A high-performance transport network platform

by G. Lebizay
C. Galand
D. Chevalier
F. Barre

This paper introduces the architecture and the technology of the new IBM transport network offering, which supports both fixed-length and variable-length packet switching. After a review of the broadband networking environment and the requirements it makes on the transport network, the transport network node is described. The hardware and software architecture is described, and performance figures are reported showing that the switch can handle a mix of traffic while sustaining maximum throughput.

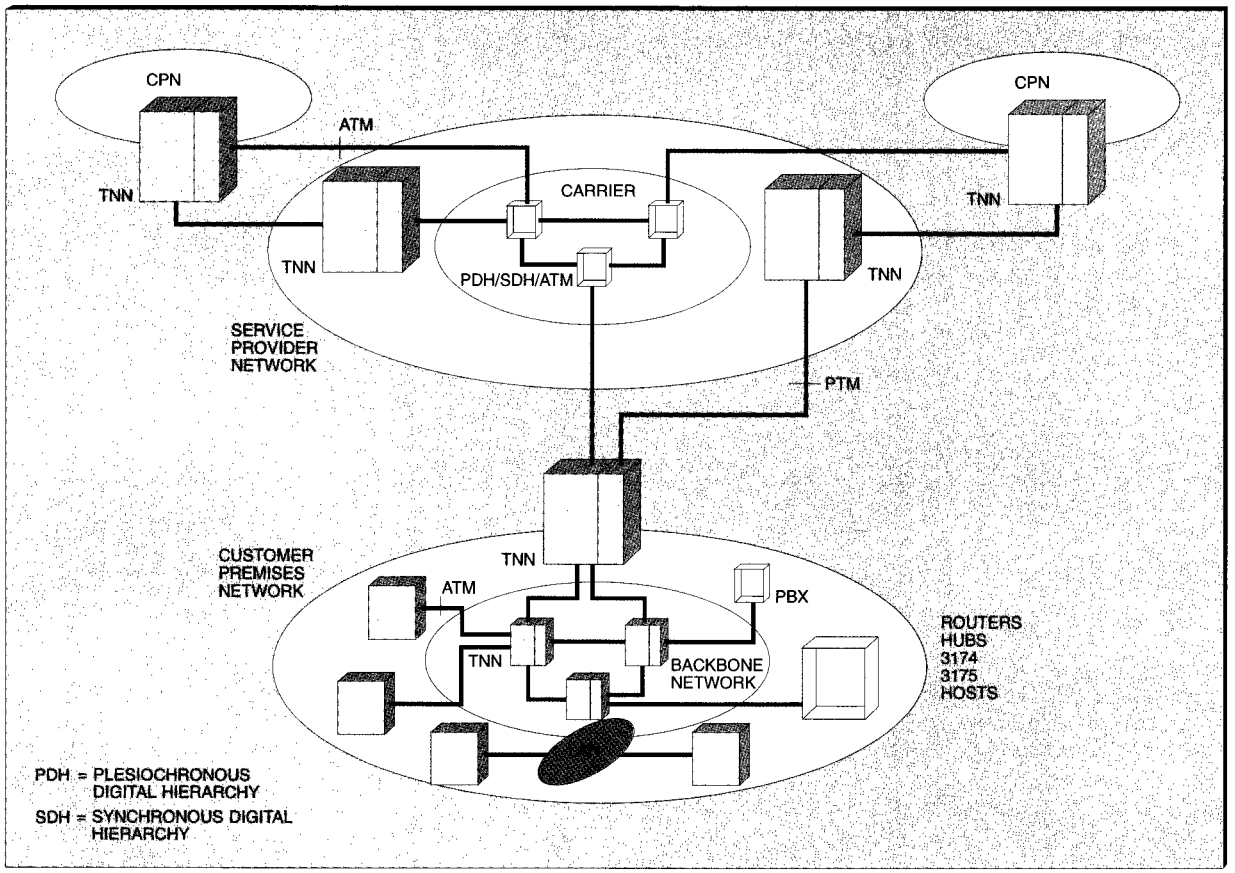
In this paper we describe the architecture and the technology of the IBM transport network offering that supports both packet transfer mode (PTM), which is the native format of existing protocols such as Internet Protocol (IP), frame relay (FR), and high-level data link control (HDLC), and asynchronous transfer mode (ATM), which is the emerging format for virtually all new communication equipment. First, we briefly review the broadband networking environment and the requirements it places on the transport network. We show that these requirements translate into new approaches for the operation and control of the transport network node (TNN): low-level switching, rate-based flow control mechanisms, extended multicast capability, bandwidth management, and fast distribution of control information. The section concludes with an overview of the IBM Networking BroadBand Services architecture that addresses these new requirements.

We then give a detailed description of the transport network node. After an overview at the system level, the switch and adapter design points are discussed. The switch is designed to avoid congestion and prevent cell loss. The adapters are modular and can be used either for port, trunk, server, or control-point functions. The adapter design point is based on a picocoded very large scale integrated (VLSI) unit, which allows for flexibility while preserving performance. Performance figures are reported to illustrate the design points. Next we describe the transport network node software implementation, which is based on the Networking BroadBand Services (NBBS) architecture. The different functions are described and positioned with reference to high-performance and reliability objectives. Finally, we give an overview of the network management architecture. Again, a distributed implementation enables a high level of performance and reliability.

Broadband environment

The last decade has seen an unprecedented growth in the availability of digital transmission technology and the widespread deployment of high-per-

Figure 1 Broadband network model



formance workstations. In the same period, equipment and service costs have decreased so much that many enterprises have increased their reliance on computer networks at the heart of their business. These changes have enabled the development of multimedia applications across all aspects of the enterprise. The net result has been a shift of the networking paradigm, from the earlier local area network (LAN) data interconnection over a wide area network (WAN), to a high-speed network able to transport a wide variety of data with a prespecified quality of service (QOS), and with greatly increased call-processing capabilities.

Figure 1 shows the broadband networking environment, featuring a customer premises network (CPN) and a service provider network (SPN). The CPN includes a wide variety of equipment, such as private branch exchanges (PBXs), hubs, routers, and communication controllers, which is locally interconnected through various types of networks, using subnetworking. Today, a number of subnetwork technologies exist, such as that defined by the Institute of Electrical and Electronics Engineers (IEEE) in standard 802.3, that allow multiple protocols to be transparently carried over shared data communication facilities. Multimedia applications have started to use these existing facilities, with requirements for real-time delivery. An approach in the industry and in the standards organizations is to define, for the CPNs, a common ATM transport network that would support all these subnetwork technologies while enabling the migration to gigabit speeds.

Today most SPNs are based mainly on time-division multiplexing (TDM) facilities offered by carriers, and are used to provide CPN interconnection

over WAN. Synchronous Optical Network (SONET) and Synchronous Digital Hierarchy (SDH) technologies are now being deployed with up to gigabit speeds. As in the CPN industry, the SPN industry is moving from earlier carrier subnetwork technologies, such as X.25 packet switching, to ATM technology for equipment interfaces and data transport across the WAN. The transport network node (TNN) can operate in both CPN and SPN environments. It provides a new way to interconnect subnetworks, ensuring QOS for any kind of data, offering standard interfaces to user and carrier equipment, and providing transparent end-to-end transport to any kind of user protocol.

The networking paradigm shift has generated a set of new challenges. As mentioned, the high-speed transport network must be able to transport a variety of data types with a prespecified QOS, which can spread over a wide range of values. For example, the maximum end-to-end delay and the maximum bit error rate may be specified as small as 5 milliseconds (ms) and 10^{-10} respectively for a voice connection, while they may be 100 ms and 10^{-6} for a file transfer. Transporting different data on the same physical layer with such different QOS specifications is the first technical challenge of the TNN.

The second challenge is related to the bandwidth requirements of these various data types, in terms of peak rate and burstiness. The burstiness of a source is defined by the ratio of peak rate to mean rate of the source, and can cover a wide range of values. For example, uncompressed voice has a peak rate of 64 kilobits per second (Kbps) and a burstiness of 1, while interactive data from a high-speed server may peak at 50 megabits per second (Mbps) with a burstiness larger than 100. Transporting data with such a wide range of characteristics, while optimizing the network resources, represents the second technical challenge of the TNN.

Perhaps a more fundamental challenge is to minimize the packet processing requirement within each switching node in the network, to enable operation at media speed. Consider a 30 millions of instructions per second (MIPS) processor. A total budget of 3 million cycles could be dedicated to the processing of each 128-byte packet transmitted on a low-speed 9.6 Kbps line. This budget drops to 50 cycles if the packet is transmitted on an STS-12 line (SONET synchronous transport signal level corresponding to 622 Mbps).

The implications of this cycle demand are clear. First, the switching functions should be implemented either in hardware or using optimized picocode. Second, the high-speed networking architecture must minimize both storage and processing requirements for the switching nodes. As a result, there can be no hop-by-hop error recovery (at each node in the path) or flow control. Errors are recovered end-to-end, and rate-based flow control is used. The architecture includes bandwidth reservation along the route of the connection prior to enabling the connection, access control at the port level, and simple congestion control at the switching nodes. Third, considering the general trend of increasing call-processing capabilities, the rate-based flow control satisfies the additional requirement of being able to distribute the control information, in a fast and efficient way, to every control point in the network.

