

Multiprotocol networking—a blueprint

by M. L. Hess
J. A. Lorrain
G. R. McGee

IBM's Open Blueprint™ is used as a framework for discussing recent developments in communications with particular emphasis on multiprotocol networking. The major parts of the Open Blueprint that pertain to networking are described, indicating advancements in user networks, networking standards, and their underlying technologies. Asynchronous transfer mode is shown to have special significance both for today's multiprotocol networks and tomorrow's multimedia networks.

The phrase “rapid pace of change” has become a cliché in the last decade of this century. However, the field of networking is truly being transformed by dramatic technological advances in *both* computing and communications. Enormous changes in computing price and performance, in the speeds of communication links, and in bit-error rates are fundamentally altering the underlying design assumptions for tomorrow's networks. In addition, advances in network protocol design and network management are allowing networks to be deployed across an ever-increasing variety of media and topologies while enjoying dramatically lower costs of administration and management.

Although the underlying technology is changing, a rich legacy of networked application software remains “mission-critical” for today's enterprises. Many of these application investments are tied to a particular networking protocol and remain so, despite convincing arguments advanced in the “protocol wars” waged in the trade press. It is not unusual in today's enterprise to find a variety of networking protocols such as Systems Network Architecture (SNA), Transmission Control Proto-

col/Internet Protocol (TCP/IP), and DECnet*. Thus, it has become part of the modern paradigm not only to seek industry-wide standards for new protocols but also to accommodate the old through convergence and coexistence within unified networks.

Networking protocols are being deployed across an ever-growing assortment of media and carrier services. On-campus (i.e., the geographic and physical facilities of an institution or business) media include coaxial cable, telephone twisted pair, fiber-optic channel, fiber-optic local area network (LAN), infrared, and radio frequencies. Each has its peculiar characteristics of speed, error rate, and distance supported. Wide area network media have also become more diverse. Carriers are augmenting traditional analog telephone facilities with many more sophisticated offerings, including X.25, Integrated Services Digital Network (ISDN), frame relay, and, most recently, asynchronous transfer mode (ATM). Cellular communications and other wireless schemes are also being offered in the wide area network context.

This diverse networking infrastructure is being used to support an increasing variety of applications, each with different requirements for security, integrity, bandwidth, response time, and dependability of service. Some examples are:

©Copyright 1995 by International Business Machines Corporation. Copying in printed form for private use is permitted without payment of royalty provided that (1) each reproduction is done without alteration and (2) the *Journal* reference and IBM copyright notice are included on the first page. The title and abstract, but no other portions, of this paper may be copied or distributed royalty free without further permission by computer-based and other information-service systems. Permission to *re-publish* any other portion of this paper must be obtained from the Editor.

- The everyday inquiry-response credit authorization at a local convenience store requires a low-bandwidth, dependable service.
- The more involved transactions of an airline reservation system require higher speed and greater dependability.
- On-line banking transactions require both integrity and security to protect against fraud.
- "Surfing" the Internet using Mosaic** requires little security or dependability but considerable bandwidth if the user is to enjoy the images available.
- Videoconferencing requires bandwidth and dependably low delay but can sacrifice some reliability in that the occasional lost frame does not compromise its value.

Few large networked systems are based on a single set of consistent technologies. Most have grown over time and include application programs and system components that are from different computing periods. The continued accommodation of such applications and hardware in today's systems is a tribute to the architectural principles that have been agreed upon and implemented across many systems.

Network architectures. A robust network architecture provides a framework in which hardware and software investments can be protected while new investments can take advantage of advances in technology.

Systems Network Architecture (SNA)¹ introduced the idea of a network architecture in 1974. Building on the layering concepts in operating systems, SNA provided a layering of networking function that allowed communications facilities to be shared across multiple applications. The layering of SNA was archetypal for the genre: communications application subsystems (such as the Customer Information Control System*, or CICS*) use an SNA session interface that isolates them from the end-to-end session pipes; these in turn are isolated by a data routing (or path control) layer that can route data across a variety of underlying subnets,² such as Synchronous Data Link Control (SDLC) multipoint, host channels, or X.25 networks. Distributing application program responsibilities into the workstation was fostered by the functions of the logical unit (LU); these functions evolved into the program-to-program protocols of LU 6.2³ which are uniquely adaptable to providing a high-level, cli-

ent/server interface, or to supporting purely peer communications.

Similar layering concepts evolved in other architectures, such as the Internet protocol suite⁴ and DECnet.⁵ The wide variety of architectures led to a quest for convergence to a single open standard. The Open Systems Interconnection (OSI)⁶ project in the International Organization for Standardization (ISO), begun in the late 1970s, attracted contributions from around the globe. It defined a model and vocabulary that are in common use today. Ironically, the breadth of worldwide participation in the OSI effort slowed consensus to a crawl. Instead of defining the preeminent standards for all layers, OSI has become only one of many sources contributing the constituent standards within an industry-standard infrastructure.

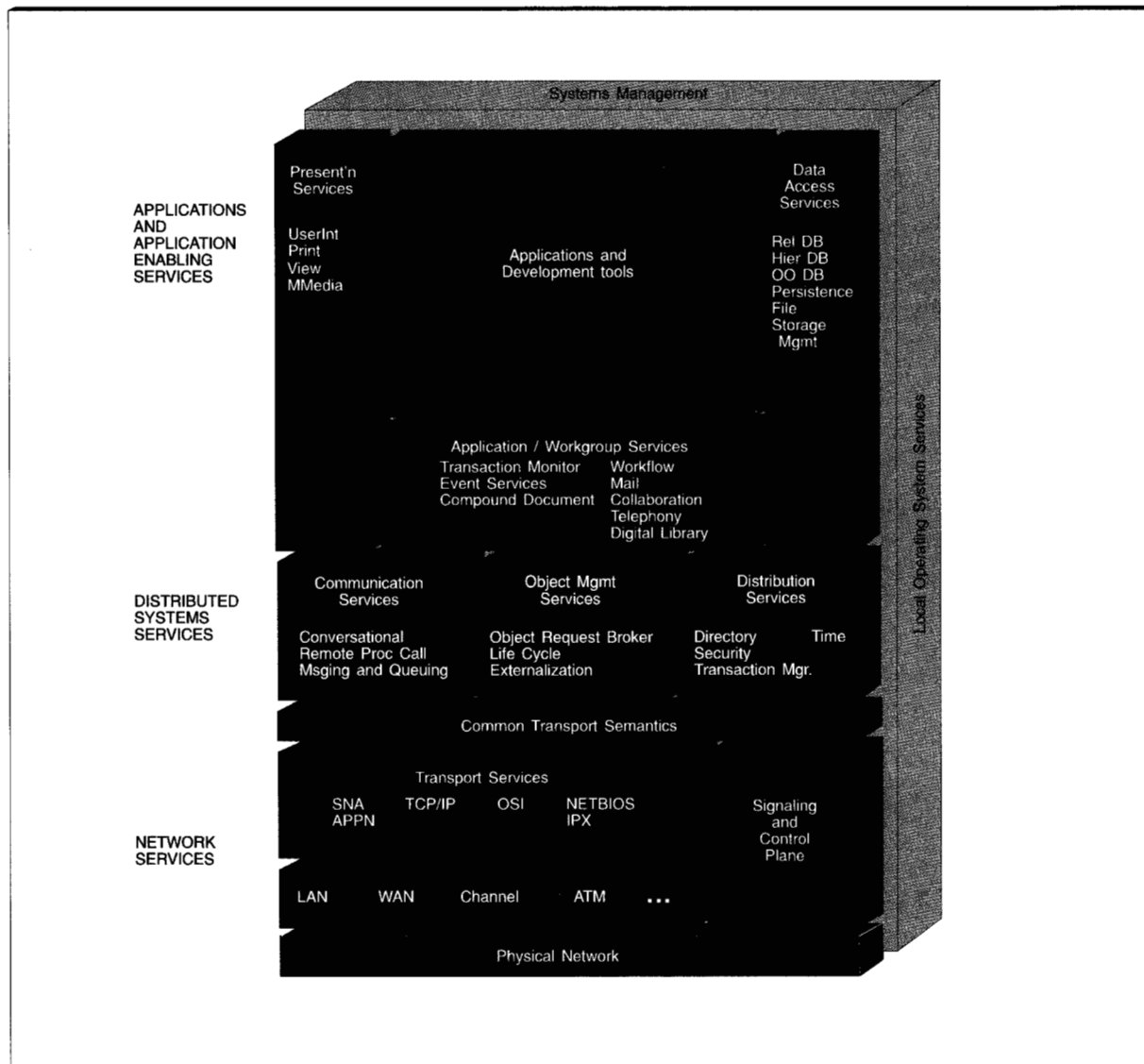
This infrastructure was captured in the Networking Blueprint⁷ that IBM announced in 1992 and has been elaborated with more application-level detail in the Open Blueprint*,⁸ introduced in 1994. The Open Blueprint, shown in Figure 1, represents the relationships of key functions and standards in the distributed computing environment. We use it, in this paper, as the framework for discussing the technological advances and trends in the various networking layers.

