A Preview of APPN High Performance Routing

Document Number TR 29.xxxx

April 8, 1993

James P. Gray
Marcia L. Peters

SNA Studies
Department F92, Building 673
- and -
Advanced Peer-to-Peer Networking
Department C74, Building 673
- at -
Networking Systems Architecture
IBM Corporation
P.O. Box 12195
Research Triangle Park, North Carolina, 27709 U.S.A.
internet address: mpeters @ ralvm6.vnet.ibm.com
Telephone: (919)-254-4380
Tie line: 444-4380

ABSTRACT

After a brief review of APPN—Advanced Peer-to-Peer Networking—and a survey of existing routing techniques, a new SNA approach to routing called HPR—APPN High Performance Routing— is introduced. Topics covered in this overview include HPR function placement within the OSI layered model, priority scheduling for multilink transmission groups, Automatic Network Routing, Rapid Transport Protocol, Adaptive Rate-Based congestion control, the relationship of effective congestion control algorithms to throughput and response time, and HPR's selection of frame relay as a preferred data link control.

ITIRC KEYWORDS

- APPN
- APPC
- Advanced Peer-to-Peer Networking
- Advanced Program-to-Program Communication
- LU 6.2
- SNA
- APPN/HPR
- HPR
- High Performance Routing
- RTP
- Rapid Transport Protocol
- ANR
- Automatic Network Routing
- Logical Link Control
- LLC
- connectionless routing
- congestion control
- flow control
- adaptive rate-based
- ARB

- Frame Relay
- SNA product directions
- multilink transmission groups
- MLTG

ABOUT THE AUTHORS

Marcia Peters leads the APPN architecture group at IBM's Networking Systems line of business in Research Triangle Park, North Carolina where she develops architecture and product strategies to commercialize new technologies. She previously worked on a new high speed networking architecture, specializing in network control algorithms for multicast and distributed directories for multiprotocol routing. She also contributed enhancements for the seamless interconnection of APPN and subarea SNA architectures. Before joining IBM in 1988, she was lead programmer at Decision Data Computer Corp., a vendor of plug-compatible equipment for the AS/400 and System/36 family, and a development programmer at Telex Computer Products and Raytheon Data Systems, producing SNA networking products. She received a B.A. in music from Swarthmore College in Pennsylvania in 1975. She is a senior member of the IEEE. She has filed applications for over 5 US patents and published over 40 technical disclosures and articles.

Dr. James P. Gray joined IBM in 1970 in Raleigh, North Carolina. After contributing to an IBM microprocessor architecture, from 1972 to 1984 he contributed to various aspects of SNA, including APPC, syncpoint, and APPN. In 1984 he was named an IBM Fellow and manager of SNA Studies, a group that explores issues in networking and distributed processing. Dr. Gray earned a B.E. in Electrical Engineering from Yale College in 1965 and a Ph.D. in communication theory from Yale's department of Engineering and Applied Science in 1970. He is a fellow of the IEEE and a member of ACM.

CONTENTS

FIGURES

INTRODUCTION

WHAT IS APPN?

APPN—Advanced Peer-to-Peer Networking—is an extension of SNA—Systems Network Architecture. APPN was first announced in 1987. At the time this paper was written, APPN was available on the following IBM products:

- AS/400 [22] [6] [7]
- 3174 Establishment Controller [11]
- OS/2 [26] [9]
- System/36 [7]
- DPPX/370

IBM has announced plans to make APPN available on the following products in 1993:

- 6611 Network Processor
- VTAM
- AIX SNA Services.

A number of other companies also offer APPN in their products, including Brixton, InSession Inc., and Systems Strategies Inc. Even more vendors currently have APPN prototypes running or are expected to offer APPN in their products, including: Advanced Computer Communications, Apple, 3Com, Cabletron, cisco systems, CrossComm, Data Connection Ltd., Network Equipment Technologies, Network Systems Corporation, Novell, Siemens-Nixdorf, Ungermann-Bass Access-One, and Wellfleet. IBM opened the APPN End Node protocols in 1991 by publication of the SNA Type 2.1 Node Reference [24]. In early 1993 IBM opened the APPN Network Node protocols by source code licensing and by publication [25].

