

The AURORA Gigabit Testbed

David D. Clark, Bruce S. Davie,
David J. Farber, Inder S. Gopal
Bharath K. Kadaba, W. David Sincoskie
Jonathan M. Smith, David L. Tennenhouse*

Abstract

AURORA is one of five U.S. networking testbeds charged with exploring applications of, and technologies necessary for, networks operating at gigabit per second or higher bandwidths. The emphasis of the AURORA testbed, distinct from the other four testbeds, BLANCA, CASA, NECTAR and VISTANET, is research into the supporting technologies for gigabit networking.

Like the other testbeds, AURORA itself is an experiment in collaboration, where government initiative (in the form of the Corporation for National Research Initiatives, which is funded by DARPA and the National Science Foundation) has spurred interaction among pre-existing centers of excellence in industry, academia, and government.

AURORA has been charged with research into networking technologies that will underpin future high-speed networks. This paper provides an overview of the goals and methodologies employed in AURORA, and points to some preliminary results from our first year of research, ranging from analytic results to experimental prototype hardware. This paper enunciates our targets, which include new software architectures, network abstractions, and hardware technologies, as well as applications for our work.

1 Introduction

AURORA is an experimental wide area network testbed whose main objective is the exploration and evaluation of technologies that may be appropriate for use in a Phase 3 National Research and Education Network (NREN) operating near or at gigabit per second bandwidths [14]. AURORA will also address the issues associated with the use of such networks, such as communications architecture and application service models. The principal research participants in AURORA are Bellcore, IBM, MIT, and the University of Pennsylvania (Penn). Collaborating telecommunications carriers are Bell Atlantic, MCI, and Nynex. These carriers are investigating the provision and operation of experimental facilities for site interconnection and cooperating in the research.

The research being carried out in AURORA may be divided into three categories:

- Exploration of Network Technology Alternatives
- Investigation of Distributed System/Application Interface Paradigms
- Experimentation with Gigabit Network Applications

The types of work being undertaken in each area are outlined below.

*Authors' Affiliations:

Dr. Clark and Prof. Tennenhouse are with MIT's Laboratory for Computer Science.
Prof. Farber and Prof. Smith are with U. Penn's Distributed Systems Laboratory.
Dr. Davie and Dr. Sincoskie are with Bell Communications Research, Inc.
Dr. Gopal and Dr. Kadaba are with the IBM Corporation.

1.1 Network Technology Alternatives

Several approaches have been proposed to achieve the next generation of network, based on different information transfer paradigms. AURORA will explore two significant options for network architecture and the interworking between them. The Asynchronous Transfer Mode (ATM) based on the transfer of small, fixed-size data cells, is the broadband methodology currently favored within the telecommunications industry [20]. Packet Transfer Mode (PTM) is the term used in this document to describe packet transport methodologies that permit a mixture of different packet sizes within the network¹. It is the method preferred by segments of the data communications industry. Each approach has its advantages, and they will coexist in the national network of tomorrow.

This project will enhance and deploy experimental switch prototypes tailored to each of the transfer modes — Bellcore’s ATM-based Sunshine switch and IBM’s PTM-based plaNET switch. Since switches are only one aspect of network technology, the project will prototype additional components, both hardware and software, to support the associated transmission, signaling, interface, operations and management functions. Transfer mode independent issues, including higher level transport protocols, are being explored, and the architecture of host interfaces is under investigation. The interworking issues between the PTM and ATM environments will also be studied.

A result of this experiment will be hands-on experience with the use of these two transfer modes, a characterization of the domain of utility for each of them, and an increased understanding of the problems of internetworking at gigabit speeds.

1.2 Distributed System/Application Interface Paradigms

An important part of network architecture is packaging the network service and presenting it to the application builder in a way that simplifies the application design without restricting or unduly complicating the operation of the network. This packaging, often called “abstraction”, is a basic aspect of computer science. The most popular abstractions for today’s networks are the reliable byte stream and the remote procedure call (RPC). Both of these seem to provide convenient and natural interfaces to applications, but both are limited in the functions they can deliver. The byte stream, because it insists on reliable delivery, cannot also control latency of delivery. Remote procedure calls, because they represent serialized rather than parallel communication across a network, degrade directly with increasing network latency. For high-speed networks, RPC makes poor use of the available bandwidth.

An alternative network abstraction is one in which the network is modeled as shared virtual memory. That is, the application makes use of the network by reading and writing parts of its address space which are replicated at the communicating sites using the network. This abstraction is a very basic one to computer science, but potentially suffers from the same latency problems as RPC. However, techniques from existing virtual memory management implementations and file systems, namely read-ahead and caching, can be adapted to the network context.

This approach stresses network transparency and assumes that the software supporting the application interface can deduce the proper action (e.g., read-ahead) from the past behavior of the application. In contrast, alternative approaches that are less transparent and require some explicit characterization of application service requirements are also being explored. This approach might serve a broader set of applications than an implicit scheme such as shared virtual memory. However,

¹It should be noted that while ATM transports data through the network in fixed-sized packets, applications may still communicate using variable length packets. Conversion between ATM cells and variable length packets is handled in a *host interface*, described in Section 4.2

previous experiments with explicit resource management by applications have not always proven successful, either in networks or in the management of virtual memory paging. The opportunity to explore both of these approaches in the context of AURORA may reveal basic issues in the packaging of the network for the application.

1.3 Gigabit Applications

The exchange of visual images represents an increasingly significant aspect of network traffic. The growing bandwidth requirement is driven both by increased display resolutions and by increased emphasis on visually-oriented computing and communication. The result is likely to be a network load dominated by transmission of visual still images, video sequences, and animated scientific visualizations.

As part of the project we are exploring the use of the testbed for video conferencing and multi-media teleconferencing applications, and for the presentation of multi-media information, including high-resolution images — all targeted at understanding their use in the business, scientific and residential environments of the future. As an adjunct to the project, we intend to encourage the use of this testbed by selected members of the research community at the participating sites. Although many of the applications may seem similarly focused upon the presentation of information to humans, both the aggregate traffic mix and collective service requirements will be far from homogeneous. For example, a real-time video conferencing application may generate high-bandwidth, potentially bursty traffic, demand little variation in delay, and tolerate a certain level of error. In contrast, a medical imaging application may generate less bursty traffic, tolerate significant variation in delay, and require completely error-free transmission. The applications identified for exploration in the AURORA project manifest the diversity of traffic models that is needed for thorough testing and understanding of tomorrow's network technologies.

1.4 Research Methodology

The research methodology for AURORA is experimental proof-of-concept for key ideas by actual prototyping and deployment of a long-haul experimental network. AURORA is not intended primarily as an applications testbed. Rather, options for providing the necessary network services are being explored and applications will be used as a vehicle to understand these technologies.

The deployment of the long-haul experimental testbed is crucial to the realization of this research goal. Because the project is fundamentally collaborative, and because the participants are focusing on distinct components of the overall solution, it is only by the assembly of these components into an integrated, functioning system that both the overall architecture and the individual components can be properly tested. Some participants are focusing on switching technology; others are addressing host interfaces and terminal devices; still others are concentrating the software aspects of gigabit networks. Proper testing of switches and measurement of their performance requires the realistic traffic generated by the terminal components. Similarly, evaluation of terminal devices requires their interconnection by a switching fabric with appropriate bandwidth, delay, and jitter characteristics. Thus, the testbed will both enable and motivate a close inter-relationship among these distinct activities.

The gigabit network will link four sites:

- Bellcore's Morristown Research and Engineering Laboratory in Morristown, NJ
- The IBM T.J. Watson Research Center in Hawthorne, NY
- MIT's Laboratory for Computer Science in Cambridge, MA

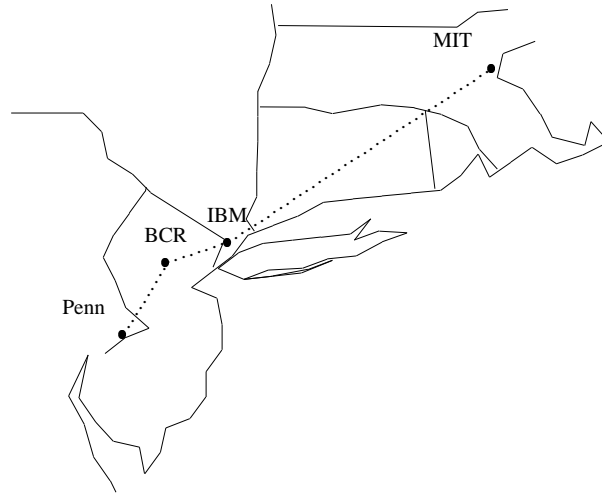


Figure 1: AURORA testbed geography

- University of Pennsylvania's Distributed Systems Laboratory in Philadelphia, PA

The geographical distribution of the testbed, illustrated in figure 1, not only adds significantly to the experimental reality of the planned research (realistic delay, jitter, error rates), but it will also afford experience regarding the performance and maintenance of such a network that will be valuable to operating companies.

