proceedings of Infocom 90*, 6/90.

[117] Mazraani, T.Y., Parulkar, G.M., "Performance Analysis of the Ethernet under Conditions of Bursty Traffic," *Proceedings of the IEEE Global Telecommunications Conference, GLOBECOMM'92.*

[118] McNally, J. G. and E. C. Cox. "Spots and Stripes: the Patterning Spectrum in the Cellular Slime Mold *Polyphondylium pallidum*," *Development*, 1989.

[119] McNally, J. G., C. Preza, J. Conchello and L. Thomas. "Artifacts in Computational Optical-Sectioning Microscopy," To appear in *Journal of the Optical Society of America*, 1993.

[120] Mermall, V., J. G. McNally and K. G. Miller. "Transport of Cytoplasmic Particles Catalyzed by an Unconventional Myosin in Living Drosophila Embryos," submitted to *Nature*, 1993.

[121] Melen, Riccardo and Jonathan S. Turner. "Distributed Protocols for Access Arbitration in Tree Structured Communication Channels," *Proceedings of ICC 88*, June 1988.

[122] Melen, Riccardo and Jonathan S. Turner. "Nonblocking Multirate Networks," *SIAM Journal on Computing*, 4/89.

[123] Melen, Riccardo and Jonathan Turner. "Distributed Protocols for Access Arbitration in Tree-Structured Communication Channels," *IEEE Transactions on Communications*, 3/91.

[124] Melen, Riccardo and Jonathan Turner. "Nonblocking Multirate Distribution Networks," *IEEE Transactions on Communications*, vol. 41, no. 2, 2/93, pp. 362-369.

[125] Miller, M. I., G. E. Christensen, Y. Amit and U. Grenander. "A Mathematical Textbook of Deformable Neuro-anatomies," *Biomedical Computer Laboratory Monograph 548*, Washington University, 1992.

[126] National Aeronautics and Space Administration (NASA), "Earth Observing System Data and Information System, Report of the EOS Data Panel," Volume IIa.

[127] National Academy of Sciences, "Issues and recommendations associated with distributed computation and data management systems in the space sciences," National Academy Press.

[128] Papadopoulos, C., "Remote Visualization on a Campus Network," MS thesis, Department of Computer Science, Washington University in St. Louis, 5/92.

[129] Papadopoulos, C. and G. M. Parulkar, "Experimental evaluation of SUNOS IPC and TCP/IP Protocol," *IEEE/ACM Transactions on Networking*, 4/93.

[130] Parulkar, Guru and Jonathan Turner. "Towards a Framework for High Speed Communication in a Heterogeneous Networking Environment," *IEEE Network* 3/90.

[131] Parulkar, Guru. "The Next Generation of Internetworking," *Computer Communication Review*, 1/90.

[132] Perlman, Radia and George Varghese. "Pitfalls in the Design of a Class of Distributed Routing Algorithms," *Proceedings ACM SIGCOMM*, 1988.

[133] Perlman, Radia, George Varghese, and Tony Lauck. "Reliable Broadcast of Information in OSI Network Layer Protocol 10589 (IS-IS)," US. Patent 5,086,428, 2/92.

[134] Pont-Lezica, R. F., J. G. Mcnally and B. G. Pickard. "Wall-to-Membrane Linkers in Onion Epidermis: Some Hypotheses," *Plant Cell and Environment*, 1993.

[135] Preza, C., M. Miller, L. Thomas and J. McNally. "Regularized Linear Method for Reconstruction of Three Dimensional Microscopic Objects from Optical Sections," *Journal of the Optical Society of America*, 2/92.

[136] Preza, C., J. M. Ollinger, J. G. McNally and L. J. Thomas, Jr. "Point-Spread Sensitivity Analysis for Computational Optical-Sectioning Microscopy," *Micron and Microscopica Acta*, 1992.

[137] Preza, C., M. I. Miller and J.-A. Conchello. "Image Reconstruction for 3-D Light Microscopy with a Regularized Linear Method Incorporating a Smoothness Prior," in *Biomedical Image Processing and Biomedical Visualization*, R. S. Acharya and D. B. Goldgof, editors, 2/92.

[138] Purdy, J. A., W. B. Harms, J. W. Matthews, J. W. Drzymala, B. Emami, J. R. Simpson, J. Manolis and F. U. Rosenberger. "Advances in 3-Dimensional Radiation Treatment Planning Systems: Room-View Display with Real Time Interactivity," to appear in *International Journal of Radiation Oncology, Biology and Physics*, 1994.

[139] Rayes, M.A., Min, P.S. and Hegde, M. V., "Analysis of State Dependent Routing (SDR) and Modified SDR," *Technical Report*, Department of Electrical Engineering, Washington University. Also, submitted *IEEE Transactions on Communications*, 10/92.

[140] Reiker, G.G., Gohel, N., Muka, E., Blaine, G.J., "Quality Monitoring of Soft-Copy Displays for Medical Radiography" *Journal of Digital Imaging*, Vol 5, No 3, pp. 161-167, 8/92.

