# The Gigabit Testbed Initiative

# Final Report

December 1996

This work was supported by the National Science Foundation and the Defense Advanced Research Projects Agency under Cooperative Agreement NCR8919038 with the Corporation for National Research Initiatives

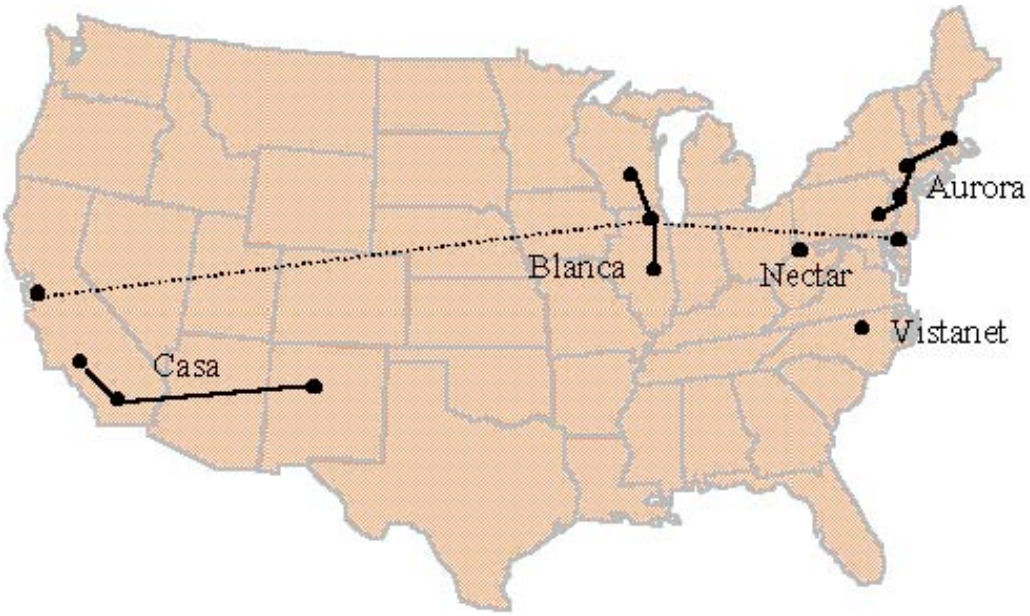
# Table of Contents

# Executive Summary

The Gigabit Testbed Initiative was a major effort by approximately forty organizations representing universities, telecommunication carriers, industry and national laboratories, and computer companies to create a set of very high-speed network testbeds and to explore their application to scientific research. This effort, funded by the National Science Foundation (NSF) and the Defense Advanced Research Projects Agency (DARPA), was coordinated and led by the Corporation for National Research Initiatives (CNRI) working closely with each of the many participating organizations and with the U.S. Government. The U. S. Government was also a participating organization insofar as testbeds were established within several Government laboratories to explore the concepts and technologies emerging from the Initiative.

Five Testbeds, named Aurora, Blanca, Casa, Nectar and Vistanet, were established and used over a period of several years to explore advanced networking issues, to investigate architectural alternatives for gigabit networks, and to carry out a wide range of experimental applications in areas such as weather modeling, chemical dynamics, radiation oncology, and geophysics data exploration. The five testbeds were geographically distributed across the United States as shown in the figure below.

## *The Gigabit Testbeds*




At the time the project started in 1990 there were significant barriers to achieving high performance networking, which was falling significantly behind advances in high performance computing. One of the major barriers was the absence of wide-area transmission facilities which could support gigabit research, and the lack of marketplace motivation for carriers to provide

such facilities. The testbed initiative specifically targeted this problem through the creation of a multi-dimensional research project involving carriers, applications researchers, and network technologists. A second (and related) barrier was the lack of commercially available high speed network equipment operating at rates of 622 Mbps or higher. Fortunately, several companies were beginning to develop such equipment and the testbed initiative helped to accelerate its deployment.

A key decision in the effort, therefore, was to make use of experimental technologies that were appropriate for gigabit networking. The emphasis was placed on fundamental systems issues involved with the development of a technology base for gigabit networking rather than on test and evaluation of individual technologies. ATM, SONET and HIPPI were three of the technologies used in the program. As a result, the impetus for industry to get these technologies to market was greatly heightened. Many of the networks that subsequently emerged, such as the NSF-sponsored vBNS and the DOD-sponsored DREN, can be attributed to the success of the gigabit testbed program.

The U.S. Government funded this effort with a total of approximately $20M over a period of approximately five years, with these funds used by CNRI primarily to fund university research efforts. Major contributions of transmission facilities and equipment were donated at no cost to the project by the carriers and computer companies, who also directly funded participating researchers in some cases. The total value of industry contributions to the effort was estimated to be perhaps 10 or 20 times greater than the Government funding. The coordinating role of a lead organization, played by CNRI, was essential in helping to bridge the many gaps between the individual research projects, industry, government agencies and potential user communities. At the time this effort began, there did not appear to be a clearly visible path to make this kind of progress happen.

## *Initiative Impacts*

In addition to the many technical contributions resulting from the testbeds, a number of non-technical results have had major impacts for both education and industry.

First and foremost was a new model for network research provided by the testbed initiative. The bringing together of network and application researchers, integration of the computer science and telecommunications communities, academia-industry-government research teams, and government-leveraged industry funding, all part of a single, orchestrated project spanning the country, provided a new level of research collaboration not previously seen in this field. The Initiative created a community of high performance networking researchers that crossed academic/industry/government boundaries.

The coupling of application and networking technology research from project inception was a major step forward for both new technology development and applications progress. Having applications researchers involved from the start of the project allowed networking researchers to obtain early feedback on their network designs from a user's perspective, and allowed network performance to be evaluated using actual user traffic. Similarly, application researchers learned

how network performance impacted their distributed application designs through early deployment of prototype software. Perhaps most significantly, researchers could directly investigate networked application concepts without first waiting for the new networks to become operational, opening them to new possibilities after decades of constrained bandwidth.

The collaboration of computer network researchers, who came primarily from the field of computer science, and the carrier telecommunications community provided another important dimension of integration. The development of computer communications networks and carrier-operated networks have historically proceeded along two separate paths with relatively little cross-fertilization. The testbeds allowed each community to work closely with the other, allowing each to better appreciate the other's problems and solutions and leading to new concepts of integrated networking and computing.

From a research perspective, the testbed initiative created close collaborations among investigators from academia, government research laboratories, and industrial research laboratories. Participating universities included Arizona, UCBerkeley, Caltech, Carnegie-Mellon, Illinois, MIT, North Carolina, Pennsylvania and Wisconsin; national laboratories included Lawrence Berkeley Laboratory, Los Alamos National Laboratory (LANL), and JPL, and the NSF-sponsored National Center for Supercomputer Applications, Pittsburgh Supercomputer Center, and San Diego Supercomputer Center, while industry research laboratories included IBM Research, Bellcore, GTE Laboratories, AT&T Bell Laboratories, BellSouth Research, and MCNC. The collaborations also included facilities planners and engineers from the participating carriers, which included Bell Atlantic, BellSouth, AT&T, GTE, MCI, NYNEX, Pacific Bell and US West.

Another important dimension of the testbed model was its funding structure, in which government funding was used to leverage a much larger investment by industry. A major industry contribution was made by the carriers in the form of SONET and other transmission facilities within each testbed at gigabit or near-gigabit rates. The value of this contribution cannot be overestimated, since not only were such services otherwise non-existent at the time the project began, but they would have been unaffordable to the research community if they had existed under normal tariff conditions. By creating an opportunity for the carriers to learn about potential applications of high speed networks while at the same time benefiting from collaboration with the government-funded researchers in network technology experiments, the carriers were, in turn, willing to provide new high-speed wide-area experimental transmission facilities and equipment and to fund the participation of their researchers and engineers.

The Initiative resulted in significant technology transfer to the commercial sector. As a direct result of their participation in the project, two researchers at Carnegie-Mellon University founded a local-area ATM switch startup company, FORE Systems. This was the first such local ATM company formed, and provided a major stimulus for the emergence of high speed local area networking products. It also introduced to the marketplace the integration of advanced networking concepts with advanced computing architectures used within their switch.

Other technology transfers included software developed to distribute and control networked applications, the HIPPI measurement device (known as Hilda) developed by MCNC as part of the Vistanet effort, and the HIPPI-SONET wide-area gateway developed by LANL for the Casa testbed. In addition, new high speed networking products were developed by industry in direct response to the needs of the testbeds, for example HIPPI fiber optic extenders and high speed user-side SONET equipment. Major technology transfers also occurred through the migration of students who had worked in the testbeds to industry to implement their work in company products.

At the system level, the testbeds led directly to the formation of three statewide high speed initiatives undertaken by carriers participating in the testbeds. The North Carolina Information Highway (NCIH) was formed by BellSouth and GTE as a result of their Vistanet testbed involvement to provide an ATM/ SONET network throughout the state. Similarly, the NYNET experimental network was formed in New York state by NYNEX as a result of their Aurora testbed involvement, and the California Research and Education Network (CalREN) was created by Pacific Bell following their Casa testbed participation.

The testbed initiative also led to the early use of gigabit networking technology by the defense and intelligence communities for experimental networks and global-scale systems, which have become the foundation for a new generation of operational systems. More recently, the U.S. Government has begun to take steps to help create a national level wide-area Gigabit networking capability for the research community.

The key technical areas addressed in the initiative are categorized for this report as transmission, switching, interworking, host I/O, network management, and applications and support tools. In each case, various approaches were analyzed and many were tested in detail. A condensed summary of the key investigations and findings is given at the end of the executive summary and elaborated on more fully in the report.

## *Future Directions*

Among the barriers most often cited to the widespread deployment of very high-speed networks, those often cited are costs of the technology (particularly the cost of its deployment over large geographic areas), the regulated nature of the industry, and lack of market forces for applications that could make use of it and sustain its advance. Moreover, most people find it difficult to invest their own time or resources in a new technology until it becomes sufficiently mature that they can try it out and visualize what they might do with it and when they might use it.

A recent National Research Council report [1] includes a summary of the major advances in the computing and communications fields from the beginning of time-sharing through scalable parallel computing, just prior to when the gigabit testbeds described in this report were producing their early results. Using that report's model, the gigabit testbeds would be characterized as being in the early conceptual and experimental development and application phase. The first technologies were emerging and people were attempting to understand what could be done with them,

long before there was an understanding of what it would take to engineer and deploy the technologies on a national scale to enable new applications not yet conceived.

The Gigabit Testbed Initiative produced a demonstration of what could be done in a variety of application areas, and it motivated people in the research community, industrial sector, and government to provide a foundation for follow-on activities. Within the Federal government, the testbed initiative was a stimulus for the following events:

- The HPCCIT report on Information and Communication Futures identified high performance networking as a Strategic Focus.
- The National Science and Technology Council, Committee on Computing and Communications held a two day workshop which produced a recommendation for major upgrades to networking among the HPC Centers to improve their effectiveness, and to establish a multi-gigabit national scale testbed for pursuing more advanced networking and applications work.
- The first generation of scalable networking technologies emerged based on scalable computing technologies.
- The DoD HPC Modernization program initiated a major upgrade in networking facilities for their HPC sites.
- The Advanced Technology Demonstration gigabit testbed in the Washington DC area was implemented.
- The defense and intelligence communities began to experiment with higher performance networks and applications.
- The NSF Metacenter and vBNS projects were initiated.
- The all-optical networking technology program began to produce results with the potential for 1000x increase in transmission capacity.

To initiate the next phase of gigabit research and build on the results of the testbeds, CNRI proposed that the Government continue to fund research on gigabit networks using an integrated experimental national gigabit testbed involving multiple carriers, with gigabit backbone links provided over secondary (i.e., backup) channels by the carriers at no cost and switches and access lines paid for by the Government and participating sites. However, costs for access lines proved to be excessive, and at the time the Government was also unable to justify the funding needed for a national gigabit network capability – instead, several efforts were undertaken by the Government to provide lower speed networks.

In the not-too-distant future, we expect the costs for accessing a national gigabit network on a continuing basis will be more affordable and the need for it will be more evident, particularly its potential for stimulating the exploration of new applications. The results of the gigabit testbed initiative have clearly had a major impact on breaking down the barriers to putting high perform-

ance networking on the same kind of growth curve as high performance computing, thus enabling a new generation of national and global-scale high performance systems which integrate networking and computing.

## *Investigations and Findings*

Four distinct end-to-end network layer architectures were explored in the project. These were a result both of architecture component choices made by researchers after the work was underway and of the a priori testbed formation process. The architectures were (1) seamless WAN-LAN ATM and (2) seamless WANLAN PTM, both used in the Aurora testbed, (3) heterogeneous wide-area ATM/local-area networks, used in the Blanca, Nectar and Vistanet testbeds, and (4) wide-area HIPPI/SONET via local switching, used in the Casa testbed.

The following summaries present highlights of the technology and applications investigations. It should be noted that while some efforts are specific to their architectural contexts, in many cases, the results can be applied to other architectures including architectures not considered in the initiative.

### *Transmission*

- OC-48 SONET links were installed in four testbeds over distances of up to 2000 km, accelerating vendor development and carrier deployment of high speed SONET equipment, establishing multiple-vendor SONET interconnects, enabling discovery and resolution of standards implementation compatibility problems, and providing experience with SONET error rates in an operational environment
- Testbed researchers developed a prototype OC-12c SONET cross-connect switch and investigated interoperation with carrier SONET equipment, and developed OC-3c, OC-12, and OC-12c SONET interfaces for hosts, gateways and switches; these activities provided important feedback to SONET chip developers
- Techniques for carrying variable-length packets directly over SONET were developed for use with HIPPI and other PTM technologies, with both layered and tightly coupled approaches explored
- An all-optical transmission system – the first carrier deployment of this technology – was installed and used to interconnect ATM switches over a 300 mile distance using optical amplifier repeaters
- HIPPI technology was used for many local host links and for metropolitan area links through the use of HIPPI extenders and optical fiber; other local link technologies included Glink and Orbit
- Several wide-area striping approaches were investigated as a means of deriving 622 Mbps and higher bandwidths from 155 Mbps ATM or SONET channels; configurations included end-to-end ATM over SONET, LAN-WAN HIPPI over ATM/SONET, and LAN-WAN HIPPI and other variable-length PDUs directly over SONET

- A detailed study of striping over general ATM networks concluded that cell-based striping should be used. This capability can be introduced at LAN-WAN connection points in conjunction with destination host cell re-ordering and an ATM-layer synchronization scheme

### *Switching*

- Prototype high speed ATM switches were developed (or made available) by industry and deployed for experiments in several of the testbeds, supporting 622 Mbps end-to-end switched links using both 155 Mbps striping and single-port 622 Mbps operation
- The first telco central office broadband ATM switch was installed and used for testbed experiments, using OC-12c links to customer premises equipment and OC-48 trunking
- Wide-area variable-length PTM switching was developed and deployed in the testbeds using both IBM's Planet technology and HIPPI switches in conjunction with collocated wide-area gateways
- Both ATM and PTM technologies were developed and deployed for both local and desk area networking (DAN) experiments, along with the use of commercial HIPPI and ATM switches, which became available as a result of testbed-related work
- A TDMA technique was developed and applied to tandem HIPPI switches to demonstrate packet-based quality-of-service operation in HIPPI circuit-oriented switching environments, and a study of preemptive switching of variable length packets indicated a ten-fold reduction in processing requirements was possible relative to processor-based cell switching

### *Interworking*

- Three different designs were implemented to interwork HIPPI with wide-area ATM networks over both SONET and all-optical transmission infrastructures; explorations included the use of 4x155 Mbps striping and non-striped 622 Mbps access, local HIPPI termination and wide-area HIPPI bridging; resulting transfer rates ranged from 370 to 450 Mbps
- A HIPPI-SONET gateway was implemented which allowed transfer of full 800 Mbps HIPPI rates across striped 155 Mbps wide-area SONET links; capabilities included variable bandwidth allocation of up to 1.2 Gbps and optional use of forward error correction, with a transfer rate of 790 Mbps obtained for HIPPI traffic (prior to host protocol processing)
- Seamless ATM DAN-LAN-WAN interworking was explored through implementation of interface devices which provided physical layer interfacing between 500

Mbps DAN Glink transmission, LAN ATM switch ports, and a wide-area striped 155 Mbps ATM/SONET network.

### *Host I/O*

- Several different testbed investigations demonstrated the feasibility of direct cell-based ATM host connections for workstation-class computers; this work established the basis for subsequent development of high speed ATM host interface chipsets by industry and provided an understanding of changes required to workstation I/O architectures for gigabit networking
- Variable-length PTM host interfacing was investigated for several different types of computers, including workstations and supercomputers; in addition to vendor-developed HIPPI interfaces, specially developed HIPPI and general PTM interfaces were used to explore the distribution of high speed functionality between internal host architectures and I/O interface devices
- TCP/IP investigations concluded that hardware checksumming and data-copying minimization were required by most testbed host architectures to realize transport rates of a few hundred Mbps or higher; full outboard protocol processing was explored for specialized host hardware architectures or as a workaround for existing software bottlenecks
- A 500 Mbps TCP/IP rate was achieved over a 1000-mile HIPPI/SONET link using Cray supercomputers, and a 516 Mbps rate measured for UDP/IP workstation-based transport over ATM/SONET. Based on other workstation measurements, it was concluded that, with a 4x processing power increase (relative to the circa 1993 DEC Alpha processor used), a 622 Mbps TCP/IP rate could be achieved using internal host protocol processing and a hardware checksum while leaving 75% of the host processor available for application processing
- Measurements comparing the XTP transport protocol with TCP/IP were made using optimized software implementations on a vector Cray computer; the results showed TCP/IP provided greater throughput when no errors were present, but that XTP performed better at high error rates due to its use of a selective acknowledgment mechanism
- Presentation layer data conversions required by applications distributed over different supercomputers were found to be a major processing bottleneck; by exploiting vector processing capabilities, revisions to existing floating point conversion software resulted in a fifty-fold increase in peak transfer rates
- Experiments with commercial large-scale parallel processing architectures showed processor interconnection performance to be a major impediment to gigabit I/O at the application level; an investigation of data distribution strategies led to use of a reshuffling algorithm to remap the distribution within the processor array for efficient I/O

- Work on distributed shared memory (DSM) for wide-area gigabit networks resulted in several latency-hiding strategies for dealing with large propagation delays, with relaxed cache synchronization resulting in significant performance improvements

### *Network Management*

- In different quality-of-service investigations, a real-time end-to-end protocol suite was developed and successfully demonstrated using video streams over HIPPI and other networks, and a `broker' approach was developed for end-to-end/network quality-of-service negotiations in conjunction with operating system scheduling for strict real-time constraints
- An evaluation of processing requirements for wide-area quality-of-service queuing in ATM switches, using a variation of the "weighted fair queuing" algorithm, found that a factor of 8 increase in processing speed was needed to achieve 622 Mbps port speeds relative to the i960/33MHz processor used for the experiments
- Congestion/flow control simulation modeling was carried out using testbed application traffic, with the results showing rapid ATM switch congestion variations and high cell loss rates; also, a speedup mechanism was developed for lost packet recovery in high delay-bandwidth product networks using TCP's end-to-end packet window protocol
- An end-to-end time window approach using switch monitoring and feedback to provide high speed wide-area network congestion control was developed, and performance was consistent with simulation-based predictions
- A control and monitoring subsystem was developed for real-time traffic measurement and characterization using carrier-based 622 Mbps ATM equipment; the subsystem was used to capture medical application traffic statistics revealing that ATM cell traffic can be more bursty than expected, dictating larger amounts of internal switch buffering than initially thought necessary for satisfactory performance
- A data generation and capture device for 800 Mbps HIPPI link traffic measurement and characterization was developed and commercialized, and was used for network debugging and traffic analysis; more generally, many network equipment problems were revealed through the use of real application traffic during testbed debugging phases

### *Applications and Support Tools*

- Investigations using quantum chemical dynamics modeling, global climate modeling, and chemical process optimization modeling applications identified pipelining techniques and quantified speedup gains and network bandwidth requirements for distributed heterogeneous metacomputing using MIMD MPP, SIMD MPP, and vector machine architectures

- Most of the applications that were tested realized significant speedups when run on multiple machines over a very high speed network; however, a superlinear speedup of 3.3 was achieved using two dissimilar machines for a chemical dynamics application; other important benefits of distributed metacomputing such as large software program collaboration-at-a-distance were also demonstrated, and major advances made in understanding how to partition application software
- Homogeneous distributed computing was investigated for large combinatorial problems through development of a software system which allows rapid prototyping and execution of custom solutions on a network of workstations, with experiments providing a quantification of how network bandwidth impacts problem solution time
- Several distributed applications involving human interaction in conjunction with large computational modeling were investigated; these included medical radiation therapy planning, exploration of large geophysical datasets, and remote visualization of severe thunderstorm modeling
- The radiation therapy planning experiments successfully demonstrated the value of integrating high performance networking and computing for real-world applications; other interactive investigations similarly resulted in new levels of visualization capability, provided new techniques for distributed application communications and control, and provided important knowledge about host-related problems which can prevent gigabit speed operation
- A number of software tools were developed to support distributed application programming and execution in heterogeneous environments; these included systems for dynamic load balancing and checkpointing, program parallelization, communications and runtime control, collaborative visualization, and near-realtime data acquisition for monitoring progress and for analyzing results.

# 1 Introduction

This report summarizes the results of the Gigabit Testbed Initiative, a project involving several dozen participants that ran from 1990 to 1995. The report attempts to put these results into perspective by providing the background, motivation, and current trends impacting the overall work. Detailed descriptions of context and results can be found in the final reports from each of the five testbeds involved in the Initiative [2-6].

The Initiative had two main goals, both of which were premised on the use of network testbeds: (1) to explore technologies and architectures for gigabit networking, and (2) to explore the utility of gigabit networks to the end user. In both cases the focus was on providing a data rate on the order of 1 Gbps to the end-points of a network, i.e., the points of user equipment attachment, and on maximizing the fraction of this rate available to a user application.

A key objective of the Initiative was to carry out this research in a wide-area real-world context. While the technology for user-level high-speed networking capability could be directly achieved by researchers in a laboratory setting circa 1990, extending this context to metropolitan or wide-area network distances at gigabit per second rates was virtually impossible, due both to the absence of wide-area transmission and switching equipment for end-user gigabit rates and to the lack of market motivation to procure and install such equipment by local and long-distance carriers.

To solve this “chicken-and-egg” problem, a collaborative effort involving both industry and the research communities was established by CNRI with funding from government and industry. NSF and ARPA jointly provided research funding for the participating universities and national laboratories, while carriers and commercial research laboratories provided transmission and switching facilities and results from their internally-funded research. Five distinct testbed collaborations were created. These were called Aurora, Blanca, Casa, Nectar, and Vistanet. (A sixth gigabit testbed called MAGIC [7] was funded by DARPA about 18 months later, but was managed as a separate project and is not further described in this report.)

Each testbed had a different set of research collaborators and a different overall research focus and objectives. At the same time, there were also common areas of research among the testbeds, allowing different solutions for a given problem to be explored.

The remainder of this report is organized as follows. Section 2, The Starting Point, briefly describes the technical context for the project which existed in the 1989-90 timeframe. Section 3, Structure and Goals, gives an overview of the Initiative structure, including the participants, topology and goals of each testbed. The main body of the report is contained in Section 4, Investigations and Findings, which brings together by technical topic the major work carried out in the five testbeds. Section 5, Conclusion, summarizes the impacts of the Initiative and how they might relate to the future of very high speed networking research. Appendix A lists reports and publications generated by the testbeds during the course of the project.

Readers are strongly encouraged to consult the testbed references and publications for more comprehensive and detailed discussions of testbed accomplishments. This report summarizes much of that work, but is by no means a complete cataloging of all efforts undertaken.

# 2 The Starting Point

## *2.1 A Brief History*

Computer networking dates from the late 1960s, when affordable minicomputer technology enabled the implementation of wide-area packet switching networks. The Arpanet, begun in 1969 as a research project by DARPA, provided a focal point within the U.S. for packet network technology development. In the 1970s, parallel development by DARPA of radio and satellite-based packet networks and TCP/IP internetworking technology resulted in the establishment of the Internet. The subsequent introduction and widespread use of ethernet, token ring and other LAN technologies in the 1980s, coupled with the expansion of the Internet by NSF to a broader user base, led to increasing growth and a transition of the Internet to a self-supporting operational status in the 1990s.

Wide-area packet switching technology has from its inception made use of the telephone infrastructure for its terrestrial links, with the packet switches forming a network overlay on the underlying carrier transmission system. The links were initially 50 Kbps leased lines in the original Arpanet, progressing to 1.5 Mbps T1 lines in the NSFNET circa 1988 and 45 Mbps T3 lines by about 1992. Thus, at the time the gigabit testbed project began, Internet backbone speeds and large-user access lines were in the 50 Kbps to 1.5 Mbps range and local-area aggregate speeds were typically 10 Mbps or less. Individual peak user speeds ranged from about 1 Mbps for high-end workstations to 9.6 Kbps or less for PC modem connections.

The dominant application which emerged on the Arpanet once the network became usable was not what had been expected when the network was planned. Conceived as a vehicle for resource sharing among the host computers connected to the network, people-to-people communication in the form of email quickly came to dominate network use. The ability to have extended conversations without requiring both parties to be available at the same time, being able to send a single message to an arbitrarily large set of recipients, and automatically having a copy of every message stored in a computer for future reference proved to be powerful stimuli to the network's use, and is an excellent example of the unforeseen consequences of making a new technology available for experimental exploration.

The computer resource sharing which did take hold was dominated by two applications-namely, file transfer and remote login. Applications which distributed a problem's computation among computers connected to the network were also attempted and in some cases demonstrated, but they did not become a significant part of the original Arpanet's use. Packetized voice experiments were demonstrated over the Arpanet in the 1970s, but with limited applicability due to limited bandwidth and long store-and-forward transmission delays at the switches.

The connection of the NSF-sponsored supercomputer centers to the Internet in the late 1980s provided a new impetus for networked resource sharing and resulted in an increase of activity in this application area, but multi-computer explorations were severely limited by network speeds.

## *2.2 State of Very High-Speed Networking in 1989-90*

Prior to the time the testbeds were being formed in 1990, very little hands-on research in gigabit networking was taking place. Work by carriers and equipment vendors focused primarily on higher transmission speeds rather than on networking. There was a good deal of interest in high-speed networking within the research community, consisting mostly of paper studies and simulations, along with laboratory work at the device level. Interest was stimulated in the telecommunications industry by ongoing work on the standardization of Broadband ISDN (B-ISDN), which was intended to eventually address user data rates from about 50 Mbps upwards to the gigabit/s region, within the scientific community, interest in remote data visualization and multi-processor supercomputer-related activities was high.

A few high speed technologies had emerged by 1989, most notably HIPPI and Ultranet for local connections between computers and peripherals. HIPPI, developed at Los Alamos National Laboratory (LANL), was in the process of standardization at the time by an ANSI subcommittee and had been demonstrated with laboratory prototypes. Ultranet was based on proprietary protocols, and Ultranet products were in use at a small number of supercomputer centers and other installations. Both technologies provided point-to-point links between hosts at data rates of 800 Mbps to 1 Gbps.

In wide-area networking, SONET (Synchronous Optical Network) was being defined as the underlying transmission technology for the U.S. portion of B-ISDN by ANSI, and its European counterpart SDH (Synchronous Digital Hierarchy) was undergoing standardization by the CCITT. SONET and SDH were designed to provide wide-area carrier transport at speeds from approximately 50 Mbps to 10 Gbps and higher, along with the associated monitoring and control functions required for reliable carrier operation. While non-standard trunks were already in operation at speeds on the order of a gigabit/s, the introduction of SONET/SDH offered carriers the use of a scalable, all-digital standard with both flexible multiplexing and the prospect of ready interoperability among equipment developed by different vendors.

A number of high-speed switch designs were underway at the time, most focused on ATM cell switching. Examples of ATM switch efforts included the Sunshine switch design at Bellcore and the Knockout switch design at AT&T Bell Labs. Exploration of variable length packet switching at gigabit speeds was also taking place, most notably by the PARIS (later renamed Planet) switch effort at IBM. These efforts were focused on wide-area switching environments-investigation of ATM for local area networking had not yet begun.

Computing performance in 1990 was dominated by the vector supercomputer, with highly parallel supercomputers still in the development stage. The fastest supercomputer, the CRAY-YMP, achieved on the order of 1-2 gigaflops in 1990, while the only commercial parallel computer available was the Thinking Machines Corporation CM-2. Workstations had peak speeds in the 100 MIPS range, with PCs in about the 10 MIPS range. I/O interfaces for these machines consisted mainly of 10 Mbps ethernet and other LAN technologies with similar speeds, with some instances of 100 Mbps FDDI beginning to appear.

Optical researchers were making significant laboratory advances by 1990 in the development of optical devices to exploit the high bandwidth inherent in optical fibers, but this area was still in a very early stage with respect to practical networking components. Star couplers, multiplexors, and dynamic tuners were some of the key optical components being explored, along with several all-optical local area network designs.

The data networking research community had begun to focus on high-speed networking by the late 1980s, particularly on questions concerning protocol performance and flow/congestion control. New transport protocols such as XTP and various lightweight protocol approaches were being investigated through analysis, simulation, and prototyping, and a growing amount of conference and journal papers were focusing on high-speed networking problems.

The regulatory environment which existed in 1990, at the time the Gigabit Testbed Initiative was formed, was quite different from that which is now evolving. A regulated local carrier environment existed consisting of the seven regional Bell operating companies (RBOCs) along with some non-Bell companies such as GTE, which provided tariffed local telephone services throughout the U.S. Long distance services were being provided by AT&T, MCI, and Sprint in competition with each other. Cable television companies had not yet begun to expand their services beyond simple residential television delivery, and direct broadcast satellite services had not yet been successfully established. And while some independent research and development activities had been established within some of the RBOCs, the seven regional carriers continued to fund Bellcore as their common R&D laboratory.

With the passage of the Telecommunications Act of 1996, a more competitive telecommunications industry now seems likely. Mergers and buy-outs among the RBOCS are taking place, cable companies have begun to offer Internet access, and provisions for Internet telephony have begun to be accommodated by Internet service providers.

## *2.3 Gigabit Networking Research Issues*

When the initiative began in 1990, many questions concerning high-speed networking technology were being considered by the research community. At the same time, telephone carriers were struggling with the question of how big the market, if any, might be for carrier services which would provide a gigabit/s service to the end-user. Cost was a major concern here. Research issues existed in most, if not all, areas of networking, including host I/O, switching, flow/congestion and other aspects of network control, operating systems, and application software. Two major questions underlie most of these technical issues: (1) could host I/O and other hardware and software operate at the high speeds involved? and (2) would speed of light delays in WANs degrade application and protocol performance?

These issues can be grouped into three general sets, which are discussed separately below:

- network issues
- platform issues
- application issues

### *Network Issues*

A basic issue was whether existing conceptual approaches developed for lower speed networking would operate satisfactorily at gigabit speeds. Implementation issues were also uppermost in mind. For example, would a radically different protocol design allow otherwise unachievable low-cost implementations. However, most of the conceptual issues were driven by the fact that speed-of-light propagation delay across networks is constant, while data transmission times are a function of the transmission speed.

At a data rate of 1 Gbps, it takes only one nanosecond to transmit one bit, resulting in a link transmission time of 10 microseconds for a 10 kilobit packet. In contrast, for the 50 Kbps link speeds in use when the Arpanet was first designed, the same 10 kilobit packet has a transmission time of 200 milliseconds. The speed-of-light propagation delay across a 1000-mile link for either case, on the other hand, is on the order of 10 milliseconds. The result is that, whereas in the Arpanet case propagation delay is more than an order of magnitude smaller than the transmission time, in the gigabit network the propagation time is more than three orders of magnitude larger than the transmission time!

This difference has both positive and negative consequences. On the positive side, store-and-forward delays introduced by packet switches and routers along an end-to-end path are directly related to transmission time, causing them to become very small at gigabit speeds (barring unusual queuing situations). This removes a major problem inherent in the early Arpanet for packetized voice and other traffic having low delay requirements, since at gigabit speeds the resulting cumulative transmission delays effectively disappear relative to the propagation delay over wide-area distances.

On the negative side, the very small packet transmission time means that information sent to the originating node for feedback control purposes may no longer be useful, since the feedback is still subject to the same propagation delay across the network. Most networks in place in 1990, and particularly the Internet, relied on window-based end-to-end feedback mechanisms for flow/congestion control, for example that used by the TCP protocol. At 50 Kbps, a 200 millisecond packet transmission time meant that feedback from a destination node on a cross-country link could be returned to the sender before it had completed the transmission, causing further transmissions to be suppressed if necessary. At 1 Gbps, this type of short-term feedback control is clearly impossible for link distances of a few miles or more.

The impact of this feedback delay on performance is strongly related to the statistical properties of user traffic. If the peak and average bandwidth requirements of individual data streams are

predictable over a time interval which is large relative to the network's roundtrip propagation delay, then one might expect roundtrip feedback mechanisms to continue to work well. On the other hand, if the traffic associated with a user `session', such as a file transfer, persists only for a duration comparable to or less than the roundtrip propagation time, then end-to-end feedback will be ineffective in controlling that stream relative to events occurring within the network while the stream is in progress. (And while we might look to the aggregation of large numbers of users to provide statistical predictability, the phenomenon of self-similar data traffic behavior has brought the prospect of aggregate data traffic predictability into question.)

Another control function impacted by the transmission/propagation time ratio is that of call setup in wide-area networks using virtual circuit (VC) mechanisms, for example in ATM networks. The propagation factor in this case can result in a significant delay before the first packet can be sent relative to what would otherwise be experienced. Moreover, for cases in which the elapsed time from the first to last packet sent is less than the VC setup time, inefficient resource utilization will typically result.

The transmission/propagation time ratio also impacts local area technologies. The performance of random access networks such as ethernet is premised on this ratio being much greater than one, so that collisions occurring over the maximum physical extent of the network can be detected at all nodes in much less than one packet transmission time. A factor of 100 increase from the original ethernet design rate of 10 Mbps to 1 Gbps implies that the maximum physical extent must be correspondingly reduced or the minimum packet size correspondingly increased, or some combination of the two, in order to use the original ethernet design without change.

More generally, as new competing technologies such as HIPPI or all-optical networks are introduced to deal explicitly with gigabit speeds, and with the prospect of still higher data rates in the future, issues of scalability and interoperability become increasingly important. Questions of whether ATM and SONET can scale independently of data rate or are in fact constrained by factors such as propagation delay, whether single-channel transmission at ever higher bit rates or striping over lower bit-rate multiple channels will prove more cost-effective, and how interoperability should best be achieved are important questions raised by the push to gigabit networking and beyond.

Along a somewhat different dimension, the proposed use of distributed shared memory (DSM) as a wide-area high speed communication paradigm instead of explicit message passing raised a number of issues. DSM attempts to make communication among a set of networked processors appear the same as if they were on a single machine using shared physical memory. A high bandwidth is required between the machines to allow successful DSM operation, and this had been achieved for local area networking environments. Issues concerning the application of DSM to a wide-area gigabit environment included how to hide speed-of-light latency so that processors do not have to stop and wait for remote memory updates and how far DSM could/should extend into the network; for example, should DSM be supported within network switches? Or, at the other extreme, should it exist only above the transport layer to provide a shared memory API for application programmers.

### *Platform Issues*

A second set of issues concerns the ability of available computer and other technologies to support protocol processing, switching, and other networking functions at gigabit speeds. We use platform here very generally to mean the host computers, switching nodes internal to a network, routers or gateways which may be used for network interconnection, and specialized devices such as low level interfacing equipment.

For host computers the dominant question is the amount of resources required to carry out host-to-host and host-to-network protocol processing -- in particular, could the computers available in 1990 support application I/O at gigabit rates, and if not at what future point might they be expected to?

Because of the dominance of TCP/IP in wide-area data networking by 1990, a question frequently asked was whether TCP implementations would scale to gigabit/s operation on workstation-class hosts. Some researchers claimed it would not scale and would have to be replaced by a new protocol explicitly designed for efficient high speed operation, in some cases using special hardware protocol engines. Others did not go to this extreme, but argued that outboard processing devices would be required to offload the protocol processing burden from the host, with the outboard processing taking place either on a special host I/O board or on an external device. Still others held that internal TCP processing at gigabit rates was not a problem if care was taken in its implementation, or that hardware trends would soon provide sufficient processing power.

For network switching nodes, a key question in 1990 was whether hardware switching was required or software-based packet switching could be scaled up to handle gigabit port rates and multi-gigabit aggregate throughputs. Another important question was how much control processing could reasonably be provided at each switch for flow/congestion control and Quality-of-service algorithms that require per-packet or per-cell operations. Routers and gateways were subject to much the same questions as internal network switches.

Switching investigations were largely focused on detailed architectural choices for fixed-size ATM cell switching using a hardware paradigm, with the view that the fixed size allowed cost-effective and scalable hardware solutions. Issues concerned whether a sophisticated Batcher-Banyan design was necessary or relatively simple crossbar approaches could be used, how much cell buffering was needed to avoid excessive cell loss, whether the buffers should be at the input ports, output ports, intermediate points within the switch structure, or some combination of these choices, and whether input and output port controller designs should be simple or complex.

For variable-length PTM switching, issues concerned how to develop new software/hardware architectures to distribute per-port processing at gigabit rates while efficiently moving packets between ports, and how to implement network control functions within the new architectures. A key question was how much, if any, specialized hardware is necessary to move packets at these rates.

Other platform issues concerned the cost of achieving gigabit/s processing in specialized devices such as those needed for interworking different transmission technologies or for SONET cross-connect switching, and whether it was reasonable to accomplish these functions by processing data streams at the full desired end-to-end rate or alternatively to stripe the aggregate rate over multiple lower speed channels.

Software issues also existed within host platforms over and above transport and lower layer protocol processing. One set of issues concerned the operating system software used by each vendor, which like most platform hardware was designed primarily to support internal computation with little, if any, priority given to supporting efficient networking. In addition to questions concerning the environment provided by the operating system for general protocol transactions, an important issue concerned the introduction of multimedia services by external networks and whether sufficiently fast software response times could be achieved for passing real-time traffic between an application and the network interface.

Another host platform software issue concerned the presentation layer processing required to translate between data formats used by different platforms, for example different floating point formats -- because the translation must in general be applied to each word of data being transferred, it had the potential for being a major bottleneck.

Highly parallel distributed memory computer architectures which were coming into use in 1990 presented still another set of software issues for gigabit I/O. These architectures consisted of hundreds or thousands of individual computing nodes, each with their own local memory, which communicated with each other and the external world through a hardware interconnection structure within the computer. This gave rise to a number of questions, for example whether TCP and other protocol processing should be done by each node or by a dedicated I/O node or both, how data should be gathered and disseminated between the machine I/O interfaces and each internal node, and how well the different hardware interconnect architectures being used could support gigabit I/O data rates.