To appreciate the influence of this infrastructure, one need only consider the changing face of the networking industry in the last decade. Ten years ago, there were relatively few vendors, whose products rarely interworked. Today, many vendors, large and small, can supply pieces of the overall networking picture. Few provide a full range of products, but architectural consistency has enabled many niche vendors, such as those who specialize in adapters, routers, LAN servers, or middleware (application support), to become enormously successful. The risk of products being incompatible has been reduced to a level acceptable to many customers.

The Open Blueprint

We can relate the Open Blueprint to the OSI model, bearing in mind that the various layers of the Open Blueprint embrace a broader set of industry standards than are included in the OSI suite. The lowest layer matches the OSI physical layer. *Subnet-working* (as the term is used in the Open Blueprint)

Figure 1 Open Blueprint



corresponds to those parts of the lower OSI layers that contain functions dealing with specific communications facilities, as well as frame- and packet-handling formats and dialing procedures. *Transport networking* corresponds to the OSI networking and transport layers. The Signaling and Control Plane is not present explicitly in the OSI model but is derived from the signaling and control plane in the Broadband Integrated Services Digital Network (B-ISDN) model. *Distributed systems services*

correspond to OSI layers 5-7.⁹ The upper layers, involving applications and application-enabling services, embrace a number of industry standards, most of which are outside the scope of this paper, but could be viewed as falling within the domain of the OSI application layer (layer 7).

Each of these layer groupings is accessed via service interfaces that have progressed from being an architectural modeling convenience to being, in

many cases, dependable product interfaces. It is particularly true at the application programming interface (API) to communications. Here, the needs of the application environment are addressed by three well-defined APIs—conversational, remote procedure call, and messaging and queuing—which represent alternative distributed computing models.

Because of its importance and unique relationship to all the layers, *systems management* is shown as a back plane that touches all the layers. This position illustrates its function in monitoring and managing components in each of the layers in the front plane.

Since today's networks incorporate many different computers, offered by many different vendors over the span of multiple computing periods, the Open Blueprint must be inclusive rather than exclusive. It incorporates today's most common protocols, not just the program of work of a particular standards group. It is a pragmatic model reflecting marketplace realities. Typically, equivalent models from a standards organization do not include protocols or interfaces that are outside its program of work. Such exclusions can distort networking realities and obscure both the multiprotocol challenge and solution.¹⁰ In fact, the Open Blueprint provides a structure for choosing applications, networking protocols, and subnetworks more or less independently, and focuses attention on pragmatic solutions that allow a mix and match of these independent choices.

In the following subsections, we explore recent developments in each of the communications layers of the Open Blueprint (starting at the lowest layers and working upward) and highlight those of special significance to multiprotocol networking. Because of important developments in subnetworking, we will revisit that topic at the end of the paper.

Subnetworking. The two lowest parts of the Open Blueprint are called the *subnetworking* layer. It contains functions dealing with specific communications facilities such as local area networks, wide area networks, and channels.

Local area networks. Over the past decade, local area networks¹¹ (LANs) have been the fastest growing subnetworks. This observation is not surprising considering the high growth rate of the personal

computer industry in the same period. But the attractive attributes of LANs have also been an important reason for their growth. The communications medium is customer-owned and relatively inexpensive, yet offers high speeds and low error rates. Because it is a shared medium, it is inherently fully connected, so special routing protocols

The Open Blueprint incorporates today's most common protocols.

are not needed to communicate with other stations on the same LAN. Also because of the shared medium, it is simple to use broadcast techniques to locate resources, thereby considerably simplifying installation and administration. Some simple LAN-specific protocols such as IBM's NetBIOS* and Novell's IPX** (Internetwork Packet Exchange) were developed to exploit these characteristics and have enjoyed wide success. Because of the local-area assumptions built into these protocols, they have not scaled well into the wide area network environment. A LAN is the prototypical subnet: any station can communicate concurrently with any number of other stations on the same LAN, with each LAN connection between two such stations potentially carrying different transport protocols.

Wide area networks. Wide area networks (WANs) have been characterized by design trade-offs very different from those in the LAN environment. The costs inherent in providing communications paths over longer distances, together with the arbitrary tariff policies of different carriers, have combined to make the WAN a low-speed, high-cost component, and hence a continuing focus for performance optimization in the design of data networks over the last two decades. This situation has resulted in numerous wide area networking standards, some of which have country-specific variations.

WAN physical connections can have both point-to-point and multipoint configurations. Well-suited to both of these configurations, IBM's SDLC,¹² intro-

duced over 20 years ago, even before SNA, and concurrently standardized as high-level data link control (HDLC) in ISO and the International Telecommunication Union (ITU) has provided the basis for virtually all bit-oriented data link control procedures in the marketplace today. SDLC and HDLC leased circuits also form a subnet. When augmented with the X.25 packet protocol, a large X.25 subnet is created, with any port being able to have simultaneous connections (virtual circuits) with any number of other ports—each connection potentially carrying different transport protocols.

Subnet commonalities and distinctions. The archetypal LAN or WAN subnetwork provides simple functions of station addressing, data transmit and receive, and optional error checking, with no application significance given to protocol handshaking.¹³

However, at the detailed level, each subnet has a different address space, quality of service, speed, error rate, etc. In addition, every subnet has a distinctive user-to-network interface (UNI) that allows the attaching entity to request specific connectivity services such as an X.25 call packet, an all-route-broadcast, or a telephone call. Typically, a subnetwork is used by one or more transport networking protocols (shown above the subnetworking layer in Figure 1). The routing layer component within each transport networking protocol must understand the specifics of the underlying subnets, such as the capability to support broadcast or not, the need for an explicit connection (dialing) phase, and so forth. Specific standards define, for each protocol suite, how the routing layer should use any given subnet.

Once these differences have been accommodated by means of subnetwork-specific adaptations, the commonality in LAN and WAN subnetworking services provides a conceptual point of convergence in the multiprotocol architecture model. Since the services are common across many subnetworks, the upper layers of the model can be insulated from the peculiarities of the physical medium.

Multiprotocol subnetworks. Subnetworks are designed to be transparent to upper-layer protocols. They are indifferent to the content of the data they transport and, hence, indifferent to upper-layer routing protocols. Frequently, a system attached to a subnet is running a variety of routing protocols. To allow a single subnet attachment (port) to

be conveniently used for multiple protocols, subnet formats are augmented with protocol identifiers. These identifiers allow a single port to be shared across different transport protocol connections in an industry-standard fashion. The inclusion of a network protocol identifier enhances the inherent protocol transparency of the subnetwork and makes it a “natural” vehicle for multiprotocol networking.¹⁴

In Figure 2, station A is communicating via SNA protocols with station B and using TCP/IP protocols to communicate with station D. The subnetwork is oblivious to the networking protocols being used. The use of a protocol identifier field facilitates the concurrent use of the same subnetwork attachment hardware in station A for both protocols.