1.5 Overview of this Document

The main purpose of this paper is to enunciate the research plans of the AURORA project. Thus, not all of the ongoing work is reported here, although we provide extensive citations. The remainder of this article comprises sections on:

- Network Infrastructure
- Backbone Network
- Local attachment
- Transport and higher layers
- Distributed systems
- Gigabit Applications
- Network control

2 Network Infrastructure

The network infrastructure is composed of the transmission facilities that interconnect the various sites. The facilities will be based on SONET [20], which is emerging as the dominant standard for

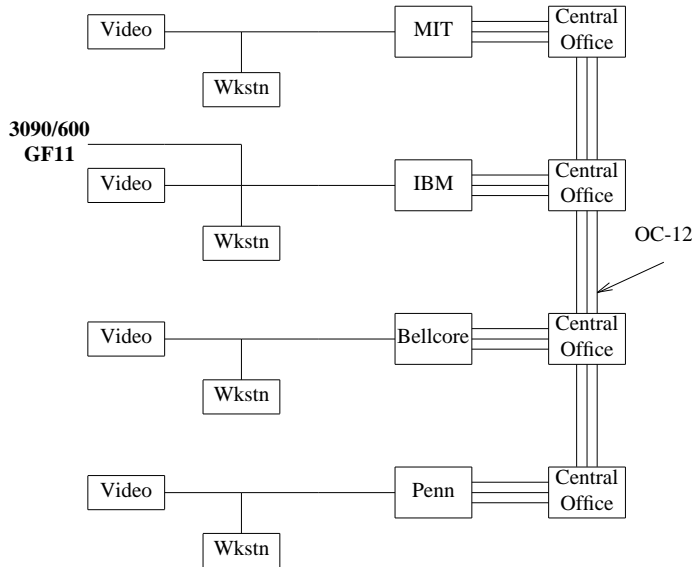


Figure 2: AURORA testbed topology

point-to-point long distance communication over fiber optic transmission links. While the switching nodes view the facilities as comprised of point-to-point links, the facilities themselves are more complex and capable of rearrangement into different patterns of connectivity by means of cross-connect switches and add-drop multiplexers within the infrastructure. In fact, the infrastructure may be viewed as a large piece of experimental apparatus that can be tuned and refined in response to changing hypotheses regarding applications and their supporting interfaces. Accordingly, the AURORA project will provide an experimental testbed for the exploration of issues related to transmission equipment, to multiplexers and cross-connects, to the management aspects of the SONET standard, and other issues related to network infrastructure.

The planned topology of the AURORA testbed is illustrated in Figure 2. Each of the four sites is connected to a central office through three OC-12 (622 Mbps) links. The central offices are themselves interconnected in a linear fashion. The links between the central offices also comprise three OC-12 links. The central offices have the ability to cross-connect the various OC-12 links independently and consequently, with this physical topology, a large number of logical topologies can be configured.

The initial use of the facilities will be to provide two separate networks, one based on plaNET and the other based on Sunshine. This will enable the two technologies to be tested and debugged before introducing the additional difficulties of interworking between them. It has been shown that with the available facilities it is possible to configure two separate networks, each of which connects all four sites, the available bandwidth between any two sites being 622 Mbps. When it becomes possible to interwork between the two network technologies, a single network with richer connectivity can be configured. It can readily be shown that the most highly connected single network topology that can be realized by the facilities is just one link less than a fully connected mesh.

2.1 Transmission Interfaces

In order to attach the switching equipment to the carrier provided facilities, work will be done at Bellcore and IBM to prototype SONET-compatible transmission link interfaces. These interfaces

will support the transfer of information within the SONET payload.

Bellcore's research effort includes two experimental custom SONET devices, a 155 Mbps STS-3c framer and a 622 Mbps STS-12 multiplexer. Both devices can function as either a transmitter or receiver. The STS-3c framer generates the SONET framing overhead and embeds user supplied data within the SONET payload. This device contains a byte-wide interface and generates the control signals which handshake with user circuitry. It performs all the pointer manipulations required to identify the synchronous payload envelope contained within a SONET frame. The framer supplies as its output either a serial stream or a byte-wide interface containing the formatted SONET signal. The STS-12 multiplexer interfaces to 4 STS-3c framers and byte interleaves these signals producing an STS-12 format. The combination of these two devices provides access to an STS-12 link through byte-parallel interfaces to four STS-3c channels. Details of the transmission and cross-connect arrangements will be finalized in conjunction with the participating telecommunications carriers. SONET STS-3c framers have been prototyped and were fabricated successfully last year.

The IBM and Bellcore interfaces will both use the SONET chip-sets described above. The interface for Bellcore's experimental switch prototype will enable ATM cells to be mapped into the SONET payload. The chip-set provides some additional control signals which facilitate this mapping. The plaNET interface developed at IBM will permit the mapping of variable sized packets into the SONET payload. The mapping and the corresponding reconstruction of packets will be performed by Programmable Gate Array devices capable of operating at 622 Mbps (the SONET STS-12 speed).

3 Backbone Network

The backbone network consists of the switching facilities and the associated transmission interfaces. The issues of switch structure, packet formats, link scheduling, routing, etc. are important research areas that will be addressed in the construction of the AURORA backbone network. As mentioned earlier, there will be two backbone networking technologies deployed in AURORA— Sunshine and plaNET.

Sunshine [18] is an experimental switch being prototyped at Bellcore. It will use the Asynchronous Transfer Mode (ATM), which has been identified within the telecommunications industry as the preferred approach for the next generation of common carrier infrastructure, known as the Broadband Integrated Services Digital Network (BISDN). Since standardization of the ATM architecture is now ongoing, practical experimentation with prototypes is an important activity.

Sunshine is a synchronous, self-routing packet switch architecture based on non-blocking Batchier/banyan networks. The ability to implement large networks within custom CMOS VLSI devices along with their simplified control and non-blocking properties makes Batchier/banyan networks extremely attractive for high speed ATM applications. Sunshine's advanced queuing strategies make it extremely robust over a wide range of traffic profiles and link utilizations.

The plaNET network being developed at IBM will serve as AURORA's PTM test-bed. PlaNET (formerly PARIS [9]) is a high-speed wide area networking system that makes use of a simplified network architecture in order to achieve the low packet delay and high nodal throughput necessary for the transport of high-speed real-time traffic. PlaNET includes several novel design features that support high-speed network operation. The design of plaNET has been targeted toward supporting heterogeneous traffic types within the network. Thus, plaNET can support packets of different sizes, priorities, routing methods, etc. Among the different packet structures supported by the plaNET hardware are the source-routed PARIS [9] packets and ATM cells. While, for the purposes of the AURORA trial plaNET will be used as a PTM system, the switching hardware can, if desired,

provide the appearance of a pure ATM switch.

It is likely that ATM and PTM will coexist, so interworking between them will be a requirement for successful networking. AURORA thus provides two opportunities: first, to investigate the operating regions of each approach and, second, to attempt to interwork between them. In the following sections, we will examine the components of these two switching systems in more detail.

3.1 The Sunshine Switch

The Sunshine Switch is a self-routing ATM packet switch, conceived at Bellcore, with output buffering and a shared recirculating queue. This combination of buffering schemes yields a switch that is robust under a wide range of incident traffic. The architecture of the switch and its experimental prototype implementation using custom CMOS chips were described in [18, 22]. More detailed descriptions of the chips have also been published [21, 23]. The current prototyping effort will produce 32×32 port switches, each port operating at the STS-3c rate of 155 Mbps. To deliver higher rates, a mechanism known as *trunk grouping* (described in Section 3.1.1) is used, in which groups of ports may be aggregated to form higher bandwidth pipes. This will allow traffic to be switched at a rate of 622 Mbps.

The 32 port Sunshine switch (excluding port controllers) is being implemented on a single circuit board. It includes twenty experimental custom CMOS VLSI chips (five different chip designs). At the time of writing, two of the five chips have been fabricated and tested at full speed, and the remainder are about to be fabricated. The physical design of this board presents some major challenges; considerations such as simultaneous switching noise that causes power-supply fluctuations and crosstalk are very significant.