### *Application Issues*

The overriding application concern for host-to-host gigabit networking was what classes of applications could benefit from such high data rates and what kind of performance gains or new functionality could be realized.

Prior to the Initiative, many people claimed to have applications needing gigabit/s rates, but most could not substantiate those claims quantitatively. It was the competition for participation in the Initiative that led to ideas for applications that required ~ Gb/s to the end user. Essentially all the applications which were selected had in common the need for supercomputer-class processing power, and these fell into two categories: ‘grand challenge’ applications in which the wall-clock time required to compute the desired results on a single 1990 supercomputer typically ranged from days to years, and interactive computations in which one or more users at remote locations

desired to interact with a supercomputer modeling or other computation in order to visually explore a large data space.

The main issue for grand challenge applications was whether significant reductions in wall-clock solution time could be achieved by distributing the problem among multiple computers connected over a wide-area gigabit network. Here again, speed-of-light propagation delay loomed large -- could remote processors exchange data over paths involving orders of magnitude larger delays than that experienced within a single multiprocessor computer and still maintain high processor utilization?

While circumventing latency appeared to be a major challenge, another approach offered the promise of major improvements for distributed computing in spite of this problem. This was the prospect of partitioning an application among heterogeneous computer architectures so that different parts of the problem were solved on a machine best matched to its solution. For example, computations such as matrix diagonalizations were typically fastest on vector architectures, while computations such as matrix additions or multiplications were fastest on highly parallel scalar architectures. Depending on the amount of computation time required for the different parts on a single computer architecture, a heterogeneous distribution offered the possibility of s*uperlinear* speedups. (One definition of superlinear speedup is “an increase by more than a factor of N in effective computation speed, using N machines over a network, over that speed which the fastest of the N machines could have achieved by itself.”)

Thus issues for this application domain included how to partition application software so as to maximize the resulting speedup for a given set of computers, which types of computers should be used for a particular solution, what computation granularities should be used and what constraints are imposed by the application on the granularities, and how to manage the overall distributed problem execution. The last question required that new software tools be developed to assist programmers in the application distribution, provide run-time execution control, and allow monitoring of solution progress.

The second class of applications, interactive computations, can range from a single user interacting with a remote supercomputer to a large number of collaborators sharing interactive visualization and control of a computation, which is itself distributed over a set of computing resources as described above and which may include very large distributed datasets. An important issue for this application class is determining acceptable user response times, for example 100 milliseconds or perhaps one second elapsed time to receive a full screen display in response to a control input. This should in general provide more relaxed user communication delay constraints than the first application class, since these times are large enough to not be significantly impacted by propagation delay, and will also remain constant as future computation times decrease due to increased computing power.

Other issues for remote visualization include where to generate the rendering, what form the data interface should take between the data generation output and the renderer, how best to provide platform-independent interactive control, and how to integrate multiple heterogeneous display

devices. For large datasets, an important issue is how to best distribute the datasets and associated computational resources, for example performing preprocessing on a computer in close proximity to the dataset and moving the results across the network versus moving the unprocessed data to remote computation points.

Each of the above issues were examined in a variety of networking and application contexts and are described more fully in the referenced testbed reports. The investigations and findings are summarized in Section 4.

# 3 Structure and Goals

## *3.1 Initiative Formation*

The origins of the testbed initiative date back to 1987, when CNRI submitted a proposal to NSF and was subsequently awarded a grant to plan a research program on very high speed networks. The original proposal, which involved participants from industry and the university research community, was written by Robert Kahn of CNRI and David Farber of the University of Pennsylvania. Farber later became an active researcher on the follow-on effort, while CNRI ran the overall initiative. As part of this planning, CNRI issued a call for white papers in October 1988. This call, published in the Commerce Business Daily, requested submissions in the form of white papers from organizations with technological capabilities relevant to very high speed networking.

The selection of organizations to participate in the resulting testbed effort was carried out in accordance with normal government practices. A panel of fourteen members, drawn largely from the government, was assembled to review the white papers and to make recommendations for inclusion in the program. Those recommendations formed the basis for determining the government-funded participants. CNRI then worked with telecommunications carriers to obtain commitments for wide-area transmission facilities and with others in industry to develop a cohesive plan for structuring the overall program.

A subsequent proposal was submitted to NSF in mid-1989 for government funding of the non-industry research participants, with the wide-area transmission facilities and industrial research participation to be provided by industry at no cost to the government. A Cooperative Agreement, funded jointly by NSF and DARPA, was awarded to CNRI later that year to carry out testbed research on gigabit networks. The research efforts were underway by Spring 1990. Government funding over the resulting five-year duration of the project totaled approximately $20M, with these funds used primarily for university research efforts, with total value of industry contributions over this period estimated to be perhaps 10 or 20 times greater than the Government funding.

## *3.2 Initiative Management*

The overall effort was managed by CNRI in conjunction with NSF and DARPA program officials. Within NSF, Darleen Fisher of the CISE directorate, provided program management throughout the entire effort. A series of program managers, beginning with Ira Richer, were responsible for the effort at DARPA. Many others at both NSF and DARPA were also involved over the duration of the effort. In addition, each testbed had its own internal management structure consisting of at least one representative from each participating organization in that testbed; the particular form and style of internal management was left to each testbed's discretion.

The coordinating role of a lead organization, played by CNRI, was essential in helping to bridge the many gaps between the individual research projects, industry, government agencies and po-

tential user communities. At the time this effort began, there did not appear to be a clearly visible path to make this kind of progress happen.

To provide an independent critique of project goals and progress, an advisory group was formed by CNRI consisting of six internationally recognized experts in networking and computer applications. A different, yet similar by constituted, panel was formed by NSF to review progress during the second year of the project.

Administrative coordination of the testbeds was carried out in part through the formation of the Gigabit Testbed Coordinating Committee ("Gigatcc"), made up of one to two representatives from each participating testbed organization and CNRI/NSF/DARPA project management. The Gigatcc, chaired by Professor Farber, met approximately 3-4 times per year during the course of the initiative. In addition, each research organization provided CNRI with quarterly material summarizing progress, and each testbed submitted annual reports at the completion of each of the first three years of the initiative. Final reports for each testbed were prepared and are being submitted along with this document.

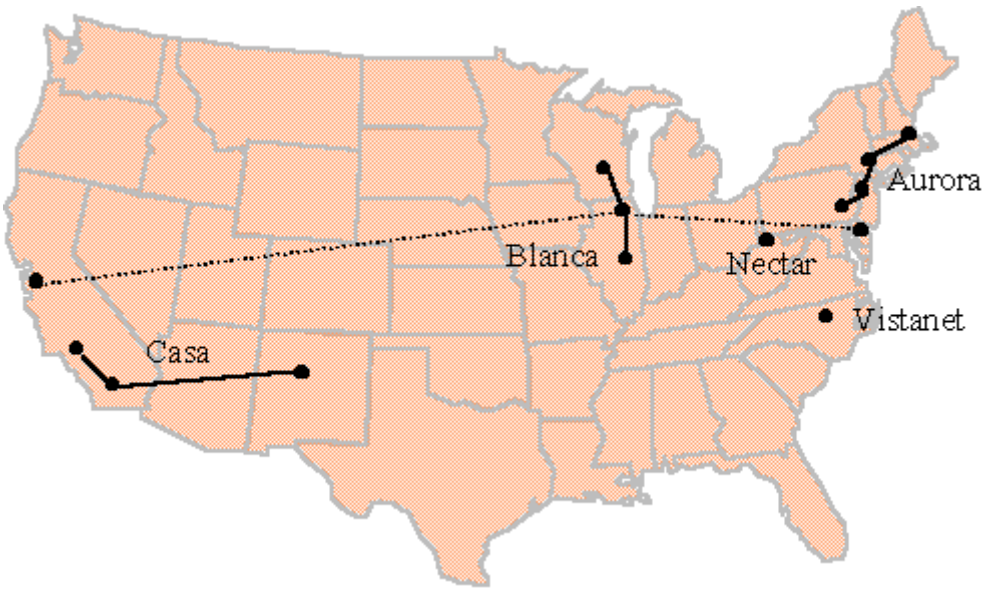
To encourage cross-fertilization of ideas and information sharing between the testbeds, CNRI held an annual three-day workshop attended by researchers and others from the five testbeds, plus invited attendees from government, industry, and the general networking research community. Attendance at these workshops typically ranged from 200-300 people, and served both as a vehicle for information exchange among project participants and as a stimulus for the transfer of testbed knowledge to industry. CNRI also assisted the U.S. Government in hosting a Gigabit Symposium in 1991,attended by over 600 individuals and chaired by Al Vezza of MIT.

A number of small inter-testbed workshops were also held during the course of the project to address specific testbed-related topics which could especially benefit from intensive group discussion. A total of seven such workshops were held on the following topics: HIPPI/ATM/SONET interworking, gigabit TCP/ IP implementation, gigabit applications and support tools, and operating system issues. In addition, an online database was established at CNRI early in the project to make information available via the Internet to project participants about new vendor products relevant to gigabit networking, and to maintain a list of publications and reports generated by testbed researchers.

## *3.3 The Testbeds*

The five testbeds were geographically located around the U.S. as shown in Figure 3-1.

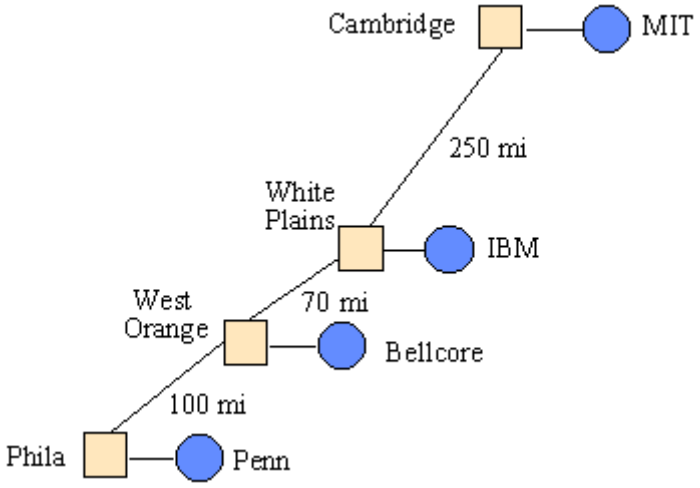
**Figure 3-1. Testbed Locations**




### *Aurora*

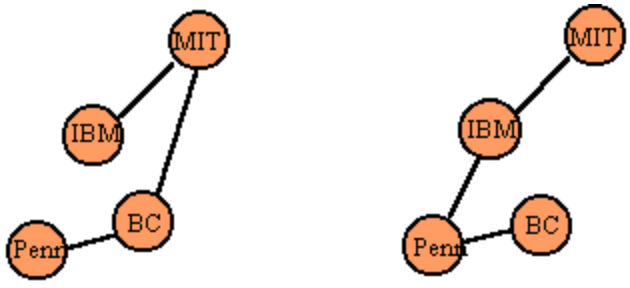
The original four Aurora testbed research participants were Bellcore, IBM Research, MIT, and Penn. Arizona was added as a research collaborator during 1992. Both Bellcore and IBM contributed state-of-the-art laboratory equipment and Bellcore provided financial support to MIT and the University of Pennsylvania. The participating carriers were Bell Atlantic and NYNEX for regional connectivity and MCI for long distance links between Bell Atlantic and NYNEX facilities.

**Figure 3-2. Aurora Physical Topology**




Four research sites were connected by carrier-provided SONET links through regional carrier equipment offices (the rectangles in above), using commercial SONET termination equipment at each research site. The research locations were Cambridge, Massachusetts; White Plains, New York; Morristown, New Jersey (via West Orange, N.J.), and Philadelphia, Pennsylvania. The facility design allowed two distinct linear topologies to be used in parallel, one for ATM experiments and one for PTM experiments, as shown in Figure 3-3.

**Figure 3-3. Aurora Logical Topologies**




(a) ATM-switched topology (b) PTM-switched topology

All the intersite physical links used SONET OC-48 2.5 Gbps transmission rates, with each link containing four SONET OC-12 622 Mbps full duplex logical channels. Three of these channels were used in the testbeds to realize the dual linear topologies shown in Figure 3-3, using manually configured SONET equipment routing connections within the carrier sites. Thus each research site could act as a switching point between two other sites for one of the topologies using two of the 622 Mbps channels, and as an endpoint in the other topology using the third 622

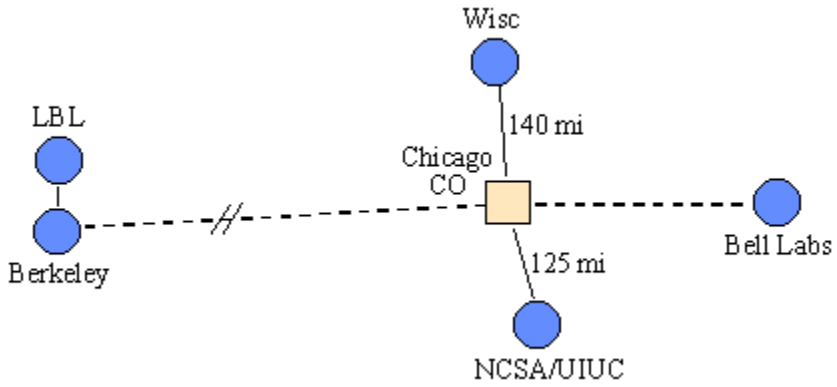
Mbps channel. Prototype wide-area ATM switches were provided by Bellcore, and prototype wide-area PTM switches were provided by IBM.

The research goals of the Aurora testbed were focused on gigabit networking technology for interconnecting workstation-class computers. Specific research topics included exploration of both ATM and PTM end-to-end wide-area switching technologies, design and evaluation of local ATM and PTM distribution technologies and their wide-area interworking, investigation of workstation interface architectures and operating system issues for gigabit networking, design and evaluation of wide-area control algorithms, and investigation of distributed shared memory for wide-area gigabit networking.

### *Blanca*

Research participants in the Blanca testbed included AT&T Bell Laboratories, UCBerkeley, Illinois, Lawrence Berkeley laboratory (LBL), the National Center for Supercomputing Applications (NCSA), and Wisconsin. AT&T provided the long distance facilities and wide-area ATM prototype switches. AT&T also provided portions of the local link facilities along with Pacific Bell and Bell Atlantic. Astronautics was also involved during part of the work. The physical topology of Blanca is shown in Figure 3-4, where solid lines indicate 622 Mbps or higher data rate links and dashed lines indicate 45 Mbps T3 links. Approximate distances are shown in the figure for the high speed wide-area portion of the testbed between Wisconsin and Illinois.

**Figure 3-4. Blanca Topology**



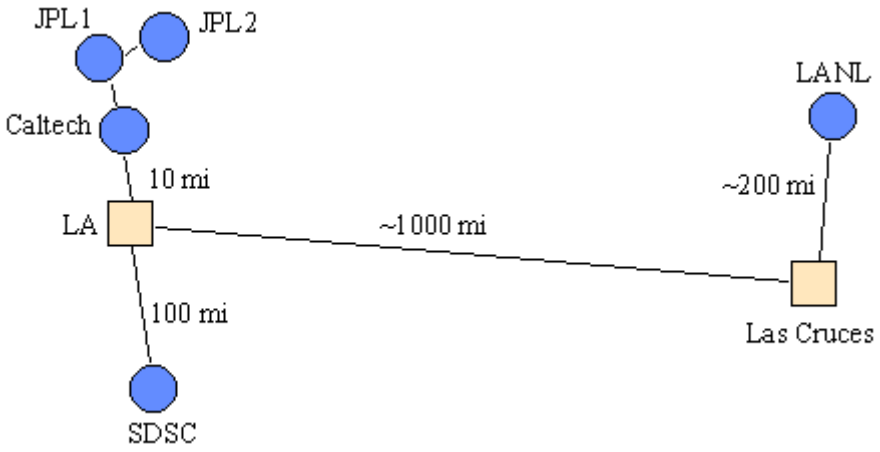

Two separate high speed portions of Blanca were established, one connecting the Illinois-NCSA campus located in Urbana-Champaign with Madison, Wisconsin via Chicago using all-optical links, and the second, using HIPPI technology, connecting the UCBerkeley Computer Science department with LBL, both located in Berkeley, California. The 45 Mbps links were part of a lower-speed university testbed previously established by AT&T called Xunet (experimental university network) which involved other sites besides those shown in Figure 3-4.

Blanca research goals included both wide-area network technology and supercomputer applications, with a strong emphasis on collaboration between the network and application researchers. Network technology research was focused on multimedia quality-of-service multiplexing strategies, flow/congestion control, ATM fast call setup, switch control software, distributed shared memory latency hiding strategies, and supercomputer gigabit-rate I/O software. Applications research was focused on remote visualization and collaboration for supercomputer modeling, for example thunderstorm modeling, and on the development of middleware software to support the remote visualization and collaboration involved in the applications.

### *Casa*

Casa testbed research participants included Caltech, JPL, Los Alamos National Laboratory (LANL), and the San Diego Supercomputer Center (SDSC) in conjunction with UCLA. Additional software development support was provided by the Parasoft Corporation. The participating carriers were MCI, Pacific Bell and US West, each providing SONET transmission and termination equipment. The physical Casa topology was that shown in Figure 3-5, with MCI providing long distance SONET OC-48 2.5 Gbps links between Las Cruces, New Mexico and a junction point in Los Angeles, and from Los Angeles to termination equipment located at SDSC in San Diego. Pacific Bell provided these SONET links between Los Angeles and Pasadena to connect Caltech and JPL, and US West provided the link from Las Cruces to Los Alamos, New Mexico. The approximate distance of each major link is shown in the figure.

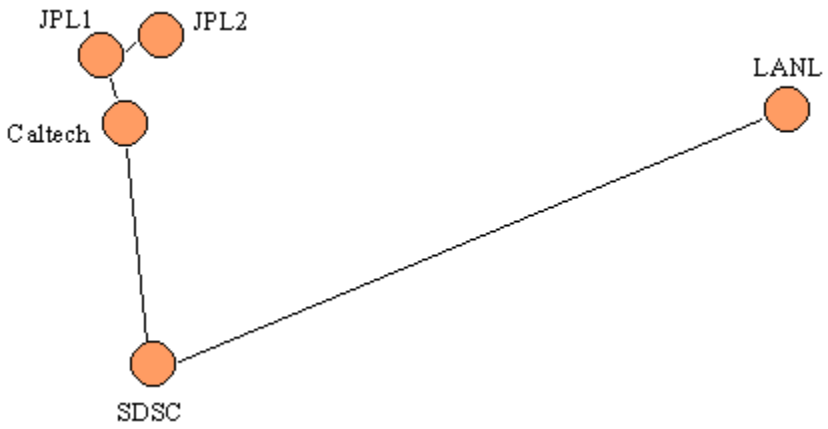
**Figure 3-5. Casa Physical Topology**




In Casa, the 2.5 Gbps links were divided into 16 SONET OC-3c 155 Mbps logical channels, with 8 of these channels made available at each site to provide a 1.2 Gbps aggregate full duplex rate for use by the researchers. The testbed logical topology is shown in Figure 3-6. While the Los Angeles junction for the LANL SONET link in Figure 3-5 was much closer to Caltech than to SDSC, carrier provisioning considerations resulted in the continuation of this link to SDSC using

1.2 Gbps of the OC-48 physical link between Los Angeles and San Diego, with the other 1.2 Gbps used to connect SDSC through Los Angeles to Caltech.

**Figure 3-6. Casa Logical Topology**

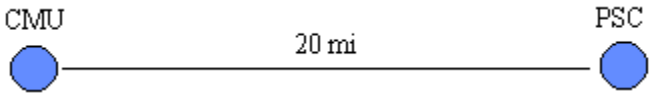



The primary focus of Casa research was on distributed heterogeneous supercomputing applications involving very large computational problems. Three representative applications were selected for investigation: global climate modeling, quantum chemical dynamics modeling, and geophysical modeling, using state-of-the-art supercomputers distributed among the sites. To support application distribution and experimentation, the Casa work included development of new software to provide distributed communication, execution control, and progress monitoring. Network technology research also played an important role in Casa, through the development of gateway technology to interface local HIPPI site distribution technology to wide-area SONET links and to use the local HIPPI sites for wide-area PTM switching, the exploration of outboard protocol processing using simple interfacing protocols, and the investigation of wide-area transport protocol performance at gigabit speeds.

### *Nectar*

The Nectar testbed research collaborators consisted of Carnegie-Mellon University (CMU), the Pittsburgh Supercomputing Center (PSC), and Bellcore. The participating carrier in this testbed was Bell Atlantic. Nectar explored issues for wide-area gigabit networking using the physical testbed shown in Figure 3-7. This configuration consisted of a carrier-supplied connection between the CMU campus and the PSC with specially designed ATM/SONET transmission equipment located at each site. The optical transmission portion was developed by Alcatel and ran at 2.4 Gbps. A HIII-ATM-SONET box developed by Bellcore allowed multiple OC-3 SONET channels to be aggregated. Each OC-3 channel contained HIPPI packets broken into multiple ATM cells.

**Figure 3-7. Nectar Topology**




This testbed was built on a previously established testbed local to the CMU campus, which originally operated at speeds of 100 Mbps. Some aspects of the earlier testbed's local networking architecture were carried over to the gigabit testbed, such as the use of crossbar switches for local distribution. The gigabit testbed connectivity allowed a number of different supercomputer architectures located at PSC during the project to be used in conjunction with workstations and an experimental parallel computer located at CMU.

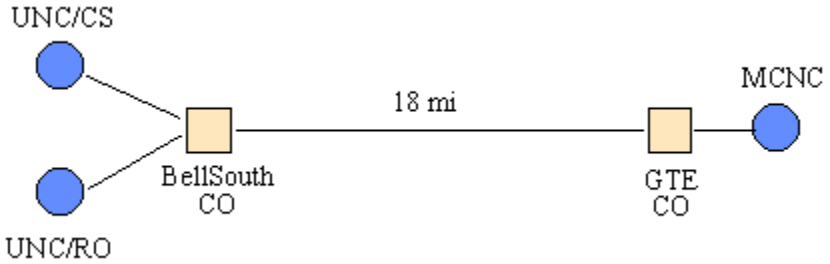
Nectar research goals included both networking technology and distributed computation, with a particular focus on developing new software programming environments for general application classes using both supercomputers and workstations. In the network technology area, the testbed investigated solutions to the problem of wide-area ATM/SONET carrier network access at gigabit rates when different local area technologies are used, and on host I/O operating system and protocol issues for both workstations and highly parallel distributed memory machines. Two types of applications were used in the work, one for distributed heterogeneous computation of chemical plant process modeling and the second for distributed homogeneous computation of large combinatorial problems using a network of workstations.

### *Vistanet*

Vistanet testbed research participants included the Computer Science and Radiation Oncology departments at the University of North Carolina (UNC), BellSouth Research, GTE Laboratories, and MCNC in conjunction with North Carolina State University (NCSU). BellSouth and GTE also provided financial support to the research participants plus SONET transmission facilities for this testbed, with the physical topology and distances involved shown in Figure 3-8. In addition, BellSouth provided a prototype ATM wide-area carrier switch in their Chapel Hill central office (CO) for testbed use and GTE provided a cross-current switch in their central office.

The Vistanet testbed was unique in its use of ATM switching within a carrier CO. This provided a star switching topology in which all ATM traffic between MCNC, UNC/CS and UNC/RO was switched within the Chapel Hill CO. The testbed was also unique in its use of synchronous OC-12c 622 Mbps SONET channels, rather than striping over lower speed channels, in conjunction with its OC-48 2.5 Gbps trunking.

**Figure 3-8. Vistanet Topology**




Vistanet research goals were centered around the use of an interactive 3-D cancer radiation treatment planning medical application, which provided a focus for investigating a number of networking as well as application issues. The testbed provided connectivity between a super-computer at MCNC, a special-purpose parallel rendering computer at UNC/CS (Computer Science), and a medical visualization workstation at UNC/RO (Radiation Oncology). In addition to investigating how to achieve interactive user response times in conjunction with the distrib-uted computation, a specific research focus involved an exploration of new software graphics techniques for providing gigabit network-based interaction. Network technology goals included developing techniques for interfacing local networks with wide-area switched ATM/SONET at synchronous 622 Mbps speeds, exploring outboard protocol processing, evaluating different transport protocols, and investigating real-time data traffic capture and characterization using actual application traffic.

# 4 Investigations and Findings

This section summarizes the testbed investigations and findings. The emphasis here is on bringing together the work by topic area -- detailed individual testbed perspectives are provided in the final reports written by each of the testbed research organizations. The topics are:

- transmission
- switching
- interworking
- host I/O
- network management
- applications and support tools

Each subsection begins with a brief summary of the investigations and findings for that area. Figures have been included to assist in conveying some of the information, but readers are encouraged to consult the individual reports for more complete expositions of the material. Specific references are given where a particular report or publication is especially relevant -- in other cases, the final report(s) for the relevant testbed or organization should be assumed as the implicit reference.

One of the stated project goals was to “investigate alternative architectures for gigabit networking”. Because of the relatively primitive stage of gigabit technology existing at the beginning of the project and the experimental nature of the testbeds, this of necessity translated into investigations focused on the elements required to realize such a network, rather than on a set of “complete” network architectures. In fact, the rapid pace of today’s technology changes effectively dictates an incremental approach to successful networking systems -- a top-down system design is almost assured of being overtaken by technology advances before it can be fully implemented.

Nevertheless, at least four distinctive end-to-end network layer architectures were used as research platforms in the testbeds. This was a result both of architecture component choices made by researchers after the work was underway and of the a priori testbed selection process discussed in an earlier section. Figure 4-1 gives a high-level view of these end-to-end architectures.

**Figure 4-1. Testbed Architectures**

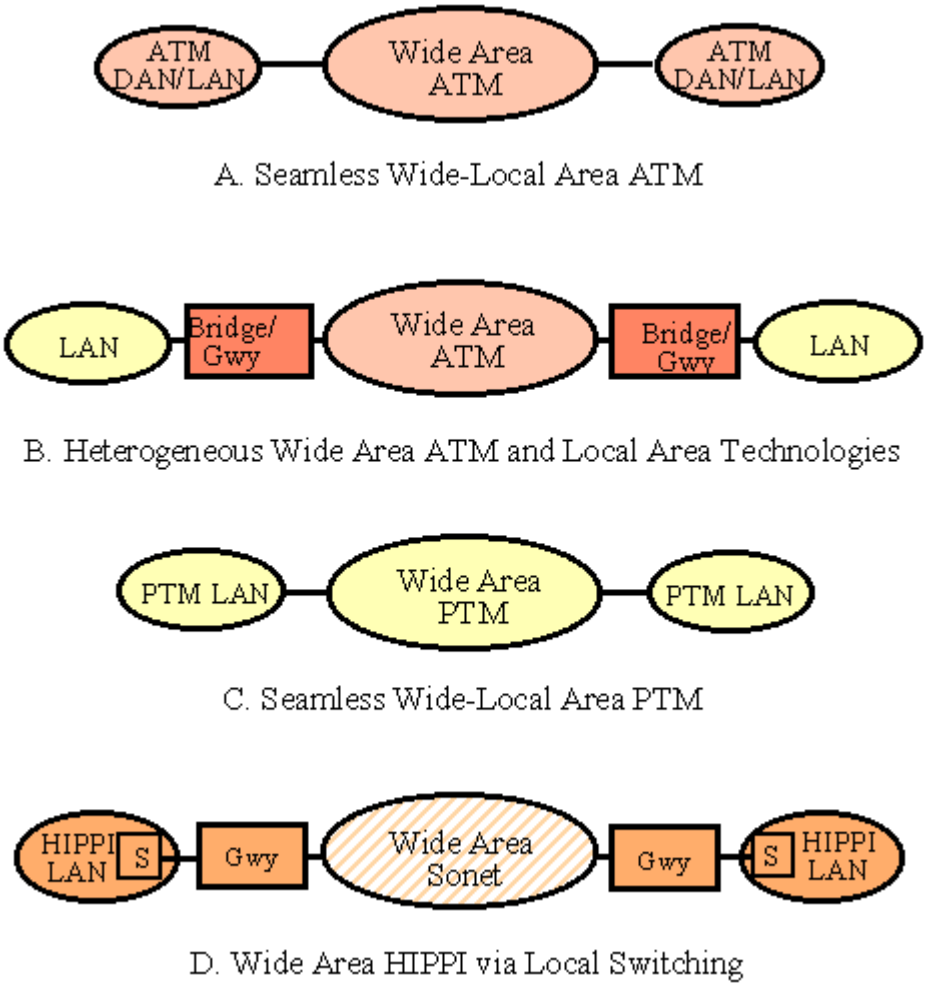



Figure 4-1A, Seamless Wide-Local Area ATM, reflects one of the resulting architectures which was investigated in the Aurora testbed. In this case all networks use ATM cell switching and are interconnected without use of an IP layer. SONET is used as the underlying transmission technology for the wide area network, and in some cases also for local distribution. In other cases other transmission technologies are used under ATM in the local area. In addition to the usual notion of LANs, Aurora introduced ATM Desk Area Networks, or DANs, as part of the overall architecture.

Figure 4-1B, Heterogeneous Wide Area ATM and Local Area technologies, shows the architecture used in the Blanca, Nectar, and Vistanet testbeds. ATM is again the switching technology used in the wide area network, but a non-ATM technology (in these instances, HIPPI) is used for local area connectivity. Both bridging and gateway approaches were investigated, and wide area transmission included SONET and all optical infrastructures.

Figure 4-1C, Seamless Wide-Local Area PTM, is the Packet Transfer Mode analogue to the seamless ATM architecture and, like the latter, was part of the Aurora testbed work. In this case variable-length packets are forwarded across both the local and wide area networks, which were designed to operate as an integrated system.

Figure 4-1D, Wide Area HIPPI via Local Switching, reflects the resulting Casa testbed architecture. In this case HIPPI was used for both local and wide area switching, with SONET providing the wide area transmission infrastructure. The combination of one or more specially designed gateways at each site, in conjunction with a site's local HIPPI switch, provided the wide area routing/switching structure for variable-length packet forwarding through intermediate sites.

The following subsections address, by topic, the technology investigations associated with these architectures, along with the work on applications and the metalevel aspects of the testbeds. It should be noted that, while some work was specific to a particular architecture, in many cases the results can be applied to one or more of the other architectures of Figure 4-1 and to other architectures not considered here.

## *4.1 Transmission*

### *Summary*

- OC-48 SONET links were installed in four testbeds over distances of up to 2000 km, accelerating vendor development and carrier deployment of high speed SONET equipment, establishing multiple-vendor SONET interconnects, enabling discovery and resolution of standards implementation compatibility problems, and providing experience with SONET error rates in an operational environment
- Testbed researchers developed a prototype OC-12c SONET cross-connect switch and investigated interoperation with carrier SONET equipment, and developed OC-3c, OC-12, and OC-12c SONET interfaces for hosts, gateways and switches, with these activities providing important feedback to SONET chip developers
- Techniques for carrying variable-length packets directly over SONET were developed for use with HIPPI and other PTM technologies, with both layered and tightly coupled approaches explored
- An all-optical transmission system was installed and used to interconnect ATM switches over a 300 mile area using optical amplifier repeaters, and was the first carrier-based service deployment of this technology
- HIPPI technology was used for many local host links and for metropolitan area links through the use of HIPPI extenders and optical fiber; other local link technologies included Glink and Orbit
- Several wide area striping approaches were investigated as a means of deriving 622 Mbps and higher bandwidths from 155 Mbps ATM or SONET channels; configu-

rations included end-to-end ATM over SONET, LAN-WAN HIPPI over ATM/SONET, and LAN-WAN HIPPI and other variable-length PDUs directly over SONET

- A detailed study of striping over general ATM networks concluded that cell-based striping should be used which can be introduced at LAN-WAN connection points, in conjunction with destination host cell re-ordering and an ATM-layer synchronization scheme

Transmission technology was fundamental to the establishment of the gigabit testbeds. The emergence of SONET/SDH standards for high speed transmission over wide area optical fiber in the late 1980s, and of HIPPI for local high speed connectivity in the same time frame, provided an opportunity both to construct experimental testbed facilities using prototype equipment and to accelerate that equipment's path to successful use in operational networks.

While SONET and HIPPI were the dominant transmission technologies used in the testbeds, other technologies were also used. These included wide area all-optical links using optical amplifier repeaters and new local area gigabit technologies such as Glink, which became available during the course of the project.

An important research focus in this area was the exploration of striping techniques to derive gigabit user rates from multiple lower-speed SONET channels. This was necessitated by the fact that most of the early SONET equipment available to the testbeds provided only 155 Mbps user ports – however, striping was also pursued as a research topic in its own right, since the aggregation of multiple lower speed network channels to achieve higher bandwidths becomes increasingly attractive as user data rate requirements increase and multichannel technologies such as all-optical WDM are introduced.

### *4.1.1 SONET*

The testbed initiative resulted in major advances in the use of SONET technology through its use in four of the five testbeds, ranging from the establishment of long-distance connections spanning over 2000 km to its use as a local interconnect technology. The major areas of activity centered around establishing and using carrier-provisioned SONET links, designing and experimenting with customer-premises SONET access, exploring the use of striping over parallel SONET channels, and interworking SONET with other technologies (the last topic is discussed in a later section).

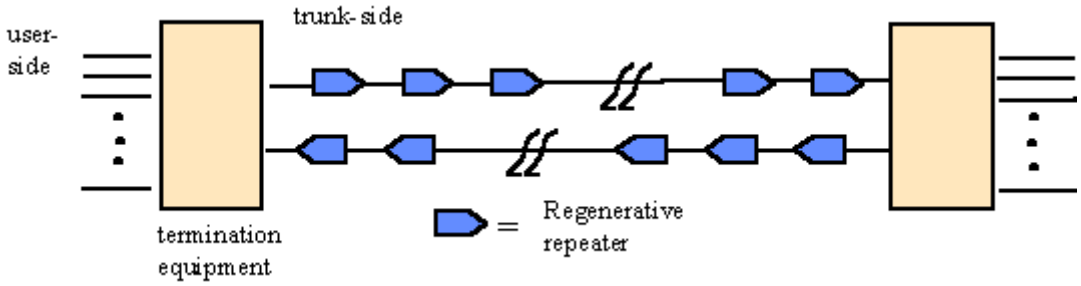
#### *Carrier-provisioned SONET Links*

To provide vendor-based SONET equipment for use in the testbeds, the participating testbed carriers worked together early in the project to develop specifications for the SONET equipment required in the two long-distance SONET-based testbeds, Aurora and Casa. At the time of this activity in 1990, a very limited range of equipment was available commercially from vendors, with minimal support for user-side interface rates above T3. The collective testbed needs for 622

Mbps user interfaces and unified procurement from the testbed carriers provided a major stimulant to SONET equipment vendors to accelerate delivery of higher speed equipment, with the result that prototype and in some cases operational versions of such equipment became available to the testbeds in the 1992-93 time frame.

The equipment which was subsequently deployed in the SONET-based testbeds operated at 2.5 Gbps on the trunks, and had either 155 or 622 Mbps user-side interfaces. The total bandwidth allocated between any two endpoints within the 2.5 Gbps trunks ranged from 622 Mbps to 1.2 Gbps, with the total endpoint rate achieved either through aggregating multiple 155 Mbps channels or through the use of synchronous 622 Mbps channels. Regenerative repeaters were used between termination points on the SONET links to make up optical fiber signal losses through digital signal recovery and amplification (Figure 4-2).

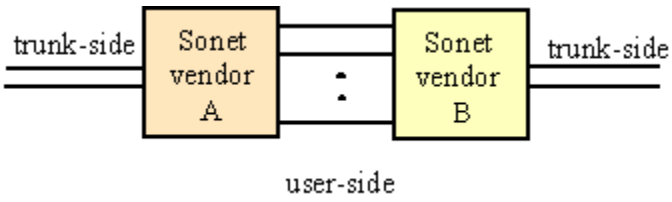
**Figure 4-2. SONET Signal Regeneration**




An area of major interest to the carriers was the potential problems which might be encountered when SONET equipment from different manufacturers were interconnected, in particular problems arising due to different interpretations of the SONET standard or of problems not foreseen by the standard developers. Both the Aurora and Casa testbeds provided an opportunity to deal with multi-vendor SONET equipment issues. In Aurora, the regional carriers NYNEX and Bell Atlantic both chose to install SONET equipment made by NEC, while MCI, the regional interconnect carrier, selected Northern Telecom equipment for its long-distance links. The Casa testbed included SONET equipment from three different vendors: Fujitsu, Northern Telecom and NEC.

In both testbeds, the multi-vendor interconnections were handled by collocating equipment from each vendor within the long-distance carrier (MCI) points of presence and using the user-side interfaces of the equipment for the connections between them (Figure 4-3)**.** This arrangement allowed a single carrier to discover and resolve interconnection issues existing between different vendor equipment without also needing to coordinate among different carriers.

**Figure 4-3. SONET Multivendor Interconnect**




The use of different vendor equipment was highly beneficial to the goals of the testbed, for example some differences of interpretation of the SONET standard were discovered and resolved through coordination between the carriers and the vendors. More generally, important experience was gained with the installation and operation of this new equipment and of its performance over both short and long distances. Valuable experience was also gained in understanding how to deal with errors occurring on the SONET links [4, 6].

### SONET Crossconnect

In addition to establishing SONET links, one of the testbed research groups developed and experimented with a SONET crossconnect switch. This switch allowed on-demand interconnection of full 622 Mbps SONET links, using synchronous OC-12c interfaces for connection to individual OC-12c SONET links provided by a carrier for that testbed. The design for this device was based on prototype lower-speed SONET chips just becoming available in the 1990-1991 timeframe, using advanced techniques to achieve synchronous 622 Mbps operation with chips operating individually at 155 Mbps. This activity clearly established the feasibility of building relatively low-cost equipment which operated with OC-12c links, and provided another valuable vehicle for exposing misinterpretations of or inconsistencies in the SONET standards through its installation and use within the Vistanet testbed.

### Customer Premises Access

The carrier-provided equipment made OC-3c, OC-12, or OC-12c SONET interfaces available to testbed researchers through location of SONET termination equipment at the researcher's sites. To connect to the SONET links, a number of different research groups within the testbeds developed host, gateway or on-premises switch interfaces using prototype SONET chips to carry out the various functions required to deal with SONET data streams such as framing and synchronization (while such chips were just becoming available in prototype form, off-the-shelf equipment with SONET interfaces were rare at that point in time). Since four of the five testbeds used SONET for long-distance transmission, this resulted in a substantial body of experience in dealing with the multiple kinds of SONET interfaces made available to the testbeds.