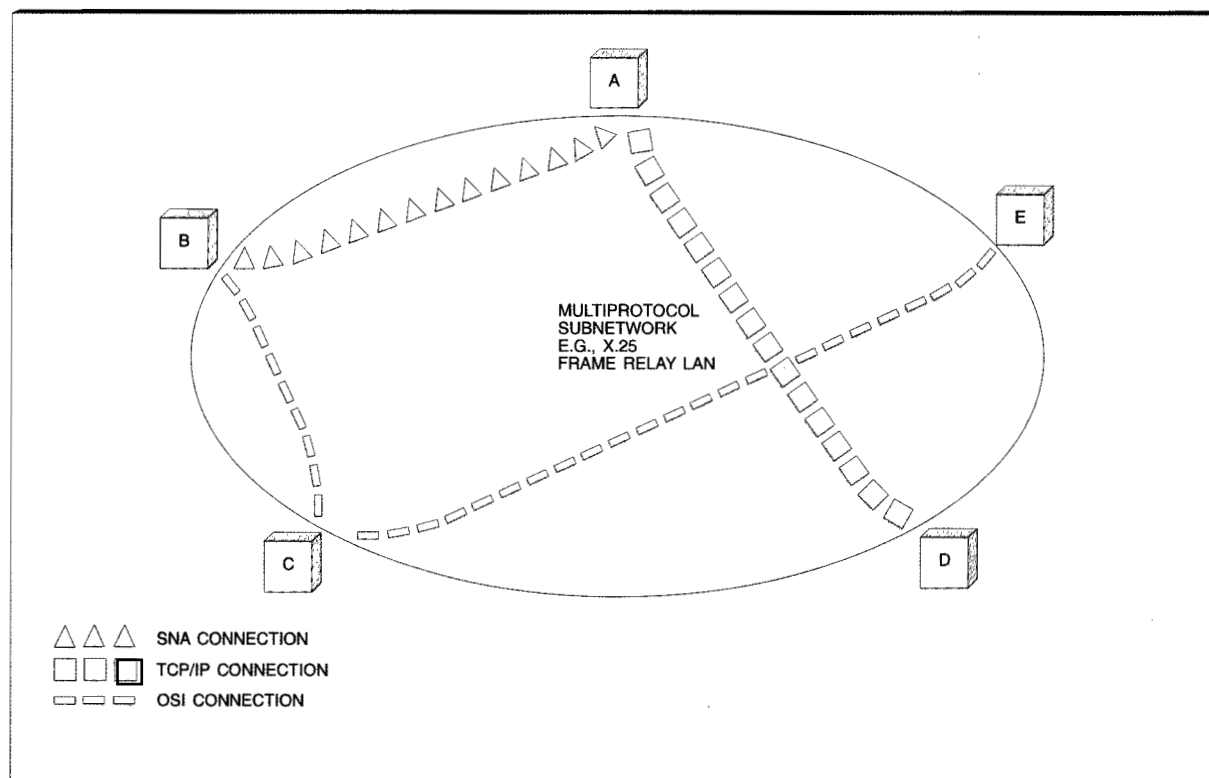
Recent advances. Multimedia applications require that multiple forms of information traditionally carried over separate facilities (telephone, data, video) be freely intermixed, dynamically, over shared facilities. This requirement has led to the concept of *cell relay*, where all forms of information are segmented using a common packet size and format, standardized in asynchronous transfer mode (ATM) communications.

ATM is the subnet of greatest generality: any ATM port can simultaneously have virtual connections with many other ATM ports, each connection being able to carry a different type of end-user medium (voice, data, video) in transparent fashion. We can loosely talk about an ATM subnet, where ATM connections traverse from switching node to switching node, with no need for the switch to look at protocol- and media-dependent information.

In effect, ATM has the potential of being the first subnet technology commonly deployed through LANs and WANs. Although today's LAN and WAN subnetworking technologies will not disappear, we can safely envisage their interoperation with ATM. Because of the significance of this expectation, we revisit ATM in more detail in the last section of this paper.

The transport network. Transport networking is shown in green directly above the subnetworking (red) layer in the Open Blueprint in Figure 1. The transport network provides end-to-end delivery of data across both local and wide area networks. It usually includes a routing function. Efforts at industry standardization have not resulted in a sin-

Figure 2 Multiprotocol subnetwork



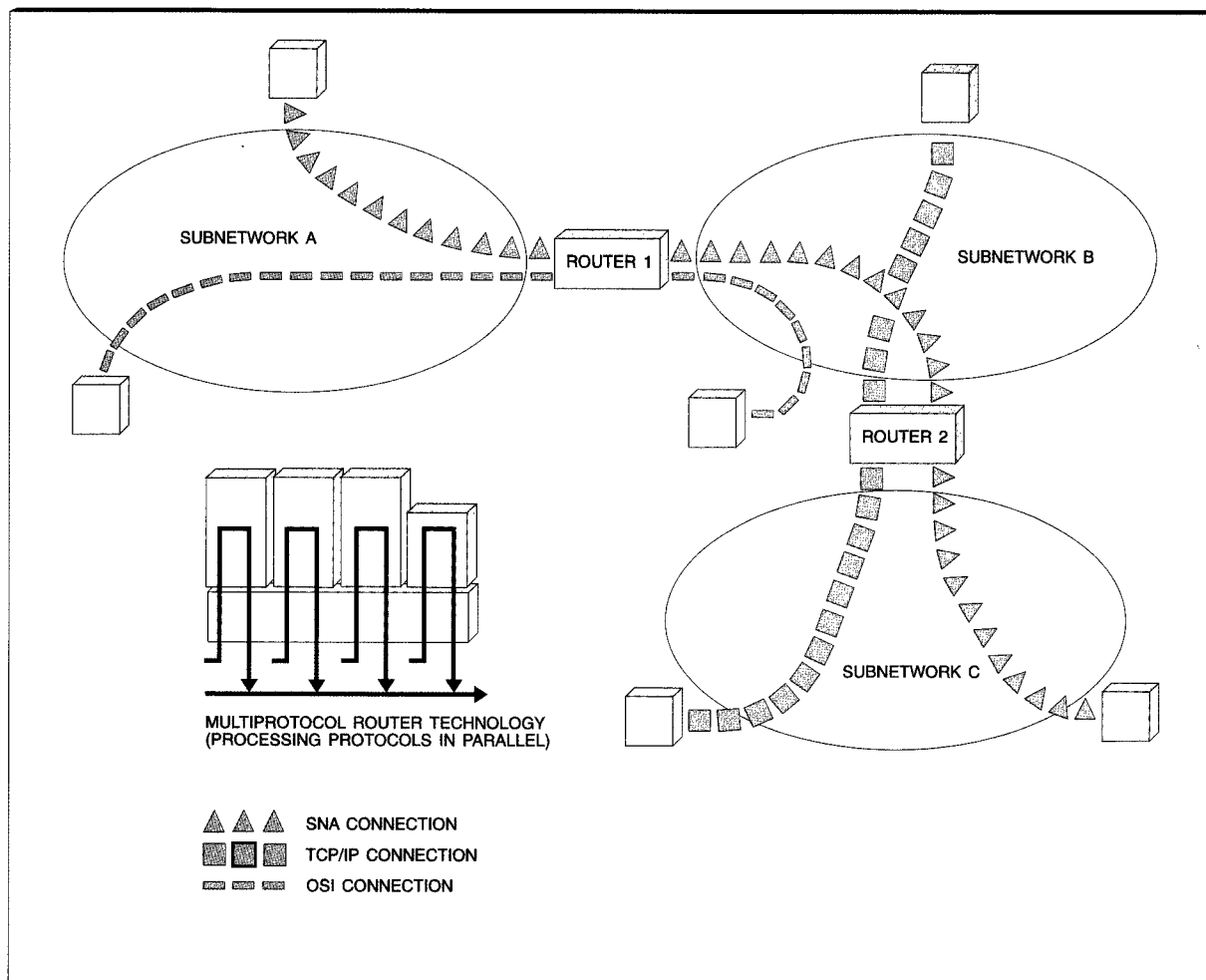
gle protocol or protocol suite (as illustrated by the set of transport networking protocols in the blueprint), but those efforts have converged remarkably on the services offered at the transport endpoints.