3.1.1 Switch Port Controllers

A major ATM component lies in the per-line controllers that are located at the interface between the transmission lines and the switch ports. On the input side of the switch, the port controller must process ATM cells at the incoming line rate. Based on information contained within the cell header and local state information, the controller must generate and prepend a self-routing string that identifies the appropriate switch output. On the output side of the switch, each port controller must control access to the output queues and format cells for transmission over the outgoing link(s). On either the input or output side of the switch, the controller must perform any hop-by-hop header mapping, accounting, and related functions that are required by the ATM-level protocol. Among the functions of the port controller are virtual circuit/datagram identifier translations, header verifications and labeling, adaptation layer processing, buffering and priority queueing, and the generation of switch control headers.

Each port of the Sunshine switch operates at the STS-3c rate of 155 Mbps. To provide switching at higher speeds, a mechanism known as *trunk grouping* is used. This enables a group of ports to be treated as a single logical unit with a bandwidth of some multiple of 155 Mbps. In the current prototype effort, trunk groups of size four, to carry traffic at 622 Mbps (the STS-12 rate), are supported. Trunk grouping is achieved by allowing the four input port controllers of a group to access a shared table, so that all members of a group can use the same information for routing, accounting, etc. Trunk grouping is implemented at the output ports by causing four output ports of the switch fabric to feed a single output port controller which in turn places cells into an STS-12 stream.

The input port controller requires a high-speed mechanism to identify and manipulate the various-sized information fields which are contained within each ATM header. A major compo-

ment of the port controller, responsible for these manipulations, is a programmable cell processor, described below.

3.1.2 Cell Processing Engine

The cell processing engine being implemented at Bellcore is a custom RISC processor for ATM cell operations. This experimental CMOS VLSI chip has several FIFO's for ATM cell I/O, and the processing unit has an instruction set tailored for header manipulation, including instructions to manipulate arbitrarily aligned bit fields in ATM or adaptation layer headers. The data path in the processor is sufficiently wide to handle entire ATM cells in a single operation. While the chip is especially tailored for handling switch input port functions, it can also be used for cell queues, multiplexors, or other high speed cell operations. It has also formed the basis of another cell processing chip, described in Section 4.2.4.

3.2 The plaNET Project

The plaNET project at IBM covers the architecture, design and prototype of a high speed packet switching network for integrated voice, video and data communications. The system includes both wide area and local area components operating as a single homogeneous network at aggregate speeds of several gigabits/sec. Various aspects of the system are described in different sections of the paper and appropriate references cited.

The plaNET project is the successor of the PARIS project [9, 5], which was successfully prototyped several years ago and provided 100 Mbps links. (The local area component, based on a 100 Mbps predecessor known as METARING [11], is called ORBIT). The plaNET switch under development will support SONET OC-12 or gigabit/second dark fiber links and will provide a nodal throughput approximately six times faster than the original PARIS switch. The ORBIT local access portion of the system will operate at a serial speed of one gigabit/second. In addition to the hardware enhancement plaNET will support significantly more functions than PARIS. For example in PARIS, intermediate node routing is performed exclusively through a source routing scheme called Automatic Network Routing. In plaNET, several new routing functions will be supported, including extensive support for multicast and for ATM. IP routing and LAN bridging functions are also being designed. The control and distributed algorithms used in the system are being optimized for the mix of traffic expected in gigabit networks.

3.2.1 The plaNET switch

The switching mechanism is based on a shared broadcast medium with an aggregate capacity of 6 Gbps. The shared medium is implemented using a 64-bit wide internal broadcast ring operating at approximately 100 million transfers per second. Access to the shared medium is arbitrated using an approximate First-Come-First-Served policy that is proven to provide minimal input delay. Numerous fault isolation and detection capabilities are supported.

The shared ring is connected to the various transmission interfaces by means of link adaptors. The switching function is implemented in a distributed fashion. Each adaptor receives every packet broadcast on the shared medium. Then, by means of the routing information in each packet, it makes an independent decision whether or not to place the packet in its local packet buffers. Broadcasting and multicasting capability is obtained therefore at no extra cost in this structure.

The adaptors are actually powerful "packet processing" engines. They contain all the packet buffers and perform management of these buffers, routing and packet header manipulation functions; they also provide support for network control and management functions. Considerable

flexibility has been built into the design of the adaptors to permit experimentation with a variety of different approaches.

The queueing structure of the plaNET switch permits it to approach the ideal output port queueing switch in terms of performance. The speed of the shared broadcast ring ensures that queueing at the input is strictly bounded by approximately three maximum sized packets. The output queues are the major point of queueing within the system. In order to provide appropriate quality of service to various classes of traffic the buffer management at the output differentiates between three delay priorities and two “loss” priorities. The delay priority influences the scheduling of packet transmissions on the output link while the loss priority influences the choice of which packet to discard in the event of buffer overflow. Most of the parameters such as discard thresholds, buffer sizes, etc. can be modified under software control.

All the routine packet handling functions are handled in programmable gate array devices on each link adaptor which are designed to keep up with the gigabit/sec link attachments. These routine functions include the queue management functions described above, checking and computing the error detecting codes, checking and updating the hop count field in the packet header, removing, adding or changing portions of the routing field, and performing a routing table lookup if required. Again, the hardware is general enough to permit different routing and packet header options to be easily incorporated.

In addition to the dedicated packet processing hardware, each adaptor contains a RISC microprocessor which is used for control and management purposes. The microprocessor initializes and updates all the registers and tables on the card. The adaptors have extensive statistics gathering and reporting capabilities which are also controlled by the microprocessor.

In addition to the source routing mode supported in the original PARIS system, several new modes have been added to plaNET. These include very general multicasting capabilities, a copy function which permits a controller to copy the packet as it is routed through the hardware, and direct support for the transport of ATM cells.

The plaNET switch will initially support three interfaces:

1. A SONET interface described previously that will operate at either 155 Mbps (STS-3c) or 622 Mbps (STS-12);
2. A gigabit/second serial optical link for attaching other nodes or workstations on a point-to-point basis;
3. A gigabit/second LAN (ORBIT — described later) used for attaching multiple workstations or other traffic sources into the plaNET backbone.

Having discussed the main components of the AURORA backbone network, the following section addresses the problem of local attachment.

4 Local attachment

In this section we address the issue of connecting end user equipment into the backbone network. We shall focus on the attachment of work-stations and personal computers as these are the primary application development platform used in the AURORA testbed. The ideas, however, should easily extend to other kinds of equipment such as mainframe computers or PBX's.

4.1 Local attachment architecture

An important issue is the architecture and topology of the local attachment. Numerous options are available and two have been selected for study in the AURORA testbed. These options represent two of the more important topologies under consideration for broadband local access: the star and the ring. In the star topology each end-user is directly attached to the backbone switch by means of a point-to-point link. In the ring topology each end user is attached in the form of a ring which is also attached into the backbone at at least one point.

The two approaches have their respective strengths and weaknesses. The star has as an advantage the capability to control and isolate individual end users. However, it requires one switch port per user. The ring attempts to share a single switch port among multiple users at the cost of some loss in control of individual end users.

4.1.1 The Sunshine “star”

Sunshine will employ ATM interfaces in a star topology. The ATM cells will travel between hosts and switches over SONET STS-12 or STS-3c point-to-point links. The host interfaces that connect hosts to the SONET links will perform the functions of segmenting packets into cells and the corresponding reassembly, in addition to other functions of buffer management and protocol support. The architecture of the interface is driven by the needs of high performance and by the necessity to allow experiments with portions of the protocol stack, e.g., congestion control and error correction strategies. The second goal dictates that the implementation of any interface should be achievable within a reasonably short time frame, to allow time for subsequent protocol experimentation as part of the AURORA project. Two host interface architectures that seek to meet these goals in somewhat different ways are described in Section 4.2.

4.1.2 The plaNET ring — ORBIT

The plaNET network uses a ring structure for local attachment. The ring known as ORBIT (Optical Ring with Buffer Insertion Technology) is a gigabit/sec local area network that permits workstations and other devices to attach directly into the wide-area network. The ring is based on a buffer insertion ring and allows spatial reuse, i.e. concurrent access to the network. This can increase the effective throughput by a significant factor over traditional token rings.

The ORBIT ring can operate in either bi-directional or uni-directional mode. In the bi-directional case, the ring can reconfigure itself in the event of failure as a bidirectional bus.

Access to the ring is controlled by a distributed fairness mechanism which has been implemented in hardware [11]. It can operate over the entire ring or, in the case of failure of one or more links/nodes, it can operate over disjoint segments of the bidirectional ring. The basic fairness mechanism has been extended for implementing multiple priority levels and the integration of asynchronous and synchronous traffic.