For example, the Vistanet testbed used synchronous 622 Mbps SONET user interfaces, and so needed to deal with the issues of interfacing to a single synchronous bit stream at that speed in developing gateway equipment to connect HIPPI-based hosts within the testbed to their ATM/SONET wide-area equipment. In the Aurora testbed, the carrier equipment provided OC-

12 interfaces at each site-in this case each SONET user port on the carrier equipment operates at a 622 Mbps rate but consists of four independent 155 Mbps logical channels. In the Casa testbed, the carrier equipment provided multiple OC-3c 155 Mbps physical user-side interfaces, while prototype equipment developed for the Nectar testbed multiplexed multiple STS-3c logical SONET channels into a SONET OC-48 2.5 Gbps trunk.

To further extend the experimentation opportunities offered by the testbeds, in many instances several different kinds of equipment were developed with SONET interfaces to explore differing approaches to dealing with particular networking problems, for example the interworking of other network technologies with SONET and in interfacing SONET links to different kinds of host architectures. These activities are discussed in more detail in their respective sections below on interworking and host interfacing.

**Variable-length Packets**

A major anticipated use of SONET was to carry ATM cells, and as a result methods were defined for mapping ATM cells into SONET payloads as part of the international B-ISDN standards. More generally, the availability of ATM formatting standards circa 1990-91 allowed testbed equipment to be developed for ATM/SONET interfacing (with respect to an individual SONET channel) in a relatively straightforward manner within the testbeds. This was carried out in the Aurora, Nectar and Vistanet testbeds at SONET channel rates of both 155 and 622 Mbps through implementations associated with ATM switches, gateways, and host interfaces.

The mapping of variable-length packets into SONET channels, on the other hand, had not been addressed prior to the testbed work. The investigation of PTM (Packet Transfer Mode) technologies in the testbeds required that techniques be developed for accomplishing PTM/SONET interfacing at the high data rates used in the testbeds. Two of the testbeds, Aurora and Casa, addressed this problem.

In the Aurora testbed, IBM explored the use of PTM networking technology through their Planet PTM switch and Orbit LAN token ring technologies. Their solution to the PTM/SONET mapping problem, based on an optimization analysis, consisted of prefixing each packet with a 32-bit header containing the packet length in the first 16 bits and a CRC in the second 16 bits, with the CRC computed on the first 16 bits. The serial bit stream delivered by the SONET link is scanned for a correct CRC result, establishing the beginning of a packet; the packet length information is used to determine the end of the packet. This process is repeated after each packet is received, providing a link-level framing protocol residing above the SONET transmission layer.

The Casa testbed provided a second opportunity for investigating variable-length packet operation over SONET, in this case driven by the use of HIPPI-based technology at each user site and a direct mapping of HIPPI packets into SONET frames for long-distance transmission over dedicated SONET links. This solution, developed by LANL as part of their HIPPI-SONET gateway work, differs significantly from the Aurora solution in that it is tightly coupled to SONET framing. In particular, 72 bits are sent by the gateway at the beginning of each SONET Synchronous Payload Envelope (SPE), the part of the SONET frame which carries user data, to provide con-

trol information to the receiving gateway. The initial HIPPI packet sent on the link is aligned to begin immediately following these control words within the SPE; the end of the variable-length HIPPI packet is determined at the receiving gateway by detecting a special set of token bits appended to the packet.

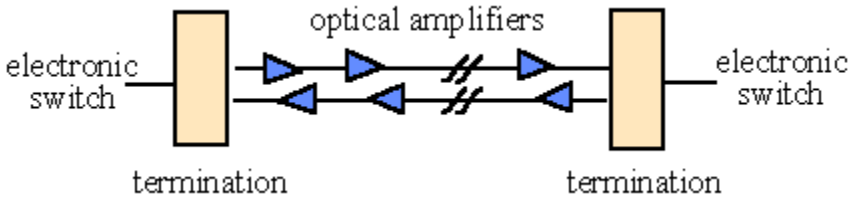
Thus in the case of Casa a combination of SONET-level framing and serial data stream scanning is used to provide robustness in the presence of errors while taking advantage of control structures present for other purposes. While this solution is tied to word sizes used in HIPPI, both this and the Aurora solution have demonstrated the viability of carrying variable-length packets over SONET links and have provided significant directions for future standardization in this area.

### *4.1.2 Other Transmission Technologies*

#### Wide Area All-optical Transmission

While four of the testbeds advanced the understanding and use of SONET transmission equipment, the Blanca testbed explored the use of all-optical wide area transmission through the use of optical fiber amplifiers over a total distance of nearly 300 miles (Figure 4-4). Eighteen amplifiers were installed on a Madison-Chicago-Champaign path, with optical termination equipment connected to Xunet switches at each of the three sites. The all-optical links provided protocol-transparent transmission, allowing the proprietary transmission format used by the Xunet switches to be replaced at a later date by SONET formats without changes to the link installations.

**Figure 4-4. All-Optical Transmission**




The link provided a 622 Mbps channel for use by the testbed in both directions between Madison and Champaign, and in addition was shared with other AT&T operational services using other capacity available on the optical link. This was the first deployment of optical fiber amplifiers as repeaters on a service basis by a carrier. The testbed-based link deployment allowed data to be collected by AT&T on link dispersion, signal-to-noise ratio, and bit error rates.

#### HIPPI

Most of the testbeds used HIPPI for their local distribution technology. This was primarily due to the fact that, at the beginning of the project in 1990, HIPPI was the only standards track gigabit-range technology on the scene. HIPPI originated in the late 1980s at LANL as a solution to local

high speed interconnection of supercomputers, providing an 800 Mbps full duplex point-to-point connection over parallel copper wire. By 1990 it was in the process of becoming an ANSI standard and was also beginning to be supported by industry, most notably Cray Research (whose supercomputers at the time dominated the computation centers participating in the testbeds) and Network Systems Corporation (NSC). The latter company had won a contract to develop a HIPPI-based device for workstation interconnection for one of the testbeds, and was also developing a local area HIPPI switch.

HIPPI thus provided a practical solution to local interconnection for the testbeds, but was significantly limited by the relatively short 25 meter distance allowed by its parallel copper wire technology and by the size, complexity and cost of the associated host interface. Two 100-wire cables were required for full duplex data transmission, resulting in a large host connector footprint and relatively inflexible and bulky cables. Nevertheless, because of its growing support for supercomputers and a commercially available switch, it became the dominant interconnection technology within supercomputer centers in the early 1990s, and by extension to the workstations used in conjunction with the supercomputers within the testbeds. The Aurora testbed was the exception, since its research was focused on workstations and ATM in both local and wide area environments, and also was not dependent on the constraints imposed by supercomputers.

**Local Area Fiber Technologies**

HIPPI's 25-meter distance constraint and the needs of the testbeds inspired the development of an optical fiber-based HIPPI extender by industry. The first extender product became available from Broadband Communication Products (BCP) through their involvement with testbed researchers in the early years of the project.

The HIPPI extender allowed a full duplex HIPPI connection to be extended across a pair of single-mode optical fibers over distances of up to 10 km. In addition to relaxing the constraints on interconnecting HIPPI devices within a campus environment, these extenders were able to support cross-town metropolitan area connections for some of the testbeds as a workaround while waiting on the development of SONET-related equipment. While the direct extension of HIPPI flow control signaling across longer distances can result in significant throughput degradation, the extender's usefulness nevertheless resulted in the definition of an optical fiber serial HIPPI standard.

Two other fiber-based technologies used for local interconnection in the testbeds were HP's Glink and IBM's Orbit, both of which operated at a rate of 1 Gbps. Glink was used to provide point-to-point links between hosts and local switches in local Aurora configurations. Orbit is a buffered ring LAN technology developed by IBM, and was used in the Planet/Orbit networking portion of Aurora.

An alternative to HIPPI, Fibre Channel (FC), was also being developed for ANSI standardization at the time the project began, and was targeted for use as the local area networking technology in the Blanca testbed. FC provided the same 800 Mbps user rate as HIPPI, but with much smaller host connector profiles and longer operating distances through its direct use of optical fiber in-

stead of HIPPI's parallel copper wires. The realization of a well-defined FC standard and useable products was significantly delayed, however, due at least in part to FC's large functional scope and complexity. When by spring 1992 FC products were still not available for 800 Mbps operation, Blanca researchers decided to use HIPPI and so FC was not used within the testbeds.

### *4.1.3 Striping*

In dealing with high performance networking applications, end-to-end bandwidth demands often cannot be met by the individual port rates of currently available wide-area transmission equipment. In particular, the Aurora, Casa and Nectar testbed SONET equipment available at the beginning of the testbed program had only 155 Mbps physical and/or logical channels for host connections, with aggregation of the channels required to fully use the total available bandwidth. In addition to addressing the immediate problem, this aggregation, or *striping*, work afforded an opportunity to investigate the wide-area rate mismatch problem more generally.

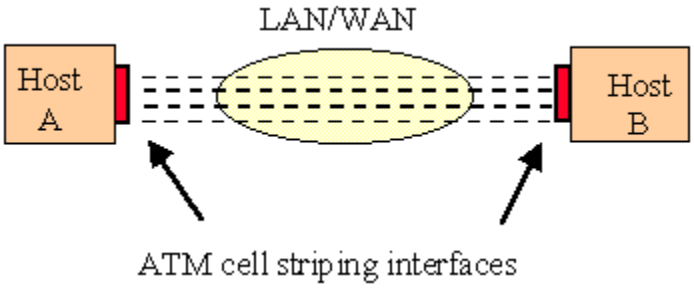
A number of choices are available for implementing a striping solution [2]. These include the amount of data to be sent in each striped increment, the protocol layer in which the striping is performed, and the topological point at which it is applied. Choices for the striping unit include byte, word, cell and packet; layer choices include physical, link, network, transport and application; topological points include physical link connections, subnet interfaces, and user endpoints. Some choices constrain others-for example choosing a physical link within an ATM network as the topological point clearly constrains the layer and striping unit choices. Because of skew introduced by different delays in each striped channel due to multiplexing equipment and switches, key factors which must be dealt with are synchronization, buffering, and reordering processing.

#### ATM Striping

The Aurora testbed had OC-12 SONET interfaces available at each site. These interfaces required the customer interfacing equipment to multiplex/demultiplex four 155 Mbps STS-3c SONET channels to/ from a single 622 Mbps STS-12 SONET channel.

For the ATM portion of Aurora, striping was implemented at the topological endpoints by Bellcore as part of their workstation host interfaces developed for the testbed, with the 53-byte ATM cell chosen as the striping unit (Figure 4-5).

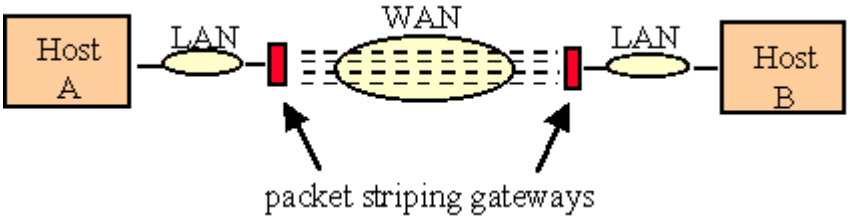
**Figure 4-5. Aurora Bellcore ATM/SONET Striping**




Since one or more ATM switches were in general part of the network path between hosts, the striping mechanism had to deal with skew introduced by both SONET equipment and switch queuing. A separate VC was used for each of the four 155 Mbps ATM channels and AAL5 used to frame the host PDU. The cells of the AAL5 frame were then striped across the four VCs in round-robin order, with the AAL5 end-of-frame bit set in the ATM header of the last cell sent in each of the four channels (rather than just in the last AAL5 frame cell, as would normally be done). To simplify synchronization at the receiver, the first cell of a new PDU is always sent in one of the four channels which has been designated the "first" one. The receiving host reorders the arriving cells into a standard AAL5 PDU.

A distinctly different approach to striping was used in the Nectar testbed. One of the goals of Nectar was to explore generic methods for interfacing local high-speed networks to a general wide-area ATM/ SONET network with lower-speed channels, and so the local-to-wide area interconnection point was chosen as the topological striping point (Figure 4-6). This testbed used 800 Mbps HIPPI for local distribution at each site, with the sites connected by a 2.5 Gbps SONET link. The link was terminated at each site by experimental ATM/SONET multiplexing equipment developed for the testbed by Bellcore and CMU, with the 2.5 Gbps channel divided into 16 STS-3c 155 Mbps SONET channels for ATM interfacing within the equipment.

**Figure 4-6. Nectar ATM/SONET Striping**

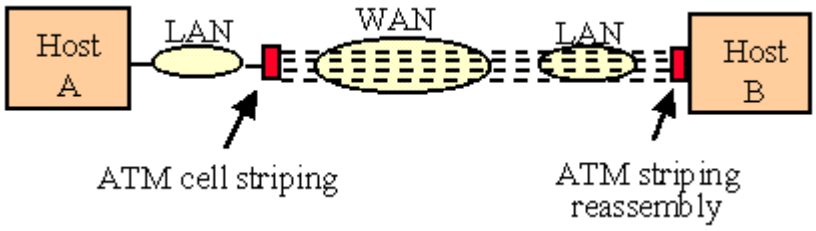



Consideration of equipment design complexity and cost associated with the choice of byte, cell, and packet striping units led to the selection of a packet striping unit, where packets in the initial implementation consisted of HIPPI frames. A distinct ATM VC is used for each 155 Mbps stripe, with the HIPPI frames which are received on the set of VCs sequentially combined into

the original HIPPI stream for delivery to the local destination. This simplification is at the expense of performance, however, since at least four packets must be available for striping to achieve the full 622 Mbps rate when four 155 Mbps channels are used-if only one packet is sent, it will be at a rate of 155 Mbps.

More general approaches to striping over ATM networks were explored by MIT as part of the Aurora testbed work, and in particular for cases where striping is introduced at points internal to the end-to-end path between hosts. This work examined different methods for achieving ATM layer striping synchronization in great detail, along with the tradeoffs associated with higher-layer striping choices. The conclusion was that ATM layer striping, introduced, say, at a point of local-to-wide area connection, coupled with destination host cell reordering will provide the broadest support for different choices of higher layer protocols and topologies (Figure 4-7).

**Figure 4-7. Integrated ATM Layer Striping**




**PTM Striping**

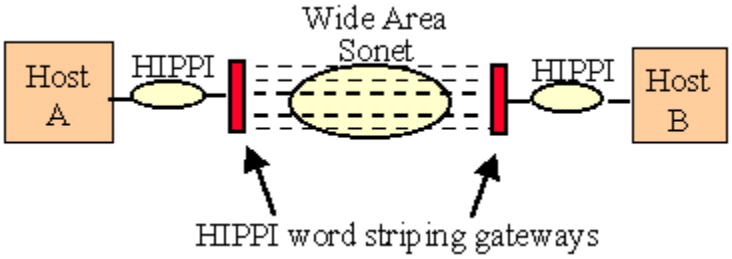
In two of the testbeds, Aurora and Casa, variable-length packets were striped directly across wide area SONET transmission channels at the local-to-wide area connection points within the testbeds.

For the PTM portion of Aurora, IBM designed and implemented a stand-alone device called the SIA (SONET Interface Adapter), which has an optical fiber connection on one side to the OC-12 interface of wide area SONET carrier equipment and an optical fiber connection on the other side to a Planet switch. In this case the striping traverses SONET equipment on the links between each Aurora site, but does not include switches within the striped paths. The PTM operation of the network made either packets or bytes the natural choice for the striping unit, with byte striping chosen for implementation. SONET-level framing information for each STS-3c payload envelope is used by the receiving SIA to determine skew and control byte reordering of the FIFO buffer outputs from each STS-3c channel. To reduce complexity of the SIA design, transmission of each new packet is always begun using the first STS-3c payload envelope of an STS-12 frame.

In the Casa testbed, SONET termination equipment at each site provided eight distinct OC-3c physical interfaces on the user side, which were multiplexed by the SONET equipment directly

into a 2.5 Gbps OC-48 intersite trunk. Thus an aggregate bandwidth of 1.2 Gbps was available for use between each pair of sites which were directly interconnected. A HIPPI-SONET gateway device was developed by LANL to interface each site to the SONET termination equipment, with striping used to send 800 Mbps HIPPI streams across the aggregated 155 Mbps channels (Figure 4-8). The remainder of the 1.2 Gbps trunk channel was used for error correction data and for spare capacity in the event of individual channel failures.

**Figure 4-8. Casa PTM Striping**




As in the Aurora PTM case, Casa striping was applied at the endpoint of each intersite link, with each link assumed to contain wide-area SONET equipment but not switches and with variable-length packets mapped directly into SONET frames. Since all Casa sites used HIPPI for local switching and host interfacing, and the gateway was designed to provide efficient HIPPI-SONET communications, the unit of striping was chosen to carry two 32-bit HIPPI words plus four per-word overhead bits, for a total striping unit of 72 bits. The actual trunk bandwidth used could be matched to a site's needs, ranging from a single 155 Mbps channel to the use of six channels to carry the full 800 Mbps HIPPI rate along with an FEC channel and a spare channel. A combination of SONET framing detection and payload control information is used by a receiving gateway to perform skew compensation and reassemble the source data stream.

Overall, the testbed implementations revealed that skew can be introduced from a number of different sources, and that considerable care must be taken to ensure that correct destriping is achieved for all cases if full striping performance is to be achieved. The general ATM-layer synchronization methods explored MIT provide a way to avoid dependence on lower and upper layer functionality while also achieving topological flexibility, and may provide a good basis for standardization in the ATM networking case. The two PTM striping cases both make use of SONET framing information to accomplish striping synchronization, but differ in the details as well as in their choice of striping unit. Further work is needed to determine what direction should be taken for standardization in the PTM case and to deal with hybrid ATM/PTM situations.

## *4.2 Switching*

### *Summary*

- Prototype high speed ATM switches were implemented and deployed for experiments in several of the testbeds, supporting 622 Mbps end-to-end switched links using both 155 Mbps striping and single-port 622 Mbps operation
- First telco central office broadband ATM switch was installed and used for testbed experiments, using OC-12c links to customer premises equipment and OC-48 trunking
- Wide area variable-length PTM switching was developed and deployed in the testbeds using IBM's Planet technology and through the use of HIPPI switches in conjunction with collocated wide area gateways
- Both ATM and PTM technologies were developed and deployed for testbed local and desk area networking experiments, along with the use of commercial HIPPI and ATM switches which became available as a result of testbed-related work
- A TDMA technique was developed and applied to tandem HIPPI switches to demonstrate packet-based QoS operation in HIPPI circuit-oriented switching environments, and a study of preemptive switching of variable length packets indicated a ten-fold reduction in processing requirements was possible relative to processor-based cell switching

Several different experimental wide area switches and newly developed local area switch products were used in the testbeds. In addition, as part of achieving overall testbed research objectives, new system-level approaches to gigabit network switching were explored for achieving Desk Area Networking, local and wide area flexible HIPPI switching, and the relaxation of switching rate requirements based on preemptive techniques.

### *4.2.1 ATM*

Experimental Wide area ATM switches included Bellcore's Sunshine switch used in Aurora, AT&T's Xunet switch used in Blanca, and a Fujitsu Fetex 150 prototype switch used in Vistanet. In addition, desk area ATM switching was developed as part of MIT's VuNet work in Aurora, and Fore Systems ATM switches were used in experiments at the University of Illinois in the Blanca testbed.

#### Sunshine Switch

The Sunshine switch had been designed and simulated by Bellcore prior to the start of the testbed project, and was intended as a prototype for carrier wide area ATM switching. The Aurora testbed provided an opportunity for Bellcore to build and experiment with the Sunshine design, with switch prototypes completed and deployed within Aurora during 1993. The design was based on a Batcher-Banyan switching fabric which switched internal 155 Mbps ATM cell streams. To provide the 622 Mbps connections required for the testbed, a set of four STS-3c channels could

be defined to be a *trunk group* and treated by the switch as a single 622 Mbps VC. This was accomplished by input and output port controller modules which interfaced the switching fabric to external OC-12 SONET interfaces. Due to the subsequent discovery by Bellcore of significant skew being introduced on the individual 155 Mbps channels by external SONET equipment, the STS-3c channels sent on the OC-12 SONET link were instead simply treated as individual VCs by the switch in later experiments. This is discussed further in the Transmission section.

The Sunshine switch also provided an important vehicle for exploring ATM Quality-of-Service (QoS) issues through implementation by Bellcore of a second-generation output port controller design later in the project (discussed further in the Network Management section). Prototype versions of the Sunshine switch were deployed by Bellcore at the Upenn, MIT, and Bellcore sites in Aurora.

From a switching perspective, lessons learned from the Sunshine development experience were that using fixed size cells significantly reduces switching fabric cost and the cost is independent of cell size, but if processors are used for output port queuing or other cell-handling functions the small ATM cell size does impose a significant cost penalty. While all-hardware port designs would eliminate this penalty, evolving network-layer QoS and congestion control algorithms may in fact benefit significantly from processor-based designs [2].

**Xunet Switch**

Wide area ATM switching in the Blanca testbed was accomplished using Xunet experimental switches developed by AT&T Bell Labs. A version of this switch known as Xunet-II had been developed prior to the start of the testbed project to provide wide area switching of 45 Mbps ATM links. For its use in Blanca, AT&T developed 622 Mbps interfaces and associated internal handling of the higher rate data streams, with this version of the switch called Xunet-III.

The Xunet switch design evolved from AT&T's earlier Datakit data switch design. It was based on a shared bus architecture and used a modular design to attach line interfacing and queuing cards to the bus structure. The switch used a modified version of the standard ATM cell format, a non-SONET proprietary trunk transmission protocol, and a direct proprietary optical link on a user-side line card of the switch to send and receive ATM cells with an ATM-HIPPI adapter.

The Xunet-III version of the switch was deployed by AT&T at Madison, Chicago, and Champaign for experimentation by Blanca researchers. A separate control computer connected to the switch architecture provided call setup, teardown and other control functions, and provided a software platform for network control algorithm investigations carried out by Blanca researchers and discussed in other sections.

**Fetex 150 Switch**

The Fetex 150 prototype switch used in Vistanet supported standard ATM cells and 622 Mbps OC-12c user ports as part of a modular, general purpose wide area carrier switch architecture. This switch was notable both for its direct support of synchronous 622 Mbps data streams and

for the fact that it was located in a telephone company central office as part of the Vistanet topology. This was the first instance of a central office broadband ATM switch, and provided an opportunity for the participating testbed carriers to gain hands-on experience in an environment involving real user data traffic.

The switch was connected over a 2.5 Gbps OC-48 SONET trunk to GTE's SONET crossconnect switch located across town, and within the central office user area to two Vistanet sites on the UNC campus. Thus Vistanet switching topology differed from the other testbeds on two significant counts: the metropolitan area switching was performed in a telephone company central office, and the switching between the two UNC campus sites was also done by the central office switch.

A prestandard SVC signaling protocol was used in the Fetex 150, allowing VCs to be dynamically established for experiments through the use of special communication channels. Traffic management was not implemented in the switch, however, and only minimal cell buffering was provided. Cell loss was avoided by relying on testbed experimenters to control their traffic through the use of acknowledgments at higher protocol layers.

**Local and Desk Area ATM Switching**

As part of its investigations into ATM networking, MIT developed what they termed a Desk Area Network (DAN) which interconnected workstations and peripheral I/O devices through a small local network. The DAN architecture was based on the use of small crossbar switches supporting 700 Mbps ports and a non-standard 56-byte ATM cell. The prototypes developed for the testbed activity consisted of 4 or 6-port switches and 500 Mbps links, where the latter used the HP Glink transmission protocol over optical fiber and other media. The switches were designed to simplify the hardware interfaces required to connect devices to the DAN switching fabric, with as much functionality as possible pushed out to host software.

The other local area ATM activity carried out in the testbeds made use of Fore Systems switches at UIUC., where they were used as part of their network control software investigations. Fore Systems in fact came into existence as a direct result of the testbed initiative-its founders were part of the Nectar testbed research group at CMU at the beginning of the project, and were stimulated to form a startup company addressing high speed networking needs as a result of their testbed involvement. The company has gone on to become a major ATM switch provider for both local and wide area environments.

### *4.2.2 PTM*

Wide area Packet Transfer Mode, or variable-length packet switching, was explored in both the Aurora and Casa testbeds. In addition, while not strictly a packet-switched technology, HIPPI was the basis for local area networking in four of the testbeds.

### Planet/Orbit Switching

A major part of IBM's research in the Aurora testbed was focused on their Planet and Orbit gigabit PTM technology (originally named Paris and Metaring respectively) which had been under development prior to the start of the project. Their architectural premise was that, because of the small transmission times involved with networking at gigabit rates, reasonably large variable length packets could be handled along with short packets while still providing real-time quality of service to the portion of traffic requiring it.

The Planet switch was intended for wide area switching, working in conjunction with 1 Gbps Orbit local area buffered rings at each site. The switch is based on a modular architecture in which link adapter cards communicate through a 6 Gbps backplane. Individual packet processing is done in hardware, with routing updates and other control functions handled in software by an RS/6000 workstation attached to the switch via an Orbit ring. In addition to Orbit rings, link adapter cards can be connected to wide area links through use of a SONET and other types of non-Orbit adapters. Also, while originally intended as a non-ATM switch, IBM added a capability to deal with fixed-size ATM cells as a special case of the switch's more general PTM capability.

Planet switch prototypes were deployed at IBM and Upenn for use in Aurora experiments. Through the use of real application traffic supplemented by artificial traffic generators, significant new insights were obtained by IBM researchers concerning traffic management under heavy switch loading, synchronization requirements, and other aspects of switch-related network performance.

### Local-Area HIPPI Switching

Network Systems Corporation (NSC) developed a commercial HIPPI switch product early in the testbed project, and these switches were used for local switching in the four testbeds which used HIPPI for local interconnection. As switches from other companies became available later in the project they were also used by some testbeds. These switches were based on non-buffered crossbar architectures and followed the ANSI standard for HIPPI switch control, typically providing either 8 or 16 ports operating at 800 Mbps.

Because the HIPPI protocol requires that a physical circuit be established prior to data transmission at each end of a HIPPI link, cascading two or more HIPPI switches in general introduces a significant potential for blocking relative to packet-switched operation. A method for avoiding this and also providing real-time services when using HIPPI switches was investigated by Blanca researchers at Berkeley, who explored a time-division multiplexing (TDM) solution to the problem. Their local portion of Blanca consisted of three cascaded HIPPI switches with two or three hosts connected to each switch.

In the Berkeley scheme one host is designated as the master and is responsible for defining a TDM frame, frame synchronization, and scheduling requests received from the other hosts in each frame. Synchronization of the hosts and switches is accomplished by a combination of the

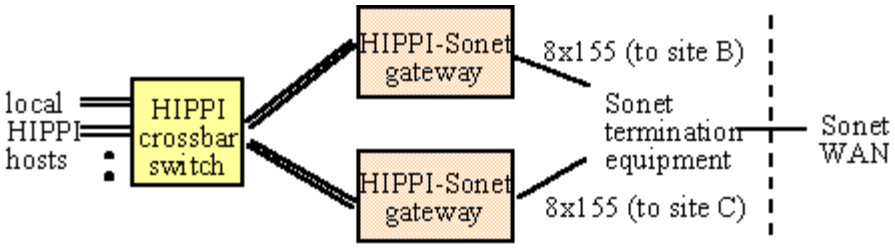
master framing definition and HIPPI switch camp-on feature. Slots are assigned to hosts in each frame according to the quality-of-service bandwidth and latency requirements contained in their requests. An end-to-end HIPPI circuit is established, traffic sent, and the circuit terminated by the assigned host in each slot.

The Berkeley TDM scheme satisfied its goal of multiplexing real-time traffic through circuit-based HIPPI switches, demonstrating stable operation and traffic delivery within real-time latency bounds. However, its developers concluded that synchronization requirements limit the scheme to small networks, it does not allow bandwidth-sharing by non-real-time traffic, and it introduces a significant bandwidth overhead penalty.

**Wide-Area HIPPI Switching**

In addition to local area switching within each site, the Casa testbed used their HIPPI switches, in conjunction with specially designed gateways, for wide area switching over the SONET links connecting the Casa sites. This was done by terminating local host HIPPI connections in a gateway device connecting the switch to an inter-site SONET link and relaying the packet through a HIPPI switch at other Casa sites on the path to the destination. Each intermediate site had a separate gateway and SONET link connecting it to each of its two neighboring sites (Figure 4-9).

**Figure 4-9. Wide-Area HIPPI Switching**




Buffering in the gateways and hosts allowed Casa to be operated as a store-and-forward packet switched network over its wide area end-to-end paths, in spite of the absence of buffering in the HIPPI switches. Routing was accomplished by defining a logical network-wide address for each host in the network and configuring a logical-physical address mapping table in each switch.

A host sending to a destination at another Casa site established a HIPPI connection between itself and the appropriate local gateway at its site, through one or more local HIPPI switches, for the duration of a single HIPPI packet transfer, with each packet constrained to the maximum IP packet size of 65 Kbytes. The local gateway is the endpoint of the HIPPI connection, exchanging HIPPI signaling with the local switch to control the flow of HIPPI bursts from the local host. A special gateway-gateway protocol is used across the wide area SONET link to send the HIPPI data and logical addressing information to the gateway at the next site and prevent buffer overflow in the receiving gateway.

Assume for this example that the destination host is at a third site. If the HIPPI switch port to the outgoing gateway at the intermediate site is not in use, a gateway-switch-gateway HIPPI connection is established at the intermediate site, and the first intermediate gateway proceeds to send its incoming data to the other gateway at that site. In this case the originating site and first intermediate gateways only buffer one complete HIPPI burst of the packet before proceeding to forward it, and do not need to buffer the entire packet. If the switch port to the outgoing gateway is in use, up to one full packet is buffered if necessary by the first intermediate gateway, which then signals the originating gateway to wait before sending another packet. Thus a form of cut-through is used, where the cut-through is applied whenever a full packet does not need to be buffered.

For the case where a higher layer retransmission protocol such as TCP is being used on the end-to-end path, a gateway will drop a packet if it cannot succeed in establishing a connection through a switch by a predefined timeout period. For situations where `raw' HIPPI packets are sent over the end-to-end path, such as was done for some Casa experiments in which TCP was not supported by one or more of the hosts, the timeout is disabled and the packet held until the connection can be made.

Gateway buffering thus allowed inter-site HIPPI switches to be decoupled in Casa, avoiding the tandem setup problem inherent in the direct connection of unbuffered HIPPI crossbar switches. HIPPI multiplexing latencies in this case were dictated by the choice of the 65 Kbyte maximum packet size allowed during a single local connection, in contrast with the real-time multiplexing achieved for local HIPPI switching by the Berkeley TDM scheme.

**Preemptive Switching**

As part of their investigations into more efficient forms of data multiplexing for high speed networking, MIT explored a scheme based on the use of arbitrarily long data units defined for application layer efficiency, where the latter was referred to as Application Layer Framing (ALF). To achieve low latencies required for some types of traffic in this context, they studied the use of preemption in network switches. Simulation results for a variety of traffic loads showed that the required switch processing rate could be reduced by about a factor of 10 relative to cell switching.

## *4.3 Interworking*

### *Summary*

- Three different designs were implemented to interwork HIPPI with wide area ATM networks over both SONET and all-optical transmission infrastructure; explorations included the use of 4x155 Mbps striping and non-striped 622 Mbps access, local HIPPI termination and wide area HIPPI bridging; resulting transfer rates ranged from 370 to 450 Mbps

- A HIPPI-SONET gateway was implemented which allowed transfer of full 800 Mbps HIPPI rates across striped 155 Mbps wide area SONET links; capabilities included variable bandwidth allocation of up to 1.2 Gbps and optional use of forward error correction, with a transfer rate of 790 Mbps obtained for HIPPI traffic
- Seamless ATM DAN-LAN-WAN interworking was explored through implementation of interface devices which provided physical layer interfacing between 500 Mbps DAN Glink transmission, LAN ATM switch ports, and a wide area striped 155 Mbps ATM/SONET network; interface functions included translation of non-standard DAN ATM formats and HEC generation
- An investigation of IP processor-based interworking at gigabit speeds concluded that, for a 622 Mbps link rate, a 100 MHz RISC processor could perform basic IP packet processing with sizes as small as 83 Bytes/packet, but that I/O port processing will in general require a dedicated processor for each flow direction, with an evolution to distributed architectures based on hardware switching fabrics

A significant amount of work was carried out in the testbeds on devices to allow the interworking of different high speed technologies. We use the term *interworking* here to distinguish this work from more general systems-level internetworking research. The testbed work in this area can be grouped into the following categories: HIPPI-ATM, HIPPI-SONET, and DAN/LAN/WAN ATM. A preliminary investigation into issues associated with using an IP protocol for generic packet forwarding at gigabit speeds was also conducted.

(The area of general internetworking at gigabit speeds was specifically not included as a research topic when the testbeds were formed. It was felt that work was first required on the underlying network technologies, with internetworking research more logically undertaken following completion of the initial testbed initiative.)

### *4.3.1 HIPPI-ATM*

Three distinct efforts were undertaken in the testbeds to interwork local HIPPI networks with wide area ATM. These efforts were undertaken in part out of necessity, since a device had to be developed in each case to allow interconnection of facilities within the testbed, and in part for the opportunities they presented to investigate high speed interworking issues. A major distinction of these interworking approaches is whether they terminate HIPPI locally, as would be done for example by an IP-based router, or extend HIPPI connections and associated control signaling across the ATM network. The Nectar and Blanca solutions used local termination, while the Vistanet solution used the extension method.

The three HIPPI efforts are represented by the architecture of Figure 4-1B.

#### Nectar HAS

The HAS design goals included allowing up to two 800 Mbps HIPPI connections at each Nectar site to use the full 2.5 Gbps SONET link bandwidth available at the wide area HAS interface, an architecture design that would allow HIPPI hosts to communicate with non-HIPPI hosts at re-

mote sites, and the ability to add support for wide area ATM network management standards as they evolved.

To maximize flexibility and allow interworking of different local area technologies, HIPPI connections were terminated locally by the HAS and HIPPI header information stripped from packets before sending them into the ATM network. PVCs were used in the testbed prototypes, with a PVC pre-established for each pair of hosts needing to communicate across the ATM network. A Management and Signaling Processor (MSP) module in the HAS allowed mappings to be established in HAS tables between VCIs and local HIPPI identifiers, and provided a means for later incorporation of SVC signaling standards.

The HAS supported AAL types 1 and 3/4 (AAL 5 had not been defined when the HAS design was begun). The data in each packet received from the HIPPI module was segmented and formatted using one of the AAL types and sent as an ATM cell stream over the ATM/SONET link. In the absence of well-defined ATM network flow control standards, a simple open-loop pacing mechanism was used at each transmitting node to prevent steady-state overflow of destination buffers.

A design choice which had a major impact on HAS complexity was the way in which striping over the available SONET channels was handled. As described in the Transmission and Switching section earlier in this report, a choice was made to stripe packets over each STS-3c SONET channel within the HAS, with 8 such channels available for carrying the data sent in an 800 Mbps HIPPI channel. A distinct ATM VC was assigned to each STS-3c channel, with all cells of a packet sent over that channel. The received cells on each VC are stripped of their ATM/AAL overhead and reassembled, and the packet passed by the ATM module to the HIPPI or other local network module for reordering if necessary, eliminating special synchronization and other striping-related processing in the ATM and SONET layers and simplifying the overall HAS design. A maximum of 1024 VCIs were available to each ATM/AAL module.

Prototype HAS devices were installed at CMU and at the Pittsburgh Supercomputer Center for testbed experimentation. Using a HIPPI tester which could generate 7 KByte maximum packet sizes and with striping used over four 155 Mbps, a maximum throughput of 420 Mbps was measured. Based on theoretical calculations, a packet size of 11 Kbytes would give very close to the maximum predicted throughput of approximately 430 Mbps (the maximum potential bandwidth available to data in a 622 Mbps path after AAL, ATM and SONET overheads are allowed for is 496 Mbps).

**Blanca HXA**

The HIPPI-ATM Adapter (HXA) was developed for use in the Blanca testbed by UIUC in collaboration with AT&T Bell Labs. Its design was driven primarily by immediate Blanca testbed interconnection needs, and thus was more limited in scope than the Nectar effort with respect to ATM network management issues. It did however implement IP-layer processing of packet

headers, providing an instance of hands-on experience in this area. Like the HAS, it also provided a direct HIPPI-ATM transfer mode for routerless operation.

The HXA terminated HIPPI connections internally, and was connected to an Xunet switch port on the ATM side via an optical fiber link. A proprietary transmission protocol was used on this link for local transport of ATM cells between an AT&T line card in the HXA and a line card on the Xunet switch. Two simplex physical HIPPI ports were provided by the HXA for communication with one HIPPI host at a time, either through a direct connection or one or more HIPPI switches. Latencies associated with multiplexing among different HIPPI hosts connected through a HIPPI switch to the HXA were determined by the maximum connection time discipline imposed upon hosts, for example breaking connections after each packet.

Routing and header processing functions were done by software running on a RISC microprocessor in the HXA. A table lookup was done to map HIPPI or IP addresses into ATM PVCs and conversely, with HIPPI headers stripped off when an IP header is present in the packet. Each HIPPI or IP packet is encapsulated with an AT&T proprietary AAL5-like protocol (AALX), segmented into cells using AT&T's 54-byte Xunet ATM format, and sent via the optical fiber line card to the Xunet switch.

Up to three 1-KB bursts of a HIPPI packet can be buffered by the HXA in the HIPPI-to-ATM direction. After initial header processing, received bursts are processed as they arrive and the resulting ATM cells sent to the Xunet switch. In the ATM-to-HIPPI direction, two AALX frames can be stored to provide a double-buffered output on the HIPPI side of the HXA, which establishes connections and transfers HIPPI bursts to a host or switch following receipt of a complete AALX frame. Multiplexing of AALX frames being received on multiple VCs is done in the Xunet switch, which provides a maximum-size AALX buffer for each VC supported for a switch output port.

Although burst flow control is used by the HXA between itself and local HIPPI hosts, explicit flow control was not implemented between the HXA and Xunet switch. Rather, TCP was assumed used by endpoints to adapt steady-state flows to available bandwidth and recover from occasional dropped packets, with large buffers provided in the Xunet switches to minimize packet losses for bursty traffic.

Prototype HXAs were deployed at Champaign and Madison Xunet switch sites. A maximum HXA transfer rate of approximately 370 Mbps was achieved using a HIPPI tester at the endpoints, which was sufficient to fill the maximum available user bandwidth on the 622 Mbps Xunet switch-trunk path.

**Vistanet NTA**

The designers of the Vistanet NTA (Network terminal Adapter) chose not to terminate HIPPI locally at each site, but rather to extend HIPPI connections directly across the intervening ATM/SONET network. This was motivated by a desire to minimize the need to define new pro-

tocols and meet the relatively short project schedule. The NTA thus functioned as a very sophisticated three-way HIPPI extender which included AAL4/ATM/SONET wide area network functionality, network control and measurement capabilities, and operation with a Fetex 150 central office ATM switch. A single unstriped 622 Mbps OC-12c SONET link was available in each direction between an NTA and the ATM switch.