Routing. In this paper we define "routing" as the set of functions needed to allow transport connections to cross multiple subnets. Routing uses the notion of "network address," which represents a higher-level entity than an address on a subnet.¹⁵ A MAC (medium access control) address identifies an adapter on a LAN: it need be unique only within its own LAN. An SNA node address (or control point name) represents a node that can have many subnet attachments, each with its own subnet address. Thus routing requires a routing header, which is the information remaining after removal of the encapsulating subnet header on an incoming message, and before addition of another subnet header for the next subnet. The IP header (in TCP/IP), or the SNA transmission header, is an invariant in a given transport connection that can span multiple subnetworks.

On large campuses, routers are frequently used to partition LANs into multiple subnets and act as firewalls (barriers) for subnet-level broadcasts (such as an IP address resolution protocol). Routers are used on campus this way, instead of bridges, to correct the deficiencies of the LAN subnet protocols.

On the WAN side, however, an inverse phenomenon is more likely. Routers (IP, SNA, IPX, etc.) traditionally use leased, point-to-point lines to build a router backbone, each line being a tiny subnet. WAN subnets, such as X.25 or frame relay, have switches within them that interconnect routers through virtual subnet links. This development essentially replaces the concept of router backbones by the concept of switched subnets, with routers relegated to the periphery of the WAN subnet. Here the router functions simply to connect LAN subnets to the WAN subnet; end-to-end transport connections traverse only two routers in these cases. WAN subnets have the advantages of being protocol-transparent (thus simplifying network configuration and administration) and of offering mea-

Figure 3 Multiprotocol routing



surable quality-of-service parameters, capabilities no router technology yet has.¹⁶

Multiprotocol routing. In a multiprotocol environment, it is advantageous to share subnetwork links and router hardware across several networking protocols. Some routers provide this function by routing each protocol "natively," i.e., implementing each of the networking protocol algorithms in parallel and multiplexing the shared use of links, hardware buffers, and processor cycles.¹⁷ Figure 3 illustrates this concept. Router 2 is multiplexing both SNA and TCP/IP traffic between subnetworks B and C. Subnetwork attachments and other router resources are shared between the two kinds of traf-

fic. Such solutions can become complex because, under heavy traffic loads, the different networking protocols have quite different procedures for handling congestion situations. Furthermore, the challenge of fairly prioritizing traffic in accord with end-user needs, a problem solved in Advanced Peer-to-Peer Networking* (APPN*),¹⁸ is made more difficult in the multiprotocol environment.

Transport network endpoints. The ultimate objective of a data network is to interconnect application programs. Many application programs can exist within a simple system. Each such program is given an addressable "plug" into the transport network. It is called a *socket number* in TCP/IP and a

transaction program name in SNA. Different types of transport connections exist, depending on the services needed by the application and on the capabilities of the underlying transport network.

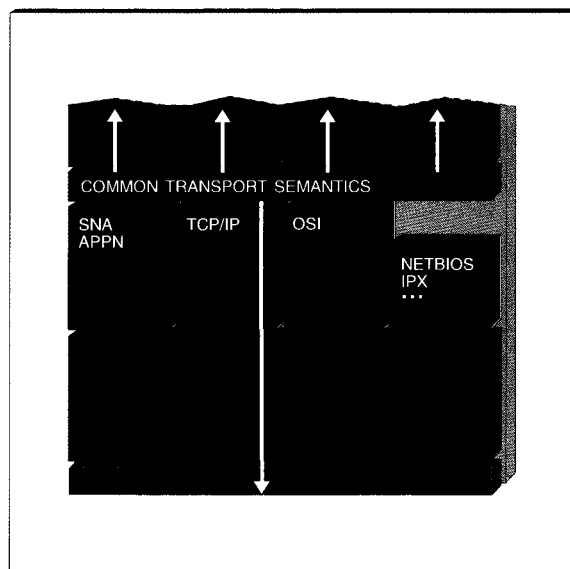
The transport endpoints are responsible for compensating for whatever services the underlying network may have lacked. If the network does not ensure the in-order delivery of packets, the transport endpoints are responsible for resequencing. If the network has segmented the data, the transport endpoints may be responsible for reassembly. Similarly, the endpoints will recover any errors that the network may not have handled. As a result, regardless of the underlying networking protocols, each protocol suite can present a remarkably consistent set of services at the transport endpoints. These services provide a flow-controlled, full-duplex, error-free logical channel between user programs.

At a more detailed level, important user-service variations exist for different transport interfaces such as in addressing conventions, out-of-band signaling capabilities, data stream support, and connection shutdown protocols. Because of these differences, a standard set of transport compensations has been designed in the Multiprotocol Transport Networking (MPTN) architecture.¹⁹ With the use of MPTN compensations, any transport protocol can be accessed, even by programs designed to run over another protocol. MPTN can be viewed as offering the union of the transport services provided by the prevalent networking protocols in the marketplace. This common set of services is shown in the Open Blueprint, in Figure 1, as *common transport semantics*.

Multiprotocol Transport Networking. By using the MPTN compensations, several different application environments can be supported on a single transport network. In this way, the application benefits of multiprotocol networking can be realized without having to administer multiple transport networks in parallel. For example, sockets applications (native to the TCP/IP environment) and NetBEUI applications (native to a NetBIOS environment) can operate over an SNA/APPN network by using the MPTN compensations. Figure 4 illustrates the same concept, but with MPTN being used to adapt several applications to a TCP/IP environment.

Multiprotocol encapsulation. Viewed at the endpoint, the services provided by the transport network are similar to those of the subnetwork. Both

Figure 4 Multiprotocol transport networking



offer full-duplex, connection-oriented, and connectionless communication across their respective interfaces. The *encapsulation* (or “tunneling”) technique of multiprotocol networking exploits this similarity. With encapsulation, one transport network serves as a subnetwork for another transport network. Although this imposes additional processing at the points of encapsulation and a duplication of headers, it is so easily implemented that numerous examples of this technique abound.²⁰

Data link switching (DLSw)²¹ is an encapsulation architecture that has been standardized for certain protocol combinations and is gaining wide support among vendors.

Application support. Applications constitute the most significant user investment and the *raison d'être* for the networking infrastructure we have been discussing. The *transport networking* and *subnetworking* layers of the Open Blueprint (the lower green and red layers in Figure 1) provide the capability to transmit and receive data between communications partners across a network. The *distributed systems services* (blue) layer makes it possible to develop portable, distributed applications by providing high-function, cross-platform application programming interfaces that are supported by popular high-level programming languages.

Before the microprocessor revolutionized the economics of distributed applications, most application logic resided in mainframe processors and was accessed from fixed-function, text-oriented terminals. This model was simple in both concept and implementation and has left a growing legacy of centralized application software critical to the success of many businesses. In some cases, these application programs are now accessed from multi-function workstations that run terminal-emulation software as part of their repertoire. Often these applications have been given a "face-lift" by more user-friendly interfaces programmed into the workstation, leaving the mainframe programming untouched. For new applications, the limited-function terminal paradigm has progressed into the support of graphical user interfaces (GUIs). The X Windows System** and Distributed Toolkit are examples of different approaches to the remote management of a GUI.²²

The limited-function-terminal paradigm touches only the fringes of the possibilities for distributed computing provided in the Open Blueprint. The program-to-program model allows the placement of application function and data anywhere in the network. Placement is facilitated by standardized programming interfaces (APIs) that allow application programs to interoperate across a network. The Open Blueprint communication services emphasize the program-to-program model.

Communication services. Experience has provided us with three useful modes for program-to-program communication: conversational, remote procedure call, and messaging and queuing. The three are briefly introduced here and are discussed and contrasted further elsewhere in this issue.²³

The conversational model is employed in TCP sockets²⁴ and in the Common Programming Interface for Communications (CPI-C). It is characterized as a connection-oriented service that insulates the programmer from awareness of all but the essential aspects of the logical communications channel. As its name implies, the conversational model allows multiple interchanges between programs in the context of the conversation. The versatility of this model has resulted in its widespread use directly for user applications and as a building block for other services.²⁵

The remote procedure call (RPC) model is employed in several RPC offerings in the marketplace. The

industry is converging on the Open Software Foundation's Distributed Computing Environment** (OSF DCE**) RPC. It is characterized by the familiar programming notion of a procedure call, except

Experience has provided us with three useful nodes for program-to-program communication.

that the called procedure need not reside on the same hardware as the calling procedure. Because the local procedure call model is already present in most programming languages, the RPC is very natural to the application programmer.