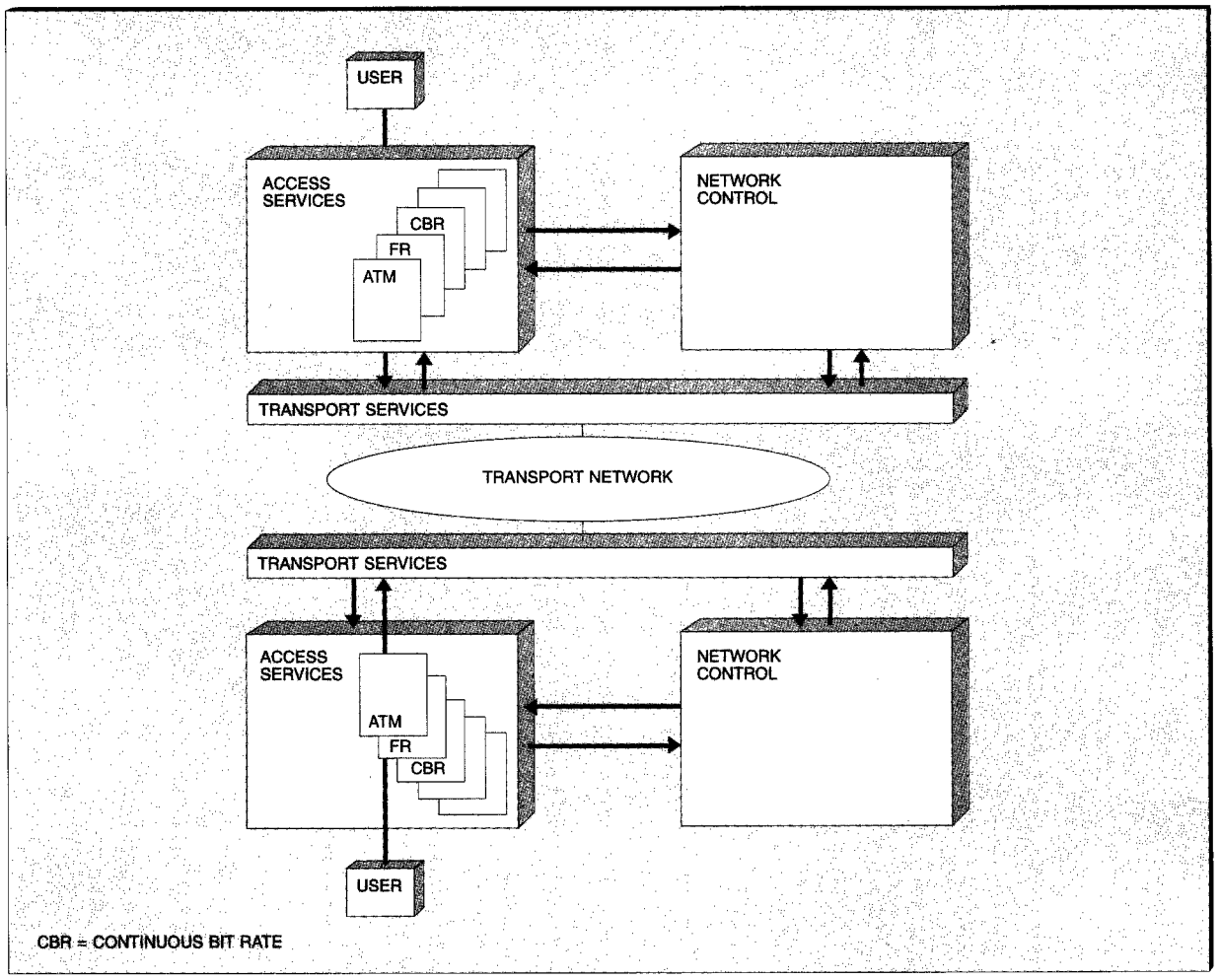
Networking BroadBand Services architecture. The Networking BroadBand Services architecture[1] addresses the challenges raised by the networking paradigm shift, and defines three functional components (Figure 2): network control, access services, and transport services.[2,3]

Network control includes classical transport network functions such as initialization, directory services, route computation, and topology database, and new functions required by the high-speed environment such as bandwidth management, congestion control, and fast distribution of control information using a powerful distributed spanning-tree algorithm.

Access services translate between the NBBS protocols and the user link protocols. At connection setup, the initiating user provides access services with the remote user name and the connection characteristics (peak rate, expected utilization and burstiness, and required QOS). Access services locate the remote user within the network and establish a network connection with reserved bandwidth that satisfies the required QOS. The amount of bandwidth that is reserved, called the connection equivalent capacity, is computed from the traffic characteristics (peak rate, average rate, burstiness), from the QOS, and from the network system parameters.[4]

The network connection is assigned a priority that matches the required QOS, and every packet of this connection will be marked with this priority, so that

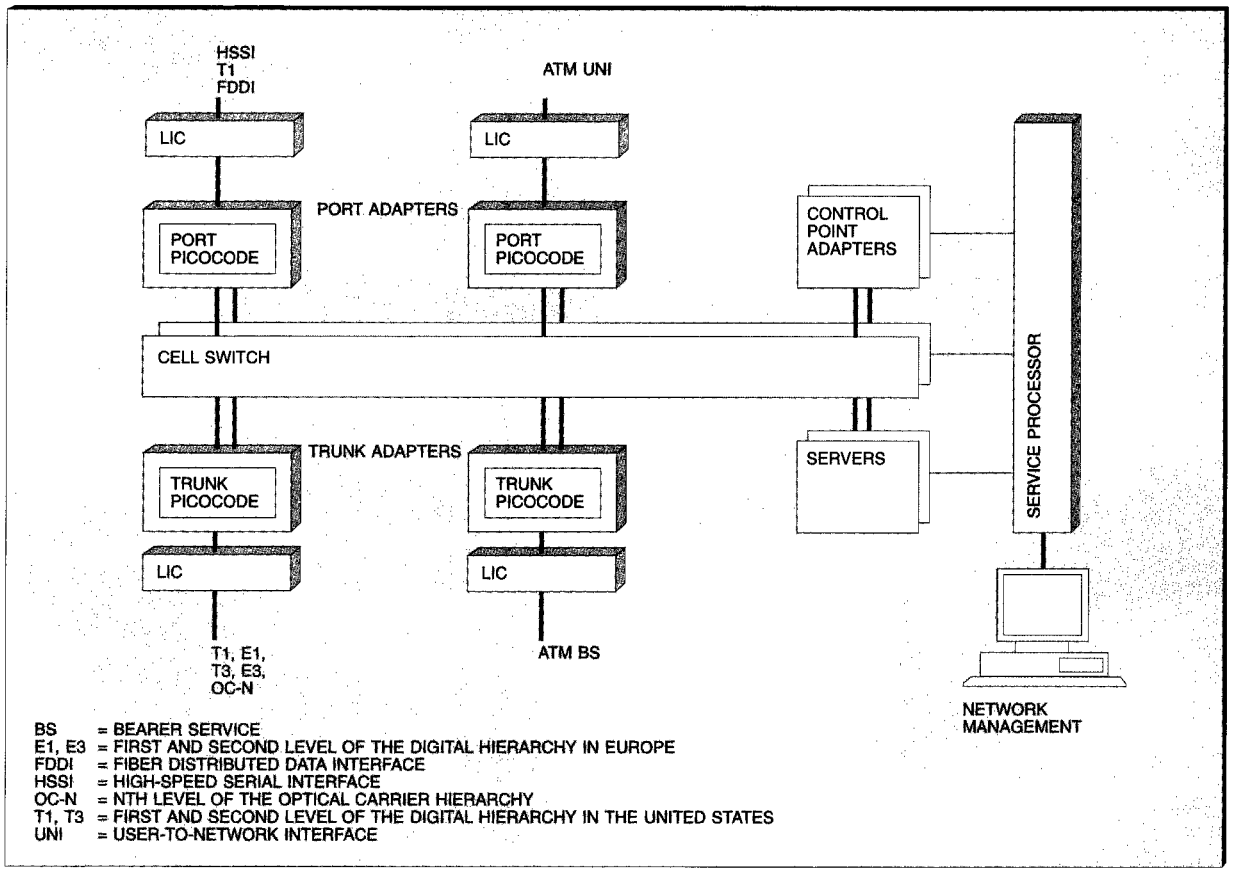
Figure 2 Networking BroadBand Services architecture



it can be scheduled appropriately at each switching node on its route. These connection setup functions are implemented in conjunction with the directory services, route computation, and bandwidth reservation functions of network control. Once a new connection has been accepted, some control must be enforced at the entry node[5,6] in order to protect the network and the other users from any excess traffic over the negotiated bandwidth granted to the user. This control is implemented in the "leaky bucket" algorithm, which actually smooths the incoming traffic according to the declared characteristics. Traffic in excess of the negotiated maximum is assigned a low priority for further selective discard in transit nodes.[7] Also implemented at the access node is the traffic monitoring that allows the bandwidth reservation to be automatically adjusted as the characteristics of the connection change. This important feature will be described later.

Transport services ensure end-to-end data transport through the most appropriate routing mode. ATM can be used natively or through an ATM adaptation layer (AAL). Audio and video conference applications are enhanced by multicast support. Transport services also support nondisruptive path switching, which allows a network connection to be rerouted in case of trunk or node failure, and path preemption, which can be used to preempt

Figure 3 Generic broadband switching platform



established low-priority connections upon failing to otherwise establish high-priority connections in heavily loaded network conditions.

Transport network node

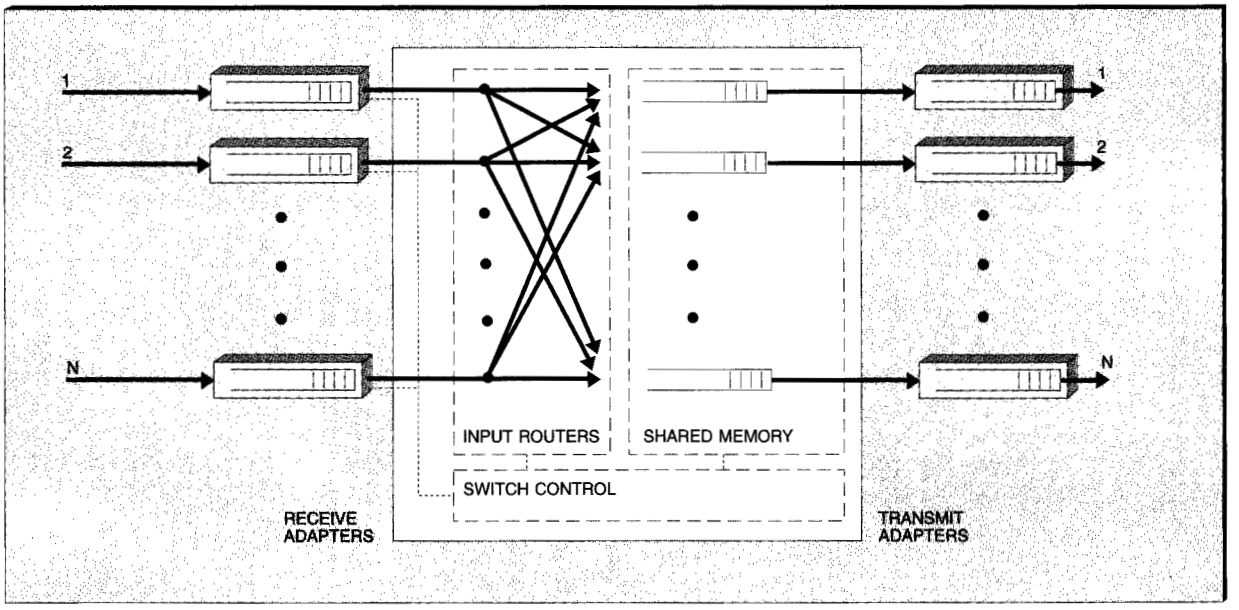
The relevance of an architecture is best demonstrated by showing its applicability to an implementation meeting the requirements of the broadband environment addressed earlier. The following sections show how the transport network node utilizes this architecture to provide efficient multiprotocol services in a broadband environment. In addition, these sections describe the technology of the TNN platform, which enables IBM to provide these multiprotocol transport services with high performance and cost effectiveness. We discuss how the general concepts apply to the provision of transport network functions for a variety of interfaces.