A key driver of NTA design was the need to provide flow control of HIPPI packets across the ATM network without relying on end-to-end protocols, since TCP or other transport layer protocols were not expected to be available for one of the key Vistanet application hosts. Since ATM flow control was not well-defined, this was solved by extending HIPPI burst-level flow control across the NTA-NTA paths along with the use of NTA buffering to eliminate the effects of path delays.

Each NTA supported one local HIPPI host and a simultaneous bidirectional ATM connection with each of the other two Vistanet sites. The NTA contained two 32KB receive buffers, each dedicated to one of the other sites, and a single 32KB transmit buffer, allowing a sufficient number of 1KB HIPPI bursts to be buffered to avoid transmission gaps due to roundtrip propagation times and other path latencies. HIPPI connection and flow control signals were carried in the AAL4 headers of the ATM cells, augmented by the use of special ATM cells when necessary to carry out control signaling.

To avoid setup delays, pairs of VCs were pre-established through the switch between each NTA and its two neighbors (a VC pair was required on each simplex data path to allow the return of control information). A host HIPPI connection request would be immediately accepted by the local NTA and data buffered by the NTAs until the destination host HIPPI connection could be established, eliminating the roundtrip setup connection delays that would otherwise occur using HIPPI extenders. HIPPI source routing was used to identify VC mappings and HIPPI destinations via inspection of packet header information.

Implementation of the NTA was partitioned into two major components, a “core NTA” portion which handled the data flows and consisted primarily of hardware, and a control and measurement subsystem (CMS) portion which consisted of a Sun workstation and software. The CMS was connected to the Internet, allowing Vistanet experimenters to configure VCs for their experiments through use of the Fetex 150 switch proprietary SVC protocol supported by the core NTA. More generally, the CMS was used both by system developers to diagnose problems and by researchers to collect traffic measurements for analysis (these topics are discussed further in later sections).

The NTAs were deployed at each of the Vistanet sites and used for Vistanet application experiments and traffic studies. NTA throughputs of approximately 450 Mbps were achieved, which was close to the maximum throughput available on the 622 Mbps SONET link after allowing for SONET, ATM, and AAL4 overhead.

### *4.3.2 HIPPI-SONET*

The HIPPI-SONET gateway developed by LANL for the Casa testbed addressed the issues associated with interconnecting islands of HIPPI-based local area networks with wide area SONET links, without use of a wide area ATM switching (Figure 4-1D). Major aspects of this gateway's operation have already been discussed in earlier sections on switching and striping in the testbeds. The striping of HIPPI data across multiple 155 Mbps SONET channels played a central role in the gateway design, allowing incremental amounts of bandwidth to be allocated to a HIPPI connection rather than just the full 1.2 Gbps available on the SONET trunks. While the gateways did not perform switching directly, they were designed to maximize the use of local HIPPI switches for wide area switching among the testbed sites, as discussed in the Switching section above.

A key difference from the Vistanet NTA approach was local termination of HIPPI connections by the Casa gateway, with a special gateway-gateway flow control mechanism used to prevent overrunning gateway buffers, and reliance on TCP for end-to-end recovery when a packet was discarded by a gateway due to destination host blocking. Another important difference was the use of forward error correction to correct single-bit errors, reducing the frequency with which hosts needed to retransmit packets to achieve reliable end-to-end operation.

The gateways were installed at Casa sites and used both to support Casa applications and for network studies. Transfer rates of 790 Mbps were measured using HIPPI testers at endpoints, and throughputs of 550 Mbps and 580 Mbps were measured when using TCP/IP and UDP/IP respectively on Cray computers at network endpoints.

### *4.3.3 Seamless ATM*

A topic of major interest to ATM developers is *seamless ATM*, the interworking of distinct ATM networks without using an explicit internetworking layer such as IP, and an initial exploration of this topic was done by MIT in the Aurora testbed (Figure 4-1A). They developed devices to interconnect their Desk Area Vunet ATM network with both the Aurora wide area ATM/SONET network and a local area ATM network.

The differences which needed to be resolved for interworking included the underlying physical transmission technologies, cell formats, and VC signaling conventions. VuNet used HP Glink single-mode fiber transmission technology operating at 500 Mbps, in contrast to the use of striped 155 Mbps SONET channels in the wide area ATM network. The VuNet ATM cell format was chosen to be 56 Bytes to simplify cell processing, and a choice was also made not to use the HEC header check defined as part of the standard ATM format, since it was felt that the relatively low error rates experienced in a DAN environment did not warrant the additional complexity associated with HEC processing. VC control signaling was also handled differently than in the Aurora Sunshine switch.

Two different devices were developed by the VuNet group for ATM-ATM interworking. The first, called AVlink, interfaced VuNet directly to a single 155 Mbps channel of the wide area

ATM/SONET network. The AVlink device converted the 56-byte VuNet cells to standard 53-byte cells and conversely, computed the HEC field in the cell header, and mapped cells to and from the Glink and SONET physical transmission formats. This was accomplished using a relatively simple design, with a resulting average latency of 4.6 microseconds and transfer rate of 150 Mbps.

A second device called Zebra was also developed to interface VuNet to a Digital Equipment AN2 ATM switch, which effectively represented the presence of a LAN between VuNet and the Aurora wide area ATM network. The latter was connected to a 622 Mbps SONET port on the AN2 switch, which was in turn connected to VuNet through the Zebra board installed on the AN2. Zebra, like AVlink, provided the required cell format mapping and Glink interfacing. More significantly, the use of Zebra with the AN2 provided an opportunity to explore solutions to the general striping problem, since VuNet supported 500 Mbps Glink channels while the wide area path consisted of multiple 155 Mbps SONET channels. This was carried out using source and destination VuNet hosts and an end-to-end path which included a loopback point within the wide area network [2].

### *4.3.4 IP-based Interconnection*

As a part of their work on high-speed networking, MIT also studied the processing costs associated with the use of the IP protocol for interconnecting different networks. They implemented an IP packet forwarder running with the x-kernel operating system on a MIPS 3000 33 MHz processor and measured resulting packet processing rates with and without network I/O drivers present.

After modifications to the x-kernel to reduce its overhead, a rate of 55 Kpackets/s was obtained with no I/O drivers installed. With the addition of an ethernet driver, throughput dropped to approximately 3 Kpackets/s, indicating that the dominant processing cost was associated with I/O rather than IP processing. Using IP instruction counts, they projected that the `pure' IP code processing on a 100 MHz RISC processor should approach 1Mpackets/s, allowing a flow of packet sizes as small as 83 Bytes to fully utilize a 622 Mbps link.

Based on these results, they designed an architecture for a high performance IP flow forwarder which makes use of multiple RISC processors. One processor is dedicated to IP level processing for each flow direction, one processor performs background management tasks, and one processor is dedicated to each flow direction of each I/O port. Thus, a forwarder providing full duplex connections between two networks has a total of 7 processors. Both a central memory and distributed memory architectures were considered, with the choice depending in part on the number of networks connected to the forwarder.

Conclusions drawn by MIT from this work were that, to scale routers up for very high speed networks, new designs should focus on providing increased hardware assistance in conjunction with RISC processing at individual ports and will need to evolve from shared memory to parallel switching fabrics for port interconnection. They also observed that the latter interconnect archi-

tecture will make it more difficult to implement global resource management, since information and control is now distributed among the port elements.

## *4.4 Host I/O*

### *Summary*

- Several different testbed investigations demonstrated the feasibility of direct cell-based ATM host connections for workstation-class computers; this work established the basis for subsequent development of high speed ATM host interface chipsets by industry and provided an understanding of changes required to workstation I/O architectures for gigabit networking
- Variable-length PTM host interfacing was investigated for several different types of computers, including workstations and supercomputers; in addition to vendor-developed HIPPI interfaces, specially developed HIPPI and general PTM interfaces were used to explore the distribution of high speed functionality between internal host architectures and I/O interface devices
- TCP/IP investigations concluded that hardware checksumming and data-copying minimization were required by most testbed host architectures to realize transport rates of a few hundred Mbps or higher; full outboard protocol processing was explored for specialized host hardware architectures or as a workaround for existing software bottlenecks
- A 500 Mbps TCP/IP rate was achieved over a 1000-mile HIPPI/SONET link using Cray supercomputers, and a 516 Mbps rate measured for UDP/IP workstation-based transport over ATM/SONET; based on other workstation measurements it was concluded that, with a 4x processing power increase relative to the circa 1993 DEC Alpha processor used, a 622 Mbps TCP/IP rate could be achieved using internal host protocol processing and a hardware checksum while leaving 75% of the host processor available for application processing
- Measurements comparing the XTP transport protocol with TCP/IP were made using optimized software implementations on a vector Cray computer; the results showed TCP/IP provided greater throughput when no errors were present, but that XTP performed better at high error rates due to its use of a selective acknowledgment mechanism
- Presentation layer data conversions required by applications distributed over different supercomputers were found to be a major processing bottleneck; by exploiting vector processing capabilities, revisions to existing floating point conversion software resulted in a fifty-fold increase in peak transfer rates
- Experiments with commercial large-scale parallel processing architectures showed processor interconnection performance to be a major impediment to gigabit I/O at

the application level; an investigation of optimal data distribution strategies led to a selection of application control for data distribution within the processor array in conjunction with use of a reshuffling algorithm to remap the distribution for efficient I/O

- Work on distributed shared memory (DSM) for wide area gigabit networks resulted in several latency-hiding strategies for dealing with large propagation delays, with relaxed cache synchronization resulting in significant performance improvements

Host I/O was one of the most challenging areas of the testbed effort. In general, it proved to be the Achilles' heel of gigabit networking -- whereas LAN and wide area networking technologies could be and were operated in the gigabit regime, many obstacles impeded achieving gigabit flows into and out of the host computers used in the testbeds.

A wide range of computers were used, ranging from vector and massively parallel supercomputers to single-processor workstations. Moreover, some testbeds experienced a dramatic change in the characteristics of the computers which were available for experiments. At the beginning of the project in 1990, state-of-the-art supercomputing was represented by Cray Research supercomputers with a peak performance of approximately 2 gigaflops. By 1994 supercomputer performance had increased by an order of magnitude or more, with the Cray vector architecture augmented by highly parallel machines such as the Paragon and CM-5.

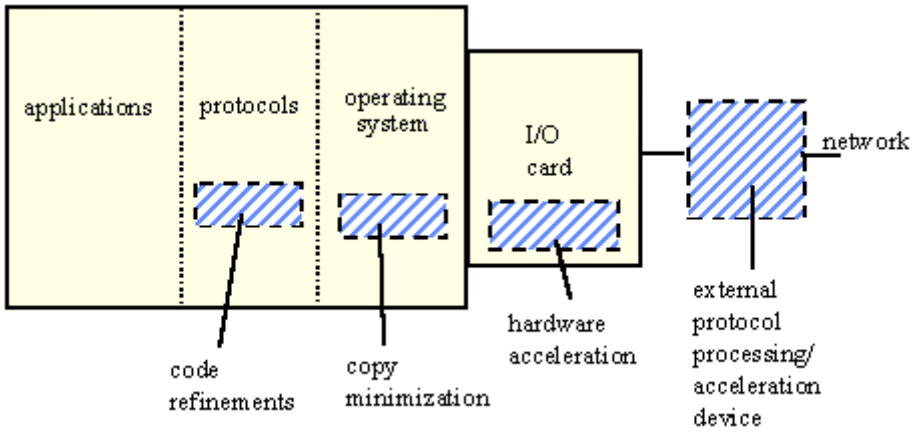
While applications work in the testbeds emphasized the use of supercomputers, workstation-class computers also played an important role. Both Digital Equipment and IBM workstations were used as platforms for extensive high speed I/O hardware and software exploration. In the workstation case, advances in processor technology also allowed replacements with higher performance machines. However, project schedules precluded the redesign of I/O boards which were specially developed for the original workstation bus architectures, and so later bus technologies were for the most part not incorporated into this area of testbed work.

Against this backdrop, researchers in all of the testbeds investigated various aspects of the host I/O problem, which for this section we take to be the movement of data between an application running on a host and an external network, exclusive of the application software itself. This work spanned a large number of individual efforts and specific topics, with the latter including:

- direct ATM connections
- PTM interfacing
- transport
- data conversion
- parallel architectures · distributed shared memory

Figure 4-10 illustrates the focal points of this work within a generic host I/O architecture, with each effort typically including only a subset of the shaded components.

**Figure 4-10. Generic Host I/O Architecture**

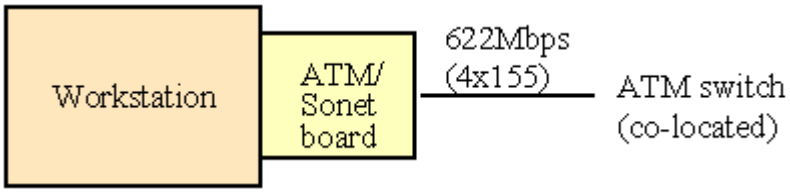



### *4.4.1 Direct ATM Connections*

Researchers in the Aurora testbed were focused on the use of workstations for gigabit user applications, dictated primarily by their view that workstation-class computers would replace supercomputers as high end platforms in the years ahead. A second major thrust in Aurora was the use of ATM as a local area technology to directly connect workstations to ATM switches. The Bellcore Sunshine ATM prototype switch was developed as part of Aurora's activities and used for the local workstation connections, as well as for wide area switching, through deployment of the switch at multiple Aurora sites. An additional opportunity for direct connections was provided by the VuNet desk area networking technology developed in Aurora by MIT.

A major challenge of the direct ATM approach was dealing with the small ATM cell size at the 622 Mbps link speed available in Aurora, including both cell transmission/reception and especially the segmentation and reassembly (SAR) of cell streams into higher layer protocol units. Several distinct efforts were undertaken to explore this domain: Penn, Bellcore and MIT developed board-level solutions which interfaced to the workstation I/O bus, and MIT also explored a novel coprocessor approach. The Bellcore and initial Penn efforts used SONET transmission (Figure 4-11), while later Penn versions and the MIT VuNet effort used Glink technology.

**Figure 4-11. Directly Connected ATM/SONET**




A key question for the board-level approaches was how much functionality should be handled by specialized hardware and software on the board itself, and how much should be done by software using the workstation's main processor (while leaving enough processor bandwidth to also run applications).

**Processing**

The three board approaches represented distinctly different choices in how SAR processing was done. The VudBoard approach developed by MIT for VuNet relegated this functionality to the main workstation processor, with the I/O board used only to transmit and receive cells on its physical layer interface. The other two board approaches both carried out the SAR and ATM layer processing functions on the I/O board. The Penn approach used an all-hardware implementation, while the Osiris board developed by Bellcore used two Intel 80960 processors and onboard software.

The AAL protocol used in VuNet was a modified version of AAL5 in which a simpler checksum was used to ease software processing requirements. The Penn and Osiris boards supported AAL 3/4, which unlike AAL5 includes an AAL header within each ATM cell of the adaptation layer frame.

The MIT coprocessor approach interfaced the network physical layer directly to the registers of a specially designed coprocessor, which was designed to work analogously to that of a floating point coprocessor.. The intent was to use the workstation processor for ATM and SAR processing while avoiding the bottlenecks introduced by the traditional I/O bus architecture [2].

**Bus Transfers and Data Movement**

While processing requirements could be dealt with in a reasonably straightforward manner through the use of special hardware or I/O board hardware/software provisioning, the workstation bus architectures presented a more formidable obstacle. Two general methods were available for moving data between the I/O interface and main host memory, programmed I/O (PIO) and direct memory transfer (DMA). For the workstations used in Aurora, the DMA choice resulted in higher data transfer rates.

The short ATM cell size and high rates combined to reveal significant latency-oriented shortcomings in the workstation I/O architectures. The VuNet and Osiris approaches were originally implemented using a DEC Turbochannel bus and one ATM cell per transfer, and found their

maximum achievable speed constrained by the latencies associated with bus hardware access and transfer control mechanisms.

The transfer of individual cells was necessitated in the VuNet case by its use of the workstation processor and memory for ATM and SAR processing in conjunction with a minimal I/O board implementation. In the Osiris case, while all cell-oriented processing was performed on the I/O board, a choice was made to transfer cell data directly into higher layer protocol buffers in host memory, eliminating additional latencies which would otherwise be incurred if the higher layer packets were first assembled in buffers on the I/O board. However, the per-cell transfer overhead of the bus significantly constrained the achievable sustained transfer rate.

The Penn approach used an IBM RS6000 MicroChannel bus and a linked-list data management architecture on the I/O board which allowed larger multi-cell segments of data to be transferred across the bus. Initial experiments revealed a bottleneck in the operation of the workstation I/O controller, which was replaced with an improved version by IBM later in the project. Of more lasting concern was the interrupt overhead associated with the RS6000 architecture, which led to the choice of a periodic rather than event-driven interrupt design in order to ensure adequate processing bandwidth for applications.

A major bottleneck to achieving gigabit transfer rates common to all of these efforts was that of memory buffer copying by the operating system, and part of the work above included exploring techniques which moved data from the I/O board directly into application memory space. This area is discussed further in later parts of this section.

**ATM Performance**

With a change to two-cell DMA transfers, use of a later-generation DEC-Alpha workstation with the Turbochannel bus, and operating system changes discussed below, the Bellcore Osiris board achieved a transfer rate of 516 Mbps using the UDP transport protocol. This was the full throughput rate available after AAL/ATM/SONET overheads are subtracted from the 622 Mbps link rate used, but was also near the maximum transfer rate which could be expected from the hardware architecture due to its bus and memory bandwidth constraints.

The initial Penn interface achieved a maximum transfer rate of about 90 Mbps using the UDP protocol and a single 155 Mbps SONET link on the RS6000, and it was estimated that this performance would scale proportionally for a 622 Mbps link. A subsequent implementation of the Penn interface on an HP PA-RISC workstation achieved a transfer rate of 215 Mbps using the TCP protocol and a Glink physical layer [2].

The MIT VuNet DMA interface achieved a maximum application transfer rate of approximately 100 Mbps using a UDP-like transport protocol and a DEC-Alpha Turbochannel workstation. While this was well below the several hundred Mbps rate allowed by the workstation's bus and memory bandwidths, the premise of the VuNet effort was to "ride the workstation processor technology curve", that is to use the simplest specialized hardware interface possible and learn

how to architect the software so that the result will naturally scale to higher rates when used with faster processors.

The MIT coprocessor implementation was not completed due to schedule problems, precluding actual measurements. Extensive simulations were carried out to predict its performance relative to other approaches, however, and are discussed in detail in [2].

In summary, the testbed work in this area demonstrated both that direct cell-based interfacing of workstations to gigabit ATM networks was feasible and that some aspects of workstation architectures needed to be redesigned to make it reasonable. In particular, the design space explorations in Aurora laid the groundwork for the subsequent development of high speed ATM SAR chipsets by industry, and provided an understanding of the improvements needed to workstation I/O architectures for gigabit ATM networking.

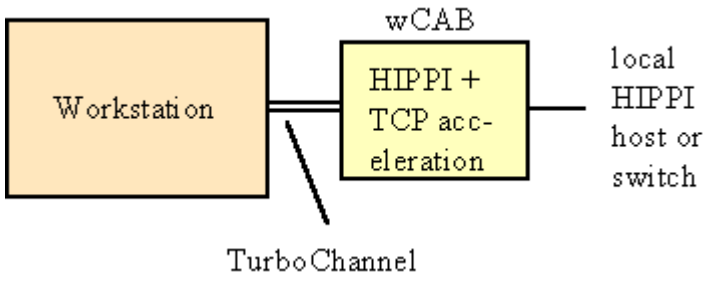
### *4.4.2 PTM Interfacing*

HIPPI variable-length packet interfaces were used to connect hosts in the Blanca, Casa, Nectar and Vistanet testbeds, and also for some hosts in Aurora. In addition, a variable-length gigabit token ring, Orbit, was developed by IBM and used in Aurora, and a general-purpose PTM host interface was also developed as part of the Aurora work.

Most of the HIPPI interfaces were provided by computer vendors as part of their supplied equipment during the course of the project. Cray Research provided HIPPI interfaces for the CRAY YMP supercomputer early in the project, and by the end of 1993 commercial HIPPI interfaces were available for most supercomputers and several high-end workstations. Commercially developed HIPPI interfaces were used in the testbeds with Sun and SGI workstations and with the CRAY YMP, C90 and T3D, the TMC CM2 and CM5, the Intel Delta and Paragon, and the MasPar and SGI Challenge supercomputers, and with frame buffer and RAID storage peripherals.

For the DECStation workstations used in the Nectar testbed, HIPPI interfaces were developed for the project as externally connected equipment (Figure 4-12).

**Figure 4-12. Nectar Workstation I/O**

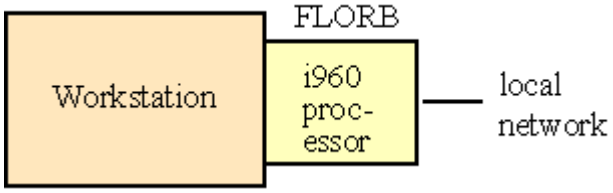



This was done as part of a more general effort under another contract to develop a device, called the Communication Accelerator Block (CAB), which could provide external I/O support to a wide range of hosts. Specific CAB implementations were developed for both the DECStation and the iWarp (discussed below). The functions of the CAB consist of buffering packets for network transmission and reception, providing TCP protocol checksumming support, and interfacing to the host and the network. The workstation unit is called the wCAB, and interfaces a HIPPI connection to the Turbochannel I/O bus of the DECStation using off-the-shelf hardware components.

Performance of the HIPPI interfaces was strongly dependent on packet length due to the various latency and overhead factors of I/O busses and software. Most of the interfaces operated at or near the full HIPPI rate of 800 Mbps when measured at the internal HIPPI I/O driver interface using a packet size of 65 KBytes. An exception was the wCAB, for which its DMA microcode implementation constrained Turbochannel transfer rates to about 200 Mbps. However, it was believed that the wCAB operation would have been able to achieve full Turbochannel rates had more time been available.

MIT developed a general PTM interface board design, called the FLORB, and implemented it for a DEC workstation using a Turbochannel I/O bus (Figure 4-13).

**Figure 4-13. FLORB PTM Interface**




Their design consisted of a shared fast SRAM, a single Intel 960 RISC processor, a FIFO at each of the four host and network input and output ports, and a DMA controller. A novel aspect of their design concerns the way in which the processor controls the data bus, allowing data to be

flexibly moved between multiple points on the bus in a single instruction cycle, for example from the host input FIFO directly to both the SRAM and the network output FIFO. Throughput rates of approximately 400 Mbps were measured for IP traffic, with the limiting factors found to be Turbochannel I/O bus arbitration and DEC memory contention effects.

### *4.4.3 Transport*

Work above the network layer was largely focused on the TCP and UDP transport protocols, along with a comparison of TCP with XTP. Key questions being asked prior to the testbed project were whether TCP in particular could run efficiently at gigabit rates on a host's native processor, or needed instead to be executed on an outboard implementation using varying degrees of specialized hardware.

The answer is obviously dependent on the power of the processor(s) used to execute the protocol, and indeed TCP/IP rates as high as 900 Mbps have been achieved on a dedicated Cray supercomputer in an I/ O driver loopback mode. More generally, however, two factors were found to be of key importance for most computers: checksum computation and data movement.

#### Checksum Computation

Since this involves computation on every word of a packet, it is in general a significant overhead factor unless it can be combined with other per-word protocol operations. While this can be done if programmed I/O is used to move the data from the network interface to host memory, as discussed above the implementations done in the testbeds concluded that DMA was more efficient for the workstations used.

Thus, some of the host interfaces developed for the testbeds used special hardware to compute the checksum, passing the result to the host for received packets and inserting it into the header for outbound packets. The latter requires that the interface buffer a complete packet before transmission, with the checksum typically computed as the packet is moved from the host into the interface memory.

The fact that TCP carries its checksum in its header was a source of much debate among high speed designers, with some arguing for a standards change to allow the checksum to optionally be carried at the end of the TCP packet to reduce the interface buffering requirement. However, those arguing that the memory required was not a significant incremental cost compared to problems associated with a standards change won out, and the checksum continues to be carried in the header.

#### Data Movement

The second dominant overhead factor was found to be data movement between the interface and the application, that is to say, the operating system. All of the workstation operating systems in use at the start of the testbed effort typically performed multiple copy operations on packets, copying them between the network interface and operating system buffer space and between the latter and application memory space, with an additional packet read/store for checksum computa-

tion also often involved. Because workstation DRAM speed had not advanced significantly relative to processor advances during the span of the project, memory bandwidth was the major hardware impediment to gigabit I/O and packet copying was thus generally very costly compared to the time required to execute per-packet TCP protocol instructions.

Three of the testbed efforts addressed this problem in detail for workstations: the UPenn and Arizona/ Bellcore ATM efforts in Aurora and the CMU workstation interfacing effort in Nectar. The result in all three cases was to reduce the data movement to a single transfer between the network interface and the application's memory space, achieved through techniques such as host memory page remapping or the use of I/O board memory as the intermediate `system memory'.

A fallout of this copy elimination is the resulting need for VCI and other demultiplexing while packets are in I/O board memory, in order to be able to map particular data streams into their associated application memory prior to DMA transfer [2].

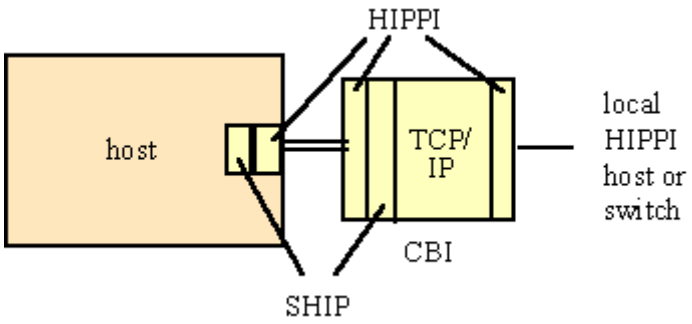
Testbed investigations using supercomputers also found significant problems associated with protocol-related memory management. Experimentation with the Data Transfer Mechanism (DTM) software developed by NCSA for distributed applications communication revealed a number of memory-related factors affecting throughput. In particular, they found that the use of page alignment and restrictions on buffer sizes could provide as much as 65% performance improvement. The primary reason for the improvement was the resulting use of DMA transfers rather than copy operations, with multiple writes of smaller buffers preferable to a single large buffer write above a certain size [3]. Similar results were obtained in the development of Express in the Casa testbed, where miss-matches of packet sizes between different protocol layers within the host significantly degraded performance [4].

**Outboard TCP**

Two full outboard TCP/IP implementations were developed in the testbeds, one by LANL for use in Casa and one by UNC for use in Vistanet.

The LANL case was motivated by the need to support MPP supercomputers used in Casa which did not contain an internal high-performance TCP implementation, for example the TMC CM-2 and the Intel Paragon. A device called the Crossbar interface (CBI), originally designed to provide HIPPI networking support for hosts within Casa, was configured with an Intel 486 computer board and a Unix operating system to perform TCP/IP protocol processing. The CBI contained two HIPPI interfaces, one for connection to a host HIPPI interface and one for connection to a HIPPI switch or other HIPPI equipment (Figure 4-14).

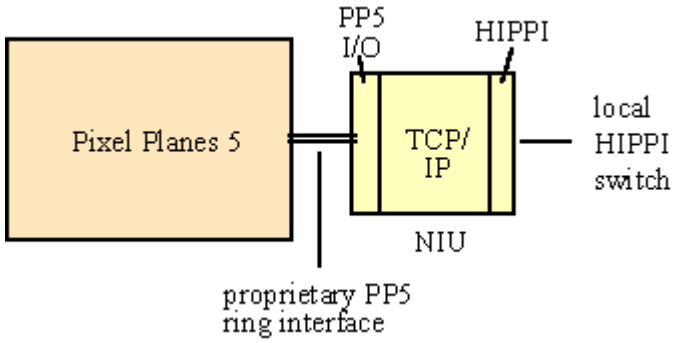
**Figure 4-14. Casa CBI**




Special hardware was used in the CBI for computing TCP checksums, and packet data buffering was handled in a flow-through manner without processor-based copying. The overall protocol processing model was thus similar to that of the workstation cases discussed above, except that a PC-class processor was used and the Unix operating system modified to allow processor interaction with the flow-through data hardware. A protocol called SHIP, for Simple Host Intersocket Protocol, was developed to present a standard TCP socket interface to applications on the host while providing relatively simple data transfers between the host and CBI. SHIP software was implemented as a library package on the Paragon and other computers.

A second instance of a full TCP outboard implementation was the NIU (Network Interface Unit) developed by UNC in Vistanet. The NIU was developed to provide external HIPPI and transport protocol support for the Pixel Planes 5 (PP5) multicomputer, a very fast graphics rendering computer developed by UNC and used for Vistanet experiments. The NIU moved data directly to and from PP5 processors using the PP5's data ring architecture, and provided a HIPPI interface for connection to a HIPPI switch (Figure 4-15).

**Figure 4-15. Vistanet NIU**




TCP/IP and UDP/IP protocol processing was supported using a 25 MHz SPARC processor, a custom multi-tasking kernel, and a custom protocol software implementation. Like the CBI, the NIU used hardware checksumming and flow-through data storage.

**Protocol Performance**

While numerous host-related bottlenecks were uncovered in the course of the testbed work, experiments nevertheless achieved record transfer rates. In particular, a TCP/IP transfer rate of 500 Mbps was measured between two Cray supercomputers over a 1000-mile HIPPI/SONET link in the Casa testbed, establishing a new high for wide area end-to-end transport. The TCP implementations used in these tests included the high bandwidth windowing and other extensions defined as part of the TCP standards.

For workstation-class machines, the Arizona/Bellcore effort achieved a UDP/IP rate of 516 Mbps using a DEC Alpha 3000/600 175 MHz processor with a Turbochannel I/O bus, the Bellcore Osiris ATM board and a collocated data source. This reflects the results of eliminating memory copying discussed above, and was obtained with the software-based checksum used in the implementation disabled. With UDP checksumming turned on, a throughput of approximately 440 Mbps was obtained.

Other results were constrained to lower rates by various factors, ranging from the hardware and software problems discussed above to the shared use of supercomputer data sources. The work by UPenn in Aurora resulted in a measured TCP/IP rate of 215 Mbps over ATM/SONET using two locally connected HP PA-RISC workstations and software checksumming. The LANL CBI outboard TCP/IP implementation in the Casa testbed gave a result of 300 Mbps over HIPPI when attached to a Cray, while the Vistanet UNC NIU outboard implementation achieved 350 Mbps for UDP/IP and approximately 200 Mbps for TCP/IP over HIPPI/ATM/SONET when used with a Cray as the data source.

The Nectar workstation effort by CMU included a careful evaluation of processor utilization, and so provides a good basis for extrapolating the testbed results in this area to newer machines.

They performed TCP/IP measurements using a DEC Alpha 3000/400 133 MHz Turbochannel workstation, an external HIPPI CAB which provided hardware TCP checksumming, and the DEC OSF/1 v2.0 operating system modified to support single-copy data transfers. While the wCAB limited the maximum interface rate to 200 Mbps, host processor utilization measurements indicated that, if 100% of host processor cycles were used for communication processing, a maximum TCP/IP rate of close to 700 Mbps could be supported by the processor for a read/write memory transfer size of 128 Kbytes, and a rate of 500 Mbps for a transfer size of 64 Kbytes.

This suggests that, with a factor of 4 increase in overall processor speed relative to the Alpha processor used in the tests and the 64KB transfer size, a 622 Mbps ATM/SONET link could be filled while leaving approximately 75% of the processor available for application processing.

**XTP**

A comparative evaluation of TCP/IP and XTP was carried out by MCNC as part of the Vistanet testbed work, using all-software protocol implementations on a Cray YMP-EL (a low-end 100 MFLOP machine) connected to a HIPPI switch.

The TCP/IP code was optimized to use a combined checksum/copy operation and vectorized checksum computation, and included the high-speed extensions to the TCP standard.

The XTP code was an optimized implementation developed by the University of Virginia and ported to the Cray Unicos 8.0.2 operating system. Two different checksums were used with the XTP code, the one originally defined as part of the XTP standard and the checksum defined for TCP. The latter was used as a result of the original XTP checksum's high computational requirements on the Cray (the TCP checksum was adopted as part of the XTP standard in July 1994).

Measurements were carried out using 64KB packets for two conditions, one using the Cray's HIPPI driver in loopback mode and the second using a loopback at the external HIPPI switch. For error-free operation, TCP/IP provided higher throughput for both test modes, even when XTP used the TCP checksum.

A second set of measurements was made to determine the effect of packet errors on throughput. Since XTP included a selective retransmission mechanism while TCP did not, it was expected that XTP might show an improved relative performance for this case. For single packet errors XTP was slightly better than TCP/IP for bit error rates greater than about $2x10^{-9}$, e.g. 118 vs 110 Mbps at a ber of $6x10^{-9}$. For a simulated burst error scenario in which three consecutive packets contained errors, XTP showed a more substantial gain over TCP/IP, giving a throughput of 115 vs 90 Mbps at a ber of $6x10^{-9}$.

Since a selective retransmission mechanism is currently undergoing standardization for TCP, the advantage shown by XTP under the above error conditions will most likely be eliminated. Thus

there does not appear to be an incentive to change from the widely used TCP/IP standard to XTP for high speed operation, at least based on throughput and computational cost.

**Data Conversion**

In addition to transport layer processing, data conversions required by hardware data representation conventions can constitute a major processing requirement at gigabit speeds. In the testbeds, conversions between Cray's 64-bit floating point representation and the IEEE 32-bit representation used by other testbed computers were found to be a significant bottleneck when standard vendor conversion software was used.

This problem was addressed by researchers in the Blanca and Casa testbeds as part of their application software support work. In both cases, Sun's XDR data representation conventions were used as a machine-independent format. Since the IEEE floating point format is used by XDR and by the non-Cray computers, they isolated the conversion processing to the Cray where its vector architecture could be exploited.

Measurements by NCSA using standard XDR conversion software on a Cray YMP resulted in a peak rate of only 11 Mbps, whereas a more efficient vector-based routine developed by NCSA for the project achieved a peak rate of 570 Mbps. SDSC and Parasoft found similar behavior in their Casa testbed work.

### *4.4.4 Parallel Architectures*

Highly parallel computers with distributed memory architectures introduce a new issue to high speed host I/O, that of dealing with their internal interconnection networks. For supercomputers such as the TMC CM-5 and Intel Paragon which were used in the testbeds, hundreds of processors, each with individual memory, communicate among themselves and with external I/O interfaces via the internal interconnection network. The latter can take many different forms, for example a two-dimensional mesh in the Paragon and a hierarchical "fat tree" in the CM-5.

Internal bandwidths between individual processors within the interconnection networks also differed according to the interconnection architecture, being on the order of 200 MBytes/s in the Paragon and 10 MBytes/s in the CM-5, with aggregate bandwidth results depending on the architecture. While HIPPI interfaces on the machines were individually capable of 800 Mbps, the ability to move data between the interface and a set of processors at that rate was a significant challenge.

To provide an initial assessment of the CM-5's I/O capabilities, NCSA measured the data rates achievable in moving data between the 512-processor machine and its scalable disk array, theoretically capable of reading data at 112 MBytes/s and writing at 100 MBytes/s. The results were a maximum read rate of 95 MBytes/s and write rate of 35 MBytes/s, with the interconnection network found to be the bottleneck for at least some of the test cases used. In the case of the Paragon, the early versions of its operating system software used in the testbeds severely constrained achievable network I/O rates. Because of the inefficiencies inherent in its protocol

support, an order of magnitude speedup was achieved using an internal raw HIPPI interface with the LANL CBI outboard TCP/IP device (discussed above), which allowed most of the normal operating system path to be bypassed.

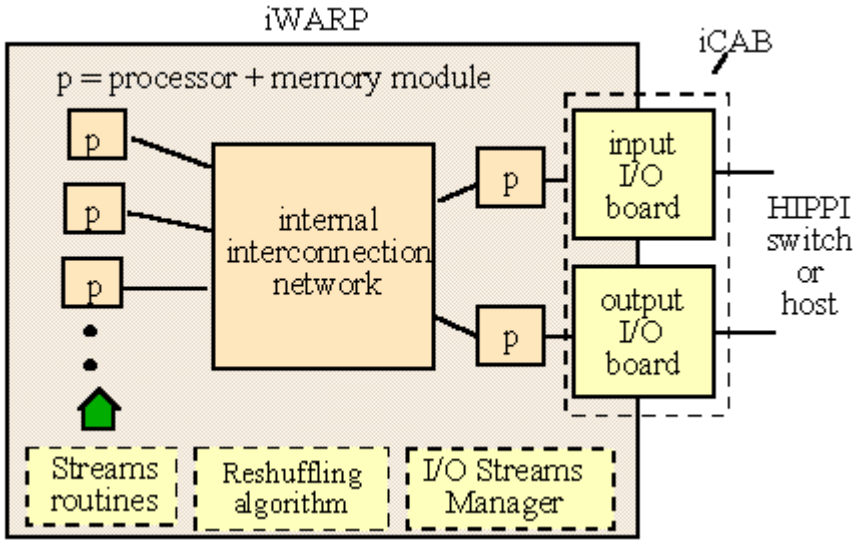
A Cray Research T3D MPP supercomputer was also used in the Casa testbed later in the project. This machine consisted of 256 processors and a three-dimensional torus interconnection network, with a peak interprocessor capability of 300 MBytes/s. Unlike the Paragon and CM-5, however, its external network connections were made to a Cray YMP used as a front-end for the T3D. Thus the HIPPI network connections were interfaced to the shared-memory YMP and did not have to deal directly with the interconnection network.

### iWarp Streams Architecture

Parallel processing work at CMU in the Nectar testbed focused on the iWarp, an experimental parallel computer developed by CMU and Intel. It used a torus interconnection network with 40 MByte/s links to each of 64 processing nodes.

The CAB outboard interfacing engine used for the DecStation HIPPI interfacing discussed earlier was also used with the iWarp, but in a different configuration. Called the iCAB, it provided an external HIPPI interface, network buffering and TCP checksum support as in the workstation CAB, but was closely coupled to two of the iWarp's processors which were used as part of the I/O interface. The processors were responsible for TCP/IP protocol processing and the coupling of the sequential network interface to the iWarp's internal distributed processor/memory system (Figure 4-16).

**Figure 4-16. iWarp Distributed Memory I/O**



The CMU work made the management of data flows between the I/O interface and the distributed memory array the responsibility of each application, which accomplished this through the use of system library code. The latter was part of a streams package, which consisted of the application library routines and a Streams Manager program executed by the I/O processors. The application (or programmer) thus decided on the best data distribution for its processor/memory array based on knowledge of the application's requirements, while the Streams Manager was responsible for efficient movement of the total set of current data streams.

A reshuffling algorithm was executed by the array processors to map the application's data distribution into one that is efficient for transfer to and from the I/O interface. Experiments with reshuffling resulted in internal aggregate transfer rates of 125 MBytes/s for array block sizes as small as 128 bytes. Without reshuffling, a block size of about 6K Bytes was required to achieve the 125 MByte/s rate, with the 128byte block size yielding only 40 MBytes/s [5].