Whereas the first two models are connection-oriented and synchronous,²⁶ the third model, messaging and queuing, is connectionless and asynchronous. It uses queues as an interprocess communication mechanism. Because of the asynchrony between processes afforded by queues, this model is attractive in applications where communications partners can be decoupled from one another's actions. Processes can be made remote transparently to the application programmer.

The ability to insulate the programmer from location-awareness is important for all three of these communication services. However, beneath these interfaces, the support software must be able to locate the target program in the network. In addition, in a multiuser networking environment, it is important that only authorized users have access to certain programs and data.

Distribution services. In order to effect a distribution of programs and data in a distributed computing environment, a core set of services is necessary for locating resources, for controlling access to resources, and for ensuring the integrity of distributed resources. Historically, these services have been provided within the context of an application subsystem such as CICS. It is possible, today, to deploy networks with multiple applica-

tion-specific or protocol-specific directory and security servers. Such networks add considerable complexity to the user and network administration wherein multiple name spaces and password schemes have to be maintained and related one to another. For this reason, the Open Blueprint promotes the concept of a unified set of services based on the industry-standard OSF DCE technology.

Systems management. As networks have become larger and more diverse in their applications, media, and protocols, the challenges presented to the network administrator have grown as well. If there had not been advances in systems management architecture, standards, and products, today's networks would be unmanageable. Instead, there are industry-standard management models, protocols, and management information that promote multi-vendor participation.

Managers and agents. Fundamental to the systems management model in SNA, OSI, and TCP/IP is the notion of a management application in one node communicating with an agent in the system being managed. They share a common understanding of the resources being managed through a common set of definitions generically called *Management Information Bases* (MIBs). Each network component has its own definitions for such things as the counters, thresholds, and control parameters specific to its operation and maintenance. For some industry-standard protocols, there are companion management information definitions.

The agent's responsibility is to report management information and to update control parameters at the request of a management application. The management application presents all the information from all its agents to the network administrator in a coherent and understandable form. More sophisticated management applications will correlate information from different agents and automatically initiate corrective actions.

Multiprotocol challenges. Although the management model and concepts described in the previous subsection are similar at a high level of abstraction, there are significant differences in the protocols used for communicating between manager and agent. More importantly, there are significant differences between the tabular MIB structures allowed by the popular Simple Network Management Protocol (SNMP) of TCP/IP and the ob-

ject-oriented structures used by the Common Management Information Protocol (CMIP) of OSI.

The tactic used by some product designers is to assume that their box will be managed using just one protocol. Others, who want to hedge their bets, are designing their management information to allow both an object-oriented view and a tabular view simple enough to fit the SNMP constraints. Another promising approach, promoted by the Desktop Management Task Force (DMTF), is to specify the

As networks have become more diverse, the challenges presented to the network administrator have also grown.

management information in a way that is independent of management protocol.²⁷ In any case, it appears that management application platforms will be supporting both models for some time to come.

Blueprint summary. We have briefly touched on the networking-related pieces of the Open Blueprint. Many other aspects²⁸ of the Blueprint are not discussed at all in this paper. The Open Blueprint presents industry standards in an industry-standard structure. The value of the common structure is proven in today's practice of mixing and matching applications, transport networks, and subnetworks to accommodate past investments in equipment and software and to exploit newer technologies.

Although we can observe technological advances in each of the areas of the Open Blueprint, none is as dramatic or potentially far-reaching as the changes currently transforming the subnetworking layer. In the next section of this paper, we focus more on these changes.

Subnetworking revolution

The invention of the laser and its application to fiber-optic communications has ushered in a new era in communications. Gigabit bandwidths and neg-

ligible error rates are already being realized in common-carrier networks and are becoming available in customer-premises equipment. The opportunity for creative new multimedia applications is capturing the imaginations of both users and vendors.

At the subnetworking layer, the technologies of data communications and telephony are converging on ATM to handle data, voice, and video communications. Thus far in this paper, we have primarily discussed data communications. In this section, we expand the set of requirements and develop the concept of multiservices virtual networks.

ATM and the information superhighway.²⁹ In today's world, most types of communications needs are met with dedicated networks: the telephone network for voice and fax; X.25 or router networks for data communications; cable networks for television; etc. Use of dedicated networks is expensive, since multiple infrastructures are required. It also makes it virtually impossible to deploy systems that are simultaneously able to deal with multiple information types: picture-phones for voice and video communications, or computers that can be used as a single user interface for all forms of information.

For multimedia applications that mix data, audio, video, imaging, etc., future communication networks will have to handle delay-sensitive information (*isochronous* flows), such as real-time voice (telephone) and video. Furthermore, networks will have to provide fully dynamic bandwidth-on-demand to allow the type of information to vary during the connection. Today's time-division multiplexing (TDM)³⁰ technology and existing standard telecommunication interfaces fall short of being able to meet this requirement because of three major limitations:

1. *Granularity of bandwidth:* It is typically 64Kbps (voice), or simple multiples of 64Kbps.
2. *Bandwidth on demand:* Once a connection is set, its bandwidth cannot be changed.
3. *Quality of service (QoS):* All connections are treated as constant-bit-rate, isochronous connections.

These limitations have spurred the development of a technology to replace TDM in carrier networks, for the user-connection-level multiplexing and switching and the specification of new service in-

terfaces, under the common rubric of Broadband Integrated Services Digital Network (B-ISDN). This new technology (and its interfaces) draws upon the lessons of packet switching. It is what is known as cell relaying.

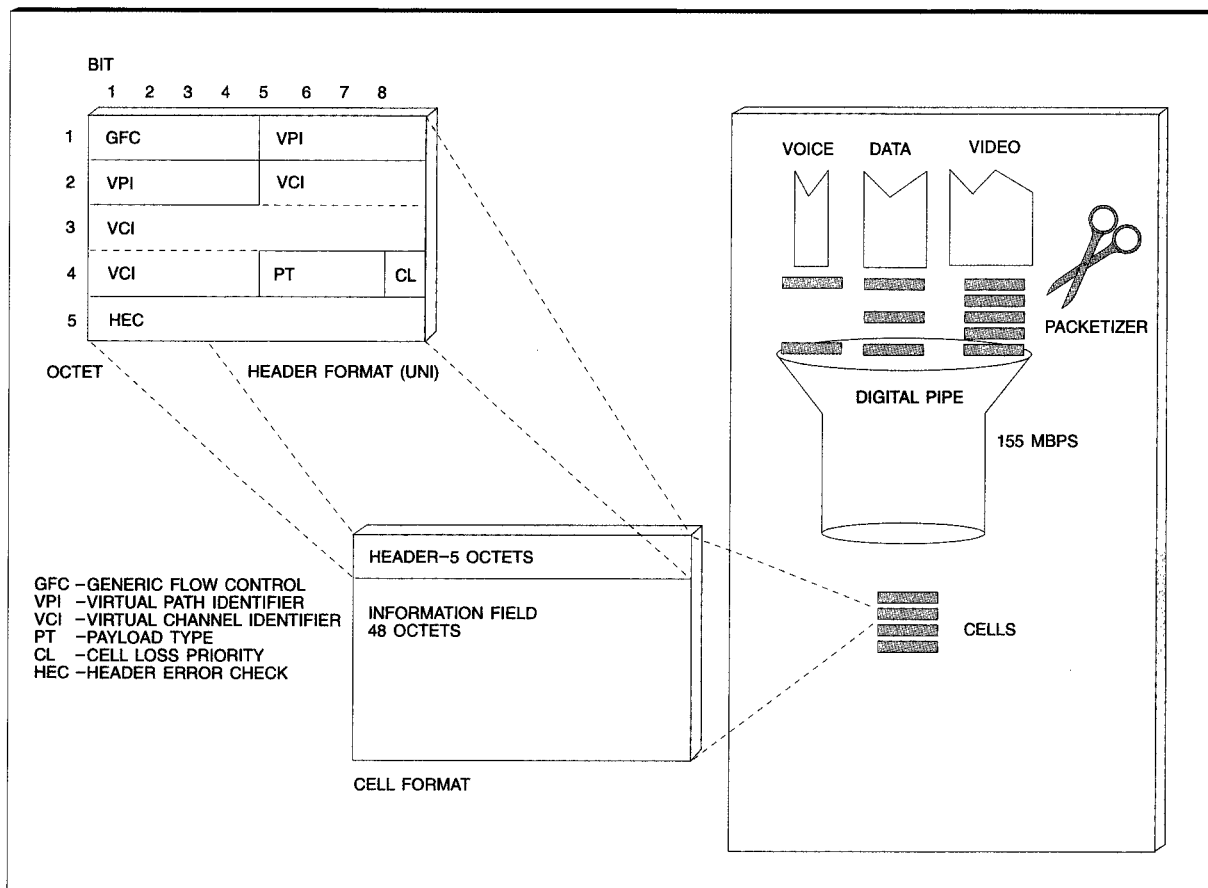
Every stream of information (including voice and video) is divided into small cells and switched using something resembling packet switching. See Figure 5. However, because some flows are delay-sensitive, relying on statistics and queuing does not work. Therefore, the ATM standard developed for the broadband networks of the future has the following characteristics:

- *Unlimited granularity* of bandwidth is available because bandwidth translates into number of cells-per-second; thus a single cell-relay switch can support connections of virtually any bandwidth, without a set increment.
- *Fixed-length* (53-byte) cells limit variations in transit delay ("jitter") created by variable packets.
- *Stable connections* use connection-oriented network technology.
- *Bandwidth reservation* along these connections guarantees that cells for isochronous connections do not need to wait in a queue more than a very short time.
- *Quality-of-service* choices are provided, so that some measure of statistical multiplexing can be obtained (with available bursty data filling whatever gaps are left unused by isochronous flows).

Thus, ATM should be looked at for what it really is: the basic multiplexing and circuit-switching technology needed to replace TDM in the forthcoming multimedia, bandwidth-on-demand networks—with the main design consideration being the support of isochronous information communications. ATM would not necessarily be a very good idea if we were considering data only or if the connections all required the same bandwidth: The overhead added by the small cells, the need to relay billions of cells per second, and the sheer complexity of managing the bandwidth would make it a technological oddity if it were not one of the necessary building blocks for multimedia.

Because of its adaptability, ATM is viewed almost universally as the key technology for the future in carrier networks, in private wide area networks, and on the campus as the multimedia LAN. One common misconception, though, is to confuse the

Figure 5 The ATM principle



technology with the services that will be offered in the broadband network of the future. Strictly speaking, "ATM network" is as meaningless a combination of words as "TDM network" would be. Just as carriers are using their TDM infrastructure today to offer a variety of services (leased/switched, circuit/packet/frame), they will use their ATM infrastructure to deliver these and new services: point-to-point/multicast/broadcast, connection-oriented/connectionless, bandwidth-reserved/committed-information-rate/best-effort, and so on. These services often will not let the ATM technology show through. Traditional 56/64Kbps leased/switched services, X.25, or I.233 FRBS (frame relaying bearer services) will still be there, side-by-side with new B-ISDN services.

Multiservices virtual networks.³¹ Although ATM can be viewed as the "ultimate technology" for

information superhighways, simple reality checks should be applied when discussing it today:

- **Application requirements:** How could anyone make a case that a VT100 or 3270 terminal has to have an SMDS, a frame-relay, or an ATM networking interface? Or that a plain telephone can make use of "bandwidth-on-demand" beyond what is provided by existing telephone facilities?
- **Cost of network attachment:** Why should a bank go through the expensive process of changing its cash dispenser devices so that they now support a "better" network technology? It is easy to forget that for any technology, more cost is buried in the attachment cards for end-user devices than in the switches themselves.
- **A single answer is rarely adequate:** Not every outlet in every wall need provide the same functions. Just because a few users need videocon-

ferencing, should an enterprise bring videoconferencing to every desk in every location?

- **Cost and availability of bandwidth:** It is not reasonable to expect that bandwidth costs will drop so dramatically that even small remote locations will suddenly qualify to be served by multimegabit links in the very near future. Thus, any technology-driven answer that mandates high-speed, high-quality links is likely to miss the actual business needs for most corporations.

Recognizing that there is no "one-size-fits-all" technical answer, corporate network managers, as well as telecommunication network operators, need to find ways to deliver different functions to different users, without having to set up multiple networking infrastructures. This requirement brings the concept of "Multiservices Networking Infrastructure" to the fore. This new concept is based on the following features:

- **A shared telecommunication infrastructure:** Packet or cell switches are used to build a shared networking fabric. They are able to multiplex and route streams of different types (voice, data, video, etc.) over shared links and are capable of dynamically establishing user-port to user-port connections with specified quality-of-service parameters (bandwidth, delay, jitter, priority, etc.).
- **Virtual networking:** Each end-to-end connection created through the network either through signaling (switched connections) or through operator intervention (permanent connections) is in fact a virtual path, created by software. The network itself will find the best route and reserve whatever resources may be required to deliver the specified QoS. Users can be added and moved, and connections created and deleted without any need to "patch-through" connections as on TDM systems. This capability offers maximum flexibility in accommodating new needs and minimizes the amount of planning required. Bandwidth is a common pool, from which systems draw the best possible overall service levels.
- **Multiple-access service interfaces:** Unlike traditional telecommunication systems that could provide only one type of service, the new switches are capable of delivering different service interfaces (UNIs) for different users. Therefore, a shared network can simultaneously deliver X.25, I.233, and B-ISDN interfaces, and more. Network designers can thus select, for each user port, the service interface that best fits the ap-

plication needs and the line attachment speed and type of any user device.

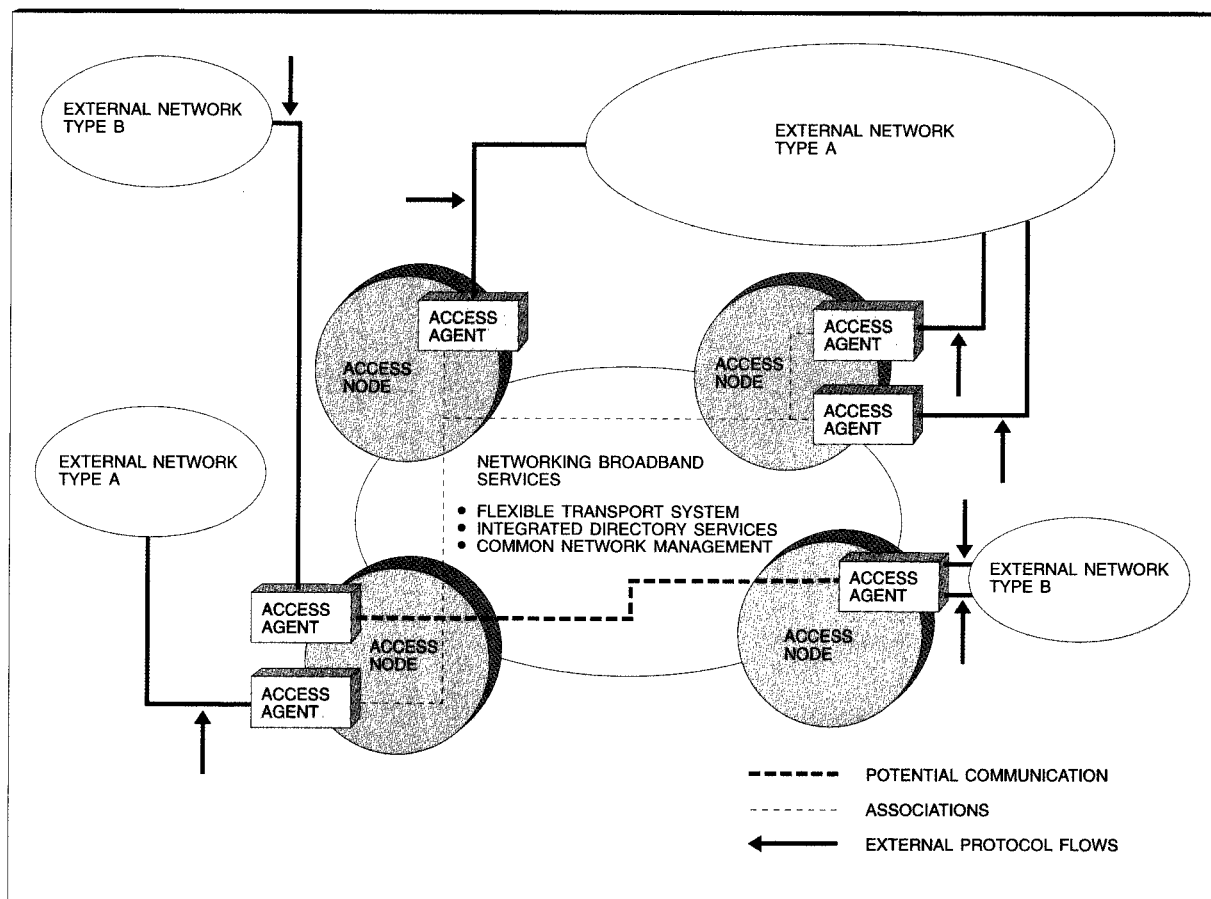
- **Multiservices networking interfaces:** As new applications become available that require bandwidth-on-demand, either for voice or video or for specific data transfer needs, new service interfaces are required. This is what the signaling and control plane in B-ISDN and the ATM UNI are all about. The new switches must thus be able to *deliver ATM UNI interfaces* to user devices, as well as to *exploit ATM services, through ATM UNIs and NNIs*, as such services become available from common carriers.