[141] Richard, W. R., Costa, P., and Sato, K., "The Washington University Broadband Terminal," *IEEE J. Selected Areas in Communications*, Special Issue on High-Speed Host/Network Interface, to appear.

[142] Richard, W.D., Jerome R. Cox, Jr., A. Maynard Engebretson, Jason Fritts, Brian Gottlieb and Craig Horn. "The Washington University Multimedia eXplorer," Washington University Department of Computer Science Technical Report WUCS-93-40, 8/93.

[143] Roman, G.-C. and Cox, K. C., "A Declarative Approach to Visualizing Concurrent Computations," *Visual Programming Environments: Applications and Issues*, E. P. Glinert (editor), IEEE Computer Society Press Tutorial, 1990, pp. 83-94. (Reprinted from Computer 22, No. 10, 10/89, pp. 25–36.)

[144] Roman, G.-C. and Cunningham, H. C., "Mixed Programming Metaphors in a Shared Dataspace Model of Concurrency," *IEEE Transactions on Software Engineering* 16, No. 12, 12/90, pp. 1361–1373.

[145] Roman, G.-C. and Cunningham, H. C., "Reasoning about Synchronic Groups," *Research Directions in High-Level Parallel Programming Languages*, Banatre, J. P., and Metayer, D. Le, (editors), Lecture Notes in Computer Science 574, Springer-Verlag, 1992, pp. 21–38.

[146] Roman, G.-C., Cox, K. C., Wilcox, C. D., and Plun, J. Y. "Pavane: A System for Declarative Visualization of Concurrent Computations," *Journal of Visual Languages and Computing 3*, No. 2, 6/92, pp. 161–193.

[147] Roman, G.-C., Gamble, R. F., and Ball, W. E. "Formal Derivation of Rule- Based Programs," *IEEE Transactions on Software Engineering 19*, No. 3, 3/93, pp. 277–296.

[148] Roman, G.-C., Plun, J. Y., and Wilcox, C. D. "Dynamic Synchrony among Atomic Actions," Washington University, Department of Computer Science, WUCS-90-3. (To appear in *IEEE Transactions on Parallel and Distributed Systems.*)

[149] Roman, G.-C., and Cox, K. C. "A Taxonomy of Program Visualization Systems," Washington University, Department of Computer Science, WUCS-93-24. (To appear in *Computer.*)

[150] Roman, G.-C., and Wilcox, C. D. "Architecture-Directed Refinement," Washington University, Department of Computer Science, WUCS-92-34. (To appear in *IEEE Transactions on Software Engineering.*)

[151] Rosenberger, F. U., J. W. Matthews, G. C. Johns, R. E. Drzymala and J. A. Purdy. "Use of Transputers for Real Time Dose Calculation and Presentation for 3-Dimensional Radiation Treatment Planning," *International Journal of Radiation Oncology, Biology and Physics*, 1992.

[152] Rosenberger, F. U., J. W. Matthews, G. C. Johns, R. E. Drzymala and J. A. Purdy. "Real Time 3-D Radiation Treatment Planning Dose Calculation Using Parallel Processing," *Medical Physics*, 1992.

[153] Schnase, J.L., Leggett, J.J., Hicks, D.L., Nuernberg, P.J., and Sanchez, J.A. "Open Architectures for Large-Scale Hypermedia-Based Information Systems," *Hawaii International Conference on System Science*, (HICSS-27), 1994, to appear.

[154] Schnase, J.L., Leggett, J.J., Hicks, D.L., Nuernberg, P.J., and Sanchez, J.A. "Design and Implementation of the HB1 Hyperbase Management System," *Electronic Publishing – Origination, Dissemination and Design*, Vol. 6, No. 2, pp. 1–29, 1993.

[155] Schnase, J.L., Leggett, J. J., Hicks, D. L., and Szabo, R. L. 1993. "Semantic data modeling of hypermedia associations," *ACM Transactions on Information Systems*, Vol. 11, No. 1, pp. 27-50.

[156] Schnase, J. L. and Leggett, J. J. "Computational Hypertext in Biological Modelling," *Proceedings of the Third ACM Conference on Hypertext/Hypermedia* (Hypertext '89), (Pittsburgh, PA, November), pp. 181–198.

[157] Sterbenz, James and Guru Parulkar. "Axon: Network Virtual Storage Design," *Computer Communication Review*, 4/90.

[158] Sterbenz, James and Guru Parulkar. "AXON: a High Speed Communication Architecture for Distributed Applications," *Proceedings of Infocom 90*, 6/90.

[159] Sterbenz, James and Guru Parulkar, "Axon: Application-Oriented Lightweight Transport Protocol Design", *Proceedings of ICCC*, 11/90.

[160] Sterbenz, James and Guru Parulkar "Host-Network Interface Architecture for Gigabit Communications," in *Protocols for High Speed Networks, II*, Marjory Johnson (ed.), Elsevier Science Publishers, 1991.