### *4.4.5 Distributed Shared Memory*

An approach to communication between applications called Distributed Shared Memory (DSM) was investigated by two research groups. In contrast to the message-passing model used by applications in most of the testbed work, DSM attempts to emulate a shared-memory communication model across a network. The major motivation for doing so is to hide the complexities of networked communication from the application programmer, making it appear to be the same as if two or more processes were executing on a single shared-memory computer.

The major obstacle to making DSM work efficiently is end-to-end network latency, which because of propagation delay is typically much greater even across a local area network than the latency encountered within a computer. And while DSM schemes have been implemented successfully for LANs, the challenge for testbed researchers was to make it work across a gigabit wide area network.

As is the case for internal computer architectures, caching provides a basic mechanism for hiding latency, but with the problem of cache consistency greatly magnified by the much larger latencies involved. Thus the testbed work was focused on investigating techniques which could either relax cache synchronization constraints or else reduce the latency seen by an application when remotely generated data is needed, in either case reducing or eliminating the time an application must suspend execution because of a network event.

As part of the work at UIUC, researchers developed a coordinated memory model in which cache synchronization is relaxed by optimistic data sharing. That is, a program continues to execute after changing shared data, optimistically assuming that changes by others are not in transit and performing recovery mechanisms only for the hopefully small number of cases in which conflicts do occur. Experiments were performed over the Blanca testbed using three representative applications to evaluate the technique's effectiveness: a matrix multiplication, a solution technique for partial differential equations, and a quicksort algorithm. The first application represented a reference case in which network latency had no effect on the result, verifying that the

coordination scheme did not in fact degrade performance. For the last two applications, a performance improvement of 50 to 60% was achieved when coordination was used with two and three nodes. This represented a near-linear processing gain versus number of nodes speedup, with computation and communication effectively fully overlapped.

In the Aurora testbed, Penn researchers investigated several different latency-hiding techniques. The first was similar to the UIUC approach in that it relaxed cache consistency constraints, using a policy of "read gets recent write" instead of the "read gets last write" policy used with strict consistency. In addition, this approach used two different page sizes, one for control and a second for data, and a mechanism which combined page access with process state control.

Three other Penn efforts investigated ways of directly reducing the latency involved in getting data to the application. The first of these used *anticipation*, in which data was sent to remote nodes before it was requested, achieving latency reduction through the use of more network bandwidth and storage. The second investigated the use of intelligent caching within the wide area network, with a minimal spanning tree used to reduce propagation delay through strategic selection of cache locations. The third approach exploited knowledge of the application through the use of program-defined objects to achieve optimized caching lookup and reference strategies, building on the ideas of the first two schemes.

## *4.5 Network Management*

### *Summary*

- In different QoS investigations, a real-time end-to-end protocol suite was developed and successfully demonstrated using video streams over HIPPI and other networks, and a `broker' approach was developed for end-to-end/network QoS negotiations in conjunction with operating system scheduling for strict real-time constraints
- An evaluation of processing requirements for wide area QoS queuing in ATM switches, using a variation of the WFQ algorithm, found that a factor of 8 increase in processing speed was needed to achieve 622 Mbps port speeds relative to the i960/33MHz processor used for the experiments
- Congestion/flow control simulation modeling was carried out based on testbed application traffic, with the results showing rapid ATM switch congestion variations and high cell loss rates; in other work, a speedup mechanism was developed for lost packet recovery in high delay-bandwidth product networks using TCP's end-to-end packet window protocol
- An end-to-end time window approach using switch monitoring and feedback was developed and evaluated to provide high speed wide area network congestion control, and performed according to simulation-based predictions

- A control and monitoring subsystem was developed for real-time traffic measurement and characterization using carrier-based 622 Mbps ATM equipment, and was used to capture medical application traffic statistics which revealed ATM cell traffic to be more bursty than expected, dictating larger amounts of internal switch buffering than initially thought necessary
- A data generation and capture device for 800 Mbps HIPPI link traffic measurement and characterization was developed and commercialized, and was used for network debugging and traffic analysis; more generally, many network equipment problems were revealed through the use of real application traffic during testbed debugging phases

This section includes testbed work on high speed networking undertaken from a systems viewpoint, and includes investigations in the areas of quality of service (QoS), congestion/flow control, and traffic measurement and characterization. Work in this area was generally constrained by the relatively limited extent of testbed facilities, in particular the existence of only a small number of network switches and hosts, and by the amount of time needed to achieve full operation of the facilities. To deal with these constraints, analytical modeling and simulations were exploited where possible by several of the efforts in the early stages of this work, with testbed experiments successfully carried out for limited contexts later in the project.

### *4.5.1 QoS*

Several testbed efforts addressed the QoS problem for high speed networks. This work included design, implementation and experiments with a new suite of protocols for handling real-time traffic, a detailed investigation of the processing required to support sophisticated QoS queuing algorithms in gigabit networks, an investigation of how to provide guaranteed response times to applications using general operating systems, and an exploration of optimal QoS dynamic packet scheduling.

**Tenet Real-Time Protocol Suite**

Researchers in the Blanca testbed at Berkeley investigated the problem of providing end-to-end real-time traffic support in a general mixed-traffic internetworking environment. Their approach consisted of developing a suite of real-time traffic protocols to provide IP-level channel setup and data forwarding and end-to-end transport, with the protocols intended to operate in conjunction with the use of TCP/IP for non-realtime traffic.

Called the Tenet suite, it consisted of four protocols: the real-time channel administration protocol (RCAP) for channel setup across a set of internetwork routers; the real-time internet protocol (RTIP) for IP-level forwarding; the real-time message transport protocol for host-to-host transfers using the underlying RTIP protocol; and the continuous media transport protocol (CMTP), used over RTIP for periodic traffic.

The protocols included admission control, priority scheduling, and distributed rate control, and were intended to provide mathematically provable performance guarantees based on user-

specified traffic parameters. Extensive simulations were used to establish the scheme's performance, with prototype software then developed for experiments in real networks. The Berkeley-LBL portion of the Blanca testbed, an 800 Mbps HIPPI network, was used for the high speed experimentation.

Because the Tenet suite was designed for operation over packet-switched networks and HIPPI is a circuit-switched technology, the RCAP setup protocol had to be substantially redesigned (the original Blanca testbed plan was for an ATM network at Berkeley-LBL, but was changed due to delays in equipment development). The HIPPI version of RCAP made use of the time division reservation system discussed in section 0, allowing RCAP to dynamically reserve time slots within each physical HIPPI link for the real-time traffic.

Three experiments were carried out over the HIPPI network using SGI and Sun workstations, a single flow for reference purposes followed by two and then three simultaneous flows. Throughput and time variance data was collected for each case, with the results generally showing stable, well-behaved operation with only short transients occurring when a stream was turned on or off.

While these experiments demonstrated good bandwidth-sharing properties for competing real-time flows, the HIPPI time division scheme prevented simultaneous transmission of asynchronous nonrealtime traffic. To obtain data in the latter context, experiments were also carried out using a local 100 Mbps FDDI network and a wide area T1 network connecting Berkeley, UC Santa Barbara, UCLA, and UC San Diego. These experiments successfully mixed real-time video traffic with varying amounts of background traffic loads, with the video streams essentially unaffected by other traffic.

**Weighted Fair Queueing for ATM**

To establish the feasibility of using sophisticated QoS algorithms for ATM cell streams at gigabit speeds, MIT and Bellcore collaborated on an investigation of this question in the Aurora testbed. As part of their Sunshine switch effort, Bellcore prototyped a second-generation output port controller which could be used both with the switch and as a standalone device for QoS and other experiments. Known as OPC-V2, it provided for two 155 Mbps SONET OC-3c data inputs and one OC-3c data output, with a second output for switch feedback signaling. An Intel i960CA 33 MHz processor controlled the actions of hardware which directly handled cell flows, allowing different queuing strategies to be applied which could control traffic flows as a function of ATM VC identifiers and other parameters. Two hardware sort modules were provided for fast sequencing of ATM cells.

To establish the gigabit-rate processing capabilities needed for QoS schemes under consideration by the IETF and ATM Forum, MIT first investigated ways to simplify weighted fair queuing (WFQ) approaches while retaining their key properties. The result was an approximation to hierarchical weighted fair queuing which could be implemented as a single-level queue structure, allowing it to be executed on the OPC-V2.

A determination of the number of instructions required by the resulting algorithm revealed that it needed slightly more processing power than was available from the I960 to maintain the full output rate of 155 Mbps -- the i960/33 could execute approximately 80 instructions in one ATM cell time, whereas the algorithm required 88 instructions per cell. Allowing also for the additional background processing required for full operation, MIT estimated that a factor of 2 increase in processing speed would satisfy the 155 Mbps rate of the OPC-V2 output port.

Extrapolating from these results, a factor of 8 increase relative to the i960/33 would be needed for a 622 Mbps output rate, or roughly an order of magnitude increase for gigabit-rate operation of the algorithm. Since the i960/33 was a circa 1990 processor, given processor cost/performance trends it seems likely that gigabit operation of a WFQ-class algorithm will be economically achievable in the 1996-1998 timeframe.

**Extending QoS to Applications**

Researchers at Penn in the Aurora testbed investigated issues in providing service guarantees to realtime applications using general operating system environments and high speed networks. A QoS service kernel was designed to provide scheduling-based operating system guarantees to meet stringent response time requirements such as those required for remote robotics control. This was generalized to a logical framework which established the relationships and requirements between application-specified QoS parameters, operating system policies and mechanisms, and network QoS services.

To bring these results together for experimentation, a QoS *broker* was implemented for use in controlling a remote robotic arm over an ATM network. The broker negotiates between applications and underlying networks in an attempt to best satisfy the desired performance, coordinating resources over the end-to-end path of the application. Conceptually, the broker is positioned as middleware in each endpoint machine, interfacing to applications, the operating system, and network I/O.

Testbed experimentation made use of initial broker software implementations on an IBM RS/6000 workstation using the AIX Unix operating system at each endpoint, with real-time robotic control computers interfaced to the RS/6000s through bus-to-bus connections. Their principle finding was that considerable additional work is needed on general operating systems such as the AIX, in order to provide the guarantees needed for robotics and other tightly constrained applications.

**Dynamic Packet Scheduling**

A dynamic packet scheduling approach to providing service guarantees in wide area network switches and routers was investigated by Wisconsin researchers in the Blanca testbed, using a combination of analysis, simulations and experiments. Their work focused on identifying optimal schedulable regions and associated scheduling algorithms for guaranteed and predictive service classes, assuming token bucket traffic descriptors. By performing dynamic scheduling to

reallocate delay among competing traffic while still satisfying delay targets, they expected to satisfy a larger domain of service requirements than would otherwise be the case.

Experiments were carried out using Blanca testbed facilities between Wisconsin's Madison campus and UIUC in Illinois. The configuration included three Xunet ATM switches and an SGI workstation used as a router at each endpoint, with special software implemented to provide controllable traffic generation, performance measurements, and scheduling algorithms. Two traffic flows were used, a `source' flow and a background flow, with both sent from Madison to UIUC. Five scheduling strategies were compared: static priority, round robin, and dynamic scheduling with three different ratios of source to background traffic. For each strategy, experiments were run with different source traffic distributions and link utilizations, with dynamic scheduling outperforming the other approaches in nearly every case [3].

### *4.5.2 Congestion/Flow Control*

The large bandwidth-delay product of wide area gigabit networks poses important questions for network congestion and flow control schemes, since for bursty traffic the transit time of a burst may typically be much smaller than the propagation delay through the network. The small number of nodes in each testbed made it difficult to carry out meaningful experiments for this problem, however, and so it was not a major focus of testbed work.

Some analytical and simulation studies were carried out on selected aspects of the problem by NCSU and Berkeley [3,6]. MIT developed modifications to existing TCP protocol mechanisms to provide improved congestion control properties for large bandwidth-delay products, in particular to avoid long retransmission timeouts when packets are lost and to return to a full throughput state more quickly after lost packet events [2]. The lost packet events are used by TCP in the current Internet to infer the existence of network congestion and invoke a traffic backoff/recovery algorithm.

In contrast to reliance on TCP or other strictly end-to-end mechanisms to deal with network congestion, Wisconsin developed a new scheme involving both source and network node mechanisms. Called *Dynamic Time Windows* (DTW), their approach is targeted towards wide area gigabit ATM networks with large bandwidth-delay products. It uses a time window at each source to control the source's burstiness, with a larger window duration allowing greater burstiness. Feedback of information from network switches along the source-destination path causes the source to dynamically adjust its window as a function of aggregate switch traffic.

DTW is predicated on the use of fixed routes between each source and destination, as is the case for ATM virtual circuits, and on the use of weighted fair queuing (WFQ) in each switch. WFQ is used to bound additional traffic burstiness introduced by the switches, allowing DTW to bound overall network congestion time through control of the source time windows. The scheme allows network administrators to trade off packet loss due to switch buffer overflow with the time required for the network to return to a stable state after congestion occurs, through definition of a global network constant.

Analysis was used to establish the scheme's general properties under simplifying assumptions, with extensive simulations used to establish its performance for a variety of traffic and network situations. The simulation results showed the system to be stable, with its steady-state behavior characterized by periodic oscillations of source time windows about their average values. In simulation comparisons to a TCP-like end-to-end packet window system, the DTW system had lower average network delays and switch packet losses, along with higher source delays. This reflects DTW's more rapid response to network congestion events, resulting in larger queuing time of original packets by the source and fewer retransmissions due to dropped packets in the network.

Experimental data on DTW was obtained using Blanca testbed facilities between Wisconsin and UIUC. A router at Wisconsin was used as the source node, sending traffic through the Wisconsin, Chicago, and UIUC Xunet ATM switches to a receiving node at UIUC. DTW monitoring and feedback algorithms were implemented in the Wisconsin switch, with a local traffic generator at the switch used to introduce bottleneck traffic. A simple weighted round robin algorithm was used with the switch's programmable queuing hardware to approximate the use of WFQ.

The results showed that, at least for the simple experimental configuration used, DTW adjusted correctly to different values of congestion thresholds, system loads, negotiated source throughput, and changing network state. Although the link data rate was limited to 45 Mbps due to problems with Blanca 622 Mbps equipment at the time of the experiments, the experiments yielded data consistent with the more extensive simulation results and provided valuable experience with the DTW algorithms under real-world conditions.

### *4.5.3 Traffic Measurement/Characterization*

From a network technology point of view, the presence of real application traffic in the testbeds had two major benefits: it allowed network problems to be discovered during the debugging stage which would not otherwise have been found, and it allowed measurement of the traffic properties which result when real high speed networks are used.

Essentially all of the testbeds experienced problems when real traffic was used which did not occur with artificial traffic sources. Many of the problems involved equipment design subtleties, for example timing effects due to traffic burstiness. The use of real applications was especially important for this, since the traffic characteristics seen by the network are shaped by the interaction of the application and the network, and so for new contexts such as gigabit networks the traffic cannot be realistically generated a priori using artificial means.

One of the research goals of the Vistanet testbed was an investigation of traffic characteristics using its dynamic radiation therapy planning application. To allow traffic measurements at the testbed's 622 Mbps OC-12c SONET link rates, a special control and monitoring subsystem (CMS) was developed by BellSouth as part of each Network Terminal Adapter (NTA) used to interface the testbed's HIPPI hosts to the ATM/SONET network. The CMS allowed information

about ATM cells flowing through an NTA to be captured in real-time and saved for subsequent analysis.

The CMS was used by Vistanet researchers to capture and analyze radiation dose traffic generated under a number of different usage conditions, with data captured both at the transmitting NTA connected to the Cray source at MCNC and at the receiving NTA collocated with the Pixel Planes 5 destination on the UNC campus. For the testbed experiments, flow control was used between NTAs communicating across the network and between each NTA and its local HIPPI host to prevent overflowing NTA buffers, but no switch-NTA flow control or traffic shaping was applied to the ATM cells entering the network.

The resulting histograms showed the application's ATM cell traffic to be highly bursty -- mean link utilization ranged from only 0.27% to 1.2%, with each new dose transfer lasting for 40 milliseconds at an aggregate data rate of 80-90% of the available ATM cell rate (the latter was approximately 600 Mbps after subtracting out SONET overhead on the 622 Mbps link). Inspection of detailed histograms of dose bursts showed an oscillatory pattern of peaks and valleys, attributed to the end-to-end HIPPI flow control used in the testbed and possibly also due to Cray computation effects.

The different mean utilizations reflect the interburst idle times measured for different usage conditions. When an experienced user was using the system for therapy planning, idle times were approximately 2 seconds long and fairly periodic, reflecting user think time before initiating a new dose calculation and its resulting 40 millisecond burst transfer. At other times interburst idle times were as long as 28 seconds and highly variable, and were associated with demonstrations of the system.

Simulations were carried out using the measured data and a model of the testbed's Fetex 150 ATM switch to investigate the impacts of the traffic burstiness on cell loss. The results showed that a high peak cell loss rate occurred in the switch for highly bursty traffic, and that increasing the amount of cell buffering in the switch provided only a relatively small improvement over a wide range of buffer sizes. The conclusion drawn from these and other results was that either traffic shaping must be used for the ATM cell traffic, or a near peak-rate virtual circuit bandwidth allocation must be used to avoid significant cell loss rates.

As part of their Vistanet work, MCNC researchers developed the HIPPI Link Data Analyzer (HILDA) to capture 800 Mbps HIPPI traffic statistics. Configured as a single VME-bus board for operation on a Sun4 or SGI workstation, HILDA could also serve as a continuous 800 Mbps traffic source for network testing and as a standard HIPPI host interface (with data rates constrained in the latter mode by the VME bus). For data capture, the HILDA board provided a passthrough connection for a HIPPI link, analyzing and displaying the resulting traffic statistics on its host workstation or on a remotely located X-windows node. It also provided a capability for programmable error insertion onto a HIPPI link.

HILDA was used to capture application traffic data on local HIPPI networks and for the evaluation of transport protocols operating over HIPPI links, with analysis of the resulting data carried out in collaboration with NCSU researchers. More generally, HILDA proved invaluable for testing and problem diagnosis of Vistanet HIPPI/ATM/SONET facilities, and was also used for these purposes by other testbeds. A technology transfer of HILDA took place during 1992, when MCNC entered into an agreement with the Avaika Corporation to make HILDA available commercially.

## *4.6 Applications and Support Tools*

### *Summary*

- Investigations using quantum chemical dynamics modeling, global climate modeling, and chemical process optimization modeling applications identified pipelining techniques and quantified speedup gains and network bandwidth requirements for distributed heterogeneous metacomputing using MIMD MPP, SIMD MPP, and vector machine architectures
- Most of the applications realized significant speedups, with a superlinear speedup of 3.3 achieved using two machines for the chemical dynamics application; other important benefits of distributed metacomputing such as large software program collaboration-at-a-distance were also demonstrated, and major advances made in understanding how to partition application software
- Homogeneous distributed computing was investigated for large combinatorial problems through development of a software system which allows rapid prototyping and execution of custom solutions on a network of workstations, with experiments providing a quantification of how network bandwidth impacts problem solution time
- Several distributed applications involving human interaction in conjunction with large computational modeling were investigated; these included medical radiation therapy planning, exploration of large geophysical datasets, remote visualization of severe thunderstorm modeling and other problems
- The radiation therapy planning experiments successfully demonstrated the value of integrating high performance networking and computing for real-world applications; other interactive investigations similarly resulted in new levels of visualization capability, provided new techniques for distributed application communications and control, and provided important knowledge on problems which can prevent gigabit speed operation
- A number of software tools were developed to support distributed application programming and execution in heterogeneous environments; these included systems for dynamic load balancing and checkpointing, program parallelization, communications and runtime control, collaborative visualization, and near-realtime data acquisition for progress monitoring and results analysis.

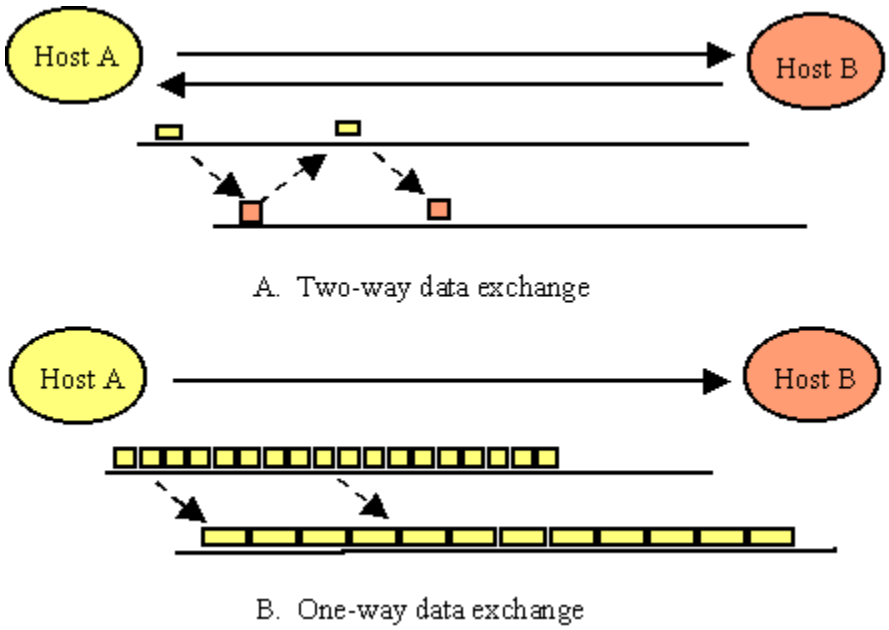
The testbeds, through experimentation using a variety of representative applications, provided major advancements in the understanding of distributed application problems and in the development of related software support tools. Work in this area led to direct scientific and technology advancements in particular application domains, and more generally mapped out new approaches to solving very large computational problems.

From a networking perspective, the application investigations can be grouped into two categories: those in which network interaction is only between computers, and those which involve human interaction. Applications in the first group ("distributed computation") attempt to minimize the wall-clock time required for a given problem, whereas applications in the second group ("human interaction") are concerned with providing a response to human input within a time deemed acceptable by the user.

For distributed computation, as processing power increases, the time for an incremental computation decreases, requiring proportionally higher network data rates to allow communication exchanges to keep up with computation (for a given problem size). As the transmission time becomes smaller due to the higher data rate, communication latency across the network becomes dominated by the speed-of-light propagation delay. For applications requiring two-way data exchanges between machines, this can cause further speedup gains to diminish as incremental computation time becomes smaller than the propagation delay.

This is illustrated in Figure 4-17A, where propagation delay causes both hosts to stop and wait for their next data input. In the one-way exchange shown in Figure 4-17B, on the other hand, propagation delay only affects the start of host B's timeline.

**Figure 4-17. Pipelining and Propagation Delay**




In many problems involving human interaction, on the other hand, a response time of from one to several seconds is acceptable to users, depending on user expectations and the cost associated with waiting for a response. Since the propagation time for light through an optical fiber spanning the US is on the order of 30 milliseconds, propagation latency should not normally pose a problem for data communications directly involving the user. However, applications in the human interaction category may also involve two-way or n-way distributed computation as part of the processing initiated by user input. Thus even though the total computation interval for the user response may be on the order of one second or more, propagation latencies could still prevent the computation from being completed within the desired user response time.

In the following, testbed results are first presented for `pure' distributed computation applications, followed by applications involving human interaction. The last part of this section summarizes testbed work which was focused on the development of software for supporting applications in a distributed heterogeneous metacomputing environment.

### *4.6.1 Distributed Computation*

Four representative applications were investigated in this category: a molecular reaction modeling problem, global climate modeling, chemical process optimization, and branch and bound combinatorial problems. In most cases, the time for obtaining results for interesting problem sizes on a single machine ranged from several hours to weeks using the fastest supercomputer available at the beginning of the project, with the problem size and/or accuracy of the results correspondingly constrained.

The nature of the first three applications allowed a functional distribution of key software components, enabling the possibility of superlinear speedups through the use of heterogeneous computer architectures, and such speedups were in fact realized during the project.

Even for cases limited to linear speedups due to problem homogeneity or because computer architecture did not make a significant difference, the resulting speedups enabled a major advance in the amount and quality of data which could be obtained. And beyond the speedup factor, testbed experience revealed or confirmed a number of other benefits of metacomputing, foremost among them the ability to take advantage of already-installed software and local expertise at a site, rather than having to port the software to another site.

Having said this, software development and porting did in fact account for a major part of testbed application activities. One reason for this was that MIMD MPP machines were not available on a production basis prior to the start of the project, precluding the use of existing application software which took advantage of the new machine architectures. A second reason was that most existing software was written for execution on a single machine, and so needed to be partitioned for distributed execution over a wide area network. Gaining an understanding of how to partition applications for heterogeneous metacomputing was in fact one of the key goals of the testbed applications research, and accounted for much researcher time in the early phases of the project.

Two important dimensions common to the applications of this group are *problem size* and *computational granularity*. Measures for problem size are specific to the type of problem, for example the number of simulated years and spatial resolution used in a global climate modeling computation. Computational granularity is a measure of the incremental computation associated with network data transfers for a given problem size.

For problems involving two-way (or n-way) interchange dependencies, one would like to find maximal speedup distributions with computational granularities which are much larger than network propagation times, thus negating speedup degradations due to speed-of-light latencies. For many problems, achieving this for the maximum-size problems which can be run with present processing power implies continuing to achieve it as processing power increases, since researchers will eagerly increase the problem size to use the new capacity.

In the following, the quantum chemical dynamics application is an example of a one-way pipeline flow in which computational granularity need not be large compared to propagation time (providing the right choices are made for the associated communication protocols). In the global climate model application, on the other hand, processing is sequentially dependent on interchanges in both directions between its distributed components, and as a result is directly impacted by propagation time.

**Quantum Chemical Dynamics**

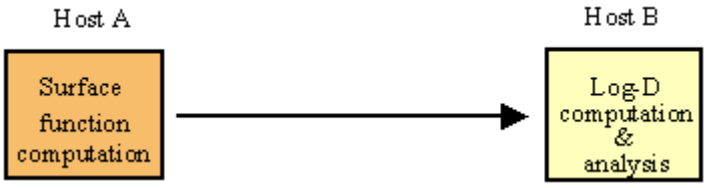
This application provides an example of basic scientific investigation through computer modeling. The objective is to predict the rates and cross-sections of chemical reactions from first principles using quantum mechanics, providing an understanding of chemical reactions at the molecular level. The work was carried out in the Casa testbed by researchers at Caltech, using supercomputers at all four Casa sites over the course of the project.

The problem solution consists of three steps: local hyperspherical surface function (LHSF) calculations, log-derivative (LOG-D) propagation calculations, and asymptotic (ASY) analysis. The first two steps are centered on matrix calculations and require large amounts of machine time for matrix diagonalization, inversion, and multiplication. The third step is not computationally intensive, and can be done in series with the other steps without impacting the total solution time.

The measure for problem size here is the number of "channels" used for the computation, with the number of channels directly determining matrix size. The channels represent the number of molecular rotational and vibrational states used in the modeling, which determines the level of detail obtained in the results. Typical problem sizes in the tested work involved on the order of 1000 channels.

This problem involves a small amount of data input, e.g. the initial conditions and size parameters, and large amounts of data output, in the range of hundreds of Megabytes to Gigabytes. Its computational structure made it a good candidate for heterogeneous metacomputing, since the LHSF and LOG-D steps provided natural architecturally-related partitioning choices. The intensive part of LHSF computation involved matrix diagonalizations, which was well-matched to vector processing, while LOG-D computation involved primarily matrix inversions and multiplications, which could be done efficiently on MPP machines (Figure 4-18).

**Figure 4-18. Quantum Chemical Dynamics**




A one-way pipelined distribution (Figure 4-17B) was developed by Caltech researchers in which a set of LHSF matrix calculations for a single hyperspherical sector were performed on a vector machine and the results sent to an MPP machine for LOG-D processing. Once the first sector's data was sent, both machines could then perform computations in parallel using this pipeline. When all matrix computations were completed, ASY processing was then carried out on the MPP machine.

Using a 1000-channel problem size, a single Cray YMP CPU at SDSC with a processing power of 0.2 Gflops, and a 64-node Intel Delta configuration at Caltech, a 5 Mbps data rate (the Delta's maximum I/O rate) allowed LHSF data to be transferred and the next LHSF computation to be completed in parallel with an incremental LOG-D computation. Using an analytical model to extrapolate this work to two machines with a processing power of 10 Gflops each, a 900 Mbps data rate would allow the problem size to be increased to 3700 channels and the pipeline again kept full.

To determine the resulting speedup, the wall-clock time for the distributed solution was compared to the wall-clock times which resulted when all execution was done on each machine alone. The problem required 29 hours to run on the single Cray C90 CPU and 28 hours on the 64-node Intel Delta. With the problem partitioned among the two machines as described above, total execution time was 8.5 hours, resulting in a superlinear speedup factor of 3.3.

In a later set of three-machine experiments, a single-processor Cray YMP located at LANL shared the LHSF processing task with an 8-processor configuration of the C90 at SDSC, while LOG-D processing was done using 128 nodes of an Intel Paragon at Caltech. Using a problem size of 512 channels, a speedup factor of 2.85 was achieved. This non-superlinear result was attributed to the imbalance of computational loads due to the relatively low-powered LANL machine used in the configuration.

To provide a comparison to the experimental results, a detailed analytical model was developed by Caltech and used to predict expected speedup gains as a function of problem and machine configuration variables. For the two-machine case the model gave excellent agreement with experimental results for a wide range of problem sizes. In the three-machine case, experimental speedup gains were about 20% less than the model's predictions due to its balanced-load assumption.

In summary, the quantum chemical dynamics work confirmed the superlinear speedup capability of wide area metacomputing and dramatically reduced the time required to solve an important class of problems, significantly advancing computational modeling capabilities for fundamental science. As proof of its effectiveness, the increased accuracy provided by the testbed experiments produced new results concerning the geometric phase effect in hydrogen reactions [4].

**Global Climate Modeling**

This application, investigated by UCLA, SDSC, and LANL in the Casa testbed, has as its objective the prediction of long-term changes to the earth's climate through the use of atmospheric and ocean computer models. Each of the models is a large software program reflecting many years of development, and each incorporates a large body of theoretical and empirical knowledge. Several different programs have been developed over time by different groups of researchers for execution on particular machines.

Because the atmosphere and oceans strongly influence each other, the model for one must exchange data with the model for the other as the computation proceeds. The problem size for this

application is determined by the spatial resolution used and the number of calendar years of time simulated by the coupled models.

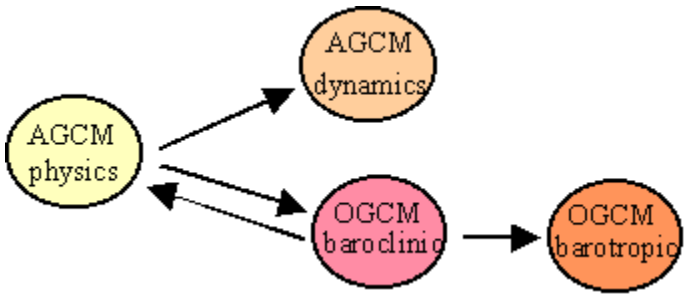
Much of the Casa work was based on an atmospheric general circulation model (AGCM) developed at UCLA and an ocean general circulation model (OGCM) developed at Princeton. The problem resolution used with these programs was 5 degrees longitude, 4 degrees latitude, and 9 atmospheric levels for the AGCM, and 1 degree longitude, 0.3 degrees latitude, and 40 depth levels for the OGCM.

A second ocean model developed at LANL was also used in Casa later in the project in conjunction with the UCLA AGCM model. Called the Parallel Ocean Program (POP), it was developed specifically for execution on LANL's CM-5 MPP to provide higher-resolution simulations. Using 256 nodes of the CM5, the POP could provide ocean modeling resolutions of 0.28 degrees longitude and 0.17 degrees average latitude with 20 depth levels.

The AGCM and OGCM programs provided a natural starting point for distributing the global climate modeling problem among the nodes of a metacomputer, since they were already separately implemented and had well-defined data exchange interfaces. However, each of these programs could also be readily decomposed into two functions, yielding a total of four distinct components: AGCM Physics, AGCM Dynamics, OGCM Baroclinic, and OGCM Barotropic. These components had significantly different machine-dependent execution times. For example, the AGCM Physics and Princeton OGCM components ran about a factor of three faster on the MPP architecture of the Intel Paragon than on a vector-based Cray C90 CPU, whereas the AGCM Dynamics was faster on the C90.

An important logical constraint on possible component distributions was imposed by the problem's data exchange requirements. The results of an AGCM Physics incremental computation had to be passed to the AGCM Dynamics and OGCM Baroclinic components before the latter could proceed with their computations, and the Baroclinic results had to be passed to the Physics and Barotropic components before they could proceed with their next computations (Figure 4-19). Thus a two-way sequential exchange was required between the Physics and Baroclinic components, constraining parallelism choices and making network latency a potentially limiting factor in the speedup which could be achieved.

**Figure 4-19. Global Climate Modeling Exchanges**




Fortunately, the incremental computation intervals for the C90-Paragon distribution were on the order of 1-2 seconds, making network speed-of-light latency insignificant. For the problem size used, if there was no overlap of computation and communication, the amounts of data exchanged required a transmission data rate on the order of 100 Mbps or higher to keep total network latency from adding significantly to solution wall-clock time. If the problem resolution and processing speed is increased, the network data rate must also be increased to maintain this result.

On the other hand, if data generation is spread out over the computation interval, the data can be sent in parallel with the computation and the data rate requirement correspondingly reduced. As part of their work, UCLA researchers developed a subdomain computation method which did in fact allow communications to take place in parallel with computation.

Taking all of these factors into account and using a configuration consisting of one CPU on the C90 at SDSC and 242 nodes on the Paragon at Caltech, a speedup of 1.55 was obtained using the best possible component distribution. The total time to run all components on only the Paragon was 9.5 hours per simulated year, which was reduced to 6.2 hours/year using both machines. Further reductions in total solution time became possible in the latter part of the project when a Cray T3D was installed at JPL. Using a combination of measurements and estimated speedups for individual components, the estimated solution time with all components running on the T3D was 6.6 hours/year. By distributing the components among the T3D and two CPUs on the SDSC C90, a total estimated solution time of 2.9 hours/year and speedup of 2.2 was obtained.

The metacomputing results for this application, while not achieving significant superlinear speedups, nevertheless can be seen to give significant reductions in solution wall-clock time. Of perhaps equal importance is the ability to couple improved-resolution models such as LANL's POP, developed explicitly for execution on the CM-5, with the UCLA or other atmospheric models. For this problem class, the enabling of collaboration-at-a-distance among research groups working different aspects of the problem may outweigh the absence of dramatic speedup gains.

In related work done by SDSC for the GCM application, a visualization tool called HERMES was developed to allow near-realtime monitoring of GCM progress during a run. The data generated by applications such as GCM is very large; for example, the LANL POP program generated nearly 50 gigabytes of data for each simulated year. Prior to HERMES this data was typically not able to be meaningfully examined until a run was completed, which was very costly in situations where errors in initial conditions or instabilities invalidated the results. HERMES allowed scientists to monitor progress during the run while introducing an overhead cost of only about 15% to the run's wall-clock time for the additional I/O processing required to send data to the visualization workstation.

**Chemical Process Optimization**

Researchers at CMU in the Nectar testbed investigated the distribution of a chemical process optimization problem using heterogeneous metacomputing. The goal of this application is to improve the economic performance of chemical processing plants through optimal assignment of resources to processing units, and is representative of a large class of stochastic optimization problems in other fields. Real problems of interest have a combinatorially large solution space, and serial algorithm solutions have been limited by their computation time to relatively small problems.

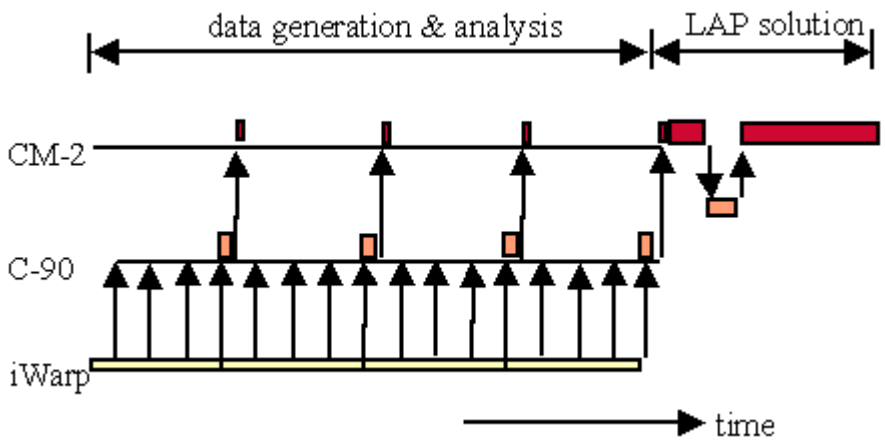
The availability of a SIMD CM-2, a MIMD iWarp array, and a vector-based C90 in the Nectar testbed allowed CMU to implement new parallel algorithm solutions for the major parts of the problem, while executing vectorized tasks where appropriate on the C90. The set of algorithms used were not previously implemented in a form which allowed execution on a single machine, precluding a direct speedup comparison for this work.

The problem size for this application is determined by parameters representing raw materials, processing units, costs, and the number of samples and grid points used in the computations. For this investigation the number of samples and grid points were fixed and a single variable, n, used to represent problem size.

The solution has two phases: a data generation and analysis phase, followed by a Linear Assignment Problem (LAP) solution phase. The data generation reflects the real world modeling of the problem, and was done on the 64-node iWarp array located at CMU. As this computation proceeded, output data was sent to the C90 at the Pittsburgh Supercomputer Center (PSC), where a single CPU was used to analyze the data and compute cost matrix elements. This data was then sent to a 32K-node CM2 at PSC.

Phase two, LAP solution, began once all data from the first phase was received by the CM2. This phase consisted of three discrete steps: first the CM2 computed a reduced cost matrix used for LAP solver initialization and sent the results to the C90, which performed the next LAP computational step; these results were then sent back to the CM2 for the final stage of LAP solution processing (Figure 4-20).

**Figure 4-20. Chemical Process Optimization**




Parallel computation was possible in the first phase, which due to its one-way data flows allowed the iWarp to compute continuously and a simple pipeline to be formed. Because the cumulative execution times for both the C90 and CM2 were only a few percent of the execution time required by the iWarp in this phase, total time was relatively insensitive to the computational granularity used by the iWarp for passing data to the C90. For example, for a problem size of n=4000, the iWarp required 13 minutes for its total computation; when divided into 8 intervals, 500 MBytes was generated for transmission to the C90 in each interval. Since transmission of the data could be spread out over the iWarp computation interval, in this case a data rate of only 50 Mbps allowed communication to the C90 to keep up with computation (with smaller rates required for C90 to CM2 transfers).