Networking BroadBand Services (NBBS). Cell relay—more specifically ATM—is the core technology for tomorrow's networks. As we have discussed, cell relay is also a way to deliver multiple services to existing devices, over a shared telecommunication infrastructure, in the form of "virtual networks." One can visualize a single network emulating a telephone network, an X.25 network, or a bridged LAN, yet being implemented over a single set of cell-relay switches.

IBM has embraced the ATM standard as the core multiplexing and switching technology in both its campus and WAN Nways* products. A common "ATM switch" chip is now used in LAN hubs (Nways 8260)³² and WAN bandwidth managers (Nways 2220). But multiservices virtual networking requires more than just a common switch, it also requires a mechanism to map various services onto a single structure and to manage and optimize bandwidth where it is scarce. NBBS (Figure 6), a new and far-reaching communications architecture, fills this role. NBBS has three different pieces:

1. **Access services:** Every user port is associated with a specific "access agent" that creates the UNI required for the service demanded by that user. The access agent deals with the specific formats and the specific service requests associated with the UNI, and maps them to the internal formats and language used within the network. For example, a telephone UNI will assume 64Kbps connections and an E.164 dialing address space; an X.25 UNI will demand virtual circuits and an X.121 address space. The respective access agents will map these address spaces to a common one, and map these different types of connections to a common cell-based format and different quality-of-service parameters.

Figure 6 NBBS components



2. Control point services: Each switch is controlled by a node control point that exchanges information with every other switch so as to understand the network topology and the available bandwidth on every link, and to make routing decisions on behalf of the various access agents when they request end-to-end connections. The control points also provide directory services for the various UNI address spaces, allowing access agents to resolve external addresses to internal network ports. These services can be viewed as operating within the signaling and control plane of the Open Blueprint.
3. A set of common transport services: Any end-to-end connection between users is mapped onto a common format, with characteristics

such as bandwidth, delay, and jitter mapped to common parameters. The transport formats supported by NBBS are the standard ATM cells, as well as a packet transfer mode that, on lower-speed links, can mix variable-size packets with standard ATM cells. Furthermore, functions such as multicast are supported, both for control point communications and for broadcast-type applications.

In essence, NBBS is a set of functions *above* ATM to build bandwidth-efficient multiservices networks and to minimize the administrative effort to define the various virtual networks. The incorporation of ATM and NBBS into the lower layers of the Open Blueprint is the first step in an evolution toward the world of multimedia. It enables client/server

information processing to exploit the emerging information superhighways. New APIs and distributed services will be added that will allow new applications (such as collaborative computing, personal videoconferencing, etc.) to directly invoke the ATM and NBBS capability to dynamically create connections with widely different quality-of-service characteristics.

Conclusion

The Open Blueprint is a structure for discussing multivendor, industry-standard distributed computing solutions. At the networking layers, the Open Blueprint illustrates the protocol choices and multiprotocol networking solutions in the marketplace today. Each of these alternatives has strengths and weaknesses that can be magnified or reduced by any given situation. A typical network today will have a variety of protocols and often a variety of multiprotocol solutions. Tomorrow's networks are being designed with a recognition of our multiprotocol legacy. Much has been learned to prepare us for the ultimate multiprotocol challenge: multimedia.

This paper has emphasized the importance of ATM from two perspectives. From the perspective of the evolving technology in traditional data networks, ATM can be viewed as a new high-speed multiprotocol subnetwork technology that is equally applicable to local and wide area networks. In this respect, ATM fits into the Open Blueprint infrastructure as just another subnetwork, although with more universal appeal.

From the broader perspective of communications in general, ATM is applicable not only to data networks, but also to telephony, videoconferencing, and video distribution because it can satisfy a range of quality-of-service requirements. Thus, Open-Blueprint-based client/server systems can share the same network infrastructure with a variety of other users not directly related to distributed computing.

The capability of an ATM subnetwork to support different quality-of-service requests on each connection makes it a natural vehicle for multimedia communications. As the Open Blueprint evolves to embrace tomorrow's multimedia standards, we can anticipate multimedia application programs³³ in the higher layers of the blueprint, signaling their

connection QOS requirements across enhanced APIs.

The notion of subnetworks that provide any-to-any connections, and more importantly the notion of quality-of-service guarantees, will fundamentally change the way public and private networks are built. Data-only routers and voice-only switches will gradually be displaced from the core of the networks by cell switches serving multiple user requirements, until the day when any form of communication will be "one ATM phone call away."

The Open Blueprint provides both the vision of this exciting new world and the migration path to it.

Acknowledgments

We would like to acknowledge the help and encouragement of G. D. Schultz, S. K. Smith, J. P. Golick, D. L. Kaminsky, J. P. Gray, H. S. Trollman, D. B. Blakeley, and the anonymous reviewers.

*Trademark or registered trademark of International Business Machines Corporation.

**Trademark or registered trademark of Digital Equipment Corporation, Novell, Inc., Massachusetts Institute of Technology, or Open Software Foundation.

Cited references and notes

1. J. H. McFadyen, "Systems Network Architecture: An Overview," *IBM Systems Journal* 15, No. 1, 4-23, (1976).
2. The term *subnetwork* or *subnet* is widely used in the industry, and in the various papers in this issue of the *IBM Systems Journal*, to refer to a particular "vertical" or "horizontal" subset of a network. The use here is in the latter sense, to refer to the "inner" network comprising the data link control and physical layers or particular instances thereof. Elsewhere, "subnetwork" is also applied to "vertically complete" portions of the network, where some common characteristic prevails. It is not uncommon for the term to be used in both ways in some discussions; usually the context or the appropriate qualifier makes its use clear.
3. J. P. Gray, P. J. Hansen, P. Homan, M. A. Lerner, and M. Pozefsky, "Advanced Program-to-Program Communication in SNA," *IBM Systems Journal* 22, No. 4, 298-318 (1983).
4. E. G. Britton, J. Tavs, and R. Bournas, "TCP/IP: The Next Generation," *IBM Systems Journal* 34, No. 3, 452-471 (1995, this issue).
5. S. Wecker, "DNA: The Digital Network Architecture," *IEEE Transactions on Communications* COM-28, No. 4, 510-526 (1980).
6. J. R. Aschenbrenner, "Open Systems Interconnection," *IBM Systems Journal* 25, Nos. 3/4, 369-379 (1986).
7. *Networking Blueprint Executive Overview*, GC31-7067, IBM Corporation; available through IBM branch offices.