A key aspect of ORBIT is its “seamless” interoperability with plaNET. Considerable attention has been paid to ensuring that the various routing modes, packet structures and priority levels supported in the backbone are supported identically in the ORBIT component. This eliminates the need for gateways or bridging.

4.2 Host interface design

Having described the high level design approaches and architecture, we now go into some detail on the host interface implementations. The speed of this interface is clearly a critical component of

the overall network performance. When viewed from a hardware perspective, it is clear that the speed of tomorrow's interface must be much higher than the technology of today. However, speed is not just a matter of fast data paths: it is more critically a matter of protocol and operating system overhead. Unless these overheads can be controlled, the raw bandwidth of network and interface will remain unused.

This requirement for speed, together with a requirement for the support of multiple services, impose a challenging set of engineering constraints. Further, since AURORA contains two sorts of switches, with two very different multiplexing paradigms, it is desirable to segregate the transfer-dependent parts of the interface, so that by substituting an ATM or PTM specific back-end, a single host interface, running the same transfer mode-independent protocols, can be used in either context.

Several options for the design of the interfaces were considered, and the suitability of a number of possible hosts was evaluated. Three host computer families were selected as the first candidates for attachment into AURORA:

- The DECstation 5000 workstation. An important characteristic of this machine is the high available bandwidth (close to 800 Mbps) on its open bus, the TURBOchannel.
- The IBM RS/6000 workstation.
- The PS/2 personal computer.

Both the RS/6000 and the PS/2 families from IBM use the Micro Channel bus architecture for I/O attachment. Both machines will be used in AURORA—the RS/6000 as a platform for scientific and engineering applications and the PS/2 for more business oriented applications.

4.2.1 ATM interface for the TURBOchannel

The characteristics of the TURBOchannel have had a substantial impact on the architecture of this interface. A host interface that will provide considerable flexibility (for example, allowing experimentation with a variety of segmentation and reassembly protocols) is being implemented using embedded controllers (the Intel 80960) and programmable logic devices [15, 16]. Whereas the ATM interface to the RS/6000 (described below) consists entirely of dedicated hardware, the TURBOchannel interface uses a combination of dedicated hardware (for functions such as cell formatting and data movement) with embedded controllers. The controllers perform those functions that require flexibility, such as scheduling of data for transmission and the reassembly of received cells into larger units. The interface also provides for flexible communication between the host and the interface; in general, they can exchange arbitrary information through an area of shared memory, and the way in which this information is interpreted is determined by the software running in the host and the interface controllers. For example, the host can specify information regarding the priority of different packets that are currently awaiting transmission, and the interface can use this information as input to its rate control algorithms.

The interface uses four STS-3c framers to provide a total bandwidth of 622 Mbps. These will feed into a 4-to-1 multiplexor to allow connection to a single STS-12 link.

4.2.2 ATM interface for RS/6000

This interface [34] migrates a carefully selected set of protocol processing functions into hardware, and connects an IBM RS/6000 workstation to an STS-3c line carrying ATM cells. It is highly parallel and a pure hardware solution. There is a clean separation between the interface functions,

such as segmentation and reassembly, and the interface/host communication. This separation should ease the task of porting the interface to other workstation platforms.

As in the TURBOchannel interface, this design offloads a considerable amount of processing from the host. The benefit of this is twofold. First, it frees the host to address applications workload, and provides concurrent processing. If the computers are high-performance workstations, and not supercomputers, this is a significant attraction. Second, the specialized hardware in the interface can often perform functions faster than the host, thus increasing the bandwidth available to applications. It is noteworthy that, unlike the TURBOchannel interface, this implementation has no software-programmable component, performing all its tasks in hardware.

The current implementation consists of two wire-wrapped Micro Channel cards (which can be reduced to one if double-sided surface-mount fabrication techniques are used) and assumes a connection to an ATM network through SONET framers. The host interface performs the following functions:

1. physical layer interface;
2. segmentation and reassembly;
3. virtual circuit support;
4. buffering for the host.

It is likely that future implementations of the Micro Channel Architecture will support an interface running at 622 Mbps (the STS-12 rate and the full AURORA bandwidth).

4.2.3 ORBIT interface for RS/6000 and PS/2

At IBM, an ORBIT interface for the Microchannel that will operate on either the RS/6000 or the PS/2 family of machines will be prototyped. The current design operates over 1 Gbps serial optical links using the Gazelle HOTROD chipset to perform the clock recovery, coding, and the serial to parallel conversion. The ORBIT access control and fairness mechanisms will be performed in Programmable Gate Array devices. The board will also contain a powerful RISC microprocessor for possible outboard implementation of protocol function and hardware support for the input rate control mechanism of the plaNET architecture. In addition, a “private” interface will be provided that will permit packets to be transmitted to and from the card without requiring them to flow over the Microchannel. This private interface will be used by the video conference hardware to transmit and receive video packets without loading the Microchannel.

4.2.4 Cell-Based Coprocessor for ATM

At MIT a cell-based coprocessor chip is being designed. This chip will provide a direct interface between the ATM network and the coprocessor interface of a conventional RISC processor. The combined RISC processor/cell coprocessor complex could form the core of an ATM-compatible workstation or be used as a stand-alone cell processor, similar in function to Bellcore’s cell processing engine, described in Section 3.1.2. To perform network operations, such as reading and writing cells, the RISC processor executes cell coprocessor instructions, much the way it performs floating point operations. The analogy is so exact that early experiments could be performed on an existing workstation by removing the workstation’s floating point chip and substituting the cell chip in its place.

This effort is closely aligned with Bellcore’s work on the stand-alone cell processing engine. A large fraction of the coprocessor chip, including the serial interfaces, cell buffers, and register file

will be directly copied from parts of the Bellcore chip implementation. MIT will substitute a simple co-processor sequencer and interface for Bellcore's on-chip processing engine. The savings resulting from the substantial re-use of chip design and layout is a clear demonstration of the benefits of the close collaborative links that have been established within the project.

This is primarily a *proof of concept* effort addressing a specific memory architecture issue — one that is largely orthogonal to the performance issues addressed by the TURBOchannel and Micro Channel interfaces. Our initial coprocessor instruction set will be a simple one, relying on substantial software support from the host processor. In practice, this software overhead will limit the overall throughput attainable with this primitive implementation. However, this performance limitation should not detract from our proof of concept objective.

5 Transport and higher layers

In this section we address the work being performed at the higher layers in the protocol stack, i.e. the transport, session, presentation and application layers. All the functions described are intended to be performed at the end user (the host computer). The basic design goal is high performance, i.e. to maximize throughput delivered through the transport protocol to the application. There are two schools of thought in this area. One school would argue that, for the most part, this high throughput is achievable through good implementation practices. For example, it is important to minimize the number of times a packet is moved from one memory location to another. The other school argues that while implementation is clearly important, new protocol concepts provide cleaner abstractions for user applications as well as providing new functions that are enabled by the high speed network.

In the AURORA testbed we hope to reach a deeper understanding of these two approaches. We will study innovative techniques for the implementation of existing protocols as well as introduce new protocol concepts and approaches. Briefly summarized below is a survey of some of the technical activity in this area.

5.1 Application Level Framing

At MIT, a new approach to protocol design is being developed [13]. This approach, called Application Level Framing or ALF, has the following high level goals:

- A more general model of protocol modularity.
- Recognition of fundamental limits to network performance, and demonstration of real systems that can approach these limits.
- A generalization of the “packet” concept, to deal with new technologies such as ATM.
- A new paradigm for providing network service to the application, which permits efficient operation even with lost data elements.
- A structure that permits the application to request and obtain a variety of qualities of service within one protocol suite.

ALF argues that the application, not the network, should control the framing of data. The data stream is broken into Application Data Units, or ADUs, which become the units of checksumming, encryption, retransmission and presentation formatting. Only as the data is moved to the network is it broken into Network Data Units. NDUs could be packets, or ATM cells, as the technology

demands. In this way, ALF can accommodate both ATM and PTM, as discussed above, and indeed can convert between the two.

ALF is an example of a *reduced constraint* protocol, where maximum flexibility has been provided to the implementer as to the timing and order of the various protocol processing steps. One way to take advantage of this to improve the performance of protocol implementations is the technique called Integrated Layer Processing, or ILP. In ILP, which is particularly useful with RISC processors, the data is fetched into the registers of the processor once, where a number of operations can be performed on it. In this way, ALF and ILP reduce the demand on memory, which is (at least in the case of RISC) the most important limit to protocol processing performance. ILP thus lets implementations approach the basic processing limits of the host machine.

A key demonstration of ALF will involve the transport of video, and an MIT objective is to demonstrate transport of compressed video over AURORA using ALF. This is discussed in more detail in Section 7.2.