In the LAP solution phase, a single two-way exchange was required between the CM2 and C90, with computation and communication occurring sequentially, precluding the use of pipelining. For the n=4000 problem size, total CM2 execution time was 3.2 minutes, C90 time was approximately 50 seconds, and for a data rate of 100 Mbps or higher the sequential data transfers did not significantly add to total LAP solution phase time.

To determine how solution times and data rate requirements scaled with more powerful machines, CMU researchers developed a set of analytical models based on trace data obtained from the experimental runs. The results showed that, as the number of iWarp nodes is increased, it ceases to be the bottleneck and total solution time becomes quite sensitive to network bandwidth. For a 1000-node iWarp machine (estimated to be roughly equivalent to a medium-sized Intel Paragon), a data rate of at least 800 Mbps is required to prevent its data transfers to the C90 from becoming a bottleneck.

Two issues of note in the experimental work concerned format conversions and machine availability. Floating point format conversions on the C90 for data received from the iWarp made up about 35% of total C90 execution time for first-phase processing. Although masked by total

iWarp processing for the n=4000 case, the impact of this processing becomes significant as iWarp execution time is reduced.

Machine availability here refers to whether the machine was dedicated or shared during the run. For these experiments, the CM2 and iWarp processor arrays were dedicated while the C90 was shared with other jobs. Since C90 execution time was a small percentage of total time in both phases, its variability did not significantly impact the results: it was masked by the continuous iWarp processing in the first phase, and its impact on total time was less than 1.5% in the sequentially-executed second phase. While the C90 variability would have a greater impact as CM2 processing power is scaled up, this result demonstrates that tight coordination of all metacomputer machine resources is not required for some problems.

**General Branch and Bound Problems**

A Nectar testbed effort involving Purdue University researchers investigated a class of large combinatorial problems using the well-established branch and bound solution framework. Unlike the other distributed computation efforts discussed above, their work was based on homogeneous metacomputing using a network of workstations. Another difference was their focus on a general problem-solving solution rather than using a specific problem as the basis for their investigation.

Building on earlier distributed application work for particular branch and bound problems, they developed a software system for rapidly prototyping custom solutions for a large class of problems. Called DCABB (Distributed Control Architecture for Branch and Bound Computations), the software provides algorithm parallelization and distribution while also allowing the user to tailor the parallelization details to his or her specific problem. DCABB handles the details of assigning computation to individual nodes and the communications among them using a message-passing paradigm, and deals with problems such as node or network failures.

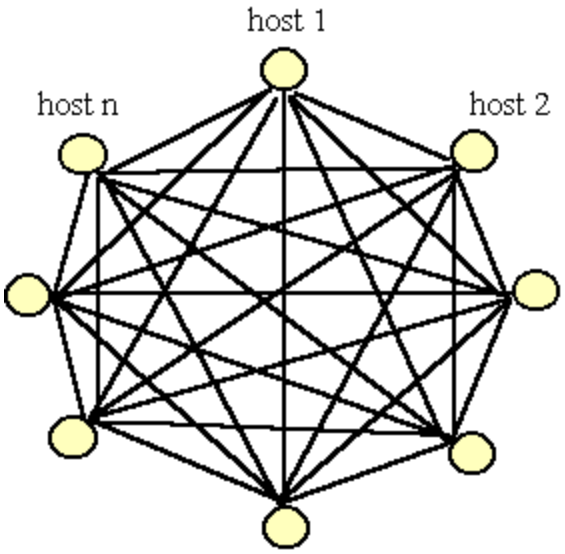
A special high level language called CLABB, Control LAnguage for Branch and Bound, was also developed to allow users to specify their particular algorithms. The CLABB specification is then parsed into a programming language such as C++, providing portability across a wide variety of platforms. The DCABB execution environment includes tools for monitoring and visualization of progress during a run, allowing the user to gain insights into changes needed for solution optimization.

The measure of problem size for this problem class is *node size*, the amount of information (in kilobytes) associated with each node of a problem's combinatorial tree. The node size reflects the number of elements and level of detail describing the problem, and is directly related to the amount of processing which must be performed for each node of the tree.

The time required to compute an individual tree node result defines the computational granularity of the distribution. DCABB distributes queues of tree nodes among the processors comprising the metacomputer, with data exchanged when a tree node computation is completed. The notion of pipelining is thus quite different for this case than the previously discussed applications --

work proceeds in parallel but asynchronously among the set of processors, with highly bursty data transfers taking place as processors become idle and tasks are redistributed. The associated data/control exchange topology is in general n-connected, as shown in Figure 4-21.

**Figure 4-21. DCABB Data Exchange**




To explore the resulting system performance, a generic branch and bound search process was developed which generated combinatorial trees similar to those for many real problems. By manipulating a number of control parameters, this approach allowed more general results to be obtained than if tests had been carried out for a few specific problems. Data was obtained for a range of key parameters using a network of eight DEC Alpha 3000/500 workstations and two network configurations, the first consisting of a shared 10 Mbps ethernet and the second consisting of switched 100 Mbps links. By comparing results for the different bandwidths, the impact of communications on total problem solution time could be characterized.

Experiments were run for problem sizes of 10, 100, and 1000 KBytes and computational granularities (node execution times) of 20-40, 200-400, and 800-1000 milliseconds, where a smaller granularity represents a processing speedup due to a faster algorithm or more powerful processor. The results showed that, for a given network bandwidth and sufficiently large problem size, there is a point beyond which decreasing the granularity results in an increase in total solution time due to increased communication requirements.

For the 1000 KByte problem size, solution time in the ethernet case monotonically increased as granularity was decreased from 800-1000 to 20-40, implying that the problem was constrained by the network bandwidth for all granularities used. When this problem size was repeated using 100 Mbps links, total solution time was reduced by a factor of two for the 20-40 granularity relative to the ethernet case. The solution time was smallest for the 200-400 granularity value, rising

again as the granularity was increased to the 800-1000 value. This implies that a still higher network bandwidth was required for this problem size to realize the speedup afforded by the 20-40 granularity.

The 100 KByte problem size showed a slight reduction in solution time with the higher bandwidth at the 20-40 granularity, indicating its bandwidth requirements were just beginning to exceed that provided by the ethernet at that point. For the 10 KByte problem size, the smallest solution time was obtained at the 20-40 point and increasing the bandwidth did not reduce this time further.

Overall, while DCABB's bursty communication requirements are difficult to characterize in detail, the experimental results demonstrated that bandwidths greater than 100 Mbps are needed for problem sizes on the order of 1000 KBytes or higher. Because of the non-overlapped n-way communication exchanges required, however, propagation latencies may impose limits on solution time reductions as bandwidths are increased beyond a certain point, particularly when wide area networks are used.

In addition to advancing our understanding of distributed computation using high speed networks, the DCABB work has produced an important methodology for solving large combinatorial problems. As a result of strong interest by industry, a company has been formed to develop and market a commercial version of DCABB directed at the solution of Mixed Integer Linear Programs, a technology widely used by manufacturing, retail distribution, and other industries.

### *4.6.2 Human Interaction*

Each of the applications described in this section involve human interaction, in which computation is initiated by a user input and terminates when the resulting response has been transferred and displayed to the user. In most cases the response consists of a visualization of a modeling computation, either a single display frame or a sequence of frames depicting motion, with generation of the latter possibly continuing until the user's next input. All involve computation requiring one or more supercomputer-class machines, with a workstation used as the user input device. The workstations were typically also used for the display, but other display devices such as stand-alone high speed frame buffer displays and a CAVE visualization facility were used as well.

**Dynamic Radiation Therapy Planning**

This Vistanet testbed application explored the use of interactive distributed computation and visualization for medical treatment planning. A collaboration of physicians and computer science graphics researchers at the University of North Carolina investigated the problem of generating radiation treatment plans for cancer patients, using Vistanet's ATM/SONET network. The goal of the application was to enable physicians to interactively explore the space of 3D treatment plans for different numbers of radiation beams and placements, working with a computerized tomography (CT) scan of the patient obtained prior to the planning session. A more general re-

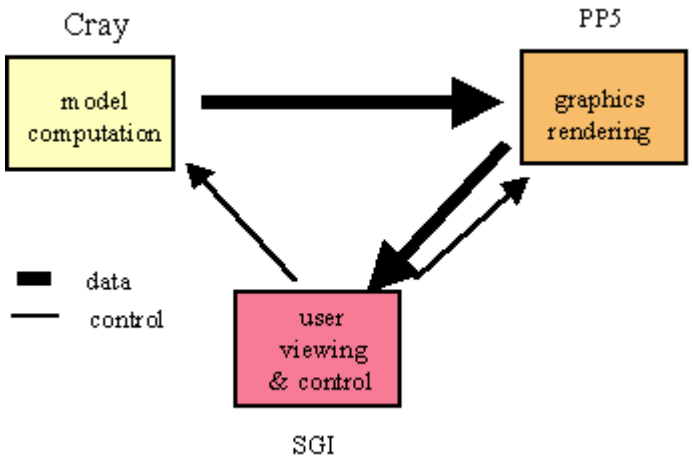
search goal of this work was the exploration of new network-based graphics generation techniques.

Treatment planning up to this time was typically limited to two-dimensional planning of beam angles, shapes and other variables, which generally led to sub-optimal treatment plans. The premise for the Vistanet work was that 3D planning would allow more accurate treatment plans to be developed, significantly improving patient cure rates while reducing problems caused by irradiation of non-targeted regions of the body. Because of the very large exploration space created by 3D planning, however, an interactive visualization-based capability was required which would let a physician quickly find good solutions. A response time of at least one frame per second was considered necessary to make the system useable, with much faster rates desirable to allow continuous visual inspection as the orientation of the 3D display is changed.

To accomplish this, two major computational steps had to be carried out. First, the tissue radiation dosage resulting from a given set of beam parameters had to be calculated relative to the patient's CT scan, then the resulting 3D dose distribution data had to be rendered into a 3D volume visualization, with the computations under interactive control of the user.

To provide the accuracy required in the results while attempting to meet the interactive response time constraints, three distinct machines were used -- an 8-processor Cray YMP located at MCNC, a specialized parallel-processor rendering machine called the Pixel Planes 5 (PP5) located in the UNC Computer Science department, and an SGI Onyx workstation located in the UNC Radiation Oncology department for physician interaction. Radiation dose data was computed on the Cray and sent to the PP5 for rendering, which sent the results to the SGI for display. User control inputs were sent from the SGI to the Cray to specify new beam parameters and to the PP5 for rendering control (Figure 4-22).

**Figure 4-22. Radiation Treatment Planning**




Both the Cray dose computations and PP5 rendering required more time for full accuracy than allowed by the one second per frame criteria (the PP5 was considered to be the most powerful rendering machine available relative to the 1990 time frame). To solve this problem within the processing power available at the time, a one-way computation pipeline was used in conjunction with progressive refinement and other rendering techniques. This allowed users to trade accuracy for speed of presentation, letting them move more rapidly through a set of beam placement choices. As the presentation reached a stationary state, the display was made more accurate.

Each display frame of dose data computed by the Cray resulted in from 4-8 MBytes of data sent to the PP5, with up to 3 MBytes per rendered frame sent by the PP5 to the SGI. Since approximately 500 Mbps of data throughput was available from the 622 Mbps ATM/SONET network link, sending 8 MBytes required approximately 0.13 seconds of transmission time. Thus, network bandwidth constrained the achievable frame rate to about 8 frames/second for the maximum data transfer conditions.

Letting the time required by the Cray to compute one frame of dose data define the pipeline computational granularity, a Cray rate of 1 frame/second provided 0.87 seconds of rendering time on the PP5. By adjusting the accuracy of its 3D volume rendering to match the available time, the rendering could be adapted to the current frame rate. This was accomplished through a combination of adaptive sampling, progressive refinement, and kinetic depth effect techniques developed by UNC researchers during the course of the project.

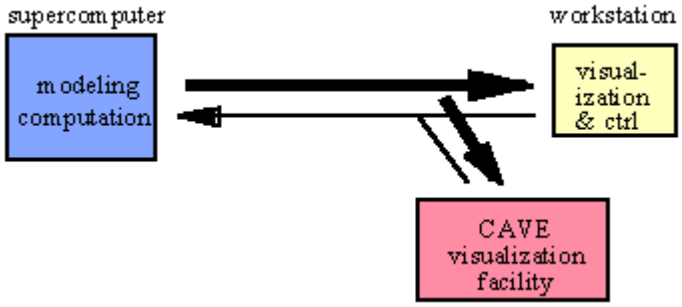
By using these and other techniques to maximize visualization detail within the imposed time constraints, the application successfully realized its goal of generating improved cancer treatment plans. More generally, it provided new insights into the importance of correct 3D perception in dealing with large and complex amounts of data through interactive visualization, and provided directions for further research in this area.

**Remote Visualization**

NCSA investigated a number of different supercomputer applications in the Blanca testbed, in order to gain an understanding of how their functionality can be best distributed for interactive visualization using a high speed network. During the course of the project, several high performance computers were available at NCSA: a Cray YMP, a TMC CM-2 and CM-5, a Convex 3880, and an SGI Power Challenge. In addition, the CAVE visualization facility developed by the University of Illinois became available in the latter part of the effort.

Much of their work was concerned with determining the data rates which could be sustained by different applications when optimized for a particular machine, with the applications partitioned to perform simulation computations on one machine and visualization rendering and display on a second machine, typically a high performance workstation (Figure 4-23).

**Figure 4-23. Remote Visualization**




An early experiment used a 3D severe thunderstorm modeling application running on a Cray YMP, to determine the maximum data rate which could be generated by the code. Using DTM for communications support, a rate of 143 Mbps was measured. The code was later ported to the CM-5 and then to the SGI Challenge, and the CAVE visualization system also used. The move to the higher-powered CM-5 in particular demonstrated the value of providing the visualization function on a physically separate machine from the modeling computations, since the CM-5 port required major restructuring of the modeling software for the highly parallel distributed memory CM-5 environment, while the visualization software remained constant and could be used with multiple modeling engines.

Other interactive-based work by NCSA included investigation of I/O rates for a cosmology modeling application. The results showed this application could generate output at a rate of 610 Mbps, requiring real-time transfer of the data to networked storage devices to avoid filling storage on the CM-5. Early work with a neurological modeling problem on the CM-2 showed that application capable of generating data at a rate of 1.8 Gbps; however, experiments which sent data from the CM-2 to a Convex 3880 over a HIPPI channel could achieve only about 120 Mbps, with CM2-HIPPI I/O and data parallel-serial transformation determined to be the bottlenecks. A general relativity application which modeled colliding black holes was run on both the CM-5 and

SGI Challenge, with the Challenge used to perform visualization processing and transfer the results to the CAVE display system. Users could also steer the modeling computation from the CAVE.

The Space Science and Engineering Center (SSEC) at the University of Wisconsin also carried out remote visualization investigations within the Blanca testbed. Their work focused on creating a distributed version of Vis-5D, an existing single-machine program for interactively visualizing modeling results of the earth's atmosphere and oceans. Experiments were carried out over Blanca facilities using an SGI Onyx workstation at Wisconsin and three different supercomputers at NCSA: a 4-processor Cray YMP, a 32-processor SGI Challenge, and a TMC CM-5.

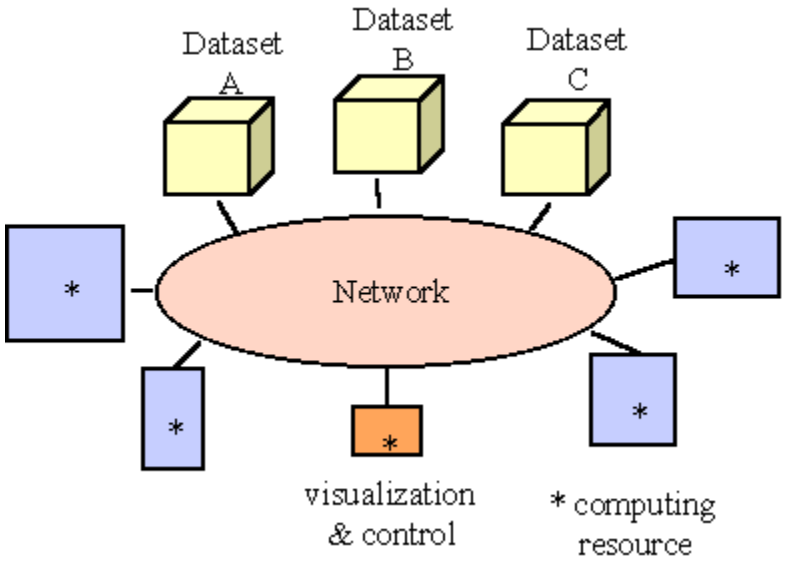
The resulting distributed version of Vis-5D partitioned the visualization functionality such that rendering was performed on the SGI workstation at the user's location, while more general computation such as isosurface generation was done on the remote supercomputer under interactive control of the workstation user. Experiments using the Cray YMP in dedicated mode showed a sustained transfer rate of 91 Mbps from the Cray to the SGI workstation, with queue size observations indicating that communication rather than computation was the bottleneck. Similar results were obtained using the SGI Challenge as the server, with a sustained rate of 55 Mbps measured.

For the CM-5, the Vis-5D server code was first ported to run in MIMD mode to maximize the isosurface generation rate. However, this approach could not be completed due to a lack of support for the MIMD mode's multiplexed I/O by the DTM communication software used for the experiments. A SIMD software port was pursued instead, but yielded poor performance -- it was found after further investigation that a critical aspect of the algorithm was executed serially by the hardware independently of the number of processors being used. Because of delays experienced in establishing an operational state for some of the facilities used for these experiments, time did not allow a resolution of these findings.

**CALCRUST: Interactive Geophysics**

Researchers at JPL in the Casa testbed used an interactive geophysics application to investigate computationally intensive problems involving very large distributed datasets. The goal was to allow interactive 3D visualization of geographic data through the integration of three distinct data sources: Landsat satellite images, elevation data, and seismic data, with the data set sizes ranging from hundreds of megabytes to several gigabytes. This effort thus combined networked databases, distributed computation, and remote visualization in a single metacomputing application (Figure 4-24).

**Figure 4-24. CALCRUST Metacomputing**




In order to achieve reasonable interactivity, an elapsed time of one second or less was desired between the user's view selection and display of a static 3D rendering. For fly-by visualizations, the goal was to generate 10-15 frames per second. However, using a single computer such as the Cray C90 for all computations required on the order of 30 seconds to generate a single frame.

The nature of the 3D rendering processing made it well-matched to a highly parallel distributed memory machine such as the Intel Paragon, while shared-memory vector machines such as the Cray were a better match for 2D preprocessing of the very large raw data sets. By assigning the Landsat, elevation, and seismic 2D processing to different machines, a pipeline could be created which allowed the three datasets to be processed in parallel while their output was sent to a fourth machine for rendering, with the latter's output sent to a frame buffer or high-resolution workstation for display.

In experiments carried out using Casa machines and connectivity available at about the mid-point of the effort, a Cray C90 at SDSC and a Cray YMP at JPL were used for 2D processing, a 512-processor Intel Delta (a precursor to the Paragon) at Caltech was used for 3D rendering, and a workstation at JPL was used to provide visualization control and display. Using the Express control software developed as part of the Casa effort, a user at the workstation could initiate the processing pipeline on the other machines, with a wireframe or other low-resolution rendering used to select a new orientation or fly-by for high resolution display. A visualization software program called Surveyor was developed by JPL as part of this effort to allow the workstation and Intel Delta to be used interactively for terrain rendering.

Because of the processing-intensive nature of the problem, the best results which could be obtained from the above configuration involved a total latency of about 5 seconds to generate a single frame, with a corresponding frame rate of 0.2 frames/second for fly-by visualizations. The dominant component of this latency was the 2D processing on the Crays. The 3D processing was successfully parallelized on the Intel Delta through the replacement of a traditional ray-casting technique with a new ray identification approach. Although the Delta's I/O limitation of 5 Mbytes/second would have been an impediment to further speedups, it proved sufficient for these experiments due to the 2D processing bottleneck.

While the goal of less than one-second total latency was not achieved with the machine complement used for the experiments, the resulting metacomputer nevertheless provided a major advance relative to previous capabilities in this area, with the techniques subsequently applied to a high-resolution visualization of the California Landers earthquake. More generally, this work is expected to have wide applicability to the scientific evaluation of very large distributed datasets produced by systems such as NASA's Earth Observing System.

### *4.6.3 General Support Tools*

In addition to the specific application area investigations discussed above, the testbed work also included the development of software tools to support general distributed applications. This activity constituted an important part of the overall testbed effort, with the resulting tools providing ongoing utility to other projects.

**DOME**

Researchers at CMU in the Nectar testbed developed a system called DOME (Distributed Object Migration Environment) to perform dynamic load balancing and checkpointing in a distributed heterogeneous computing environment. Application programmers incorporate objects from the DOME object library into their High Performance FORTRAN or C++ programs using a single-program-multiple-data (SPMD) execution model, with special DOME data object classes defined to ease the work of programming. As a measure of the latter, DOME programs are typically much shorter than those using PVM for achieving distributed execution.

At runtime DOME distributes the program over the given set of machines and initiates execution, performing dynamic load balancing and checkpointing as the application proceeds. Time is divided into repeated cycles consisting of a work phase and a load balancing/checkpointing phase. Data on task execution times is collected on each machine during the work phase, and is then exchanged between neighbor machines during the balancing phase, with portions of DOME objects migrated to neighbors exhibiting better performance for a particular task. Checkpointing is accomplished in a machine-independent manner, so that a task can be restarted on a different machine architecture if necessary in the event of a machine or network failure.

To evaluate the network overhead costs introduced by DOME in carrying out the load balancing and checkpointing, experiments were run using a particular application on a workstation cluster. For a phase interval resulting in good load balancing, the total volume of traffic exchanged by

the program increased 80%, with the traffic due to load balancing migration very bursty. Thus higher network data rates are required when using DOME to prevent the increased data exchanges from becoming a bottleneck and slowing overall execution.

**Express**

The Express effort in the Casa testbed had as its goal the development of a comprehensive software system for supporting applications running in distributed heterogeneous environments. Its starting point was existing software which had evolved out of parallel processing research at Caltech for supporting applications running on a single machine. Work during the testbed effort expanded on the original package to provide support for wide area metacomputing application algorithm design, code implementation and debugging, execution control, and performance analysis.

Communication among machines was accomplished using a message passing model, with both TCP/IP and raw HIPPI transfers supported by Express. A unified computation model was used in which the set of machines were seen by applications programmers as a single machine with multiple processors. Runtime control was centralized in an Express software "console" module which was run on one of the machines used for the application. Load balancing was provided by pre-assigning tasks based on user-supplied data about individual machine processing power, or by running a special test mode to obtain the data for the pre-execution assignments.

Express was implemented on all major machines used in Casa testbed application experiments: the Cray YMP, Cray C90, Intel Delta, Intel Paragon, TMC CM-5, and on selected workstations. In addition to its testbed use, the results of this work were incorporated into a new commercial version of Express for use by the general application community.

**DICE/DTM**

DICE, a Distributed Interactive Collaboration Environment, was developed by NCSA as part of their Blanca testbed work. It consists of a highly modular set of software objects which provide multiple-user control and data visualization for a simulation or other application running in a distributed environment. DICE separates the transmission of control and data streams as part of its communication management, applying the most appropriate handling to each stream, for example minimum latency or maximum bandwidth.

DTM, Data Transfer Mechanism, was developed by NCSA to provide a general communication environment for distributed heterogeneous applications. It uses a message-passing model consisting of block data transfers and a simple handshaking protocol for flow control, with TCP used to provide transmission reliability. DTM provides a simplified API for application programmers, while at the same time optimizing application communications by managing multiple TCP streams through a single DTM user socket.

Data transfers are optimized by DTM for the large data exchanges typical of high-end applications, and for the use of high speed networks. Messages are transferred directly in and out of

application buffers without additional buffering by DTM, minimizing latencies which would otherwise be introduced due to copying activities. Data format conversions required between different machine architectures are handled directly by DTM, which contains special code to minimize conversion processing overhead. As noted in the section on Host I/O, this can be a significant source of processing overhead if standard vendor code is used.

**BEE**

To provide a general distributed application monitoring capability, CMU researchers in the Nectar testbed developed BEE (Basis for Distributed Event Environments). In large computationally intensive distributed environments, collecting and processing information about events can add large processing overheads to the program. BEE attacks this problem by removing event interpretation from the monitored program and performing it instead on a different node, minimizing local event processing. In addition, BEE provides a uniform platform for three kinds of event analysis: runtime monitoring, runtime analysis, and post-mortem analysis.

BEE allows event data to be collected in realtime from the different execution points of a distributed application and displayed using visualization techniques. Past events during the run can be included in the current visualization display, and more generally the visualization of a program's execution can be browsed at past time points without losing current visualization information through the use of caching and synchronization mechanisms.

A second version of BEE, called BEE++, was also created during the project. BEE++ is an object-oriented framework which allows application programmers to customize BEE's performance analysis tools. In addition, BEE++ allows users to dynamically control the monitoring process during execution.

**HERMES**

The HERMES near-realtime data acquisition tool mentioned earlier in the discussion of the global climate modeling work was also applied by SDSC to general application support. It provides support for programs written in FORTRAN and C and is designed to minimize the processing overhead introduced to applications running in distributed heterogeneous environments. HERMES provides the needed program hooks and data transmission and collection mechanisms, and uses the AVS software system for visualization processing and display. A second version of HERMES added functionality to allow interactive steering of distributed simulations and other features, and was demonstrated at Supercomputing 95.

# 5 Conclusion

The Gigabit Testbed Initiative, by creating a new model for network research, has had a major impact on both education and industry. Bringing together network and application researchers, integrating the computer science and telecommunications communities, creating academia-industry-government research teams, and leveraging government to obtain substantial contributions from industry, all part of a single, orchestrated project spanning the country, provided a new type of research collaboration not previously seen.

The coupling of application and network technology research from project inception was a major step forward for both new technology development and applications progress. Having applications researchers involved from the start of the project allowed network researchers to obtain early feedback on their network designs from a user's perspective, and allowed network performance to be evaluated using actual user traffic. Similarly, application researchers could learn how the network impacts their application designs through early deployment of prototype software. Perhaps most significantly, they could proceed to investigate networked application concepts without first waiting for the new networks to become commercially available.

The coupling of computer network researchers, who have largely come from the field of computer science, with the carrier telecommunications community provided another important dimension of integration. The development of computer communications networks and carrier-operated networks have historically proceeded along two separate paths with relatively little cross-fertilization. The testbeds allowed the two communities to work together, allowing each to better appreciate the problems and solutions of the other.

From a research perspective, the testbed initiative created close collaborations among investigators from academia, government research laboratories, and industrial research laboratories. In addition to participants from leading universities, national laboratories included Lawrence Berkeley Laboratory, Los Alamos National Laboratory, and JPL, and the NSF-sponsored National Center for Supercomputer Applications, Pittsburgh Supercomputer Center, and San Diego Supercomputer Center, while industry research contributors included IBM Research, Bellcore, GTE Laboratories, AT&T Bell Laboratory, BellSouth Research, and MCNC.

Another important dimension of the testbed model was its funding structure, in which government funding was used to leverage significantly larger contributions by industry. A major industry contribution was made by the carriers in the form of SONET and other transmission facilities within each testbed at gigabit or near-gigabit rates. The value of this contribution cannot be overestimated, since not only were such services non-existent at the time, but they would have been unaffordable if they had existed under normal tariff conditions. By creating an opportunity for the carriers to learn about potential applications of high speed networks while at the same time benefiting from collaboration with the government-funded researchers in network technology experiments, the carriers were, in turn, willing to provide new high speed wide-area experimental facilities.

The Initiative resulted in significant technology transfers to the commercial sector. As a direct result of their participation in the project, two of the researchers at Carnegie Mellon University founded a local area ATM switch startup company, FORE Systems. This was the first such local ATM company formed, and provided a major stimulus for the emergence of high speed local area networking products. Applications research in the testbeds also resulted in technology transfers to industry -- examples are the DCABB software developed in the Nectar testbed for distributing large combinatorial problems over a network, which spawned a startup company to commercialize the technology for use in manufacturing, retail distribution and other industries, and the Express metacomputing control software developed in the Casa testbed, which was transferred to the marketplace.

Other technology transfers involved the Hilda HIPPI measurement device, developed as part of the Vistanet effort by MCNC, and the HIPPI-SONET wide-area gateway developed by LANL for the Casa testbed. Both of these systems have been successfully commercialized by the private sector. In other cases, new high speed networking products were developed by industry in direct response to the needs of the testbeds, for example HIPPI fiber optic extenders and high speed user-side SONET equipment. Additionally, major technology transfers occurred through the migration of students who had worked in the testbeds to industry to implement their work in company products.

On a larger scale, the testbeds directly led to the formation of three statewide high speed initiatives undertaken by carriers participating in the testbeds. The North Carolina Information Highway (NCIH) was formed by BellSouth and GTE as a result of their Vistanet testbed involvement to provide a 622 Mbps ATM/SONET network throughout the state. Similarly, the NYNET experimental network was formed in New York state by NYNEX as a result of their Aurora testbed involvement, and the California Research and Education Network (CalREN) was created by Pacific Bell as a result of their Casa testbed participation.

## *5.1 Testbed Results and Technology Trends*

While the testbed work spanned a five-year period, its experimental technology base was formed largely by the platform and component technologies available in the 1990-93 timeframe. Moreover, while the testbed final reports generally reflect platform processor capabilities available in 1993 (for example, the first generations of the Alpha workstation processor and of the Paragon supercomputer), most testbed hardware prototypes were actually based on 1990 component technology. Prototypes which interfaced with vendor platforms also constrained platform upgrades of components such as workstation I/O buses, and so the latter were also representative of circa 1990 technology.

Some of the testbed results which are likely to be impacted by platform and transmission technology advances are discussed in the following paragraphs.

## *5.2 Host I/O*

Workstation-class platforms used in the testbeds did not support gigabit speeds, even with the hardware and software optimizations introduced by the testbed work. They were constrained primarily by their DRAM memory speed and I/O bus and interrupt architectures, but also by their processing speed when simultaneously handling protocol and application processing. Maximum achievable speeds were on the order of several hundred Mbps.

For platform technology circa 1996, on the other hand, memory speed has improved by about a factor of five through the use of SDRAM, peak I/O bus speed has improved from about 500 Mbps to 2 Gbps (64-bit PCI), and Alpha processor clock speed has increased from 133 MHz to 500 MHz. Since the work by CMU predicted that a factor of four Alpha speed improvement would allow full TCP/IP use of a 622 Mbps ATM/ SONET link using only 25% of the processor, 1966 workstation platforms should in fact support gigabit speeds in parallel with application execution. Further, the improvements in memory and I/O bus bandwidths could obviate the need to eliminate operating system copying and for some of the other optimizations required in the testbed platforms.

MPP supercomputers used in the testbeds, while representing state-of-the-art machine architectures, all had difficulty in making gigabit port speeds usable by applications. In some cases this was due to bottlenecks caused by operating system software not developed for high speed network I/O, and in other cases was due to insufficient internal hardware interconnect bandwidth. While new-generation hardware and operating systems can be expected to alleviate these problems, a more difficult one may remain: how to distribute an application over a large set of internal MPP nodes so that data can be moved at high speeds between the application and the network. Some initial solutions to this problem were developed in the testbeds, but much remains to be learned.

## *5.3 Striping*

Although striping was used for local area distribution over SONET in one of the testbeds, other high speed transmission technologies were available in the 1990-92 timeframe which allowed non-striped gigabit operation over local distances, in particular HIPPI and Glink. Additional commercial technologies have since evolved which provide 622 Mbps and higher speeds for both transmission and switching within the local area. Given also the advances in host platform and component hardware, striping would not appear to be necessary for local area networking at gigabit speeds.

For wide-area transmission, the testbeds made use of the highest speed SONET equipment becoming available in 1990, which in most cases meant that striping over multiple 155 Mbps channels was required to achieve a 622 Mbps user rate. SONET has since become the dominant new carrier technology and equipment speeds have increased, but will the rate of this increase render striping unnecessary for future high speed end-users? There are at least three factors to consider in answering this question.

First, the high cost of wide-area carrier facilities slows the rate at which older equipment is replaced, making it likely that 155 Mbps or lower SONET user access rates will be around for a while. Second, while new SONET equipment can generally support synchronous 622 Mbps user access, overall user demand and cost considerations may result in continued deployment of 155 Mbps or lower user access rates by carriers. Third, even if 622 Mbps user access is widely deployed, applications such as high performance metacomputing might drive up the speeds some users need at a rate faster than that at which wide-area access rates increase.

Another factor which may make striping attractive in the future is the use of optical wavelength division multiplexing (WDM) to exploit the inherent but largely still untapped bandwidth of optical fiber, both in wide and local area environments. New WDM technology is being deployed by carriers which provides 16x2.5 Gbps channels in a single optically-amplified fiber with an aggregate bandwidth of 40 Gbps. Plans for WDM equipment which will provide 40x2.5 Gbps channels, for an aggregate bandwidth of 100 Gbps, have been announced.

Whether striping will be necessary and/or desirable in a WDM environment depends in part on whether individual user requirements increase beyond that of a single WDM channel, on whether carriers deploy WDM facilities, and, in part, on which technology proves the most cost-effective for a given total rate. Given the additional hardware ports needed for a striping solution, the latter is more likely to be invoked in situations pushing the state-of-the-art of expensive high bit-rate hardware or to deal with legacy equipment.

## *5.4 Switching*

Switches used in the testbeds were dominated by specialized hardware architectures, in contrast to lower speed software packet switches. ATM cell switches ranged from Batcher-Banyan to simple crossbar hardware architectures, and HIPPI switches also consisted of crossbar hardware. This hardware approach is now finding its way into high speed vendor router products for the Internet, with a combination of hardware switching and traditional software switching being introduced to the marketplace.

While the hardware advances discussed above for host I/O might also be applied to some switch processing functions, the short duration of ATM cells at gigabit speeds will likely require that hardware continue to play an important role for ATM, both in switching and in host I/O cell operations. For PTM switching, this question may depend on whether trunk rates continue to rise through advances in TDM technology or will flatten out through use of multiple WDM channels, allowing continued processing advances to make software packet switching an economical alternative.

## *5.5 Network Protocols and Algorithms*

While technology advances cannot change the speed of light, they have increased the choices that can be made in congestion control and quality-of-service protocols and algorithms. Testbed results predicted that about a factor of eight was required relative to 1990 processor technology to execute a simplified weighted fair queuing algorithm on a 622 Mbps ATM cell stream, and

1996 processor technology has roughly provided that increase. To the extent that more sophisticated algorithms can help with this problem area, then, processing technology trends should have a positive impact.

Transmission advances could also have an important impact here. The quality-of-service and congestion control problems would be simplified if bandwidth were plentiful and inexpensive, allowing resource contention to be reduced through larger bandwidth allocations. To date this has not occurred, but the future exploitation of optical fiber bandwidth could conceivably bring this situation about if user demand does not increase correspondingly – a big if, however, given the recent history of computer networking.

## *5.6 Future Research Infrastructure*

Since the beginning of computing, the communications dimension has not been able to keep pace with the rest of the field. Among the barriers most often cited are costs of the technology and its deployment over large geographic areas, the regulated nature of the industry, and market forces for applications that could make use of it and sustain its advance. Moreover, most people find it difficult to invest their own time or resources in a new technology until it becomes sufficiently mature that they can try it out and visualize what they might do with it and when they might use it.

A recent National Research Council report [1] includes a summary of the major advances in the computing and communications fields from the beginning of time-sharing through scalable parallel computing, just prior to when the gigabit testbeds described in this report were producing their early results. Using that report's model, the gigabit testbeds would be characterized as being in the early conceptual and experimental development and application phase. The first technologies were emerging and people were attempting to understand what could be done with them, long before there was an understanding of what it would take to engineer and deploy the technologies on a national scale to enable new applications not yet conceived.

The gigabit testbeds produced a demonstration of what could be done in a variety of application areas, and educated people in the research community, industrial sector, and government to provide a foundation for the next phase. Within the federal government, the testbed initiative was a stimulus for the following events:

- The HPCCIT report on Information and Communication Futures identified high performance networking as a Strategic Focus.
- The National Science and Technology Council, Committee on Computing and Communications held a two day workshop which produced a recommendation for major upgrades to networking among the HPC Centers to improve their effectiveness, and to establish a multi-gigabit national scale testbed for pursuing more advanced networking and applications work.

- The first generation of scalable networking technologies emerged based on scalable computing technologies.
- The DoD HPC Modernization program initiated a major upgrade in networking facilities for their HPC sites.
- The Advanced Technology Demonstration gigabit testbed in the Washington DC area was implemented.
- The defense and intelligence communities began to experiment with higher performance networks and applications.
- The NSF Metacenter and vBNS projects were initiated.
- The all-optical networking technology program began to produce results with the potential for 1000x increases in transmission capacity.

To initiate the next phase of gigabit research and build on the results of the testbeds, CNRI proposed that the Government continue to fund research on gigabit networks using an integrated experimental national gigabit testbed involving multiple carriers, with gigabit backbone links provided by the carriers at no cost using secondary (i.e., backup) channels and switches and access lines paid for by the Government and participating sites. However, costs for access lines proved to be excessive, and at the time the Government was also unable to justify the funding needed for a national gigabit network capability -- instead, several efforts were undertaken by the Government to provide lower speed networks.

The role for a national gigabit network within the research community is clear. In the not too distant future, we expect the costs for accessing a national gigabit network on a continuing basis will be more affordable and the need for it will be more evident, particularly its potential for stimulating the exploration of new applications. The results of the initial testbed project have clearly had a major impact on breaking down the barriers to putting high performance networking on the same kind of growth curve as high performance computing, thus enabling a new generation of national and global-scale high performance systems which integrate networking and computing.

# References

1. “Evolving the High Performance Computing and Communications Initiative to Support the Nation’s Information Infrastructure”, National Research Council, 1995.

2. Aurora Testbed Final Report*

3. Blanca Testbed Final Report*

4. Casa Testbed Final Report*

5. Nectar Testbed Final Report*

6. Vistanet Testbed Final Report*

7. Magic Testbed: http://www.magic.net

*Testbed final reports are available from CNRI for the cost of reproduction. They may also be requested directly from the relevant testbed organizations.