Generic broadband platform. Figure 3 shows the general structure of the broadband switching platform, which is organized around two cell switches connected in parallel for reliability. There is an active switch and a standby switch, and switch-over operation is automatically triggered upon detection of switch failure.

The port lines (user side) and the trunk lines (network side) are attached to line interface cards (LICs), which are connected to the switch through trunk-port adapters (TPAs). There is a different LIC for each type of line and, as will be described below, the TPA adapter is software configurable to handle either a port or a trunk, or both. For reliability, each TPA is connected to both switches.

The node control point is duplicated and is implemented on two LIC-less TPAs called control point adapters (CPAs). There is an active CPA and a

Figure 4 ATM switch



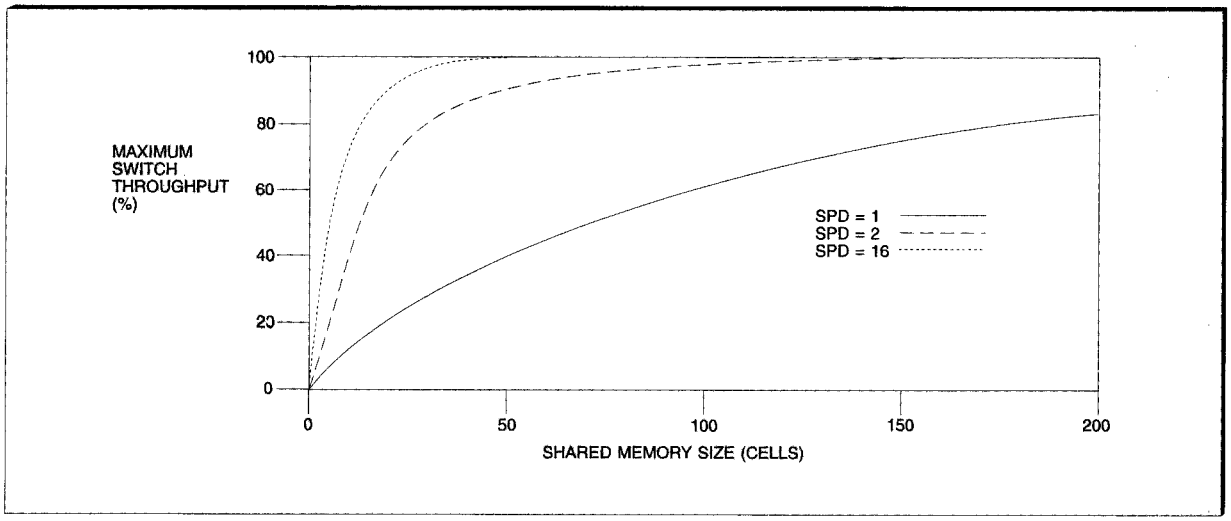
standby CPA, and switch-over operation is automatically triggered upon CPA-failure detection. Each CPA includes network control functions such as route computation, directory, topology, and spanning tree. Other network control functions such as bandwidth management and congestion control are implemented in the TPAs. Optional servers can be implemented on LIC-less TPAs to support IP, switched multimegabit data service (SMDS), LAN emulation, or voice services.

Finally, the service processor has the primary responsibility for controlling the operation of every component of the node. It manages a disk unit containing all the loadable software, configuration tables, error logs, initialization data, and other information. In addition, it is the point within the node that communicates with the network manager and the remote support center.

Switching technology. The switch is able to handle either ATM or PTM traffic. When a PTM packet arrives at a receive adapter, it is segmented into ATM cells, and the header of each cell is set to include the information required by the switch to route the cell. A cell can be forwarded to several transmit adapters by properly setting bits in the header, which will trigger the embedded multicast function of the switch. The transmit adapter receives the cells from the switch and is in charge of the reassembly process before transmission on a PTM line. PTM traffic to be transported on an ATM line is not reassembled at the transmit adapter, but transmitted as ATM traffic. ATM traffic is switched and transmitted natively.

Figure 4 shows the logical structure of the N×N switch, which is based on the architecture described by Denzel, et al.[8] The N input ports are connected to the receive adapters and the N output ports are connected to the transmit adapters. A 1-to-N router for each input provides full, contention-free connectivity to all output ports. The routing is determined by the destination address in the cell header. Each output queue is a logical queue with N inputs and one output. All N output queues are located in a block of shared memory where the space allocated to a specific output varies dynamically with the load of the switch. In fact, switch output queues are implemented through queues of pointers, with each pointer addressing a cell in the shared memory. This allows better memory utilization than can be implemented with fixed memory allocated to each output queue.

Figure 5 Switch throughput as a function of shared memory size and speed-up factor (spd)



For reasons of fairness, the maximum share of memory that can be allocated to each output must be limited to prevent a temporarily overloaded connection from using the entire memory space and degrading the throughput of other connections to other outputs. This policy is implemented in the control section of the switch. When the number of cells stored reaches a given threshold, a back-pressure mechanism sends a signal to the receive adapter, which stores the waiting cells in an input buffer. Actually, the TNN adapters implement four input buffers, one per transport priority supported in the network, and these buffers are scheduled for cell transmission to the switch according to their relative priorities.

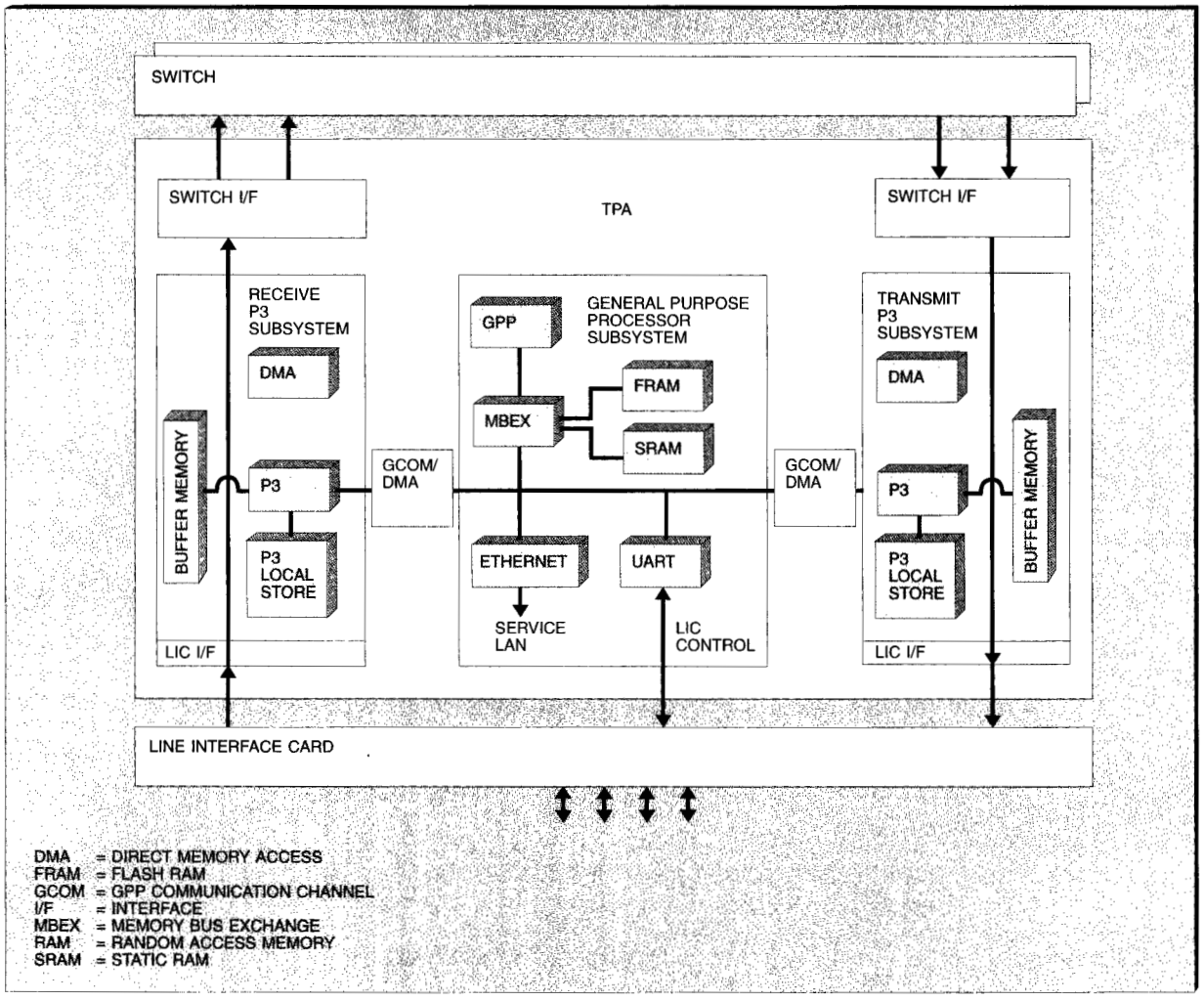
The trade-off between the shared memory size of the switch core and the speed-up factor, *spd*, which is defined by the ratio between the switch port rate and the line rate, has been extensively studied for both ATM and PTM traffic. For illustration purposes, Figure 5 assumes PTM traffic only (exponential distribution of packet size, with 500-byte average size), and shows for different speed-up factors (*spd* = 1,2,16) the switch throughput as a function of the core memory size expressed in number of cells. One can see that the maximum throughput can be reached, for fairly small shared memory sizes, with only a small speed-up factor. Note that switching ATM traffic with the same speed-up factor would require a much smaller shared memory.

The first model of the IBM 2220 Nways* BroadBand Switch[1] uses a 16×16 switch fabric implemented on a single chip, which is clocked at 266 Mbps and is able to switch up to 16 STS-3 lines. Higher range switches of the Nways family use the same fabric to offer much higher connectivity and speeds, by a simple combination of several switch fabrics, because of expansion capabilities (speed expansion, memory expansion, or port expansion).[8] The 16×16 switch chip includes circuitry that enables the designer to combine several such switches to build larger switches, or to increase memory or speed.