8. *Open Blueprint Technical Overview*, GC23-3808, IBM Corporation; available through IBM branch offices.
9. Although the seven-layer model of OSI once included strict notions of layer independence, more recent work in the OSI committees on the upper three layers has treated the upper-layer functions somewhat like a shared subroutine library. (See ISO 9545.) The functions in the distributed systems services of the Open Blueprint are a subset of those in OSI layers 5-7.
10. Of course, one unlikely solution to the multiprotocol challenge is to converge on a single "standard" protocol, but to do even that requires some form of multiprotocol networking as a transition step.
11. K. J. Christensen, L. C. Haas, F. E. Noel, and N. C. Strole, "Local Area Networks—Evolving from Shared to Switched Access," *IBM Systems Journal* 34, No. 3, 347-374 (1995, this issue).
12. R. A. Donnan and J. R. Kersey, "Synchronous Data Link Control: A Perspective," *IBM Systems Journal* 13, No. 2, 140-162 (1974).
13. The absence of application semantics from link-level acknowledgments is a cornerstone of network architecture layering concepts. The "streamlined" use of link-level acknowledgments by the application in certain LAN protocols has introduced problems of scalability as the LAN stations were connected into larger networks.
14. ISO/IEC TR 9577 provides Network Level Protocol ID values that are used as protocol discriminators in several environments, including ATM and frame relay.
15. The network routing techniques can be dependent on the structure and values in the network address or can be decoupled from the network address. In the latter case, equipment can be moved from one location to another without the need for network administrators to reassign network addresses.
16. The ability of some routers to prioritize traffic is valuable for data-only traffic but does not satisfy the needs of multimedia.
17. A more thorough description of this and other multiprotocol techniques is provided in GG24-4338, *Introduction to Networking Technologies*, available through IBM branch offices.
18. R. Bird, C. Brotman, R. Case, G. Dudley, R. E. Moore, and M. Peters, "Advances in APPN Architecture," *IBM Systems Journal* 34, No. 3, 430-451 (1995, this issue).
19. D. Pozefsky, R. Turner, A. K. Edwards, S. Sarkar, J. Mathew, G. Bollella, K. Tracey, D. Poirier, J. Fetvedt, W. S. Hobgood, W. A. Doeringer, and D. Dykeman, "Multiprotocol Transport Networking: Eliminating Application Dependencies on Communications Protocols," *IBM Systems Journal* 34, No. 3, 472-500 (1995, this issue).
20. IBM's SNALINK function uses this approach to carry TCP/IP traffic through an SNA network. Similarly IBM's non-SNA Interconnect (XI) product routes X.25 traffic over an SNA network using this technique. Many router vendors use the encapsulation technique to carry non-TCP/IP traffic as data in TCP/IP packets.
21. P. Gayek, "Data Link Switching: Present and Future," *IBM Systems Journal* 34, No. 3, 409-429 (1995, this issue).
22. The X Windows technique communicates individual key-strokes and sampled mouse movements to the remote node that is responsible for updating the user's workstation display in real time. Because of the intense interplay between client and server in the X Windows model, it is widely used only in a LAN environment. IBM's Distributed Toolkit (DT) reduces the intensity of interaction between workstation and the application by handling most GUI events locally at the workstation and only passing application events to the remote node. Within the distributed computing paradigm, DT offers a standard way to distribute presentation management functions. DT shows promise for both local and wide area environments.
23. W. S. Arnette, A. D. Kshemkalyani, W. B. Riley, J. P. Sanders, P. J. Schwaller, J. C. Terrien, and J. Q. Walker, "CPI-C: An API for Distributed Applications," *IBM Systems Journal* 34, No. 3, 501-518 (1995, this issue).
24. The supporting infrastructure of most sockets implementations does not include all the distributed services support that would qualify it as a full-function conversation interface. However, the popularity of sockets as an application programming interface invites its classification into one of the three distributed computing models. The conversation model is the best fit.
25. For example, the Open Software Foundation's remote procedure call builds on a (conversational) sockets interface to gain access to the transport network, and the Distributed Relational Database Architecture™ (DRDA™) of the Open Blueprint uses CPI-C to gain access to the transport network.
26. The conversational model can be employed in a full-duplex mode and thereby allow asynchrony between the send and receive streams, but this mode assumes no immediate logical relationship between data sent and data received.
27. The Management Information Format (MIF) of DMTF is a simply understood guideline that relieves developers of the requirement to become expert in management protocol technologies.
28. J. A. Colosimo, "The Role of IBM's Open Blueprint Approach in Distributed Computing," Technical Forum, *IBM Systems Journal* 34, No. 1, 138-141 (1995).
29. The material in this subsection of the paper was previously published in: "The Networking Blueprint—An Update," *Proceedings of GUIDE/SHARE Europe*, Session AA-002, Vienna (October 1994).
30. Time division multiplexing is widely used in telephone networks to divide high bandwidth channels into multiple lower bandwidth subchannels by allotting the common channel to each of the subchannels, one at a time.
31. The material in this subsection of the paper was previously published in: "The Networking Blueprint—An Update," *Proceedings of GUIDE/SHARE Europe*, Session AA-002, Vienna (October 1994).
32. R. A. Sultan and C. Basso, "ATM: Paving the Information Superhighway," *IBM Systems Journal* 34, No. 3, 375-389 (1995, this issue).
33. B. K. Aldred, G. W. Bonsall, H. S. Lambert, and H. D. Mitchell, "An Architecture for Multimedia Communication and Real-Time Collaboration," *IBM Systems Journal* 34, No. 3, 519-543 (1995, this issue).

Accepted for publication March 15, 1995.

Matthew L. Hess *IBM Networking Hardware Division, 800 Park Offices Drive, Research Triangle Park, North Carolina 27709 (electronic mail: mhess@vnet.ibm.com).* Dr. Hess is a senior engineer and manager in the Networking Architecture group at IBM's Research Triangle Park laboratory, responsible for management architecture and standards. A graduate of McGill University and the University of Birmingham in England, he started his IBM career in 1968 as a systems engineer

in his native Montreal. In 1979, he joined the U.S. company and has since worked on several networking architectures and standards, including X.25, SNA Network Interconnect, SNA/DS, OSI, MPTN, the Networking Blueprint, and network management.

Jean A. Lorrain *C.E.R. IBM France, Le Plan du Bois, 06610 La Gaude, France (electronic mail: lorrain@vnet.ibm.com).* Mr. Lorrain graduated in cybernetics at the Institut des Sciences de l'Ingenieur de Nancy (I.S.I.N.) and joined IBM La Gaude in 1969. His main contributions have been in the areas of private telephone systems; modems; network management; communication controllers (FEPs); APPN architecture; and more recently, high-speed networking, including frame relay and ATM support in IBM products. He is a Senior Technical Staff Member and a member of the IBM Academy of Technology. He is currently responsible for the technical strategy in La Gaude.

G. Richard McGee *IBM Networking Hardware Division, 3039 Cornwallis Road, Research Triangle Park, North Carolina 27709 (electronic mail: mcgee@vnet.ibm.com).* Mr. McGee joined IBM in 1974 with degrees in mathematics and industrial engineering from Auburn University. He later earned an M.S. in the management of technology from the Massachusetts Institute of Technology. His IBM career has included programming, design, and management responsibilities across a variety of networking architectures and products including SNA, APPC, APPN, CPI-C, MPTN, VTAM, NCP, and TPNS. In 1992, he became Director of Networking Systems Architecture with cross-divisional responsibility for IBM's networking strategy, architectures, and standards. Mr. McGee is currently Director of Networking Controllers, responsible for the development and worldwide support of network processor products such as the IBM 3745, 3746, 3174, NCP, and the 2217 NWay Multiprotocol Concentrator.

Reprint Order No. G321-5571.