# Appendix A: Publications and Reports

Adam, J., “The Vidboard: A Video Capture and Processing Peripheral for the ViewStation System”, Master’s Thesis, September 1992 (MIT, Aurora)

Adam, J., H. Houh, M. Ismert, and D. Tennenhouse, “A Network Architecture for Distributed Multimedia Systems”, Proc. of the International Conf. on Multimedia Computing and Systems, May 1994 (MIT, Aurora)

Adam, J., H. Houh, and D. Tennenhouse, “Experience with the VuNet: A Network Architecture for a Distributed Multimedia System”, Proc. of the IEEE 18th Conf. on Local Computer Networks, Sept. 1993 (MIT, Aurora)

Adam, J. and D. Tennenhouse, “The Vidboard: A Video Capture and Processing Peripheral for a Distributed Multimedia System”, Proc. of the 1993 ACM Multimedia Conf., August 1993 (MIT, Aurora)

Adam, J., H. Houh, M. Ismert, and D. Tennenhouse, “Media-Intensive Data Communications in a Desk-Area Network”, IEEE Communications, August 1994 (MIT, Aurora)

Ahmadi, H., R. Guerin and K. Sohraby, “Analysis of a Rate-Based Access Control Mechanism for High-Speed Networks”, Proc. GLOBECOM ‘90 and IBM Research Report No. RC 15831, IEEE Trans. Comm., 1992 (IBM, Aurora)

Ahmadi, H., J-S. Chen and R. Guerin, “Dynamic Routing and Call Control in High-Speed Integrated Networks”, Proc. Workshop Sys. Eng. Traf. Eng., ITC’13, 1991 (IBM, Aurora)

Alexander, D., C. Brendan S. Traw, and J. Smith, “Embedding High Speed ATM in UNIX IP”, in USENIX High-Speed Networking Symposium, August 1994 (Penn, Aurora)

Arabe, J., A. Beguelin, B. Lowekamp, E. Seligman, M. Starkey and P. Stephan, “DOME: Parallel Programming in a Heterogeneous Multi-User Environment”, Technical Report CMU-CS-95-137, Carnegie Mellon University, April 1995 (CMU, Nectar)

Bagheri, M., D-T. Kong, Wayne S. Holden, F-C. Irizarry, D-D. Mahoney, D-C. Larson, “An Experimental 2.488 Gigabit/sec Sonet STS-3c to STS-48 Byte Multiplexer and Demultiplexer”, GLOBECOM ‘91, December 1991 (Bellcore, Aurora)

Bala, K., I. Cidon and K. Sohraby, “Congestion Control for High-Speed Packet Switched Networks”, Proc. INFOCOM ‘90 (IBM, Aurora)

Banerjea, A., and B. Mah, “The Real-Time Channel Administration Protocol”, Proc. 2nd Int’l. Workshop on Network and Operating System Support for Digital Audio and Video, November 1991 (Berkeley, Blanca)

Banerjea, A. and S. Keshav, “Queuing Delays in Rate Controlled ATM Networks”, Proc. INFOCOM’ 93, March-April 1993 (Berkeley, Blanca)

Banerjea, , A., E. Knightly, F. Templin, and H. Zhang, “Experiments with the Tenet Real-Time Protocol Suite on the Sequoia 2000 Wide Area Network”, Proc. 2nd Annual ACM Multimedia Conf., October 1994 (Berkeley, Blanca)

Banerjea, A., C. Parris and D. Ferrari, “Recovering Guaranteed Performance Service Connections from Single and Multiple Faults”, Technical Report TR-93-066, International Computer Science Institute, Berkeley, CA, Nov. 1993 (Berkeley, Blanca)

Banerjea, A., D. Ferrari, B. Mah, M. Moran, D. Verma and H. Zhang, “The Tenet Real-Time Protocol Suite: Design, Implementation, and Experiences”, IEEE/ACM Trans. on Networking, 1995 (Berkeley, Blanca)

Basch, B., W. Bruwer, D. Casey, W. Smith, and D. Spears, “Vistanet: A BISDN Field Trial”, IEEE LTS, Vol. 2, No. 3, August 1991 (GTE/BellSouth, Vistanet)

Bassett, M., G. Kudva, J. Pekny and S. Subrahmanyam, “Using Distributed Computing to Support Integrated Batch Process Scheduling, Planning, and Design Under Market Uncertainty”, Foundations of Computer Aided Process Design Conference Proceedings, Snowmass CO, 1995 (CMU, Nectar)

Bauer, M., “Self-Framing Packets in the ATM Adaptation Layer”, BS Thesis, May 1992 (MIT, Aurora)

Bajcsy, R., D. Farber, R. Paul, and J. Smith, “Gigabit Telerobotics: Applying Advanced Information Infrastructure”, Technical Report MS-CIS-93-11, January 1993; also in 1994 International Symposium on Robotics and Manufacturing, August 1994 (Penn, Aurora)

Bajcsy, R., D. Farber, R. Paul, and J. Smith, “Gigabit TeleManufacturing: Applying Advanced Information Infrastructure”, Journal of High-Speed Networks, 1993 (Penn, Aurora)

Becker, D., R. Singh and S. Tell, “An Engineering Environment for Hardware/ Software Co-Simulation”, Design Automation Conf. ‘92, June 1992 (UNC, Vistanet)

Becker, D., R. Singh, and S. Tell, “Software/Hardware Co-Simulation with Verilog and C++”, Proc. First Open Verilog International User’s Group Meeting, March 1992 (UNC, Vistanet)

Bergman, L., “Casa Gigabit Network -- Distributed Supercomputer Applications,” National Net’92, March 1992 (JPL, Casa)

Bergman, L., “Spinoff Applications of Casa CALCRUST in Bio-Med Imaging,” SDI Workshop on Bio-Medical Imaging Networks, Jan 1992 (JPL, Casa)

Bettati, R. and A. Nica, “Real-Time Networking over HIPPI”, Joint Workshop on Parallel and Distributed Real-Time Systems, Santa Barbara, CA, April 1995 (Berkeley, Blanca)

Biersack, E., C. Cotton, D. Feldmeier, A. McAuley and W. Sincoskie, “Gigabit Networking Research at Bellcore”, IEEE Network, March 1992 (Bellcore, Aurora)

Biersack, E., “Performance Evaluation of Forward Error Correction in ATM Networks”, Proc. ACM/ SIGCOMM ‘92, August 1992 (Bellcore, Aurora)

Biersack, E., C. Cotton, D. Feldmeier, and A. McAuley, “An Overview of the TP++ Transport Protocol Project”, March 1991 (Bellcore, Aurora)

Biersack E. and D-C. Feldmeier, “Transport Protocol Issues for ATM-based Networks”, Proc. EFOC/LAN 90, June 1990 (Bellcore, Aurora)

Biersack, E., “Efficient Connection Management using Synchronized Clocks”, 3rd Int’l. Conf. on High Speed Networking, February 1991 (Bellcore, Aurora)

Biersack, E. and D. C. Feldmeier, “A Timer-Based Connection Management Protocol with Synchronized Clocks and its Verification”, Computer Networks and ISDN Systems, June 1993 (Bellcore, Aurora)

Binder, R, “Issues in Gigabit Networking”, Globecom ‘92 Proceedings (CNRI)

Blom, R., L. Bergman, R. Crippen, E. Frost, K. Hussey, P. Lyster, D. Okaya, and D. Stanfill, “Interactive, Regional-Scale Geological Data Exploration and Analysis Across a Gigabit Computing Network: A Part of the Casa Gigabit Network Testbed,” 9th Thematic Conference on Geologic Remote Sensing, Feb 1993 (JPL, Casa)

Blom, R., K. Bertan, R. Crippen, J. Ford, R. Dokka, and E. Frost, “Observations on the Use of Landsat TM and SPOT Image Data in Tectonic Studies of the Southwestern United States.” Pro-

ceedings of the International Geoscience and Remote Sensing Symposium (IGARSS'91) June 1991 (JPL, Casa)

Bottomley L. and A. Nilsson, "Traffic Characterization in a Wide Area Network", Proc. of Tri-Comm '92, February 1992. (NCSU, Vistanet)

Braun, H., B. Chinoy, K. Claffy and G. Polyzos, "Analysis and Modeling of High-Speed Networks: Project Status Report", April 1992 (SDSC, Casa)

Broscius, A., "Hardware Analysis and Implementation of the NBS Data Encryption Standard", MS Thesis, May 1991 (Penn, Aurora)

Broscius A. and J. Smith, "Exploiting Parallelism in Hardware Implementation of the DES", July 1991, Proc. Crypto 91 (Penn, Aurora)

Bruegge, B., H. Nishikawa, and P. Steenkiste, "Computing over Networks: An Illustrated Example", 6th Distributed Memory Computing Conference, April 1991 (CMU, Nectar)

Bruegge, B. and T. Gottschalk, "A Framework for Dynamic Program Analyzers", OOPSLA '93 (CMU, Nectar)

Bruegge, B., "A Portable Platform for Distributed Event Environments", SIGPLAN/SIGOPS Workshop on Parallel and Distributed Debugging, May 1991 (CMU, Nectar)

Bruegge B. and P. Steenkiste, "Supporting the Development of Network Programs", International Conference on Distributed Computing Systems, May 1991 (CMU, Nectar)

Bruegge, B., "Proceedings of the ACM/ONR Workshop on Parallel and Distributed Debugging", May 1991 (CMU, Nectar)

Bruwer, W., W. Smith, and D. Spears, "A B-ISDN Field Trial with Medical Applications", National Fiber Optic Engineers Conf., April 1991 (Bellsouth, Vistanet)

Butler, R., J. Godsil, and V. Welch, "Host Network Performance Measurements on the NCSA Ultranet Network", NCSA Technical Report, April 1992 (NCSA, Blanca)

Bryant, D., D. Casey, and D. Stevenson, "Broadband Technology and Gigabit Data Networks", Tricomm 90, March 1990 (Bellsouth, GTE, MCNC, Vistanet)

Caceres, R., P. Danzig, S. Jamin, and D. Mitzel, “Characteristics of Wide-Area TCP/IP Conversations”, SIGCOMM ‘91, Sept. 1991 (Berkeley, Blanca)

Caceres, R., “Efficiency of Asynchronous Transfer Mode Networks in Transporting Wide-Area Data Traffic”, Technical Report TR-91-043, International Computer Science Institute, Berkeley, July 1991 (Berkeley, Blanca)

Caceres, R., “Multiplexing Traffic at the Entrance to Networks”, Ph.D. Dissertation, Technical Report, UCB/CSD 92 Dec. 1992 (Berkeley, Blanca)

Campbell, R., N. Islam, P. Madany, D. Raila, and A. Sane, “Experiences Building an Object-Oriented system in C++”, Communications of the ACM, 1993 (UIUC, Blanca)

Campbell, R. and D. Pointer, “HIPPI to XUNET Adapter Function Board (HXA-FB) Design Specification”, Technical Report, July 1992 (UIUC, Blanca)

Campbell, R., S. Dorward, A. Iyengar, C. Kalmanek, G. Murakami, R. Sethi, C. Shieh, and S. Tan, “Control Software for Virtual-Circuit Switches: Call Processing”, In Future Tendencies in Computer Science, Control and Applied Mathematics, 1992 (UIUC, Blanca)

Catlett, C. and L. Smarr, “Metacomputing”, Communications of the ACM, June 1992 (NCSA, Blanca)

Catlett C. and J. Terstriep, “The Use and Effect of Multi-Media Digital Libraries in a National Network”, April 1991 (NCSA, Blanca)

Catlett, C., “In Search of Gigabit Applications”, IEEE Communications Magazine, April 1992 (NCSA, Blanca)

Catlett, C., “Internet Evolution and Future Directions,” Chapter 19 in “Internet System Handbook,” Edit, by M. Rose and D. Lynch, Addison Wesley 1992 (NCSA, Blanca)

Catlett, C., “Balancing Resources”, IEEE Spectrum, September 1992 (NCSA, Blanca)

Catlett C. and L. Smarr, “Life After Internet: Making Room for New Applications”, Building Information Infrastructure, 1992 (NCSA, Blanca)

Chang, Y., “High-Speed Transport Protocol Evaluation in the Vistanet Project”, TriComm ‘92, February 1992 (MCNC, Vistanet)

Chang, Y., “Protocol Evaluation for TCP, XTP, and VMTP: Phase I Report”, Vistanet Gigabit Project Report, February 1992 (MCNC, Vistanet)

Chao, H., D. Kong, N. Cheung, M. Arnould and H. Kung, “Transport of Gigabit/sec Data Packets Over the SONET/ATM Network”, Globecom ‘91 (Bellcore/CMU, Nectar)

Chao, H., “Design of Leaky Bucket Access Control Schemes in ATM Networks”, ICC ‘91, June 1991 (Bellcore, Aurora)

Chao H. and D. Smith, “Design of a Virtual Channel Queue in an ATM Broadband Terminal Adaptor”, IEEE Infocom ‘92, May 1992 (Bellcore, Nectar)

Chao, H., “A General Architecture for Link-Layer Congestion Control in ATM Networks”, Proc. ISS 92, October 1992 (Bellcore, Aurora)

Charny, A., D. Clark and R. Jain, “CongestionControl with Explicit Rate Indication”, ICC ‘95, June 1995 (MIT, Aurora)

Chen, M., Z. Shae, D. Kandlur, T. Barzilai and H. Vin, “A Multimedia Desktop Collaboration System”, Globecom ‘92 Proc. (IBM, Aurora)

Cheung N. and H. Kung, “Gigabit/sec Wide Area Computer Networks: Potential Applications and Technology Challenges”, OFC ‘91, February 1991 (Nectar)

Cheung, N., “SONET/ATM -- The Infrastructure for Gigabit Computer Networks”, IEEE Communications Magazine, April 1992 (Bellcore, Nectar)

Chinoy, B. and K. Fall, “TCP/IP Performance in the Casa Gigabit Network”, Usenix Symposium on High-Speed Networking”, August 1994 (SDSC, Casa)

Chipman, K., P. Holzworth, J. Loop, N. Ransom, D. Spears, and B. Thompson, “Medical Applications in a B-ISDN Field Trail,” IEEE Journal on Selected Areas in Communications, Vol. 10, No. 7, September, 1992 (BellSouth, Vistanet)

Chipman, K., P. Holzworth, J. Loop, and D. Spears, “High-Performance Applications Development for BISDN”, Proc. 1992 International Switching Symposium, October 1992 (BellSouth, Vistanet)

Cidon I. and I. Gopal, “Control Mechanisms for High-Speed Networks”, Proc. ICC ‘90 (IBM, Aurora)

Cidon, I., J. Derby, I. Gopal and B. Kadaba, “A Critique of ATM from a Data Communications Perspective”, Proc. ICCC ‘90, November 1990 (IBM, Aurora)

Cidon, I., I. Gopal and S. Kutten, “Optimal Computation of Global Sensitive Functions in Fast Networks”, Distributed Algorithms, Proc. 4th Int’l Workshop on Distributed Algorithms, September 1990 (IBM, Aurora)

Cidon, I., I. Gopal and R. Guerin, “Bandwidth Management and Congestion Control in PlaNET”, IEEE Commun. Mag., October 1991 (IBM, Aurora)

Cidon, I., R. Guerin and A. Khamisy, “An investigation of Application Level performance in ATM Networks”, INFOCOM ‘95, April 1995 (IBM, Aurora)

Cidon, I., I. Gopal, M. Kaplan and S. Kutten, “Distributed Control for PARIS”, Proc. 9th Annual ACM Symp. on Principles of Distributed Comp., 1990 (IBM, Aurora)

Cidon, I. and Y. Ofek, “Metaring -- A Full Duplex Ring with Fairness and Spatial Reuse”, Proc. Infocom ‘90 (IBM, Aurora)

Cidon, I., I. Gopal and A. Segall, “Fast Connection Establishment in High-Speed Networks”, Proc. SIGCOMM ‘90, September 1990 (IBM, Aurora)

Claffy, K., H. Braun, and G. Polyzos, “Measurement Considerations for Assessing Unidirectional Latencies”, Internetworking: Research and Experience, 1993 (SDSC, Casa)

Clark, D., B. Davie, D. Farber, I. Gopal, B. Kadaba, W. Sincoskie, J. Smith, and D. Tennenhouse, “The Aurora Gigabit Testbed”, Computer Networks and ISDN Systems, vol 25, No. 6, February 1993 (Aurora)

Clark, D., S. Shenker and L. Zhang, “Supporting Real-Time Applications in an Integrated Services Packet Network: Architecture and Mechanism”, ACM/SIGCOMM ‘92, August 1992 (Aurora)

Clark, D., B. Davie, D. Farber, I. Gopal, B. Kadaba, W. Sincoskie, J. Smith and D. Tennenhouse, “An Overview of the Aurora Gigabit Testbed”, Proc. IEEE Infocom’ 92, May 1992 (Aurora)

Clay, R., “Scheduling in the Presence of Uncertainty: Probablistic Solution to the Assignment Problems”, M.S. Thesis, 1991 (CMU, Nectar)

Clay, R. and P. Steenkiste, “Distributing a Chemical Process Optimization Application over a Gigabit Network”, Supercomputing ‘95, December 1995 (CMU, Nectar)

Clay, R. and G. McRae, “Solution of Large Scale Linear Assignment Problems in the Presence of Uncertainty”, AICHE Summer National Meeting, August 1991 (CMU, Nectar)

Comer, M., M. Condry, S. Cattanach and R. Campbell, “Getting the Most for your Megabit”, ACM CCR Journal, July 1991 (UIUC, Blanca)

Condry, M. and S. Lim, “The Object-Oriented Advantage in Prototyping a Remote File System”, Proc. 2nd International Workshop on Object-Orientation in Operating Systems, September 1992 (UIUC, Blanca)

Crippen, R., “Regional Exploration in Desert Terrains: A Guide to the Use of Landsat Thematic Mapper Imagery,” Proceedings of the Eighth Thematic Conference on Geologic Remote Sensing (ERIM) April 1991 (JPL, Casa)

Crippen, R., and R. Blom, “Imageodesy: A Tool for Mapping Subpixel Terrain Displacements in Satellite Imagery”, Intnl. Union of Geodesy and Geophysics Meeting, Boulder CO, July 1995 (JPL, Casa)

Curkendall, D., P. Li, and E. DeJong, “Planetary Scientific Visualization- Using the Intel Delta Parallel Supercomputer”, Int’l. Conf. on Earth and Space Science Info. Systems, February 1992 (JPL, Casa)

Danzig, P, S. Jamin, R. Caceres, D. Mitzel, D. Estrin, “An Empirical Workload Model for Driving Wide-Area TCP/IP Network Simulations”, Journal of Internetworking Research and Experience, March 1992 (Berkeley, Blanca)

Davie, B., “A Host-Network Interface Architecture for ATM”, SIGCOMM ‘91 (Bellcore, Aurora)

Davie, B., “ATM Network Modeling and Design for Bursty Traffic”, ISS ‘90, May 1990 (Bellcore, Aurora)

Davie, B. and A. Descloux, “The Effects of Bursty Traffic on ATM Switching Systems”, Bellcore Symposium on Performance Modeling, May 1990 (Bellcore, Aurora)

Davie, B., “The Architecture and Implementation of a High-Speed Host Interface”, IEEE JSAC, February 1993 (Bellcore, Aurora)

Davie, B., J. Smith, D. Clark, D. Farber, I. Gopal R. Guerin, W. Sincoskie, and D. Tennenhouse, “Aurora: An Experiment in Gigabit Network Technologies”, High Performance Communications, January 1993, High Performance Networks: Frontiers and Experience, 1994 (Aurora)

Davie, B., J. Smith and C. Traw, “Host Interfaces for ATM Networks”, In A. Tantawi, editor, High Performance Networks: Frontiers and Experience, 1994 (Bellcore/Penn, Aurora)

Davie, B., “The costs of ATM: Some empirical observations”, Proc. 6th ERCIM Workshop, June 1994 (Bellcore, Aurora)

Davie, B., “Network-Friendly Workstations”, SPIE Photonic East ‘95 Symposium, Philadelphia PA, October 1995 (Bellcore, Aurora)

DeAddio, M., “TCP/IP UNIX Evaluation in a High-Bandwidth ATM Network, MS Thesis and BS Thesis, May 1992 (MIT, Aurora)

Delp, G., D. Farber, R. Minnich, J. Smith, and M-C. Tam, “Memory as a network abstraction”, July 2, 1991, IEEE Network (Penn, Aurora)

Descloux, A., “Buffer Requirements for Switches with Saturating Periodic Inputs”, 7th ITC Seminar on Broadband Technology, October 1990 (Bellcore, Aurora)

Descloux, A., “Stochastic Models for ATM Switching Networks”, J. on Selected Areas in Communications, April 1991 (Bellcore, Aurora)

Druschel, P., L. Peterson, and B. Davie, “Experiences With a High-Speed Network Adaptor: A Software Perspective”, Proc. ACM SIGCOMM ‘94, August 1994 (Arizona/Bellcore, Aurora)

Effelsber, W., E. Muller-Menrad, “Dynamic Join and Leave for Real-Time Multicast”, Technical Report TR-93-056, International Computer Science Institute, Berkeley, CA, Oct. 1993 (Berkeley, Blanca)

Faber, T., L. Landweber and A. Mukherjee, “Dynamic Time Windows: Packet Admission Control with Feedback”, Proc. ACM SIGCOMM ‘92, August 1992 (Wisconsin, Blanca)

Faller, N., “Measuring the Latency Time of Real-Time UNIX-like Operating Systems”, Technical Report TR-92-037, International Computer Science Institute, Berkeley, CA, June 1992 (Berkeley, Blanca)

Feldmeier, D., “A Survey of High Performance Protocol Implementation Techniques”, High Performance Communication, Kluwer Academic Publishers, 1993 (Bellcore, Aurora)

Feldmeier D., “An Overview of the TP++ Transport Protocol Project”, High Performance Communication, Kluwer Academic Publishers, 1993 (Bellcore, Aurora)

Feldmeier, D., “A Framework Architectural Concepts for High-Speed Communication Systems”, IEEE JSAC, April 1993 (Bellcore, Aurora)

Feldmeier, D., “Multiplexing Issues in Communication System Design”, SIGCOMM ‘90, September 1990 (Bellcore, Aurora)

Feldmeier, D. and A. McAuley, “Reducing Protocol Ordering Constraints to Improve Performance”, 3rd IFIP Int. Workshop, May 1992 (Bellcore, Aurora)

Feldmeier, D. and E. Biersack, “Comparison of Error Control Protocol for High Bandwidth-Delay Products Networks”, Second WG6.4 Workshop on Protocols for High Speed Networks, November 1990 (Bellcore, Aurora)

Ferrari, D., “Design and Application of a Delay Jitter Control Scheme for Packet-Switching Internetworks”, Proc. 2nd Int’l. Workshop on Network and Operating System Support for Digital Audio and Video, November 1991; also in Computer Communications, July-August 1992 (Berkeley, Blanca)

Ferrari, D., “Real-Time Communication in an Internetwork”, Technical Report TR-92-001, International Computer Science Institute, Berkeley, CA, January 1992; also in Journal of High Speed Networks, 1992 (Berkeley, Blanca)

Ferrari, D., “Distributed Delay Jitter Control in Packet-Switching Internetworks”, Technical Report TR-91056, International Computer Science Institute, Berkeley, CA, Oct. 1991; also in Journal of Internetworking: Research and Experiences, 1993 (Berkeley, Blanca)

Ferrari, D., A. Gupta, M. Moran, and B. Wolfinger, “A Continuous Media Communication Service and its Implementation”, Proc. GLOBECOM ‘92, December 1992 (Berkeley, Blanca)

Ferrari, D., J. Ramaekers, and G. Ventre, “Client-Network Interactions in Quality of Service Communications Environments, Proc. 4th IFIP Conf. on High Performance Networking, December 1992 (Berkeley, Blanca)

Ferrari, D., “Client Requirements for Real-Time Communication Services”, Technical Report TR-90-007, International Computer Science Institute, Berkeley, CA, March 1990; also in Proc.

International Conference on Information Technology, Oct. 1990; Request for Comments 1193, Nov. 1990; and IEEE Communications, Nov. 1990 (Berkeley, Blanca)

Ferrari, D., “A New Admission Control Method for Real-Time Communication in an Internetwork”, Real-Time Systems, 1994 (Berkeley, Blanca)

Ferrari, D., A. Banerjea, and H. Zhang, “Network Support for Multimedia–A Discussion of the Tenet Approach”, Technical Report TR-92-072, International Computer Science Institute, Berkeley, CA, Oct. 1992; also in Computer Networks and ISDN Systems, special issue on Multimedia Networking, 1994 (Berkeley, Blanca)

Ferrari, D. and A. Gupta, “Resource Partitioning for Real-Time Communication”, Proc. IEEE Symp. on Global Data Networking, Dec. 1993 (Berkeley, Blanca)

Ferrari, D. and D. Verma, “A Scheme for Real-Time Channel Establishment in Wide-Area Networks”, IEEE J. Selected Areas in Communications, April 1990 (Berkeley, Blanca)

Ferrari, D. and D. Verma, “Buffer Space Allocation for Real-Time Channels in a Packet-Switching Network”, Technical Report TR-90-022, International Computer Science Institute, Berkeley, CA June 1990 (Berkeley, Blanca)

Ferrari, D. and D. Verma, “Quality of Service in ATM Networks”, Technical Report TR 90-064, International Computer Science Institute, December 1990 (Berkeley, Blanca)

Ferrari , D.and D. Verma, “Real-Time Communication in a Packet-Switching Network”, Proc. Second International Workshop on Protocols for High-Speed Networks, Nov. 1990; also in M.J. Johnson, Protocols for High-Speed Networks, II, 1990 (Berkeley, Blanca)

Fisher, D., “Getting up to Speed on Stage-Three NREN”, Globecom ‘91 Proceedings (NSF)

Fisher, T., “Real-Time Scheduling Support in ULTRIX-4.2 for Multimedia Communication”, Proc. Third International Workshop on Network and Operating System Support for Digital Audio and Video, Nov. 1992 (Berkeley, Blanca)

Fraser, A.,, C. Kalmanek, A. Kaplan, W. Marshall and R. Restrick, “XUNET 2: A Nationwide Testbed in High-Speed Networking” (AT&T, Blanca)

Gautam, N., “Host Interfacing: A Coprocessor Approach”, SM Thesis, Feb 1993 (MIT, Aurora)

Georgiadis, L., R. Guerin, V. Peris and K. Sivarajan, “Effect of Traffic Shaping in Efficiently Providing End-to-End Guarantees”, 1st Intl. ATM Traffic Expert Symp., Basel Switzerland, April 1995 (IBM, Aurora)

Ghil, M. and C. Mechoso, “Data Assimilation and Predictability Studies for the Coupled Ocean-Atmosphere System”, 1992 (UCLA, Casa)

Giacopelli, J., W. Sincoskie, “Sunshine: A High Performance Self-Routing Broadband Packet Switch Architecture”, ISS ‘90, May 1990 (Bellcore, Aurora)

Giacopelli, J., T-T. Lee and W-E. Stephens, “Scalability Study of Self- Routing Packet Switch Fabrics for Very Large Scale Broadband ISDN Central Offices”, GLOBECOM 90, December 1990 (Bellcore, Aurora)

Giacopelli, J., M.Littlewood and W. Sincoskie, “Sunshine: A Broadband Packet Switch Architecture”, Proc. ISS ‘90, May 1990 (Bellcore, Aurora)

Gilge, M. and R. Gusella, “Motion Video Coding for Packet Switching Networks -- An Integrated Approach”, SPIE Visual Communications and Image Processing ‘91, Nov. 1991 (Berkeley, Blanca)

Gopal, I., I. Gopal and S. Kutten, “Broadcast in Fast Networks”, Proc. INFOCOM ‘90, June 1990 (IBM, Aurora)

Gopal, I., R. Guerin, J. Janniello and V. Theoharakis, “ATM Support in a Transparent Network”, IEEE Networks, Vol. 6, No. 6, November 1992 (IBM, Aurora)

Gopal, I. and R. Guerin, “Network Transparency: The PlaNET Approach”, Proc. INFOCOM ‘92 (IBM, Aurora)

Greenberg, M., “A Proposed Implementation of Distributed Shared Memory In a VAXcluster Computing Environment”, M.S.E. Thesis, November 1990 (Penn, Aurora)

Guerin, R., H. Ahmadi and M. Naghshineh, “Equivalent Capacity and It’s Application to Bandwidth Allocation in High-Speed Networks”, IEEE J. Select. Areas Commun., September 1991 (IBM, Aurora)

Guerin, R. and L. Gun, “A Unified Approach to Bandwidth Allocation and Access Control in Fast Packet-Switched Networks”, Proc. INFOCOM ‘92 (IBM, Aurora)

Gun, L. and R. Guerin, “A Framework for Bandwidth Management and Congestion Control in High Speed Networks”, Computer Networks and ISDN Systems, 1992 (IBM, Aurora) Note: See also Proc. Tricomm ‘92,

Gupta, A. and M. Moran, “Channel Groups: A Unifying Abstration for Specifying Inter-Stream Relationships”, Technical Report TR-93-015, International Computer Science Institute, Berkeley, CA, March 1993 (Berkeley, Blanca)

Gupta, A., W. Heffner, M. Moran, C. Szyperski, “Network Support for Realtime Multi-Party Applications”, Proc. Fourth International Workshop on Network and Operating System Support for Digital Audio and Video, Nov. 1993 (Berkeley, Blanca)

Gusella, R., “A Characterization of the Variability of Packet Arrival Processes in Workstation networks”, Technical Report UCB/CSD 90/612, Univ. of California, Berkeley, Dec. 1990 (Berkeley, Blanca)

Gusella, R., “A Measurement Study of Diskless Workstation Traffic on an Ethernet”, IEEE Trans. Communications, Sept. 1990 (Berkeley, Blanca)

Gusella, R., “Characterizing the Variability of Arrival Processes with Indices of Dispersion”, Technical Report TR-90051, International Computer Science Institute, Berkeley, CA, Sept. 1990; also in IEEE Journal on Selected Areas in Communications, Feb. 1991 (Berkeley, Blanca)

Hemy, M. and P. Steenkiste, “Gigabit I/O for Distributed-Memory Systems: Architecture and Applications”, Supercomputing ‘95, December 1995 (CMU, Nectar)

Hibbard, W., B. Paul, J. Terstriep, and C. Catlett, “Distributed Scientific Visualization on High Performance Networks”, course notes for SIGGRAPH Course 7, Distributed Scientific Visualization on High-Performance Networks, July 1992 (Wisconsin, Blanca)

Hibbard, W., D. Santek and G. Tripoli, “Interactive Atmospheric Data Access via High Speed Networks”, Computer Networks and ISDN Systems, 1991 (Wisconsin, Blanca)

Hibbard, W. and B. Paul, “Distributed Visualization at the Space Science and Engineering Center”, Notes for SIGGRAPH Course 7, Distributed Scientific Visualization on High-Performance Networks, 1992 (Wisconsin, Blanca)

Hibbard, W. and C. Dyer, and B. Paul, “Display of Scientific Data Structures for Algorithm Visualization”, IEEE ‘92, October 1992 (Wisconsin, Blanca)

Hibbard, W., B. Paul, “Energy Generation by Controlled Thunderstorm (Video)”, SIGGRAPH Video Review, 1992 (Wisconsin, Blanca)

Hibbard, W. and C. Dyer, and B. Paul, “Using VIS-AD to Visualize a Cloud Discrimination Algorithm (Video)”, IEEE ‘92, October 1992 (Wisconsin, Blanca)

Hibbard, W., B. Paul, D. Santek, C. Dyer, A Battaiola, M-F. Voidrot-Martrinez, “Interactive Visualization of Earth and Space Science Computations”, IEEE Computer, 1994 (Wisconsin, Blanca)

Hickey J. and W. Marcus, “The Implementation of a High Speed ATM Packet Switch Using CMOS VLSI”, ISS ‘90, May 1990 (Bellcore, Aurora)

Hickey, J., “A 50 MIP ATM Cell Processor for B-ISDN”, in Proc. IEEE Custom Integrated Circuits Conf., May 1992 (Bellcore, Aurora)

Hickey, T. Bogovic, B. Davie, W. Marcus, V. Massa, L. Trajkovic and D. Wilson, “The Architecture of the Sunshine B-ISDN Network”, Proc. ISS ‘92, October 1992 (Bellcore, Aurora)

Hoe, J., “Start-up Dynamics of TCP Congestion Control and Avoidance”, MIT Masters Thesis, May 1995 (MIT, Aurora)

Holtsinger, D. and H. Perros, “Performance Analysis of Leaky Bucket Policing Mechanisms”, 2nd ORSA Telecommunications Conf. March 1992 (NCSU, Vistanet)

Holtsinger, D., “Design and Analysis of the Dual Leaky Bucket Policing Mechanism for ATM Networks”, ICC ‘93 (NCSU, Vistanet)

Holtsinger, D. and H. Perros, “Performance of the Buffered Leaky Bucket Policing Mechanism”, High Speed Communication Networks (edited by H. Perros) Plenum (NCSU, Vistanet)

Holtsinger, D., “Performance Analysis of Leaky Bucket Policing Mechanisms”, Ph.D. thesis, Dec. 1992 (NCSU, Vistanet)

Houh, H., J. Adam, M. Ismert, C. Lindblad, and D. Tennenhouse, “The VuNet Desk Area Network: Architecture, Implementation, and Experience”, IEEE Journal of Selected Areas in Communications, May 1995 (MIT, Aurora)

Houh, H., D. Tennenhouse, “Reducing the Complexity of ATM Host Interfaces”, Proc. of Hot Interconnects II Symposium, August 11-12, 1994 (MIT, Aurora)

Ismert, M., “The AVLink: An ATM Bridge Between the VuNet and Sunshine”, MIT Bachelor’s Thesis, June 1993 (MIT, Aurora)

Ismert, M., “ATM Network Striping”, MIT Masters Thesis, February 1995 (MIT, Aurora)

Iqbal, M., M. Stern, J. Young, H. Izadpanah, R. Standley, and J. Gimlett, “A 2.5 Gb/s SONET Datalink with STS-12c Inputs and HIPPI Interface for Gigabit Computer Networks”, GLOBECOM ‘92, December 1992 (Bellcore, Nectar)

Johnston, C., K. Young, Jr., D. Smith, K. Walsh and N. Cheung, “A Programmable ATM/AAL Interface for Gigabit Network Applications”, Globecom ‘92, December, 1992 (Bellcore, Nectar)

Johnston, C., D. Smith, and K. Young Jr., “A generic ATM/AAL terminal adaptar”, ICC ‘93, May, 1993 (Bellcore, Nectar)

Johnston, C. and H. Chao, “The ATM Layer Chip: An ASIC for B-ISDN Applications”, IEEE JSAC, 1991 (Bellcore, Aurora)

Johnston, C., K. Young, K. Walsh, and N. Cheung, “A Programmable ATM/AAL Interface for Gigabit Network Applications”, GLOBECOM ‘92, December 1992 (Bellcore, Nectar)

Johnston, C., D. Smith, and K. Young., “A Generic ATM/AAL Terminal Adapter”, ICC ‘93, May 1993 (Bellcore, Nectar)

Johnston, C., “Architecture and Performance of HIPPI-ATM-Sonet Terminal Adapters”, IEEE Communications, April 1995 (Bellcore, Aurora)

Johnston, C., D. Smith and K. Young, “ATMTraP: An ATM Traffic and Perfomance Measurement Tool”, INFOCOM ‘95, April 1995 (Bellcore, Aurora)

Johnston, W., “The Use of HIPPI in an IP Network Environment”, ACM/IEEE Supercomputing, November 1992 (LBL, Blanca)

Johnston, W., V. Jacobson, D. Robertson, B. Tierney and S. Loken, “High Performance Computing, High Speed Networks, and Configurable Computing Environments”, CRC Press, November 1992 (LBL, Blanca)

Jou, F., A. Nilsson, and F. Lai, “Performance Analysis of an ATM Switching System under Bursty Arrivals”, IFIP workshop, January 1993 (NCSU, Vistanet)

Jou, F., A. Nilsson, and F. Lai, “A Refined Approximation of a Finite Capacity Polling System under ATM Bursty Arrivals”, Integrated Broadband Communication Networks and Services, April 1993 (NCSU, Vistanet)

Jou, F., A. Nilsson, and F. Lai, “Tractable Analysis of a Finite Capacity Polling System under Bursty and Correlate Arrivals, ICC’ 93 (NCSU, Vistanet)

Kalmanek, C., H. Kanakia, and S. Keshav, “Rate Controlled Servers for Very High-Speed Networks”, Conference Record, GlobeComm ‘90, Dec. 90 (Berkeley, Blanca)

Keeton, K., B. Mah, S. Seshan, R. Katz, and D. Ferrari, “Providing Connection-Oriented Network Services to Mobile Hosts”, Proc. 1993 USENIX Symposium on Mobile and Location-Independent Computing, August 1993 (Berkeley, Blanca)

Keshav, S., “A Control-theoretic Approach to Flow Control”,Technical Report TR-91-015, International Computer Science Institute, Berkeley, CA, March 1991; also in Proc. ACM SIGCOMM 91, Sept. 1991 (Berkeley, Blanca)

Keshav, S., “Congestion Control in Computer Networks”, Ph.D. Dissertation , Univ. of California, Berkeley, September 1991, also, Technical Report UCB/CSD 91/649, Univ. of CA (Berkeley, Blanca)

Kiamilev, F., J. Morris, J. Childers, R. Sharma, V. Badoni, M. Feldman, and D. Stevenson, “Optically Interconnected MCM for Gigabit ATM Switches” SPIE 93 (MCNC, Vistanet)

Kim, H., “A Non-Feedback Approach to Congestion Control for High-Speed Data Networks, PhD Thesis, University of Pennsylvania, CIS Dept., 1995 (Penn, Aurora)

Kim, H. and D. Farber, “The Failure of Conservative Congestion Control in Large Bandwidth-Delay Product Networks”, INET ‘95, Honolulu Hawaii, June 1995 (Penn, Aurora)

Kleinpaste, K., P. Steenkiste and B. Zill, “Software Support for Outboard Buffering and Checksumming”, SIGCOMM ‘95, September 1995 (CMU, Nectar)

Knightly, E. and G. Ventre, “Galileo: A Tool for Simulation and Analysis of Real-Time Networks”, Proc. IEEE 1993 International Conference on Network Protocols, October 1993 (NCSA, Blanca)

Knightly, E. and R. Mines, “Test Applications for a Heterogeneous Real-Time Network Testbed”, 1994 International Conference on Computer Communications and Networks, Sept. 11-14, 1994 (NCSA, Blanca)

Kolawa, A., “Portable Programming for Parallel/Distributed Computers: EXPRESS,” Supercomputer ‘92, Nov 1992 (Caltech, Casa)

Kong, D., “2.488 Gb/s SONET Multiplexer/Demultiplexer with Frame Detection Capability”, IEEE Journal, 1991 (Bellcore, Nectar)

Kudva, G. and J. Pekny, “A Distributed Exact Algorithm for the Multiple Resource Constrained Sequencing Problem”, 1992 (CMU, Nectar)

Kudva, G. and J. Pekny, “DCABB: A Distributed Control Architecture for Branch and Bound Calculations”, Computer and Chemical Engineering, Vol. 19, 1995 (CMU, Nectar)

Kung, H., P. Steenkiste, M. Gubitoso, and M. Khaira, “Parallelizing a New Class of Large Applications over High-speed Networks”, Proceedings of Third ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming (PPoPP) April 1991 (CMU, Nectar)