Trunk-port adapter technology. Figure 6 shows the general block diagram of the trunk-port adapter, which interfaces the lines and the switch. The adapter has been designed with five objectives in mind:

1. Support for media-speed packet processing up to STS-3 (155 Mbps) in the first version, and the ability to support higher rates with technology enhancements
2. The flexibility to adapt to still-evolving protocols and to support a wide range of existing or new protocols without hardware change
3. The ability to be configured for either trunk or port functions, which speeds up the development and reduces the costs
4. The ability to process either ATM or PTM traffic with little or no impact on performance

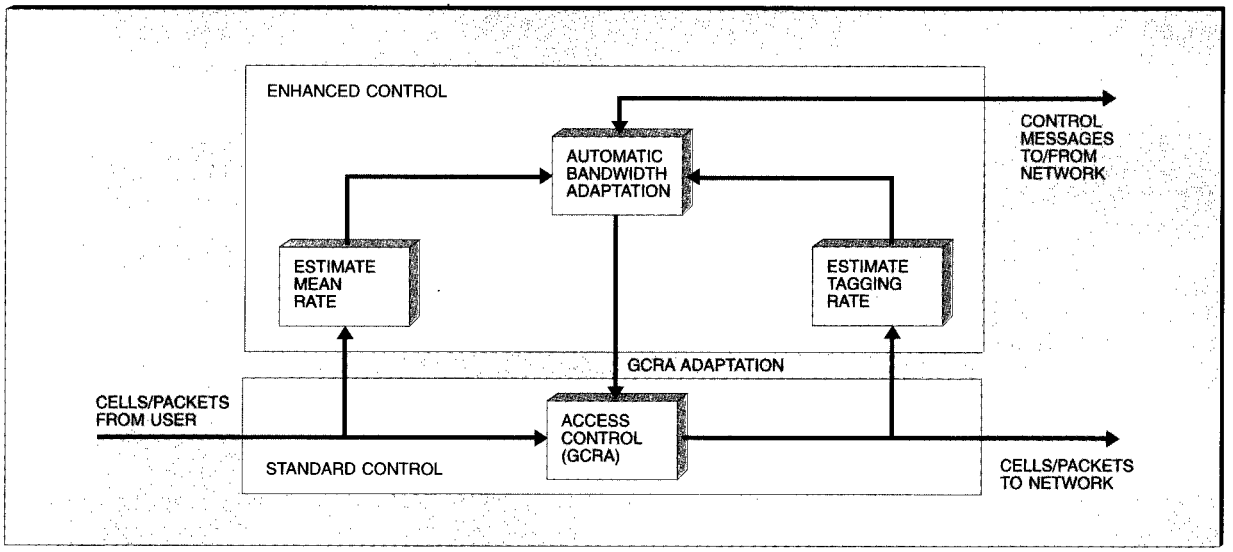
Figure 6 Trunk-port adapter (TPA)



5. The ability to offer enhanced services using standard interfaces

These objectives have been met through the design of a powerful parallel picoprocessor (P3). The instruction set has been defined to optimize all the packet-processing operations, such as segmentation, reassembly, queuing, and buffer management. For example, one can queue a packet or release a buffer in just a single machine cycle. In addition, arithmetic operations have been optimized for functions such as header processing, leaky bucket algorithms, buffer threshold processing, and spacing, and can be executed in parallel. One P3 subsystem is dedicated to each data path: one to the receive side from the line to the switch and another to the transmit side from the switch to the line. These are used for steady-state functions. In addition, the adapter includes a general-purpose processor (GPP) subsystem, which is used to implement control functions such as access control initialization and monitoring in the ports, updating of the connection tables and bandwidth management in the trunks, and statistics. In addition, the GPP controls the processor located on the LIC, and is connected to an Ethernet service LAN that links together all the GPPs of the switching node and the service processor.

Figure 7 NBBS access control



Enhanced port functions. We now show how the TPA design addresses another objective: the ability to offer services to the network operator of a service provider or a private network.

Figure 7 shows a simplified view of the access control that is implemented at the port adapter. The cells (or packets) are received from the user and processed by the P3, which implements the generic cell rate algorithm (GCRA) as specified by the International Telecommunication Union (ITU) and ATM Forum standards. The GCRA is a form of leaky bucket algorithm and is used to limit the traffic to the bandwidth that has been specified by contract with the user. According to the standards, compliant traffic is forwarded, while traffic in excess is tagged as discardable, or even discarded at the port.

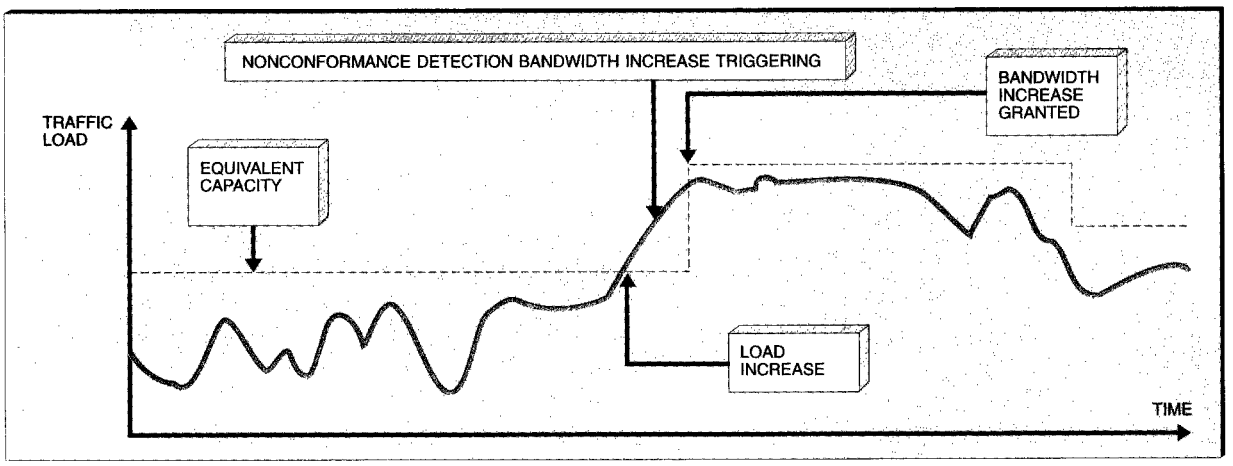
The TPA design makes possible two enhanced access control services on top of this standard interface. The first one, enabled by the large buffer memory (512 kilobytes) of the P3, provides traffic smoothing at high-speed ports. Packets or cells are received at line rate, then spaced to the network at a lower rate, further reducing the bandwidth requirement in the network.

The second service monitors the real user traffic and automatically adapts the connection according to the change in the traffic load.[9] This function is implemented in the GPP, and uses periodically sampled P3 counters to estimate the mean traffic rate and the probability of tagging traffic for discard for each connection. From these estimates, the bandwidth manager may make a decision to either increase or decrease the equivalent capacity that is reserved in the network for this connection.

For illustration purposes, Figure 8 shows the time variations of the load offered by a connection, and the corresponding evolution of the equivalent capacity. This function, which is provided in addition to the standard access control mechanism, is key in the bandwidth optimization of the network. It is also extremely helpful in sizing the initial requirements of the users and managing the evolution of the network topology accordingly.

This brief discussion has demonstrated that enhanced access control can be offered at the network ports because of the NBBS bandwidth management algorithms and the versatile design of the TPA port adapter. Similarly, functions at trunk adapters can be enhanced. For example, the large buffering available in the packet processor has been used to implement sophisticated queuing systems managed by priority schedulers, preempt and resume algorithms, and provide end-to-end flow control for the transport of available bit-rate traffic.

Figure 8 Automatic bandwidth adaptation



Access services. The TNN platform provides multiprotocol access for a number of interfaces, including ATM, frame relay, HDLC, and voice, using the concepts described in the NBBS architecture. In addition, the TNN hardware architecture makes the TPAs totally configurable. This allows a number of transport services to be provided to meet customer requirements. For example, TNN can offer two transport services for voice corresponding to different user requirements:

1. Clear channel service can be provided, where the bandwidth is reserved at peak level. This service is used to interconnect PBXs over the SPN or CPN. All the voice circuits are transported to the same destination either in ATM or in PTM mode, using circuit emulation to reduce the end-to-end delay.
2. Channelized service can be provided, where the voice circuits are transported to different destinations, either in ATM or PTM mode, using either circuit emulation to optimize the end-to-end delay or voice compression to optimize the bandwidth.

Performance. The adapter and switch performance depends on the type of traffic (packet-length distribution), on the kind of protocol (port or trunk adapter), and on the technology used. The call setup performance depends in addition on the network topology and on the route computation algorithms. Because of these multiple dependencies, we give a range of performance achieved by the platform.

The adapter throughput ranges from 200 000 packets per second for a frame-relay port adapter connected to a 50-Mbps access line to 365 000 cells per second for an STS-3 port or trunk. A 14×14 switch populated with STS-3 adapters can therefore sustain an aggregate throughput of 5.1 million cells per second in each direction.

With basic path-selection algorithms supported in the first release of the TNN platform,[1] the call-processing capability is 10 calls per second per node for a network size of 100 nodes. Advanced path selection algorithms and CPA technology available in later releases bring this figure to more than one thousand calls per second per node, whatever the network size.

NBBS software architecture

The NBBS software architecture was proposed to meet specific requirements on switch and network scalability, fault tolerance, and high performance:

1. Switch scalability: Up to hundreds of adapters would be supported. Adapters should be self-sufficient in terms of memory and MIPS.
2. Network scalability: Large networks would be enabled by access nodes and internetworking. A unique software platform would support all types of nodes.
3. Fault tolerance: The failure of a software or hardware component should affect only the failed component. For example, a failure of the

network control services should not impact the access and transport services. Moreover, the duplication of sensitive components would ensure fault tolerance.

4. High performance: The control point should be able to set up a connection in a very short time (a few milliseconds) and to achieve a very high call rate (hundreds to thousands of calls per second). In addition, each node should be able to support a very large number of simultaneous connections (several hundred thousand to millions of connections).

A distributed design direction was taken to meet these requirements, following the architecture direction. The implementation was split into three sets of services—control, access, and transport services—which can be located according to the performance and cost design points of the switch.