5.2 Rapid Transport Protocol (RTP)

At IBM, we are developing a transport protocol [24] that will permit operation at the gigabit/second speeds expected from the network. It is a “lightweight” transport protocol in the sense that it has a very small number of states and timers and has been designed to minimize the amount of buffer copying and interface crossings. In addition to these features, RTP provides some key new functions. It has a fast connection setup capability, wherein data can be sent in the first packet. Thus datagram and connection based services are both provided in a single, consistent framework. Error recovery is optional and is implemented with a single timer at the receiver. Both Go-Back-N and selective repeat modes are supported. RTP also provides multicast support (see below).

5.3 Multicast protocols

Multicast services, assisted by special hardware, are considered an important part of a high-speed network in order to support new generations of multi-user applications. However, the issues of the levels of integrity, connection establishment, multicast semantics, etc. associated with such multicast services are not clearly understood. The AURORA testbed will provide us with a method of investigating different multicast service approaches.

At IBM, we are designing into the plaNET network hardware support for multicast. A variety of network control and transport layer services are also being designed to provide a complete end-user multicast service. While these higher level functions assume plaNET style multicast hardware, many of the algorithms and structures are more generally applicable.

Numerous semantic models of multicast are possible. Important distinctions include:

1. what combinations of parties may be connected by multiparty connections and at whose initiative;
2. what degree of reliability is provided on a multiparty connection, and what global temporal orderings of message arrivals may be imposed.

Although almost any combination of requirements can be met, there may be a high associated cost. A key design principle in our service is to find a relatively low-cost set of ‘building blocks’ which should be provided in a basic network multicast service. Higher levels of functionality can then be built up by combining these building blocks appropriately.

5.4 Protocol conversion

While it is hoped that many new high speed applications will be written directly to the new transport interfaces being developed in the AURORA testbed, it is likely that some existing applications will require the network to deal with existing protocols. Thus, one area of interest will be to examine how existing protocols such as TCP, TP4, SNA, DECNET, etc., and evolving new protocols such as SMDS and Frame Relay, can be best supported across the ATM and PTM networks being implemented in AURORA. Such support involves interpreting the packet formats and control mechanisms used by the external protocol and mapping them into the appropriate internal ATM or PTM network mechanisms.

At IBM, the RS6000 attached through an ORBIT ring will be viewed as the primary protocol conversion gateway. The nature of the network permits considerable flexibility in the design of the protocol conversion. For example, in the support of datagram style protocols such as IP or SMDS, questions that would be investigated include: whether it is better to pre-establish connections between likely end-points; how much bandwidth (if any) to allocate to the pre-established connections; whether it is better to transmit cells or packets across the plaNET backbone.

Another important issue to be resolved is the interworking between plaNET and Sunshine. The hardware and software that will be required to effect this interworking at gigabit speeds is currently being investigated by researchers from each of the four sites.

6 Distributed Systems

An important part of network architecture is packaging the network service and presenting it to the application builder in a way which simplifies the application design without restricting or unduly complicating the operation of the network. The abstractions by which network services are provided to applications are especially important when a wide variety of potentially high-bandwidth services are to be supported by a single network. The AURORA project explores both the performance and functional capacities of service abstractions in the context of gigabit-per-second wide-area communications between computers — specifically the construction of *distributed computing systems*.

6.1 Distributed Shared Memory

In distributed computing systems, an essential abstraction is application-application communication, sometimes called “interprocess communication” (IPC). Particularly important is the character of the IPC primitives presented to computer users and applications.

The Penn approach is to use Distributed Shared Memory (DSM) as the IPC paradigm. DSM provides the illusion that an ensemble of computers connected by a network have a shared address space. The idea is to view networking in terms of a new abstraction, that of addressable memory, and as a consequence, the coupling between communication and computation. Evidence is accumulating that treating a computer network as an extension of a computer address space offers compelling advantages in both abstraction and performance. The performance results from the similarity between the network abstraction and the abstraction of addressable memory used by processing units. This similarity would tend to reduce the costs of IPC incurred due to layers of processing. For example, a typical protocol stack, moving towards the physical layer, might involve:

1. Converting a floating point number to ASCII representation
2. Storing the ASCII in a memory area

3. Passing the data to the operating system via a *write()* system call
4. Copying the user data into an operating system managed data area (a buffer)
5. Segmenting the buffer into frames of a size acceptable to the network
6. Copying the frames into an area reserved for network interface hardware

Each of these activities requires some processor intervention, although current protocol stacks are more memory-bandwidth constrained than processor constrained, due to the number of copies that must be performed. Even with fast processors, IPC over fast communications networks has often achieved only a small fraction of the bandwidth of which the network is capable.

Since AURORA seeks to exploit a significant fraction of the bandwidth available, communication with each machine instruction, as occurs with shared memory, is an extremely attractive goal[33]. DSM has thus served as the focus of much of the Penn applications interface work. Particular issues include:

- Security of Distributed Shared Memory, since many attractive applications, such as medical image analysis, are untenable without protection and privacy;
- Hardware support for high-bandwidth privacy transformations [6, 7];
- Operating System (OS) support for high-bandwidth networks using DSM, and OS support for real-time traffic such as multimedia; the research vehicle for OS mechanisms is UPWARDS, described below;
- Architectures for extending LAN DSM models to WANs in an efficient manner, e.g., by modifications to switch fabrics to enhance the performance of such systems. Broadcasting is inefficient in WANs, and strategies for locating pages and maintaining cache coherency are thus different from those used in a LAN. CapNet's solution is to distribute the page table in the network and place directory information in the switches, so that page requests can always be routed to the current owner of the page [31]. Thus, CapNet envisions a modest amount of extra hardware support in the network switch fabric. This lookup hardware is generally useful. The same mechanism, hardware and general approach, in principle is needed to enable the extended bridge architecture for ubiquitous personal telephone service. Such support was also needed to enable assisted message coupled inter-process communications such as described in [17]. Thus, such lookup, rather than a convenient optimization for the current setting, may represent a general solution to a wide variety of name resolution problems in distributed systems.

We are interested in testing the viability of DSM as an IPC mechanism on networks with a high *bandwidth × delay* product; we expect some significant insights into protocol performance will result. We intend to further develop and evaluate the DSM approach to storage management and interprocess communication on the AURORA testbed.

6.2 U Penn Wide-Area Distributed System (UPWARDS)

The UPWARDS Operating System [29] is a research vehicle for experimenting with applications of DSM, as well as managing devices and scheduling. As a base for applications, it defines the service primitives available to programmers for processor control, interprocess communication, and external interaction. We have taken advantage of the lack of existing applications by positing

an ideal, and designing towards it. We began by examining our assumptions about the future scientific and engineering computing environment. UPWARDS assumes high performance personal workstations connected to a high-speed WAN. Such workstations are used by a small number of users, typically one. The user emphasis is thus on response time and not on aggregate throughput.

The following design choices have been made. UPWARDS scheduling is almost entirely synchronous; the only synchronously-serviced “interrupt” is that of the system clock driving the scheduler. Hardware interrupts are serviced by creating an *event* that is later serviced in a scheduled manner. Traditional interrupt service strategies defeat caches, use memory bandwidth, and can add a large variance to execution times. UPWARDS will support multimedia traffic, which requires real-time scheduling, a natural outgrowth of our scheme.

UPWARDS *Address Spaces* are distinct from processes, which can share an address space. Each process must be associated with at least one address space. For a computing environment comprised of high-performance personal workstations and network connections, heavyweight address spaces are not needed on a per-process basis; these address spaces are largely a protection mechanism, and the individual virtual address space provides this protection at a considerable performance penalty, e.g., when context switches are required. Context switches are traditionally an expensive operation and are performed quite often [26]. One way to reduce this expense is to form extremely lightweight threads, and this is one UPWARDS approach.

The UPWARDS interprocess communication mechanism is shared memory, upon which other mechanisms such as message-passing or RPC can be constructed. We have shown experimentally, for example, that shared memory and synchronization primitives can be used to implement streams, which are useful for many IPC tasks, as illustrated by UNIX pipelines.

Many visual applications have a shared memory style of communication with a frame buffer, used to display complex objects. Real-time voice and video require specification of the real-time data delivery requirements. Such multimedia applications are a focus of intense research, as (1) they are expected to be a major source of applications traffic; and, (2) a simple shared-state abstraction is insufficient. In particular, we must understand service provision for applications with timing requirements, and incorporate this into the DSM model. We are unsure at this point how to include real-time continuous media such as digital video. One possibility is to tag address ranges used for such streams as “volatile”.