Kung, H., “Network-Based Multicomputers: An Emerging Parallel Architecture”, IEEE ‘91, November 1991 (CMU, Nectar)

Kung, H., “Gigabit Local Area Networks” A Systems Perspective”, IEEE Communications Magazine, April 1992 (CMU, Nectar)

Kuo, H., A. Nilsson, D. Winkelstein, and L. Bottomley, “Traffic Measurements on HIPPI Links in a Supercomputing Environment”, TriComm ‘93, April 1993 (NCSU/MCNC, Vistanet)

Kuppermann, A., “Ab Initio Quantum Mechanical Calculations of the Cross Sections and Rates of Chemical Reactions,” Proceedings of the First Intel Delta Applications Workshop, Technical Report CCSF-14-92, California Concurrent Supercomputing Facilities, February 1992 (Caltech, Casa)

Kuppermann, A. and Y-S Wu, “The Quantitative Prediction and Lifetime of a Pronounced Reactive Scattering Resonance”, Chem. Phys. Letters, No.241, 1995 (Caltech, Casa)

Kuppermann, A. and Y-S. Wu, “Casa Gigabit Network Testbed: Quantum Chemical Reaction Dynamics,” Technical Report CCSF-2-91, Caltech Concurrent Supercomputing Facilities, 1991 (Caltech, Casa)

Kwan, T. and J. Terstriep, “Experiments with a Gigabit Neuroscience Application on the CM-2”, Supercomputing 93 (NCSA, Blanca)

Kwan, T. and D. Reed, “Performance of the CM-5 Scalable File System”, 8th ACM Intl. Conf. on Supercomputing, July 1994 (NCSA, Blanca)

Lalk, G., L. Gluck, T. Banwell, C. Johnston and K. Young, “A Highly Integrated ATM/SONET User-Network Interface”, OFC ‘92 Technical Digest (Bellcore, Aurora)

Lalk, G., B. Davie, K. Young and W. Marcus, “An OC-12/STS-3c/ATM Interface for Gigabit Network Applications”, Electronics Letters (Bellcore, Aurora)

Lalk, G., B. Davie, K. Young and W. Marcus, “An OC-12/STS-3c/ATM Interface for Gigabit Network Applications”, Proc. ICC 93, June 1993 (Bellcore, Aurora)

Li, P. and D. Curkendall, “Parallel Three Dimensional Perspective Rendering”, Proc. 2nd European Workshop Parallel Computing, March 1992 (JPL, Casa) Liao, W., “An Implementation of RTP (Version 1) for the x-Kernel”, Technical Report, Department of Computer Science, UIUC, August 1994 (UIUC, Blanca)

Lim, S. and M. Condry, “Optimization of a Remote File System for Heterogeneous Network Topologies”, IECON 93, Nov. 1993 (NCSA, Blanca)

Lim, S., “Adaptive Caching in a Distributed File System”, PhD Thesis, U. Illinois Dept. of Computer Science, November 1995 (UIUC, Blanca)

Lindblad, C., “VuSystem Performance Measurements”, NOSSDAV ‘95, Durham NH, April 1995 (MIT, Aurora)

Lindblad, C., D. Wetherall, W. Stasio, J. Adam, H. Houh, M. Ismert, D. Bacher, B. Phillips and D. Tennenhouse, “ViewStation Applications: Implications for Network Traffic”, IEEE JSAC, June 1995 (MIT, Aurora)

Lockwood, J., C. Cheong, S. Ho, B. Cox, S. Kang, S. Bishop, and R. Campbell, “The iPOINT Testbed for Optoelectronic ATM Networking”, Conference on Lasers and Electro-Optics, May 1993 (UIUC, Blanca)

Lockwood, J., H. Duan, J. Morikuni, S. Kang, S. Akkineni, and R. Campbell, “Scalable Optoelectronic ATM Networks: The iPOINT Fully Functional Testbed”, IEEE Journal of Lightwave Technology, 1994 (UIUC, Blanca)

Loop, J., B. Basch, J. Symon, D. Becker, D. Winkelstein, R. Singh and S. Bharrat, “Vistanet Deployment and System Integration Experiences” (GTE/BellSouth, Vistanet)

Lowery, C., “Protocols for Providing Performance Guarantees in a Packet-Switching Internet”, Technical Report TR-91-002, International Computer Science Institute, Berkeley, CA, Jan. 1991 (Berkeley, Blanca)

Lyster, P., L. Bergman, P. Li, D. Stanfill. B. Crippen, R. Blom, C. Pardo, and D. Okaya, “Casa Gigabit Supercomputing Network: CALCRUST Three- Dimensional Real-Time Multi-Dataset Rendering,” Supercomputing ‘92 November 1992 (JPL, Casa)

Lyster, P., “Casa-CALCRUST Distributed Supercomputing,” Proceedings of the First Intel Delta Applications Workshop, Technical Report CCSF-14-92, California Concurrent Supercomputing Facilities, February 1992 (JPL, Casa)

Ma, C-C., Y. Chao, C. Mechoso, and W. Weibel, “Comparison of Vertical Mixing Schemes for Ocean General Circulation Models”, Con. on Climate Variations, Oct 91 (UCLA, Casa)

Mah, B., “A Mechanism for the Administration of Real-Time Channel”, MS Report Technical Report UCB/CSD 93/735, March 1993 (Berkeley, Blanca)

Mah, B., S. Seshan, K. Keeton, R. Katz, and D. Ferrari, “Providing Network Video Service to Mobile Clients”, Proc. Fourth Workshop on Workstation Operating Systems, Oct. 1993 (Berkeley, Blanca)

Makrucki, B., “On the Performance of Submitting Excess Traffic to ATM Networks”, Globecom 91 (Bellsouth, Vistanet)

Makrucki, B., “A Study of Source Traffic Management and Buffer Allocation in ATM Networks”, 7TH ITC Specialist Seminar (Bellsouth, Vistanet)

Makrucki, B., “Analysis of the Vistanet Network Terminal Adapter Buffer System”, Bellsouth technical memorandum TM-ATSEC-08-91-035, August 1991 (Bellsouth, Vistanet)

Marcus, W., “A CMOS Batcher and Banyan Chip Set for B-ISDN Packet Switching”, IEEE J. Solid-State Circuits, Dec. 90 (Bellcore, Aurora)

McAuley, A., “Reliable Broadband Communication Using a Burst Erasure Correcting Code”, SIGCOMM ‘90, September 1990 (Bellcore, Aurora)

McAuley, A. and C. Cotton, “A Self-Testing Reconfigurable Cam”, IEEE Journal, March 1991 (Bellcore, Aurora)

Mechoso, C., C-C. Ma, J. Farrara, J. Spahr, R. Moore, W. Dannevik, M. Wehner, P. Eltgroth and A. Mirin, “Distributing a Climate Model Across Gigabit Networks”, Proc. HPDC-1, IEEE Computer Society Press, September 1992 (UCLA, Casa)

Mechoso, C., C-C. Ma, J. Farrara, J. Spahr, R. Moore, “Parallelization and Distribution of a Coupled Atmosphere-Ocean General Circulation Model”, Mon. Wea. Rev., in press 1992 (UCLA,Casa)

Mechoso, C-C. Ma, J. Farrara, and J. Spahr, “Simulations of Interannual Variability with a Coupled Atmosphere-Ocean General Circulation Model”, Conf. on Climate Variations, Oct 91 (UCLA, Casa)

Mechoso, C., J. Farrara, C-C. Ma, J. Spahr and R. Moore, “Distribution of a Climate Model Across High-Speed Networks”, Supercomputing 91, November 1991 (UCLA/SDSC, Casa)

Mechoso, C., C-C. Ma, J. Farara and J. Spahr, “Climate Studies Using a Coupled Atmosphere-Ocean Several Circulation Model”, Research Activities in Atmospheric and Oceanic Modeling, World Climate Research Programme, 1994 (UCLA/SDSC, Casa)

Mechoso, C., J. Farara and J. Spahr, “Running a Climate model in a Heterogeneous Distributed Computer Environment”, Parallel and Distributed Techonology, 1994 (UCLA/SDSC, Casa)

Messina, P., “Casa Gigabit Network Testbed”, Supercomputer ‘91, November 1991 (Caltech, Casa)

Messina, P., “Parallel Computing in the 1980s: One Person’s View”, Special Issue: Practical Parallel Computing: Status and Prospects, December 1991 (Caltech, Casa)

Messina, P., “Parallel and Distributed Supercomputing at Caltech,” Procedings of COMPCON ‘91, February 1991 (Caltech, Casa)

Messina, P., “Casa Gigabit Network,” Optical Fiber Communication (OFC) Conference, February 1992 (Caltech, Casa)

Minnich, R. and D. Farber, “Reducing Host Load, Network Load, and Latency in a Distributed Shared Memory”, 10th Int’l. Conference on Distributed Computing Systems, June 1990 (Penn, Aurora)

Minnich, R., “Memory Systems for Network Multiprocessors”, Ph.D Thesis, 1991 (Penn, Aurora)

Minnich, R., J. Shaffer and J. Smith, “Mether: A Network Shared Memory Which Supports Application-Controlled Consistency”, October 1992 (Penn, Aurora)

Moore, R., “File Servers, Networking, and Supercomputers”, Adv. Info. Storage Syst., Vol. 4, 1992 (SDSC,Casa)

Moran, M. and B. Wolfinger, “A Continuous Media Data Transport Service for Real-time Communication in High Speed Networks”, Proc. Workshop on System Support for Continuous Time Media, June 1991 (Berkeley, Blanca)

Moran, M. and R. Gusella, “System Support for Efficient Dynamically-Configurable Multi-Party Interactive Multimedia Applications”, Proc. 3rd Int’l. Workshop on Network and Operating Systems Support For Digital Audio and Video, November 1992 (Berkeley, Blanca)

Mukherjee, A., L. Landweber, and T. Faber, “Dynamic Time Windows and Generalized Virtual Clock: Combined Closed-Loop/Open-Loop Congestion Control”, Proc. IEEE INFOCOM ‘92, May 1992 (Wisconsin, Blanca)

Murakami, G., “Non-Blocking Packet Switching with Shift-Register Rings”, Ph.D. Thesis, Technical Report UIUCDCS-R-91-1711, TTR91-1758, Department of Computer Science, UIUC, October 1991 (UIUC, Blanca)

Murakami, G., R. Campbell and M. Faiman, “Pulsar: Non-Blocking Packet Switching with Shift-Register Rings”, ACM SIGCOMM ‘90, Sept. 1990, In Computer Communications Review, Sept. 1990 (UIUC, Blanca)

Nahrstedt, K. and J. Smith, “The Integrated Media Approach to Networked Multimedia Systems”, January 1992 (Penn, Aurora)

Nahrstedt, K. and J. Smith, “An Integrated Multimedia Architecture for High-Speed Networks”, Proc. Multimedia ‘92 Conference, 1992 (Penn, Aurora)

Nahrstedt, K. and J. Smith, “Revision of Qos Guarantees at the Application/Network Interface”, Technical Report MS-CIS-93-34, 1993 (Penn, Aurora)

Nahrstedt, K. and J. Smith, “Experimental Study of Issues in End-to-End Qos”, ACM Transactions on Office Information Systems, August 1994 (Penn, Aurora)

Nahrstedt K. and J. Smith, “Qos Negotiation in a Robotics Environment”, in Proc. Workshop on Distributed Multimedia Applications and Quality of Service Verification, May-June 1994 (Penn, Aurora)

Nahrstedt, K. and J. Smith, “A Service Kernel for Multimedia Endstations”, in Proc., 2nd International Workshop on Advanced Teleservices and High-Speed Communication Architectures, Sept. 1994 (Penn, Aurora)

Nahrstedt, K. and J. Smith, “The Qos Broker”, IEEE Multimedia Magazine, Spring 1995 (Penn, Aurora)

Nahrstedt, K. and R. Steinmetz, “Resource Management in Networked Multimedia Systems”, IEEE Computer, May 1995 (Penn, Aurora)

Nahrstedt, K., “An Architecture for Provision of End-to-End QoS Guarantees”, PhD Thesis, University of Pennsylvania, CIS Dept., 1995 (Penn, Aurora)

Neumann, U., “Interactive Volume Rendering on a Multicomputer”, Proc. 1992 Symposium on Interactive 3-D Graphics, special issue of Computer Graphics, March 1992 (UNC, Vistanet)

Neumann, U., A. State, H. Chen, H. Fuchs, T. Cullip, Q. Fang, M. Lavoie, and J. Rhoades, “Interactive Multimodal Volume Visualization for a Distributed Radiation-Treatment Planning Simulator”, UNC-CS Technical Report TR-94-040, June 1994 (UNC, Vistanet)

Nilsson, A., L. Bottomley and A. Blatecky, “Traffic Measurements of a Working Wide Area Network”, ITC 13, June 1991 (NCSU, Vistanet)

Nilsson, A., Y. Jou and F. Lai, “The Upper Bounds for Delay and Cell Loss Probability of Bursty ATM Traffic in a Finite Capacity Polling System”, Proc. 24th Southeastern Symposium on System Theory and the 3rd Annual Symposium on Communications, Signal Processing, Expert Systems, and ASIC VLSI Design, March 1992 (NCSU, Vistanet)

Nilsson, A. and B. Lee, “Performance Analysis of Finite Buffer System: Instantaneous Single Input”, 2nd Annual Symposium on Communications, Signal, Processing Expert Systems and ASIC VLSI Design, March 1991 (NCSU, Vistanet)

Nilsson, A., F. Lai and H. Perros, “Performance Evaluation of a Bufferless N N Synchronous Clos ATM Switch with Priorities and Space Preemption”, ICC ‘91, June 1991 (NCSU, Vistanet)

Nilsson, A. and Z. Cui, “ATM Adaptation Layer Issues”, Proc. 24th Southeastern Symposium on System Theory and the 3rd Annual Symposium on Communications, Signal Processing, Expert Systems, and ASIC VLSI Design, March 1992 (NCSU, Vistanet)

Nilsson, A. and Z. Cui, “On the ATM Adaptation Layer”, 1992 International Conference on Communication Technology, September 1992 (NCSU, Vistanet)

Nilsson, A. and M. Huterer, “End-to-End Performance of ATM and Private Broadband Networks”, Proc. 24th Southeastern Symposium on System Theory and the 3rd Annual Symposium on Communications, Signal Processing, Expert Systems, ad ASIC VLSI Design, March 1992 (NCSU, Vistanet)

Nilsson, A. and M. Huterer, “Interconnected ATM and Private Broadband Networks: Delay Analysis”, 10th Nordic Telectraffic Seminar, August 1992 (NCSU, Vistanet)

Nilsson, A. and M. Huterer, “Performance of ATM and Private Broadband Networks”, 1st International Conference on Computer Communications and Networks, June 1992 (NCSU, Vistanet)

Nilsson, A., F. Lai and H. Perros, “A Queuing Model of a Bufferless Synchronous Clos ATM Switch with Head-Of-Line Priority and Push-Out”, Journal of High- Speed Networks, 1992 (NCSU, Vistanet)

Nilsson, A. and B. Lee, “Performance Analysis of Finite Buffer System: Instantaneous Single Input”, 2nd Annual Symposium on Communications, Signal Processing Expert Systems, and ASIC VLSI Design, March 1991 (NCSU, Vistanet)

Nilsson, A. and B. Lee, “A Performance Study of the XTP Error Control”, 4th IFIP Conf. on High Performance Networking, December 1992 (NCSU, Vistanet)

Nilsson, A., H. Perros, and D. Stevenson, “Tools and Methodologies for High Speed Network Design”, Integrated Broadband Communication Networks and Services, April 1993 (NCSU, Vistanet)

Nilsson, A., H. Perros and F. Lai, “An Approximate Analysis of a Bufferless N N Synchronous Clos ATM Switch”, ITC 13, June ‘91 (NCSU, Vistanet)

Nilsson, A. and H. Park, “Protocol for High-Speed MANs CRMA with Node Reservation Control for DQDB”, First International Conference on Computer Communications and Networks, June 1992 (NCSU, Vistanet)

Olsen, R. and L. Landweber, “NoW (No Waiting) Virtual Channel Establishment in ATM-like Networks”, Technical Report #1082, February 1992 (Wisconsin, Blanca)

Olsen, R. and L. Landweber, “Design and Implementation of a Fast Virtual Channel Establishment Method for ATM Networks”, Proc. IEEE INFOCOM ‘93 (Wisconsin, Blanca)

Olsen, R. and L. Landweber, “The Design and Implementation of Network Traffic Reporting (NTR): Providing QOS Guarantees in Packet Switched Networks”, Submitted for publication, Sept. 1994 (Wisconsin, Blanca)

Padkjaer, S., “Linear Algebra on the Touchstone Delta,” CRPC Seminar, California Institute of Technology, April 1992 (Caltech, Casa)

Parris, C., G. Ventre, and H. Zhang, “Graceful Adaptation of Guaranteed Performance Service Connections”, Proc. IEEE GLOBECOM ‘93, Nov. 1993 (Berkeley, Blanca)

Parris, C., S. Keshav and D. Ferrari, “A Framework for the study of Pricing in Integrated Networks”, Technical Report TR-92-016, International Computer Science Institute, Berkeley, CA, March 1992 (Berkeley, Blanca)

Parris, C., H. Zhang, and D. Ferrari, “A Mechanism for Dynamic Re-routing of Real-time Channels, Tech. Report TR-92-053, International Computer Science Institute, August 1992 (Berkeley, Blanca)

Parris, C. and D. Ferrari, “A Resource Based Pricing Policy for Real-Time Channels in a Packet-Switching Network”, Technical Report TR-92-018, International Computer Science Institute, Berkeley, CA, March 1992 (Berkeley, Blanca)

Parris, C., “The Dynamic Management of Guranteed-Performance Connections in Packet-Switched Integrated-Services Networks”, Ph.D. Dissertation, Univ. of CA at Berkeley, Sept. 1994 (Berkeley, Blanca)

Parris, C. and A. Banerjea, “An Investigation into Fault Recovery in Guaranteed Performance Service Connections”, Technical Report TR-93-054, International Computer Science Institute, Berkeley, CA, Oct. 1993 (Berkeley, Blanca)

Parris, C., C., H. Zhang, and D. Ferrrari, “A Dynamic Management Scheme for Real-Time Connections”, Proc. INFOCOM ‘94, June 1994 (Berkeley, Blanca)

Parris, C., H. Zhang, and D. Ferrari, “Dynamic Management of Guaranteed Performance Multimedia Connections”, ACM/Springer-Verlag Multimedia Systems, 1994 (Berkeley, Blanca)

Paxson, V., “Empirically-Derived Analytic Models of Wide-Area TCP Connections: Extended Report”, IEEE/ACM Transactions on Networking, 1994 (Berkeley, Blanca)

Paxson, V., “Growth Trends in Wide-Area TCP Connections”, IEEE Network, 1994 (Berkeley, Blanca)

Paxson, V. and S. Floyd, “Wide-Area Traffic: The Failure of Poisson Modeling”, Proc. ACM SIGCOMM ‘94, Sept. 1994 (Berkeley, Blanca)

Pekny, J. and D. Miller, “A Staged Primal-Dual Algorithm for Finding a Minimum Cost Perfect Two-Matching in an Undirected Graph”, July 1991 (CMU, Nectar)

Pekny, J., D. Miller, and G. Kudva, “An Algorithm for Resource Constrained Sequencing with Application to Production Scheduling under an Aggregate Deadline”, Computers and Chemical Engineering, 1992 (CMU, Nectar)

Periyannan, A., “Design of a Reactive Congestion Control Mechanism for Gateways in an ATM Environment”, M.S. Thesis, August 1992 (NCSU, Vistanet)

Perry, M., C. Sargor, and A. Nilsson, “Assignable Grade of Service Using Time Dependent Priorities: N Classes”, TriComm ‘93, April 1993 (NCSU, Vistanet)

Ransom, N., “The Vistanet Gigabit Network Testbed,” Journal of High-Speed Networks, Vol 1, No. 1, 1992 (BellSouth, Vistanet)

Ransom, N., “Vistanet Gigabit Testbed”, Workshop on Very High Speed Networks, March 1990 (Bellsouth, Vistanet)

Ransom, N. and D. Spears, “Applications of Public Gigabit Networks”, IEEE Network, March 1992 (Bellsouth, Vistanet)

Ransom, N. and K. Talmar, “ATM Traffic Measurement in the Vistanet Network”, Globecom ‘92 Proc. (Bellsouth, Vistanet)

Reed, D., C. Catlett et al, “Parallel I/O: Getting Ready for Prime Time”, IEEE Parallel and Distributed Technology, Summer 1995 (NCSA, Blanca)

Rehman, N. and J. Smith, “Using Detailed Traffic Statistics for More Effective Wideband Multiplexing”, Silicon Valley Networking Conference, April 1991 (Penn, Aurora)

Robe, T. and K. Walsh, “A SONET STS-3c User Network Interface IC”, Proc. IEEE 91 Custom Integrated Circuits Conference, May 1991 (Bellcore, Aurora)

Robertson, A., C.-C. Ma, C. Mechoso and M. Ghil, “Simulations of the Tropical- Pacific Climate With the UCLA Coupled GCM”, 17th Annual Climate Diagnostics Workshop, 1992 (UCLA, Casa)

Robertson, A., C. Mechoso and C-C. Ma, “The Seasonal Cycle in Surface Heat Fluxes Over the Tropical-Pacific in a Coupled GCM, Proceedings of the 18th Climate Diagnostics Workshop, 1994 (UCLA, Casa)

Rosenman, J., “A Gigabit Medical Application”, Workshop on Very High Speed Networks, March 1990 (UNC, Vistanet)

Rosenman, J., E. Chaney, T. Cullip, J. Symon, H. Fuchs, and D. Stevenson, “Vistanet: Interactive Real Time Calculation and Display of 3D Radiation Dose: An Application of Gigabit Networking”, Int. J. Radiat. Oncol. Biol. Phys. (UNC, Vistanet)

Russo, C., D. Moore, C. Traw and J. Smith, “Early Experiences with ATM over the HP HDMP-1000”, ACM SIGCOMM Computer Communications Review, 1995 (Penn, Aurora)

St. John, W. and D. DuBois, “Wide-area Gigabit Networking: Los Alamos HIPPI-Sonet Gateway”, Supercomputing `95, December 1995 (LANL, Casa)

Sane, A. and R. Campbell, “Subclassing state machines”, Technical Report UIUCDCS-R-92-1773, October 1992 (UIUC, Blanca)

Sane, A., K. MacGregor and R. Campbell, “Distributed Virtual Memory Consistency Protocols: Design and Performance”, Proc. of the Second IEEE Workshop on Experimental Distributed Systems, 1990 (UIUC, Blanca)

Scarbnick, C., M. Chang, M. Schultz and A. Sherman, “A Parallel Software Package for Solving Linear Systems”, Proc. 4th IEEE Symposium on the Frontiers of Massively Parallel Computation, 1992 (SDSC, Casa)

Schmidt, A., “Network Management of Asynchronous Transfer Mode Switching Systems”, Master’s Thesis, 1991 (UIUC, Blanca)

Schmidt, A. and R. Campbell, “Internet Protocol Traffic Analysis with Applications for ATM Switch Design”, Technical Report UIUCDCS-R-92-1735, May 1992, Computer Communication Review, April 1993 (UIUC, Blanca)

Shaffer, J. and J. Smith, “Experimental Evaluation of Distributed Shared Memory on a High-Speed WAN, IEEE Journal Selected Area of Communications Special Issue on Distributed Systems using Gigabit, October 1994 (Penn, Aurora)

Shaffer, J., “Experimental Evaluation of Distributed Shared Memory on a High-Speed WAN”, PhD Thesis, University of Pennsylvania CIS Dept., 1995 (Penn, Aurora)

Siegell, B. and P. Steenkiste, “Controlling Application Grain Size on a Network of Workstations”, Supercomputing ‘95, December 1995 (CMU, Nectar)

Siegall, B., “Automatic Generation of Parallel Programs with Dynamic Load Balancing for a Network of Workstations, PhD thesis, Dept. of Computer Science and Electrical Engineering, Carnegie Mellon University, 1995 (CMU, Nectar)

Singh, R., S. Tell and D. Becker, and S. Bharrat, “Vistanet Network Interface Unit: Future Communications Research”, UNC-CH DCS Technical Report TR91-018 (UNC, Vistanet)

Singh, R., S. Tell, S. Bharrat, D. Becker, and V. Chi, “A Programmable HIPPI Interface for a Graphics Supercomputer”, Supercomputing 93, Nov. 1993 (UNC, Vistanet)

Sivakumar, S., “Hypercube Message Library Simulator Over a Network of Machines”, Master’s Thesis (UIUC, Blanca)

Sivakumar, S., “HXA Software Design: An Overview”, Technical Report, February 1993 (UIUC, Blanca)

Smarr, L. and C. Catlett, “Life After Internet”, Harvard Info. Technology Quarterly, 1991 (NCSA, Blanca)

Smarr, L. and C. Catlett, “Metacomputing”, Communications of the ACM, June 1992 (NCSA, Blanca)

Smith, R., R., J. Dukowisz, and R. Malone, “Parallel Ocean General Circulation Modeling”, Technical Report LA-UR-92-200, 1992 (LANL, Casa)

Smith, D. and H. Chao, “Buffer Sizing at a Host in an ATM Network”, IEEE Infocom ‘92, May 1992. (Bellcore, Nectar)

Smith, J., “Distributed Systems of the Future?”, Workshop on Architectures for Very-High-Speed Networks, January 1990 (Penn, Aurora)

Smith, J., “Protection in Distributed Shared Memories”, 4th Int’l. Workshop on Distributed Environments and Networks, October 1991 (Penn, Aurora)

Smith, J., “Anticipation in Very High Speed Networks”, Distributed Systems Laboratory Technical Report and Working Paper, 1991 (Penn, Aurora)

Smith, “Remote Backups for Very High Speed Networks”, DSL Tech Report and Working Paper, 1991 (Penn, Aurora)

Smith, J. and D. Farber, “Traffic Characteristics of a Distributed Memory System”, Computer Networks and ISDN Systems, Vol 22, No. 2, September 1991 (Penn, Aurora)

Smith, J., B. Traw and D. Farber, “Cryptographic Support for a Gigabit Network”, Proc. INET ‘92, June 1992 (Penn, Aurora)

Smith, J., C. Brendan Traw, “Operating Systems Support for End-to-End Gbps Networking”, IEEE Network, July 1993 (Penn, Aurora)

Smith, J., “PDIS and The Information Highway”, in Proc. Parallel and Distributed Information Systems, Sept. 1994 (Penn, Aurora)

State, A., J. Rosenman, H. Fuchs, T. Cullip and J. Symon, “Vistanet” Radiation Therapy Treatment Planning Through Rapid Dose Calculation and Interactive 3D Volume Visualization”, Proc. Visualization in Biomedical Computing 1994 (UNC, Vistanet)

State, A., S. Balu, and H. Fuchs, “Bunker View: Limited-range head-motion- parallax visualization for complex data sets”, Proc. Visualization in Biomedical Computing 1994, Oct. 1994 (UNC, Vistanet)

Steenkiste, P., “A Symmetrical Communication Interface for Distributed-Memory Computers”, IEEE Proc. Eleventh Conference on Distributed Computing Systems, April 1991 (CMU, Nectar)

Steenkiste, P., H.T. Kung, S. Schlick, B. Zill, J. Hughes, B. Kowalski, and J Mullaney, “A Host Interface Architecture for High-Speed Networks”, Proc. of 4th IFIP Conf., December 1992 (CMU, Nectar)

Steenkiste, P., “Analyzing Communication Latency using the Nectar Communication Processor”, Proc. of the SIGCOMM ‘92, August 1992 (CMU, Nectar)

Steinmetz, R. and K. Nahrstedt, “Multimedia: Computing, Communications and Applications”, Prentice-Hall, Englewood Cliffs NJ, 1995 (Penn, Aurora)

Stevenson, D., “Communications Research in the Vistanet Gigabit Network Testbed”, Cray Users Group, Oct 90 (MCNC, Vistanet)

Stevenson, D., “MCNC High Speed Networking Research”, Workshop on Very High Speed Networks, March 1990 (MCNC, Vistanet)

Stevenson, D., “Supercomputer Communications as an Application for Broadband Networks”, GLOBECOM ‘91, December 1991 (MCNC, Vistanet)

Stevenson, D. and J. Rosenman, “Vistanet Gigabit Testbed”, IEEE Journal of Selected Applications for Communications, December 1992 (MCNC, Vistanet)

Stevenson, D., “Electropolitical Correctness and High-Speed Networking, or, Why ATM is like a Nose”, TriCom ‘93, April 1993 (MCNC, Vistanet)

Stevenson, D., “Supercomputer Communications over ATM: Lessons and Future Directions”, Maryland Workshop on Very High Speed Networks, March 1993 (MCNC, Vistanet)

Stevenson, D., “Broadband Public Network Technologies Meet the Internet”, North Carolina GIS Conference, April 1993 (MCNC, Vistanet)

Stevenson, D., “Vistanet Results, Lessons, New Directions, Interop 94, May 1994 (MCNC, Vistanet)

Stevenson, D., “Technical Challenges of Gigabit Networking”, IEEE LEOS, May 1994 (MCNC, Vistanet)

Stolorz, P., et al, “Fast Spatiotemporal Dataminimg of Large Geophysical Datasets”, 1st Intnl. Conf. on Knowledge Discovery and Datamining, Montreal, Canada 1995 (JPL, Casa)

Stolorz, P., C. Dean, R. Crippen and R. Blom, “Photographing Earthquakes from Space”, CSCC Annual Report, Caltech CACR-116, December 1995 (JPL, Casa)

Subrahmanyam, S., G. Kudva. M. Bassett and J. Pekny, “Application of Distributed Computing to Batch Plant Design and Scheduling”, American Institute of Chemical Engineering Journal, 1995 (CMU, Nectar)

Tam, M., J. Smith and D. Farber, “A Taxonomy-Based Comparison of Several Distributed Shared Memory Systems”, ACM Operating Systems Review, July 1990 (Penn, Aurora)

Tam, M., “CAPNET – Using a Gigabit Network as a High Speed Backplane”, Ph.D. Thesis, CIS Department, University of Pennsylvania, 1994 (Penn, Aurora)

Tam, M. and D. Farber, “CapNet – Shared Memory Distributed Computing on Wide Area High Speed Networks”, IEEE Journal Selected Areas of Communications Special Issue on Distributed Systems using Gigabit Networks, October 1994 (Penn, Aurora)

Tamashunas, B., “Supporting Service Classes in ATM Networks”, MS Thesis, May 1992 (MIT, Aurora)

Tan, S., “An Architecture for Call Processing”, MS Thesis, April 1994 (UIUC, Blanca)

Tan, S. and R. Campbell, “Efficient Signalling Algorithms for Broadband-ISDN”, Technical Report, UIUC, Department of Computer Science, 1994 (UIUC, Blanca)

Tan, S. and R. Campbell, “Efficient Signalling Algorithms for ATM Networks”, IFIP Third Workshop on Performance Modelling and Evaluation of ATM Networks, Bradford, UK, July 1995 (UIUC, Blanca)

Tell, S., R. Singh and D. Becker, “Design Management, Modeling, and Simulation of PLD-Based Systems”, 1st open Verilog International User’s Group Meeting, March 1992 (UNC, Vistanet)

Tennenhouse, D., J. Adam, D. Carver, H. Houh, M. Ismert, C. Lindblad, W. Stasior, D. Wetherall, D. Bacher, and T. Chang, “A Software-Oriented Approach to the Design of Media Processing Environments”, Proc. of the International Conference on Multimedia Computing and Systems, May 1994 (MIT, Aurora)

Terstriep, J., C. Catlett, M. Normal, and P. Moran, “Networking and Distributed Computing”, SIGGRAPH, 1992 (NCSA, Blanca)

Terstriep, J. and D. Weber, “NCSA DTM Programming Manual”, Technical Report (NCSA, Blanca)

Terstriep, J., D. Weber, and T. Kwan, “DTM, A Message Passing Library for Distributed Supercomputing”, Supercomputing ‘93 (NCSA, Blanca)

Todorova, P. and D. Verma, “Delay Constraints and Admission Control in ATM Networks”, Conf. Record, GlobeComm ‘90, Dec. 1990 (Berkeley, Blanca)

Traw, C. and J. Smith, “A High-Performance Host Interface for ATM Networks”, SIGCOMM 91, September 1991 (Penn, Aurora)

Traw, C. and J. Smith, “Implementation and Performance of an ATM Host Interface for Workstations”, IEEE Workshop on High Performance Communications Subsystems, February 1992 (Penn, Aurora)

Traw, C. and J. Smith, “Hardware/Software Organization of a High-Performance ATM Host Interface”, IEEE Journal on Selected Areas in Communications, 1993 (Penn, Aurora)

Traw, C., “A Host Interface Architecture and Implementation for ATM Networks”, M.S. Thesis (Penn, Aurora)

Traw, C., “Applying Architectural Parallelism in High Performance Network Subsystems”, PhD Thesis, University of Pennsylvania, CIS Dept., 1995 (Penn, Aurora)

Traw, C. and J. Smith, “Striping within the Network Subsystem”, IEEE Network, August 1995 (Penn, Aurora)

Udani, S., “Architectural Considerations in the Design of Video Capture Hardware”, MS Thesis, April 1992 (Penn, Aurora)

Udani, S., “Experimental Evaluation of a Video Capture Board for Networked Workstations”, Technical Report MS-CIS-93-31, 1993 (Penn, Aurora)

Umemura, K. and A. Okazaki, “Real-Time Transmission and Software Decompression of Digital Video in a Workstation”, Technical Report TR-91-004, International Computer Science Institute, Berkeley, CA, Jan. 1991 (Berkeley, Blanca)

Verma, D. and H. Zhang, “Design Documents for RTIP/RTTP”, May 1991 (Berkeley, Blanca)

Verma, D., H. Zhang, and D. Ferrari, “Design and Implementation of the Real-Time Internet Protocol”, Proc. IEEE Workshop, February 1992 (Berkeley, Blanca)

Verma, D. and D. Ferrari, “Evaluation of Overflow Probabilities in Resource Management”, Technical Report, TR-91-051, International Computer Science Institute, Berkeley, CA, October 1991 (Berkeley, Blanca)

Verma, D., “Guaranteed Performance Communication in High Speed Networks”, Ph.D. Thesis, December 1991; also Technical Report UCB/CSD 91/663, Univ. of CA, Berkeley, December 1991 (Berkeley, Blanca)

Verma, D. and D. Ferrari, “Variation of Traffic Parameters in ATM Networks”, Proc. ICC” 92, June 1992 (Berkeley, Blanca)

Verma, D., H. Zhang and D. Ferrari, “Delay Jitter Control for Real-Time Communication in a Packet Switching Network”, Technical Report TR-91-007, International Computer Science Institute, Berkeley, CA, Jan. 1991; also in Proc. TriComm ‘91, April 1991 (Berkeley, Blanca)

Widyono, R., “The Design and Evaluation of Routing Algorithm for Real-Time Channels”, M.S. Report, Univ. of CA, Berkeley, May 1994 (Berkeley, Blanca)

Williams, R., J. Flower, T. Metzger, and C. Shenton, “HIPPI and Graphics for the Intel Delta,” First Intel Delta Applications Workshop, February 1992 (Caltech, Casa)

Winkelstein, D. and D. Stevenson, “HIPPI Link Data Analysis System: Test Equipment for High Speed Network Analysis”, TriComm ‘91, April 1991 (MCNC, Vistanet)

Winkelstein, D. and D. Stevenson, “Supercomputer Communications as an Application for ATM Local Area Networks”, TriComm ‘92, February 1992 (MCNC, Vistanet)

Wolfinger, B. and M. Moran, “A Continuous Media Data Transport Service and Protocol for Real-Time Communication in High Speed Networks”, Proc. 2nd Int’l. Workshop on Network and Operating System Support for Digital Audio and Video, November 1991 (Berkely, Blanca)

Wu, Y-S, S. Cuccaro, P. Hipes, and A. Kuppermann, “Quantum Chemical Reaction Dynamics on a Highly Parallel Supercomputer,” Theor. Chim. Acta, vol. 79, 225, 1991 (Caltech, Casa)

Wu, Y-S and A. Kuppermann, “Prediction of the Effect of the Geometric Phase on Product Rotational State Distributions and Integral Cross Sections,” Chem. Phys. Lett., vol. 201, 178, 1993 (Caltech, Casa)

Wu, Y-S, A. Kuppermann, and B. Lepetit, “Theoretical Calculation of Experimentally Observable Consequences of the Geometric Phase on Cemical Reaction Cross-sections,” Chem. Phys. Lett., vol. 186, 319, 1991 (Caltech, Casa)

Wu, Y-S and A. Kuppermann, “Importance of the Geometric Phase Effect for the H + $D_2$ -> HD + D Reaction”, Chem. Phys. Letters No. 235, 1995 (Caltech, Casa)

Young Jr., K., “Gigabit Networking: Using the SONET/ATM Public Network for Gigabit Applications”, Bellcore Digest of Technical Information, August, 1992 (Bellcore, Nectar)

Young, Jr., K., C. Johnston, D. Smith, J. Mann, J. DesMarais, M. Iqbal, K. Young, K. Walsh, and W. Holden, “A HIPPI/ATM/SONET Network Interface for the Nectar Gigabit Testbed”, LEOS Summer Topical Meeting on Gigabit Networks, July 1993 (Bellcore, Nectar)

Young, Jr., K., N. Cheung, “Recent Advances in Gigabit Networking”, to be presented as invited paper at the LEOS Annual Meeting, November 1993 (Bellcore, Nectar)

Zhang, H. and S. Keshav, “Comparison of Rate-Based Service Disciplines”, Proc. SIGCOMM ‘91, Sept. 1991 (Berkeley, Blanca)

Zhang, H. and D. Ferrari, “Rate-Controlled Static-Priority Queueing”, Technical Report TR-92-003, International Computer Science Institute, Berkeley, CA, January 1992; also in Proc. INFOCOM’ 93, March-April 1993 (Berkeley, Blanca)

Zhang, H. and T. Fisher, “Preliminary Measurements of the Real-Time Internet Protocol”, Proc. 3rd Int’l. Workshop on Network and Operating System Support for Digital Audio and Video, November 1992 (Berkeley, Blanca)

Zhang, H., “Service Disciplines For Integrated Services Packet-Switching Networks”, Ph.D. dissertation, Univ. of CA at Berkeley, Nov. 1993 (Berkeley, Blanca)

Zhang, H. and D. Ferrari, “Providing Deterministic Guarantees For Bursty Traffic”, Proc. Workshop On the Role of Real-Time in Multimedia/Interactive Computing Systems, IEEE Real-Time Systems Symposium ‘91, Nov. 1993 (Berkeley, Blanca)

Zhang, H. and D. Ferrari, “Improving Utilization for Deterministic Service in Multimedia Communication”, Proc. 1994 IEEE International Conference on Multimedia Computing and Systems, May 1994 (Berkeley, Blanca)