In large switches, the network control services are centralized in a control point adapter (CPA), which can be duplicated, while the transport services are distributed on the trunk adapters, and the access services are distributed on the port adapters. In medium switches, performance requirements may be lower and cost aspects more sensitive, so the network control services may be merged with the access services or with the transport services in a single adapter. In small switches all services are merged in a unique adapter.

In the following section, the NBBS control point software implementation is described with reference to Figure 9, which represents two nodes. Each node includes three adapters: one control point adapter that contains the network control services (topology and spanning tree, set management, and path selection), one port adapter that contains the access services (protocol agent, connection agent, and directory agent), and one trunk adapter that contains the transport services (trunk manager and transit connection manager).

Network control services. Network control services manage the network resources for the transport services and the access services, and therefore are logically centralized within each network node. They include four components, topology database and control point (CP) spanning tree, path selection, set management, and duplex control point support, which are now described.

Topology database and CP spanning tree. Each node contains a copy of the network topology database, which includes information, used to control the network operation, about the link characteristics, status, and bandwidth reservation level. This database is managed by the topology algorithm, which is designed to maintain a consistent view of the network in each node. The algorithm ensures that every node in the network acquires, and stores, the same picture of the network. Changes to the network topology database are triggered by changes in a link characteristic or state, or by significant changes in the bandwidth reservation level of a link, or when two subnetworks join.

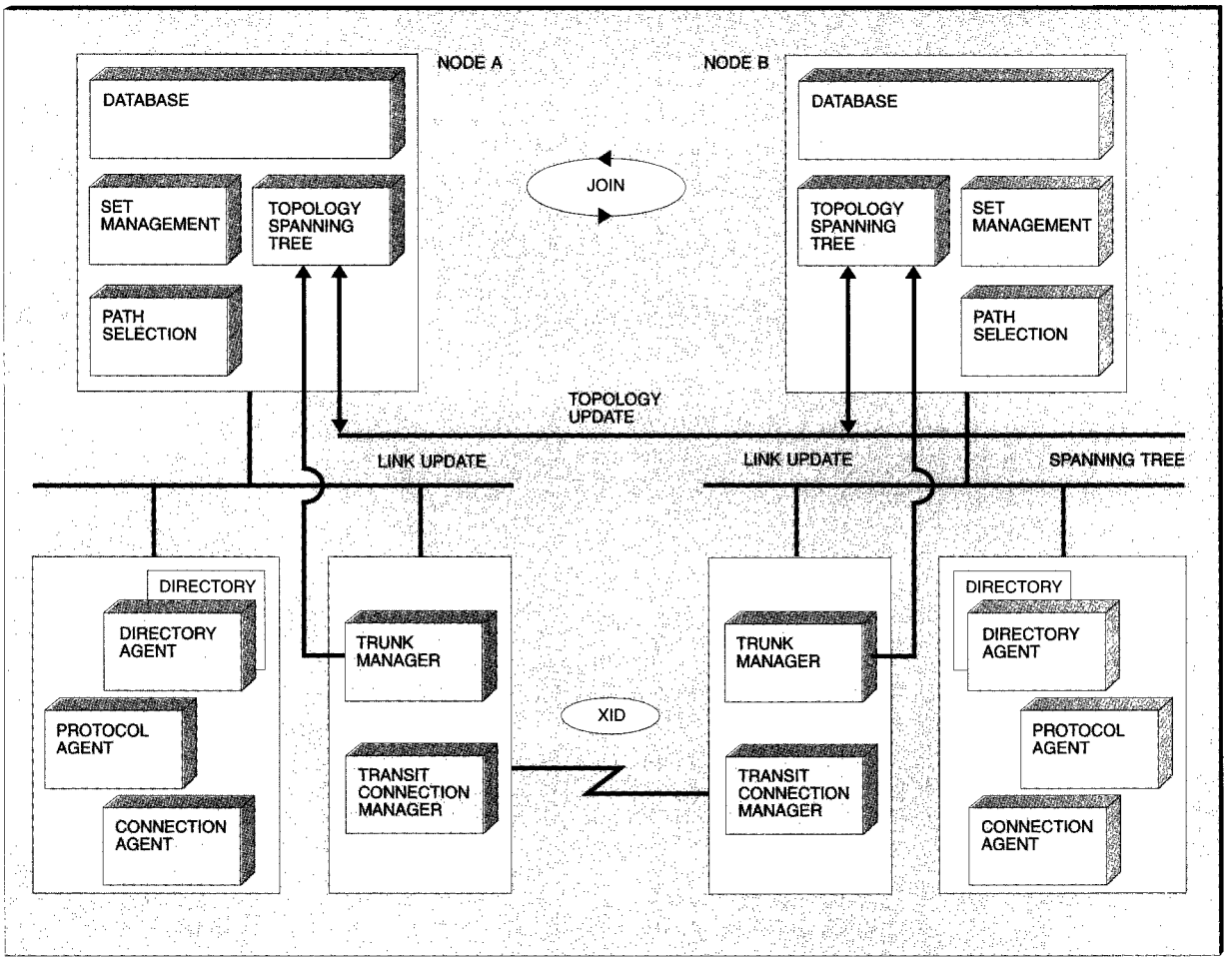
The topology algorithm works in conjunction with the CP spanning tree algorithm, which is distributed in every node and is used to maintain a tree at the network level. This tree links all the control points in the network and efficiently distributes network control messages using the multicast function of the switch. Using the CP spanning tree algorithm, the tree can automatically and rapidly reconfigure upon failure of one or more resources in the network, or upon setup of a new node in the network.

For example, Figure 9 shows a new node, A, being connected to a second node, B, that is already part of an operational network. When the new node is set up, the trunk between both nodes is initialized by the trunk exchange identifier (XID), as described later. As a result, node B receives a link update message, which is processed by the topology algorithm, which triggers an updating of the database. Then, the spanning tree algorithm starts a join procedure, which includes the exchange of the topology database and the definition of a new spanning tree that links all the nodes of the network.

Path selection. The path selection function determines the best way to allocate network resources to connections. It guarantees that user QOS requirements are satisfied and optimizes the overall throughput of the network. The path selection function supplies the requesting port with a path over which a point-to-point or a point-to-multipoint network connection will be established and bandwidth reserved if needed.

Inputs to the path selection function algorithm come from the client request and from the status

Figure 9 Control point spanning tree



of the network links as maintained in the topology database. The client of the path selection function server specifies the network connection metric, the QOS, and the routing mode requested.

1. The connection metric is defined by the triplet (R,m,b), where R is the peak rate, m is the average bit rate, and b is the average burst size of the connection.
2. The QOS specifications include the connection setup delay, the connection release delay, the connection security, the packet-loss probability, the transfer delay, and the maximum number of hops (connections between adjacent nodes).
3. The network connections use one of the supported routing modes: automatic network routing (ANR), used primarily for control traffic and datagrams, label swap routing for variable-length user data, ATM routing for native ATM user data, and tree routing for point-to-multipoint connections.

The computation of a path is based on the hop-by-hop exploration of the graph representing the network topology. The path selection function is a modified Bellman-Ford algorithm. It selects the path that supports the QOS requirements with as few links as possible, minimizing the amount of network resources involved in each connection.

In addition, the path selection function ensures load balancing of the network, giving all links about the same average utilization. This property optimizes rerouting time in case of link failure. Finally, the path selection function is optimized for performance by a routing table that is automatically built and maintained.[10]

Set management. Each node has a set management function for grouping resources. A closed set includes a predefined list of resources, an open set can accept new resources on request. Set management is used to efficiently implement applications like multiparty call, virtual LAN, video distribution, and network control (the CP spanning tree and distribution trees).

Duplex control point support. The switch supports both simplex and duplex mode of operation. In simplex mode a single CPA is installed and active; in duplex mode two CPAs are installed with one active and the other on standby. The duplex control point (DCP) function is implemented in each CPA. It detects CPA failures and triggers the switch-over from the active to the standby CPA when required. It is based on the following mechanisms:

1. Both DCPs periodically exchange their status through the active switch.
2. Each CPA tests its connection to the active switch by sending frames to itself.
3. Each DCP periodically exchanges messages with vital CPA software components. If the component is still alive, it sends back a message. If not, the DCP detects the failure.

After a switch-over, the centralized functions recover the state of the local links from the trunk adapters, reinitialize the access services in the port adapters, and execute the topology and spanning tree algorithms.

Access services. The access services[3] include three basic functions: the protocol agent, the directory agent, and the connection agent. These functions are distributed in each port adapter, increasing performance and reliability. Upon failure of the centralized functions, the port adapter continues to support the basic functions. Network connections established prior to the failure are not disturbed, and new call requests destined to this port can still be accepted. However, the port adapter cannot initialize new calls before the centralized functions have restarted.

Protocol agent. The protocol agent maps the standard protocols, like frame relay, HDLC, or ATM that are used by external resources, to NBBS protocols. It collaborates with the packet processor for data relay and for bandwidth management functions like GCRA initialization (see Figure 7).