Networks with high bandwidth-delay product pose several problems for UPWARDS in providing interactive distributed computing. Most important of these is latency. Wide-area networks have large latency (delay) due to their large geographical scope. For example, in a nationwide network, the transcontinental delays are tens of milliseconds (roughly comparable to disk latencies). The important goal is the reduction of the *amortized latency* which is the average latency per reference. Two latency-reduction strategies are *caching* and *anticipation* [30]. With caching, a fetched object is saved for reuse, and with anticipation, an object is pre-fetched for future use. Both techniques reduce the average latency, not the worst-case. Neither of these schemes seem viable with message-passing systems or remote procedure call, as in either case a detailed model of the application is necessary to decide what should be cached or prefetched. Cache management strategies (e.g., invalidation and write-through) are much more difficult with point-to-point than broadcast topologies. While caching has been extensively studied, Penn believes that anticipation is a logical candidate for examination where delays are large and bandwidth is plentiful. A “back-of-the-envelope” calculation shows that sending extra data on each request-reply becomes more attractive as (1) latency increases, and (2) bandwidth increases. Traces of program executions [28] support the latency-reduction strategies we will incorporate into the memory manager.

7 Gigabit Applications

AURORA will experiment with several applications that will stress the testbed infrastructure and exercise its gigabit capabilities. The applications identified for exploration in the AURORA project manifest the diversity of traffic models that is needed for a convincing evaluation of tomorrow's network technologies.

Of particular interest are medical imaging and collaborative efforts that require multiparty interaction, such as

- education,
- group discussions in a software development effort (especially the large project co-ordination activities involved in *software manufacturing*),
- laboratory experiments,
- business meetings, or
- collaboration within the AURORA project itself.

7.1 Medical Imaging

Several of the AURORA sites are located near to major medical research facilities, and we see several possible applications for AURORA, especially in the area of radiological imaging [3].

- Rural access to urban medical centers. For example, Pennsylvania has world-class medical facilities in Philadelphia, Harrisburg, and Pittsburgh. However, there are about 2 million Pennsylvanians who do not have easy physical access to the urban care facilities. Excellent trauma care is provided in the urban areas, but many lives are being lost in areas that have poor access to modern trauma care facilities. Often these areas are served by community hospitals that do not even have a physician on-site at all times. The most recent attempt to solve the problem was by using helicopters to speed trauma patients to a trauma center. This does not work in cases where the patient is too unstable to be transported. In these cases, "tele-medicine" would be invaluable. Possible modes of communications would include: video (doctor observing patient), image (XRAY), voice, and data (telemetry). The goal would be to direct a nurse or paramedic to stabilize the patient before transportation.
- Digitization of the film library in a major hospital. Several technical challenges must be met to allow the digital library to compete with conventional film libraries. Basically, they fall into two classes: increasing communications throughput, and scaling the storage requirements of the system. The storage requirements of a full-hospital system are estimated as 13 TB (13,000 gigabytes) of on-line or near-line storage.
- Communication between satellite facilities and major medical centers. The Penn Hospital (HUP) does MR (magnetic resonance) image reading for Rhode Island. The images are collected in RI, then transmitted to HUP for reading. Reports are transmitted in the other direction.
- A final application is to provide patients with remote, urgent, access to their older films. The remote access is required when a patient is being treated away from home, such as someone who is traveling, or who changes residence.

7.2 Video Conferencing

Bellcore has provided experimental prototype Video Windows, which will be used for interactive video conferencing, to each site. The Video Window is an experimental video conferencing terminal comprised of two large screen projection televisions mounted side-by-side creating the illusion of one large screen. Two cameras co-located with the screens are arranged to produce images that when viewed at the remote terminal are blended to appear as one large image. The life-size images, combined with high-quality directional sound, create an effective teleconferencing facility.

At Penn, a Digital Video Interface for the Micro Channel Architecture has been designed and implemented. The card interfaces the IBM RS/6000 to NTSC video, which is the video standard used by the Video Windows present at all AURORA sites.

MIT is investigating a number of video-related issues. One objective of the MIT research is to demonstrate the transport of video over AURORA, using the ALF protocol approach described in Section 5.1. This demonstration has several goals, relating to ALF, to video compression schemes, and to bandwidth allocation in networks.

Traditional video compression for transmission is based on the idea of circuit switching. The packet or cell switching alternative places different requirements on the coding scheme, in particular the opportunity to take advantage of statistical bandwidth allocation but the need to deal with lost packets. To carry video information, the compression protocol must provide a structured means to deal in real time with the loss of information. ALF provides an explicit framework for this task. The project would involve demonstrating the use of an ALF protocol to carry real-time compressed video over a packet switched network. The MIT approach to bandwidth allocation would be used to intermix this video with more traditional data transfer. The plan is to identify some suitable compression algorithm, modify it as necessary to match the packet switching context, realize it in software or using existing compression chips as appropriate, and demonstrate it using the ALF protocol approach.

7.3 Multimedia Multiparty Teleconferencing

Work is under way at IBM on multimedia, multiparty teleconferencing using a workstation-based system. We are developing a system that enables people to conduct effective meetings without physically getting together. Sitting in their offices, conferees will see each other via real-time motion videos on their multimedia workstation display, talk and listen to all the conferees via real-time audio, and view presentations via an electronic blackboard (EB) that supports on line editing and handwriting. We are currently looking into the relevant issues in supporting real time packet video and audio and other media. Several examples are described below.

In playing back packet videos, we are studying algorithms that are capable of compensating packet loss, corruption, and delay jitter with small end-to-end delay, buffer requirement, and motion distortion. We are also looking into the issue of minimizing the impact on video quality of corruption in compressed video, particularly frame-to-frame compression. In supporting the electronic blackboard (EB), we are investigating architectures for integrating an editing and foil-making system with multicast network connections to support cooperative editing among multiple parties. One issue to be addressed, for example, is to compare two alternatives — a distributed approach in which each party maintains an identical workspace or a central server-client approach.

The system being built at IBM will be based on a PS/2 with a VGA display attached to a M-Motion video adaptor. The M-Motion video adaptor has the ability to display a moving video image within a window on the VGA display. We will build a video interface card that will attach to the M-Motion adaptor on one side and the ORBIT adaptor on the other. To display video,

the interface card will perform the functions of receiving packets from ORBIT, reassembling the packets into a video stream, decompressing (using JPEG standard compression) and writing the video stream into the frame buffer on the M-Motion video adaptor for display. All information is transferred through direct interfaces that do not cross the Microchannel. On the transmit side, video is received from a camera attached to the M-motion adaptor, compressed, packetized and send into the network over the ORBIT adaptor.

At Penn, a variety of issues in teleconferencing are being studied, including synchronization of networked multiparty conversations. Group discussions present several problems which are *control* problems, e.g., who speaks next, or, where do I focus my attention. The problems likewise exist in the domain of teleconferencing, and were addressed in the earliest attempts to provide such facilities, such as voice conference-calling [4]. When computer display facilities are involved, or when computers are used in a multimedia environment, these control issues become computer-human interface issues. A research project at Penn was initiated to investigate multiparty conversations, and in particular, the associated control problems. We have also studied voice traffic to gain insight into multimedia and real-time traffic issues, by performing voice experiments over the Internet, using Ethernet-based telephones for conversations between participants at such sites as the University of Washington, the Jet Propulsion Laboratory, and Stanford University.

7.4 Service Integration

In the belief that a higher level of service integration must be a goal for the next generation of network, a variety of approaches to this problem will be studied at MIT and IBM. True service integration means much more than simple physical integration, i.e., carrying several sorts of services over one set of trunks and switches. It implies that parallel information flows, carrying a variety of media-specific services, can be multiplexed over the same host interface and can be utilized by multi-service applications. Furthermore, it should be possible to “cross-connect” these flows to permit generalized accessibility to media-independent services such as bulk storage servers.

Obvious examples of application requirements for service integration are multi-media information transfer (in support of live video conferencing) or the storage and retrieval of multi-media documents. Such documents might combine fragments of text, graphics, video and audio, all of which should be stored in a common storage server, and must be retrieved in a coordinated manner as the application presents the document to the reader. Clearly, the transfer of the document components must be coordinated and must occur in real-time to support interactive viewing and listening. Furthermore, the full range of supported media services, including video, must be stored in a coherent manner, which requires a file system that can cope with a range of service, from text to full motion video. All of these services, and others not yet identified, must be carried across the common network infrastructure, from the disk to the display.

Although the problem of service integration may not be currently well-understood, if service integration is to be a goal of the next generation of network, it is critical that we now demonstrate the need and, at the same time, demonstrate that it can be accomplished. The AURORA project will attempt to meet this goal.