[161] Sterbenz, James. "AXON: a High Speed Communication Architecture for Distributed Applications," Washington University Computer Science Department, DSc thesis, 10/91.

[162] Sterbenz, James, Anshul Kantawala and Guru Parulkar, "Axon Network Host Network Interface Functional Simulation," Washington University Computer and Communications Research Center, WUCCRC-91-2, 1991.

[163] Sterbenz, James, Anshul Kantawala, Millind Buddhikot, Guru Parulkar., "Hardware Based Error and Flow Control in the Axon Gigabit Host-Network Interface," *Proceedings of the IEEE Conference on Computer and Communications, INFOCOM'92*

[164] Strayer, T., Dempsey, B., and Weaver, A. *XTP — The Xpress Transfer Protocol*, Addisson Wesley Publishing Company, Inc. 1992.

[165] V.S.Sunderam. PVM: A Framework for Parallel Distributed Computing. *Concurrency: Practice and Experience*, 2, 12/90.

[166] Swaminathan, Bala and Kenneth J. Goldman. "Hierarchical Correctness Proofs for Recursive Distributed Algorithms Using Dynamic Process Creation," Washington University Computer Science Department, WUCS-92-10, 1993.

[167] Swaminathan, Bala and Kenneth Goldman. "Dynamic Reconfiguration with I/O Abstraction," Washington University Computer Science Department, WUCS-93-21, 1993.

[168] Turner, Jonathan S. "New Directions in Communications," *IEEE Communications Magazine*, 10/86.

[169] Turner, Jonathan S. "The Challenge of Multipoint Communication," *Proceedings of the ITC Seminar on Traffic Engineering for ISDN Design and Planning*, 5/87.

[170] Turner, Jonathan S, "Fluid Flow Loading Analysis of Packet Switching Networks," *Proceedings of the International Teletraffic Congress*, June 1988.

[171] Turner, Jonathan S. "Design of a Broadcast Packet Network," *IEEE Transactions on Communications*, June 1988.

[172] Turner, Jonathan S. "Practical Wide-Sense Nonblocking Generalized Connectors," Washington University Computer Science Department, WUCS-88-29.

[173] Turner, Jonathan. "Resequencing Cells in an ATM Switch," Washington University Computer Science Department, WUCS-91-21.

[174] Turner, Jonathan. "A Proposed Bandwidth Management and Congestion Control Scheme for Multicast ATM Networks," Washington University Computer and Communications Research Center, WUCCRC-91-1.

[175] Turner, Jonathan. "Multimedia ATM Network for Computer and Communications Research," NSF academic research infrastructure grant, 1/93–12/94.

[176] Turner, Jonathan, "Managing Bandwidth in ATM Networks with Bursty Traffic," *IEEE Network Magazine*, vol. 6, no. 5, 9/92, 50–58.

[177] Turner, Jonathan, "A Practical Version of Lee's Multicast Switch Architecture," *IEEE Transactions on Communications*, Vol. 41, No. 40, 4/93.

[178] Turner, Jonathan, "Queueing Analysis of Buffered Switching Networks," *IEEE Transactions on Communications*, vol. 41, no. 2, 2/93, pp. 412-420.

[179] Turner, Jonathan, 'An Optimal Nonblocking Multicast Virtual Circuit Switch," WUCS-93-30, 6/93.

[180] Valdimarsson, Einir "Evaluation of Blocking Probability in Multirate Networks," Washington University Computer Science Department, MS thesis, 4/90.

[181] Valdimarsson, Einir "Blocking in Multirate Networks," *Proceedings of Infocom*, 4/91.

[182] Van Essen, D.C., Anderson, C.H., and Felleman, D.J., "Information Processing in the Primate Visual System: An Integrated Systems Perspective," *Science*, Vol 255, pp. 419-423, 1993.

[183] Varghese, George. "Self-Stabilization by Local Checking and Correction," MIT Laboratory for Computer Science, Ph.D. thesis, 10/92.

[184] Waxman, Bernard. "Probable Performance of Steiner Tree Algorithms," Washington University Computer Science Department, WUCS-88-4.

[185] Waxman, Bernard. "Routing of Multipoint Connections," *IEEE Journal on Selected Areas of Communications*, 12/88.

[186] Waxman, Bernard. "New Approximation Algorithms for the Steiner Tree Problem," Washington University Computer Science Department, WUCS-89-15.

[187] Waxman, Bernard. "Performance Evaluation of Multipoint Routing Algorithms," *Proceedings of Infocom*, 3/93.

[188] Witte, Ellen. "The Clos Network as a Multirate Distributor with a Greedy Routing Algorithm," Washington University Computer Science Department, WUCS-92-13.

[189] Zegura, Ellen. "Architectures for ATM Switching Systems," *IEEE Communications Magazine*, 2/93.

[190] Zegura, Ellen. "Evaluating Blocking Probability in Distributors," *Proceedings of Infocom*, 3/93.