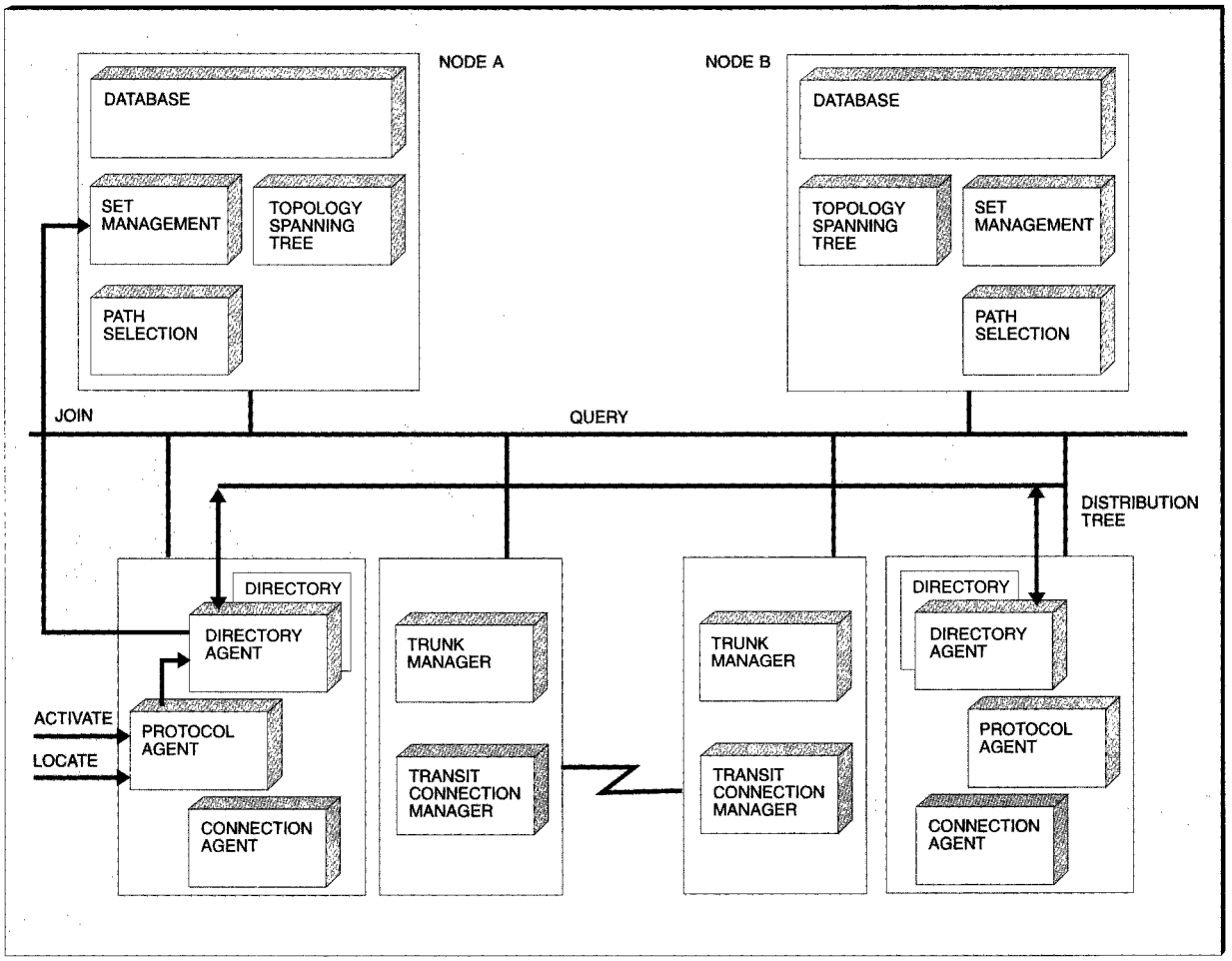
Directory agent. The directory agent locates an external resource connected to a network, first finding the node through which the external resource can be reached, and then getting the characteristics associated with the resource (availability and connection support capability, for example). Each network node has a directory agent that maintains a local directory database containing information about external resources owned by the access agent on the node. The database also contains information about external resources that have been already queried.

As shown in Figure 10, a request from a protocol agent can trigger a directory agent to send a query message on a distribution tree and to use the received information to update a set (join operation). To minimize the overhead introduced by unsuccessful queries into the network, directory agents are organized in sets using the set management function. For example, there is one set for frame relay ports, one set for ATM ports, etc.

To illustrate, suppose that at node A, located in France, a request is received to place a telephone call to the United States. The protocol agent at node A receives the request, with the phone number, and asks the directory agent at node A to find the address of the node connected to the PBX owning this number. The directory agent does not know the address, so it sends a query message, with the phone number, to other directory agents. The message is sent over a distribution tree, multicasted only to nodes connected to external PBXs. The directory agent in node B, connected to the PBX in the United States owning the requested number, will recognize the number and send back the address of node B to the directory agent at node A. The directory agent at node A then adds the address of node B, with the requested phone number, to its own directory.

Connection agent. The connection agent initializes and maintains network connections. As shown in Figure 11, the connection agent receives from a protocol agent a connection setup request, which includes the origin and destination addresses, the connection metrics, and the quality of service. It

Figure 10 Directory



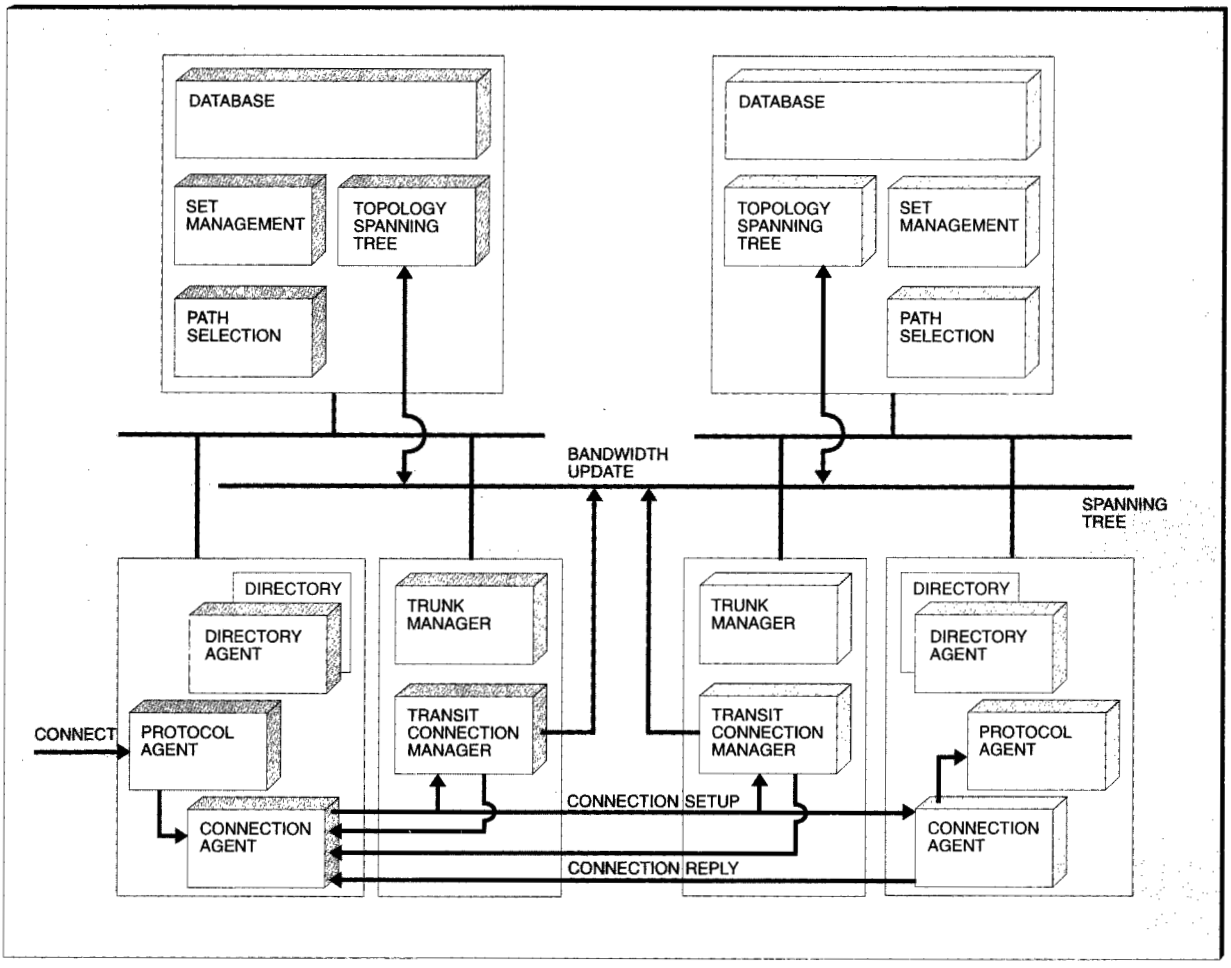
requests a path from the path selection function, then manages the connection setup flows from the connection agent at the node originating the connection, to the connection agent at the destination node, through the transit connection managers in each transit node.

The connection agent was designed for fast connection setup. A unique message is sent on the path and is copied at each transit node for resource allocation. After proper processing, including bandwidth and label reservation, the connection manager at the transit node and the connection agent at the destination node reply to the connection agent at the origin node, which activates the connection. Finally, each connection agent is responsible for rerouting the connections it owns. Rerouting can be triggered either by the topology algorithm upon link failure detection, or by the transit connection managers upon path preemption.

Transport services. The transport services are implemented in each trunk adapter. There are two basic functions, the trunk XID and the transit connection manager. Again, these functions have been distributed in each trunk adapter for performance, scalability, and reliability.

Trunk XID. The trunk XID function connects directly to the hardware of the trunk. It includes a simple finite state machine that initializes the trunk