8 Network Control and Management

The control and management of high speed networks is an extremely active area of research. The AURORA testbed will provide us an opportunity to experiment with many of the ideas proposed in the literature and to evaluate their effectiveness in realistic environments.

We distinguish between network control and management on the basis of time-scale. Network control typically refers to real-time decisions made on a call-by-call or even a packet-by-packet basis. These decisions include route computation, congestion control, bandwidth management, etc. Network management typically operates on a slower time-scale and may involve a human operator. The issues relevant here include problem diagnosis, maintenance of large network related databases, graphical presentation of information to the operator, billing etc. In the following sections, we briefly discuss some of the research activities underway.

8.1 Distributed Route Computation

When a new call is admitted into the system several functions have to be performed. These functions include the call acceptance function that makes decisions on whether or not to permit a new call access to the network and the route computation function that determines the path that a call is to be routed over. These functions of routing and admission control have to take into account the parameters of the call (eg. bandwidth), its desired quality of service, and the status of the network in terms of the loading and availability of its links and nodes.

At IBM, a decentralized approach to this problem is being investigated. We use a distributed route computation where each source maintains enough information to compute a suitable path to any destination. This requires a topology and utilization maintenance algorithm that keeps in each node a global view of the network status. This view is used by the node to compute a path that can satisfy the new call. The key to this approach is to ensure that the global view is as current as possible.

In traditional networks, this is done using a flooding procedure. The flooding procedure uses excessive computational resources and introduces software delays in the delivery of messages which can cause inefficiencies in the route selection process. At IBM we are exploring the use of hardware multicast support to perform the “flooding”. A distributed algorithm sets up a spanning tree in the network and topology information is broadcast through hardware over the spanning tree. This method ensures minimal software delay and processing load.

In the testbed, the topology maintenance algorithm (and other control functions) will be implemented in an RS6000 that attaches to every planET node. This platform will permit us to test out and experimentally refine our ideas.

8.2 Flow and congestion control

Closely associated with the work on route computation and admission control is the research in flow and congestion control. The idea here is to avoid over-allocation of network resources.

At IBM, we are investigating variants of the “leaky bucket” style input rate controls [2]. The basic idea being studied is to introduce a new class of traffic which is given lower loss priority within the network. In other words, in the event of congestion this lower loss priority traffic is always discarded first. The leaky bucket is being modified to incorporate this notion. In the event of a source requiring more capacity than it requested, the leaky bucket makes a decision if it is best to accept the packet and send it into the network as a lower loss priority packet, if the packet should be discarded at the source or if the packet should be queued at the source. There are interesting trade-offs between loss and delay and the optimal solution is very dependent on the nature of the application.

The MIT research on flow and congestion control is closely tied in with the work on ALF, described in Section 5.1. In previous efforts, MIT has explored alternatives to window-based flow control (for example rate-based controls) which may perform better on high-speed long-delay net-

works. The current objective is to develop and evaluate a practical scheme to permit controlled sharing of network bandwidth. The current approach should permit realization in software at current packet forwarding rates, as well as realization in hardware at gigabit speed. Based on our work to this point, we believe the scheme can meet the following requirements:

- Support diverse classes of traffic, including traditional data transfer and video.
- Permit traffic with similar characteristics to be aggregated into a single control class, to reduce the amount of control state in large networks.
- Couple low-level resource allocation decisions to a higher-level accounting scheme.
- Detect and regulate abuse of network bandwidth.

Our current plan is first to explore these control concepts at lower speeds using a software platform and then transfer these ideas to the AURORA context. The overall goal for the MIT research is to continue to prove and elaborate the ALF concept, by performing a number of demonstration projects, and by taking the results in flow and congestion control and integrating these into ALF to produce a complete protocol scheme addressing performance issues related both to host implementation and resource sharing inside the network. These concepts will be demonstrated over the AURORA facilities.

8.3 Bandwidth Management

At IBM, the issue of bandwidth management and allocation in a network carrying connections with possibly widely different traffic characteristics is under study [1]. Because of the statistical multiplexing of all connections at the physical layer and the variations of connections bit rate, it is important to characterize, for a given Grade-Of-Service (GOS), both the effective bandwidth requirement of a single connection and the aggregate bandwidth usage of multiplexed connections. The main focus of this work is a computationally simple approximation for the “Equivalent Capacity”, or bandwidth requirement, of both individual and multiplexed connections. The approximation takes into account the connection characteristics, the existing network traffic, and the desired Grade-Of-Service. It provides a unified metric to represent the actual bandwidth requirements of connections, and the corresponding effective loads on network links. This metric can then be used for real-time implementations of various network control functions, e.g., routing, call admission, etc.

8.4 Call Repacking

Call repacking, under investigation at Penn [32], is a mechanism that rearranges virtual circuits from one path to another. Such a mechanism provides network management with flexibility and opportunities. Typically, the cost of a path is evaluated by some cost function, e.g, hops or utilization. The repacking mechanism rearranges a circuit from the current path to a path with lower cost. The overall effect of rearrangement may be to optimize the throughput of the network, to survive node or link failure or to enable certain network management functions to be performed.

Call repacking can be viewed as a powerful and unified mechanism that improves the throughput, survivability and maintainability of networks. Further research will explore this mechanism and obtain further understanding of its costs and benefits.

8.5 Management Architecture

We plan to explore issues related to the management and operation of a gigabit-per-second networks. The emerging OSI network management standard will be used as a starting point and, enhancements to support gigabit-per-second network will be made. We will build a prototype, workstation-based system for monitoring and managing the AURORA testbed. It will rely on OSI-style “agent” processes in each of the network nodes to collect information about performance, throughput, communications errors and hardware failures. Standard protocols (e.g., SNMP and CMIP) will be used to transport information collected in this way to the management workstation. Within the workstation, a base for management application development will be provided. Its most important aspect will be a management database built using available database technology and implementation of appropriate parts and extensions of the OSI-defined Management Information Base (MIB).

Building on existing work allows other practitioners to take advantage of our innovations. It also allows us to focus our development effort on aspects unique to gigabit per second networks.

8.6 Billing

Billing is an important consideration, and service providers for the next generation of networks will expect, at minimum, some method of cost recovery. This requirement gives rise to a number of architectural questions.

- What is the metric by which billing is done (e.g., traffic, connect, average rate used, etc.)?
- Where should data be collected (at the switches, at the edge of the network)?
- Is there some set of data which is needed independent of billing metrics and policy?
- What sort of logging and storage overhead will be incurred? The high traffic capacity may generate huge records which must be retained until a summary invoice is generated.
- How do we maintain configuration data? For both operations and billing, configuration data for the network must be maintained; we should understand how to add nodes and lines as the network grows, without interrupting service.

9 Discussion and Conclusions

9.1 Experimental Evaluation

An important part of the AURORA project is the evaluation of the installed testbed in its various forms, involving each sort of switch both separately and cross-connected, as well as the various sorts of protocols and application interfaces. The key question to be answered for the testbed is how effectively the various technologies and protocols can support the desired range of applications requirements. This question can be answered by experimentally exploring the operation of the facility and by assessing the relative complexity of the various approaches in the testbed.

The evaluation depends to a great extent on the traffic model for the load that the network is expected to carry. Our assumption in AURORA is that the network of tomorrow will support a variety of applications, with varying communications service requirements. Because AURORA includes experiments with actual applications, we will have ready access to actual sources and sinks that can be used to drive the network. This component of the testbed is critical, for it

enables exercising the network with traffic loads founded more in real application requirements than in untested assumptions. We will, as well, test the various AURORA configurations with test loads that attempt to simulate expected traffic classes such as video, voice and bulk and interactive data transfer.

Real traffic models will be critical in comparing the two transfer mode alternatives. Work in IBM has indicated that, while ATM and PTM can provide similar services, there may be some significant differences in terms of efficiency, processing overhead, and ease of control. For example, consider link transmission efficiency (defined as the percentage of useful carried user traffic to the total link bandwidth). The source routing features in PTM typically force it to use a somewhat larger header than ATM. On the other hand ATM has a constant header overhead for every cell as opposed to the single header overhead per user packet in PTM. In addition, the cell integrality requirements for ATM forces padding the last cell in a packet, a trivial overhead for a large user packet but significant for short packets.

Depending on the user traffic model, it is possible to have both systems at equal efficiency or one system considerably outperforming the other (in some extreme cases by a factor of two). On one hand, for some classes of traffic, link transmission efficiency is the most relevant performance measure, for poor transmission efficiency may in turn manifest itself as increased differences in delay, buffer requirement and packet loss. On the other hand, alternative sources of traffic that are sensitive to jitter (variation in delay) may be better served in an ATM environment. It is hoped that experimentation in AURORA will shed some light on the tradeoffs between ATM and PTM. We also hope to evaluate the relative merits of alternative solutions to many other networking problems.