Figure 11 Connection setup



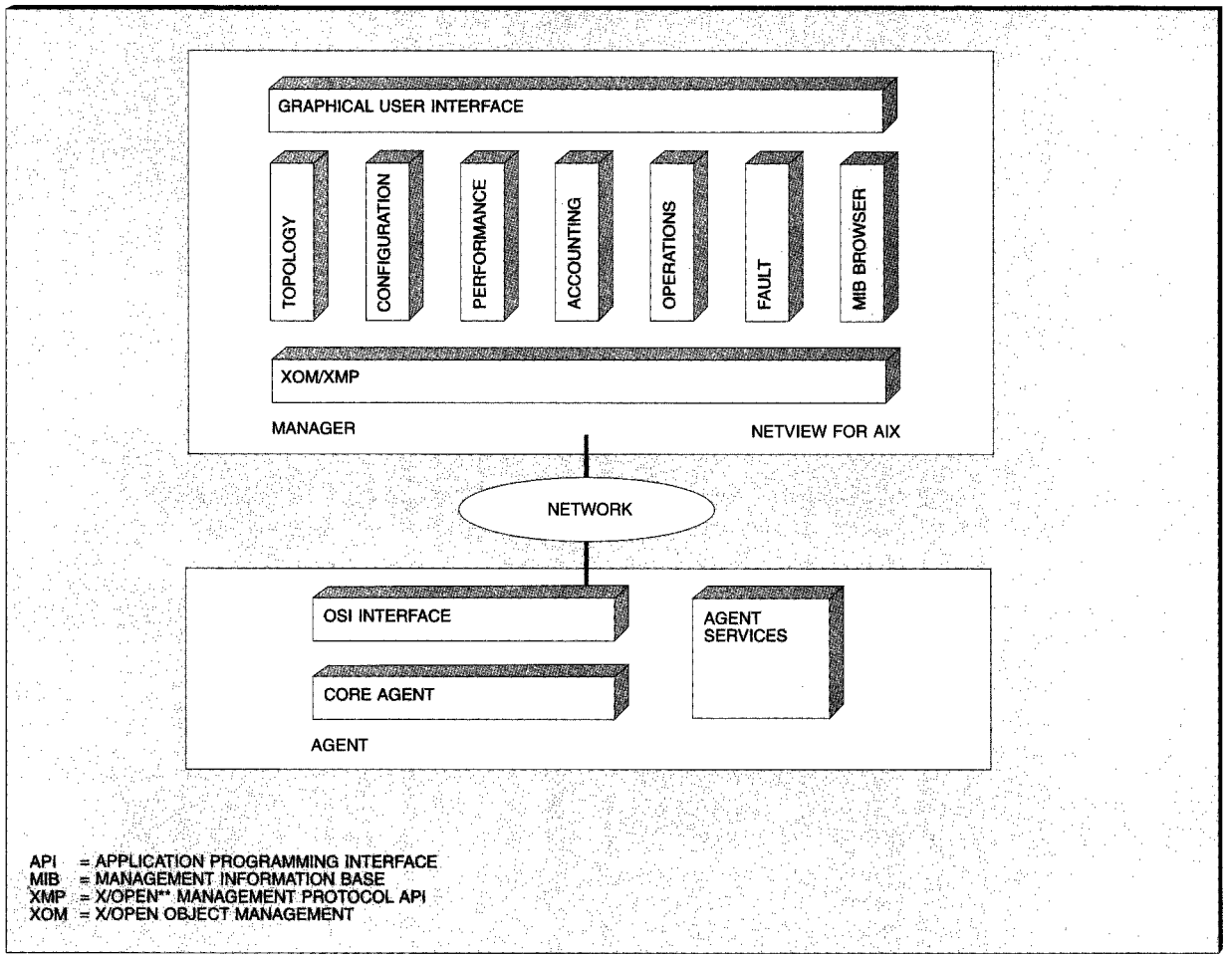
and monitors its state. Changes in the trunk state may happen as a result of a network control command from the network operator, or an event occurring at physical line level (for example, a link failure), or expiration of a time-out in the liveliness message protocol (indicating the failure of a software component). A message is then sent on the link. Both ends of the link get knowledge of their partner characteristics and state. All changes in the characteristics or state of the trunk are reflected in the topology database.

Transit connection manager. The transit connection manager is the partner of the connection agent in each transit node on the path of a connection setup. It manages the local resources (bandwidth and labels), warns the topology database when a link bandwidth threshold is reached, and triggers path preemption of lower priority connections to accept a higher priority connection request.

Network management

Network management is the set of monitoring, reporting, and control tasks that are required to operate a network. The need for compliance to standards clearly dictated the choice of Common Management Information Protocol (CMIP) and Simple Network Management Protocol (SNMP) for internal communication. The need for performance optimization led to a distributed model including agents and managers (Figure 12). The service pro-

Figure 12 Network management model



cessor of every switch implements the network management agent, while the network manager is implemented on a powerful centralized platform.[11]

The network agent is based on an object-oriented model, and includes an Open Systems Interconnection (OSI) interface, a core agent, and the agent services. This splitting allows the core entity to communicate efficiently and in a generic manner with its interfaces. The OSI interface is the entity that takes care of all the peculiarities of the OSI management model—filtering, naming-tree maintenance, and syntax conversion from ASN1/BER (Abstract Syntax Notation 1/Basic Encoding Rule) to a more exploitable format.[12] The core agent contains the set of all the managed object instances that communicate on one side with the physical resources and on the other side with the manager through the OSI interface. The agent services contain all the generic services needed by the core agent and provide basic functions such as startup, shutdown, and task supervision.

The network manager is based on an event-driven model, which means that a specific process is registered for every event the application is interested in. There are three types of events: user events, network events, and internal events. User events are triggered on user request, and result in an action on the managed objects in the network (creation, deletion, configuration setup, activation, deactivation, or status), or with a physical or a logical

resource in the network. Network events are generated by the responses associated with the requests sent, and by the agents when unattended events, called event reports under CMIP, are detected. Internal events are generated by a network management application and are to be handled by another network management application.

The network management applications are implemented under Netview* for the Advanced Interactive Executive* (AIX*). They include a management information base browser and functional tasks that manage topology, configuration, operations, performance, accounting, and faults (Figure 12). These components are discussed in the following subsections.

Topology management. Topology management is responsible for building and maintaining the display of the network. The status of each switch and of each trunk can be displayed on a graphical user interface. Any change in the network is discovered and reported to the operator console. Figure 13 shows the NBBS screen that can be obtained from the standard Netview/6000* root map screen. Here M8, M9, and M11 represent nodes in an NBBS network named USIBMNR. A special icon indicates that M9 is the network gateway (the node to which the network manager is attached).

Configuration management. Configuration management provides mechanisms for displaying and modifying the configuration of any managed object in the network. It also offers a way to create new instances of a managed object class. The switch configuration management supports four kinds of objects: trunks, ports, connections and their quality of services, and hardware FRUs (field replaceable units) such as adapters, LICs, and physical lines.

Figure 14 shows the screen that appears when the network operator selects M8, representing an Nways 500 switch, from the screen shown in Figure 13. It represents a shelf of the switch, with twelve slots. Eight of the slots can be populated with switch adapters and four with a combination of switch cards, alarm and power control cards, control point adapters, and clock cards.

On this screen, slots 1, 2, 4, 5, 7, and 10 are populated with trunk-port adapter (TPA) cards. High-speed adapters (HSA1, HSA2) attach high-speed lines (T3, E3, HSSI), and low-speed adapters (LSA1, LSA2) attach low-speed lines (T1, E1, etc.). Unlabeled slots 3, 6, 8, and 12 are not populated. Behind the TPA slots are line interface cards. Slot 9 is used by a switch (SWI) card and slot 11 contains the alarm and power control (APC) card. Although not shown here, the front of slots 9 and 11 can be populated by clock cards.

The icon above each populated slot represents the function of the card in that slot. On the right are icons representing fans and a power source. The network operator can click on the icons to configure, operate, or request status for elements of the switch.

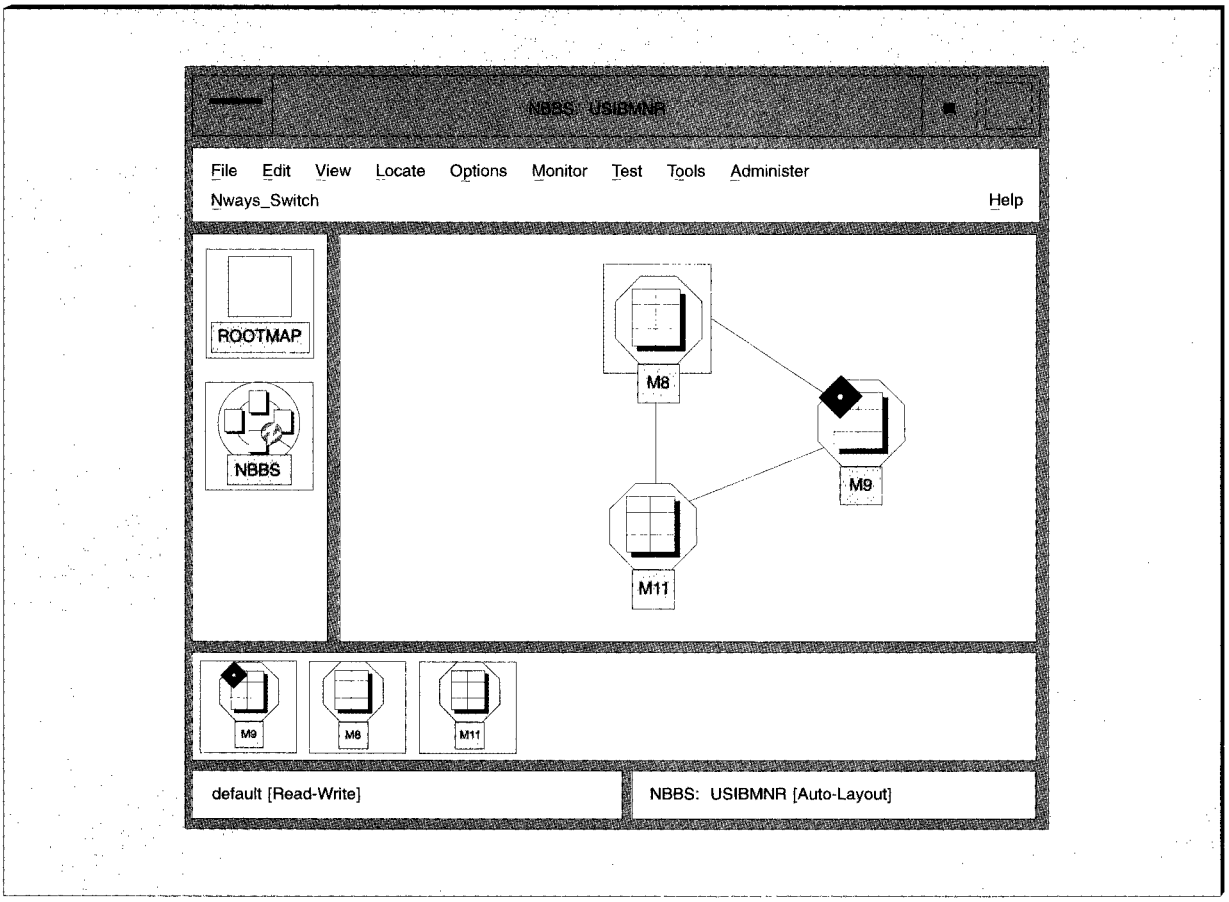
Operations management. Operations management provides a way to control each managed object instance in the network. The operator can display the status of the object, activate or deactivate the object, or lock or unlock it. The operator can also display and change the path of any activated connection.

Performance management. Performance management is the process of quantifying and measuring the responsiveness, availability, and utilization of a network. The performance manager provides the operator with functions to monitor, analyze, and control any physical or logical entity in the network. For example, it is possible to automatically include a resource in a list for further monitoring when the topology algorithm discovers it, and to automatically remove the resource from the list when the topology algorithm no longer manages it.

The operator can decide whether or not a newly discovered resource should be monitored, and update the monitoring profiles. For example the operator can select the counters and thresholds to be monitored, or define customized values, like counter thresholds, or set up thresholds for every monitored parameter and ask for the generation of an audible signal or a logging event upon reaching the threshold. The operator can also display all collected and derived data graphically or store them in a file for further processing.