9.2 Summary

The AURORA testbed will provide a platform in which researchers can explore business and scientific applications of gigabit networks, while evolving the network architecture to meet the needs of these emerging applications. Through the existence of sites with different switching equipment, workstations, and software architectures, important lessons about interworking will be learned.

We see the immediate contributions of the research as being:

- High-performance switching technologies and supporting experiments;
- Hardware support options for protocol architectures, where appropriate, such as the host interfaces described in this article;
- Interworking strategies for dissimilar high-speed architectures;
- Protocol architectures which can service high-speed networks with reasonable processing and economic cost;
- Networking abstractions which enable applications to access full network bandwidths;
- Operational experience with gigabit per second WANs and their applications.

The existence of the AURORA testbed will stimulate further research into applications and terminal devices. Such research will provide concrete feedback for the future evolution of the telecommunications infrastructure of the nation, including standards efforts, carrier direction, and networks vendors. Furthermore, the operational experience gained in this testbed will bear directly upon the deployment and operation of broadband switching installations — be they carrier central office exchanges or private customer premises switches (PBXs).

We are clearly excited about our research, both because of the near-term scientific interest and the long-term impact of our results on networking infrastructure. The importance of infrastructure cannot be overstated, as has been conclusively demonstrated by the U.S. Internet and its effect on collaboration, work environments, and interpersonal communication. Unfortunately, even the highest speed interconnects of the Internet represent a small fraction of the bandwidth needed for advanced applications. Analogous to the post-World War II U.S. Highway construction, infrastructure which can gracefully handle loads of widely varying traffic is necessary. We hope to point the way to such an infrastructure.

Acknowledgments

People

As might be imagined the number of people necessary to pursue a project of AURORA's magnitude is large as well, and we invoke the all-too-frequent excuse that they are too numerous to mention. The authorship of this paper is more a function of the leadership role we have taken in AURORA than it is of the research contributions. We have tried to indicate the extent of the effort to some degree with citations of these contributions.

Trademarks

IBM, Micro Channel, RS/6000, and PS/2 are trademarks of the International Business Machines Corporation. DECstation and TURBOchannel are trademarks of the Digital Equipment Corporation. Intel is a trademark of the Intel Corporation. UNIX is a trademark of AT&T. Ethernet is a trademark of Xerox Corporation. HOT ROD is a trademark of Gazelle Microcircuits, Inc.

Support

Much of the support for this ambitious research agenda comes from the corporate participants, namely Bell Atlantic, Bellcore, IBM, MCI and NYNEX, in the form of equipment and especially service provision. Support to the academic participants has come from Bellcore (through Project DAWN), and from the Corporation for National Research Initiatives (CNRI), which is funded by the National Science Foundation and the Defense Advanced Research Projects Agency.

References

- [1] H. Ahmadi and R. Guerin. Bandwidth allocation in high-speed networks based on the concept of equivalent capacity. In *Proc. 7th Int. Teletraffic Seminar*, Morristown, NJ, October 1990.
- [2] H. Ahmadi, R. Guerin, and K. Sohraby. Analysis of leaky bucket access control mechanism with batch arrival process. In *Proc. Globecom'90*, San Diego, CA, December 1990.
- [3] R. Arenson, U. Penn Dept. of Radiology. Personal Communication. March 1991.
- [4] AT&T Bell Laboratories, Murray Hill, NJ. *Engineering and Operations in The Bell System*, 2nd edition, 1983.
- [5] B. Awerbuch, I. Cidon, I. S. Gopal, M. A. Kaplan, and S. Kutten. Distributed control in PARIS. In *ACM Principles of Distributed Computing*, Quebec City, Canada, 1990.

- [6] A. G. Broscius and J. M. Smith. Exploiting parallelism in hardware implementation of the DES. To appear in *Proc. CRYPTO '91*, Santa Barbara, CA, August 1991.
- [7] A. G. Broscius. Hardware Analysis of the NBS Data Encryption Standard. M.S. Thesis, University of Pennsylvania, Computer and Information Science Department, May, 1991.
- [8] I. Cidon, J. H. Derby, I. S. Gopal, and B. Kadaba. A critique of ATM from a data communications perspective. In *Proc. ICC 90*, New Delhi, India, 1990.
- [9] I. Cidon and I. S. Gopal. PARIS: An approach to integrated high-speed private networks. *Int. Journal of Digital and Analog Cabled Systems*, 1, 1988.
- [10] I. Cidon and I. Gopal. Control mechanisms for high speed networks. In *Proc. IEEE Int. Conf. on Communications*, Atlanta GA, 1990.
- [11] I. Cidon and Y. Ofek. METARING — a full duplex ring with fairness and spatial reuse. In *Proc. IEEE INFOCOM*, San Francisco, CA, 1990.
- [12] I. Cidon, I. S. Gopal, and A. Segall. Connection setup protocols for high speed networks. In *Proc. ACM SIGCOMM '90*, Philadelphia, PA, September 1990.
- [13] D. D. Clark and D. L. Tennenhouse. Architectural considerations for a new generation of protocols. In *Proc. ACM SIGCOMM '90*, Philadelphia, PA, September 1990.
- [14] Computer Staff. Gigabit Network Testbeds. *IEEE Computer*, September, 1990
- [15] Bruce S. Davie. Host interface design for experimental, very high speed networks. In *Proc. Compton Spring '90*, San Francisco, CA, February 1990.
- [16] B. S. Davie. A host-network interface architecture for ATM. To appear in *Proc. ACM SIGCOMM '91*, Zurich, September 1991.
- [17] D. J. Farber. The distributed computing system. In *Proc. Compton '73*, 1973.
- [18] J. Giacomelli, M. Littlewood, and W. D. Sincoskie. Sunshine: A broadband packet switch architecture. In *Proc. ISS'90*, Stockholm, May 1990.
- [19] A. Gopal, I. Gopal, and S. Kuttan. Broadcast in fast networks. In *Proc. IEEE INFOCOM*, San Francisco, CA, 1990.
- [20] A. Hac and H. B. Mutlu. Synchronous Optical Network and Broadband ISDN protocols, *IEEE Computer*, 22(11), November 1989.
- [21] J. J. Hickey and W. S. Marcus. A CMOS batcher and banyan chip set for B-ISDN. In *Proc. ISSCC*, San Francisco, CA, February 1990.
- [22] J. J. Hickey and W. S. Marcus. The implementation of a high speed ATM packet switch using CMOS VLSI. In *Proc. ISS'90*, Stockholm, May 1990.
- [23] J. J. Hickey and W. S. Marcus. A CMOS batcher and banyan chip set for B-ISDN packet switching. *IEEE Journal of Solid State Circuits*, 25(6), December 1990.
- [24] M. A. Kaplan Rapid Transport Protocol. IBM Research Report, to appear.

- [25] David A. Martin. A VLSI cell interface and forwarder for broadband ISDN. Master's thesis, M. I. T., 1991.
- [26] J. C. Mogul and A. Borg. The effect of context switches on cache performance. In *Proc. Fourth International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS-IV)*, Santa Clara, CA, April 1991.
- [27] T. J. Robe and K. A. Walsh. A SONET STS-3c user-network interface IC. In *Proc. Custom Integrated Circuits Conference*, San Diego, CA, May 1991.
- [28] J. M. Smith and D. J. Farber. Traffic characteristics of a distributed memory system. To appear in *Computer Networks and ISDN Systems*, 1991.
- [29] J. M. Smith. UPWARDS: Towards an operating system toolkit. In *Proc. LLNL/SRC Workshop on Supercomputer Operating Systems*, Livermore, CA, July 1990.
- [30] J. M. Smith. Distributed systems of the future? In *Internet Research Steering Group Workshop on Architectures for Very-High-Speed Networks*, Cambridge, MA, January 1990.
- [31] M. Tam and D. J. Farber. CapNet — an alternative approach to ultra high speed networks. In *Proc. ICC '90*, 1990.
- [32] M. Tam and D. Farber. On call repacking. Technical report, U. Penn Department of Computer and Information Science, 1991.
- [33] M. Tam, J. M. Smith, and D. J. Farber. A taxonomy-based comparison of several distributed shared memory systems. *ACM Operating Systems Review*, 24(3), July 1990.
- [34] C. B. S. Traw and J. M. Smith. A high-performance host interface for ATM networks. To appear in *Proc. ACM SIGCOMM '91*, Zurich, September 1991.