Accounting management. Accounting management collects and processes data so that end users can be charged with the cost of using the network. The accounting is done on a connection basis, and includes start and stop times, endpoint names, the number of bytes, packets, and frames that have been transported, and the connection QOS specifications.

Figure 13 Topology



In addition to accounting operations, recorded data can be further processed by customized applications. For example, one can check the requested traffic characteristics versus the actual traffic usage, or determine the cost of usage by time of day, class of service, route taken, and bandwidth used.

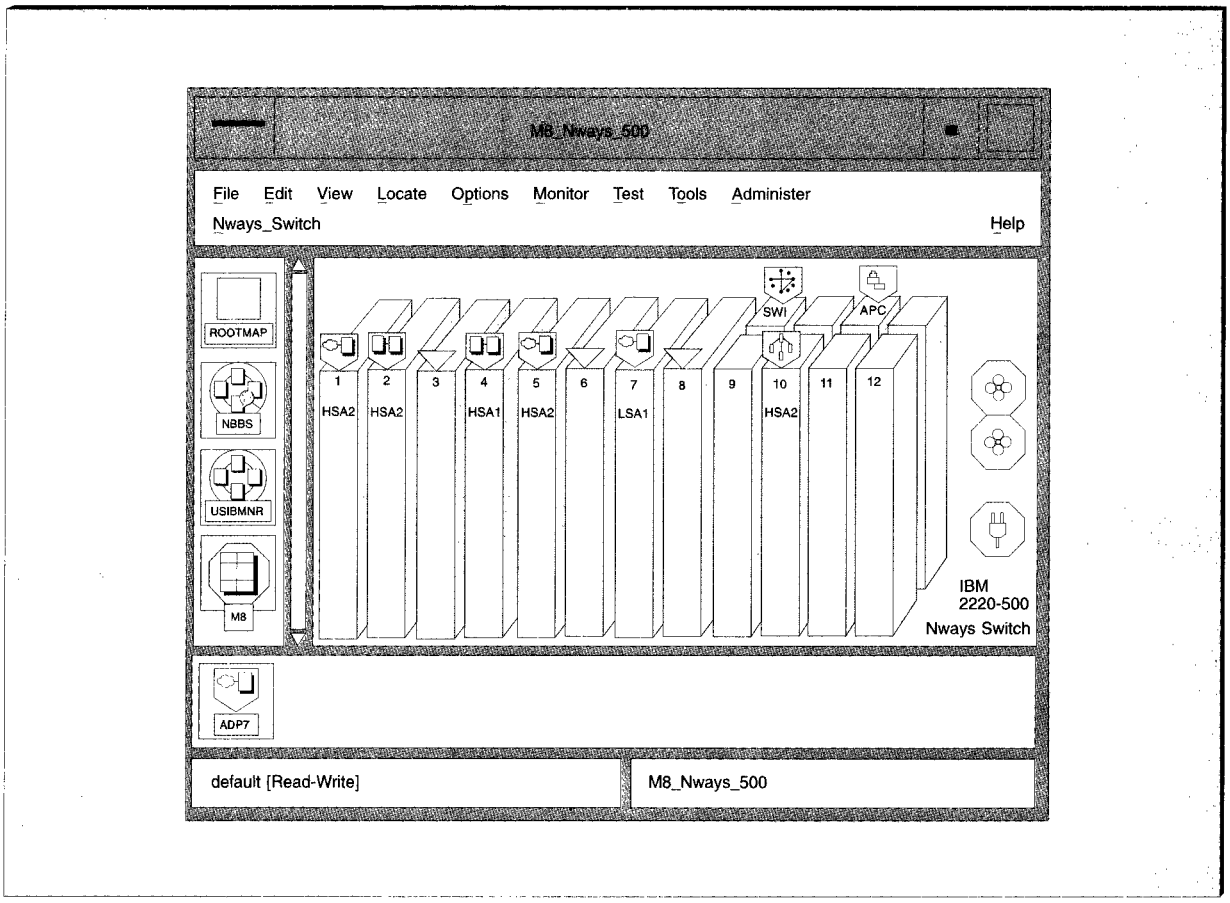
Fault management. Fault management detects and reports unexpected network behavior. The fault management application displays CMIP event notifications sent by the agents, such as notifications of object creation or deletion, changes of attribute values, and various alarms. These events may be filtered by the network management agent to minimize the network management traffic and by the network manager to select an appropriate action, for example the display of the event on the console.

Management Information Base (MIB) browser. The MIB browser displays each network resource as a CMIP instance of the network management agent via a graphical or a textual interface. With a graphical interface, the operator can trigger any operation on instances and their attributes by dragging the instance icon to the selected operation icon. With a textual interface, the operator keys in the operations. The MIB browser also provides a way to record the selected operation in a user file which can be further processed.

Conclusion

In this paper we have described the architecture and the technology of the new IBM transport network node, which supports both the native format for existing protocols and the emerging format for virtu-

Figure 14 Shelf configuration



ally all new communication equipment. We have shown that the paradigm shift of the broadband networking environment raises major challenges. It requires the transport, on the same physical layer, of multiple data streams, with bandwidth, burstiness, and QOS characteristics in a very wide range, and the optimization of network resources for such mixed traffic. These new challenges have been addressed by a powerful hardware platform featuring three key technologies:

1. A modular switch technology that offers high performance at low cost, and that can be exploited in different design points to cover the full range of link speeds and connectivity required by service providers and private network applications
2. A versatile adapter design that is used for port, trunk, server, and control point functions, and that enables enhanced functions on top of standard interfaces
3. A new architecture for broadband networking services, featuring a unique bandwidth management approach for optimization of the network resources

The transport network node platform can be used either by service providers or by private network operators to efficiently support standard services like voice, HDLC, frame relay, and ATM bearer services, on a full range of standard interfaces. The IBM Nways 2220-500 is based on this platform and can switch up to 16 STS-3 lines fully loaded. It is the middle-range model in a family of switches, ranging from low-cost access nodes to high-range switches offering much greater connectivity at STS-48 rate.

*Trademark or registered trademark of International Business Machines Corporation.

**Trademark or registered trademark of X/Open Co. Ltd.

Cited references and notes

1. *IBM 2220 Nways BroadBand Switch: Concepts and Products,* GG24-4307, IBM Corporation (1994); available through IBM branch offices.
2. M. Peyravian, R. Bodner, C.-S. Chow, and M. Kaplan, "Efficient Transport and Distribution of Network Control Information in NBBS," *IBM Systems Journal* **34**, No. 4, 640–658 (1995, this issue).
3. C. P. Immanuel, G. M. Kump, H. J. Sandick, D. A. Sinicrope, and K. V. Vu, "Access Services for the Networking BroadBand Services Architecture," *IBM Systems Journal* **34**, No. 4, 659–671 (1995, this issue).
4. R. Guérin and L. Gün, "A Unified Approach to Bandwidth Allocation and Access Control in Fast Packet-Switched Networks," *Proceedings: IEEE Infocom '92,* Florence, Italy, pp. 1–12.
5. An entry node may be an NBBS access node.
6. N. Budhiraja, M. Gopal, M. Gupta, E. A. Hervatic, S. J. Nadas, P. A. Stirpe, L. A. Tomek, and D. C. Verma, "The NBBS Access Node," *IBM Systems Journal* **34**, No. 4, 694–704 (1995, this issue).
7. NBBS network nodes are often referred to as transit nodes in this paper, where the transport capability of these nodes is the emphasis.
8. W. E. Denzel, A. P. J. Engbersen, and I. Iliadis, "A Flexible Shared-Buffer Switch for ATM at Gb/s Rates," *Computer Networks and ISDN Systems* **27**, No. 4, 611–624 (January 1995).
9. T. Tedijanto and L. Gün, "Effectiveness of Dynamic Bandwidth Management Mechanisms in ATM Networks," *Proceedings: IEEE Infocom '93,* San Francisco, CA, pp. 358–367.
10. T. E. Tedijanto, R. O. Onvural, D. C. Verma, L. Gün, and R. A. Guérin, "NBBS Path Selection Framework," *IBM Systems Journal* **34**, No. 4, 629–639 (1995, this issue).
11. S. A. Owen, "NBBS Network Management," *IBM Systems Journal* **34**, No. 4, 725–750 (1995, this issue).
12. D. E. McDysan and D. L. Spohn, *ATM Theory and Application,* McGraw-Hill Inc., New York (1995).

Accepted for publication May 25, 1995.

Gerald Lebizay *CIE IBM-France, Le Plan du Bois, 06610 La Gaude, France (electronic mail: lebizay@vnet.ibm.com).* Mr. Lebizay graduated in 1961 from the Brussels University Polytechnicum School of Engineering. He became part of IBM in 1962 in its then newly created La Gaude Laboratory in the south of France. Mr. Lebizay has been a pioneer of telecommunication technology from packet-switching networks to microcontroller, signal processor, digital PBX, high-bandwidth switch, Systems Network Architecture FEPs, and recently ATM switches. Mr. Lebizay is currently System Design Manager of IBM Nways ATM switch products.

Claude Galand *CIE IBM-France, Le Plan du Bois, 06610 La Gaude, France (electronic mail: galand@vnet.ibm.com).* Dr. Galand graduated in 1974 from Nice University, where he received the Ph.D. in electronic engineering and the State Doctorate of Sciences. He joined IBM in December 1976, at the La Gaude Laboratory, where he occupied various positions in research and development in speech and signal processing and in broadband networking. He is currently in charge of the architecture of IBM Nways ATM switch products.

Denis Chevalier *CIE IBM-France, Le Plan du Bois, 06610 La Gaude, France (electronic mail: dchevalier@vnet.ibm.com).* Mr. Chevalier graduated from Marseille University, France, where he received the DEA degree in computer sciences and mathematics. He joined IBM France in 1970. He has worked as a software developer and designer, then as development and system design manager of several telecommunication networking projects in the IBM La Gaude Laboratory: PBXs, SNA switches, and recently ATM switches. Mr. Chevalier is currently software development manager of the IBM Nways ATM switch products.

Frederic Barre *CIE IBM-France, Le Plan du Bois, 06610 La Gaude, France (electronic mail: barre@lgevm2.vnet.ibm.com).* Mr. Barre graduated from Marseille University, France, where he received the DEA degree in computer sciences and mathematics. He joined IBM France in 1984. He has been working as a software designer and development manager in several telecommunication and networking projects in the IBM La Gaude Laboratory. Since April 1995 he has been manager of network management application system design, planning, and strategy for the IBM Networking Hardware Division.

Reprint Order No. G321-5590.