WHAT IS APPN/HPR?

APPN/High Performance Routing, which IBM called *APPN+* when revealing its future networking directions in March, 1992, is a further extension of SNA to take advantage of fast links with low error rates. It replaces ISR (intermediate session routing)—the routing technique in current APPN products—with HPR (high performance routing). One of its key benefits is the ability to nondisruptively reroute sessions around failed nodes and links. APPN/HPR can improve intermediate routing performance by 3 to 10 times, greatly reduce network node storage requirements, and augment existing LU-based flow control with an advanced congestion avoidance algorithm called ARB—Adaptive Rate-Based congestion

control. In late 1992, IBM said it intends to make APPN/HPR technology available in a future release of its network node source code license.

APPN'S CLIENT-SERVER MODEL

The term *peer-to-peer* is misleading because it captures only one of two equally important aspects of APPN. Equally descriptive terms could have been *distributed* or *client/server*. *Distributed* means decentralized. APPN is not locked into the strict hierarchical topology required by its predecessor, SNA subarea architecture: APPN architecture supports a variety of physical topologies such as star, hub, mesh, and hierarchical, over a variety of transmission media including most popular LAN and WAN media. *Distributed* says that important data and important applications may reside on any computer in the network. In other words, computers other than mainframes are running significant applications. Such applications are network-centric rather that mainframe-centric. The term *peer* accurately characterizes this aspect of APPN. *Client-server* says that in application design, function placement is flexible [17]. It recognizes that computers differ in terms of physical location and physical security, the level of attended support (e.g. operator intervention, regular backup, and so forth), connectivity to other computers, permanent storage media (disk, tape, and the like), the amount of RAM for running large applications, support for multi-tasking, and computing capacity (MIPs, FLOPs). Client-server principles encourage flexible application design so that smaller computers can take advantage of the capabilities of larger computers. It is the client-server aspect of APPN that the term *peer* does not suggest.

With client-server concepts in mind, and to support emerging networked applications for distributed computing, APPN defines two main node types: the end node (EN) and the network node (NN). APPN also interoperates well with a pre-APPN node type called the Low-Entry Networking end node (LEN EN), a subset of the APPN EN. LEN was a precursor to APPN; APPN is the strategic base for enhancements to SNA. (All three of these node types are classified as SNA Type 2.1 nodes.) This general architectural structure makes sense. The control plane in an NN requires more physical resources (CPU, RAM, disk) to support the network control applications and services that it provides. By offloading most of these functions to an NN, an APPN EN can be relatively small and inexpensive, and/or have more of its resources left for applications, and still partake of APPN's automation and dynamics. An NN may also have specialized routing hardware and attachments to many WAN links.

A network node is a **server**, in the sense of providing network services to its end node clients. The services are directory services and route selection services. An brief overview will illustrate the respective client and server roles of ENs and NNs.

Directory services means locating a partner application: determining what computer the application currently resides on. This avoids the EN having that information predefined. This also gives network administrators flexibility in moving an application to the computer best able to support it. With APPN, moving an application requires only local definition

changes at the computer where the application used to be, and at the computer it moves to. All other computers learn this information dynamically, as needed.

Route selection services means picking the best route for the session, based on a user-specified class of service and transmission priority, what possible routes are currently available through the network, and the destination computer's location. Route selection also avoids congested nodes, and randomizes among equivalent routes, in order to distribute the load.

These network control applications of APPN have been described extensively in the literature [31], [32], [10], [8], [12], [37], [21]. Less often discussed are its lower layers.

A network node is also a **router**: it forwards traffic that crosses it on the way to somewhere else. The second half of this article discusses APPN routing techniques in more detail.

First though, let's examine the differences in these APPN node types using a layered architecture model. We will cover differences in network control applications (in the uppermost, or transaction, layer) and differences in the lower layers (Layers 2, 3 and 4—the data link, network and transport layers). With this foundation we can examine APPN high performance routing, describe its benefits, show how it complements existing APPN intermediate session routing functions, and consider what kinds of networking products it is appropriate for.

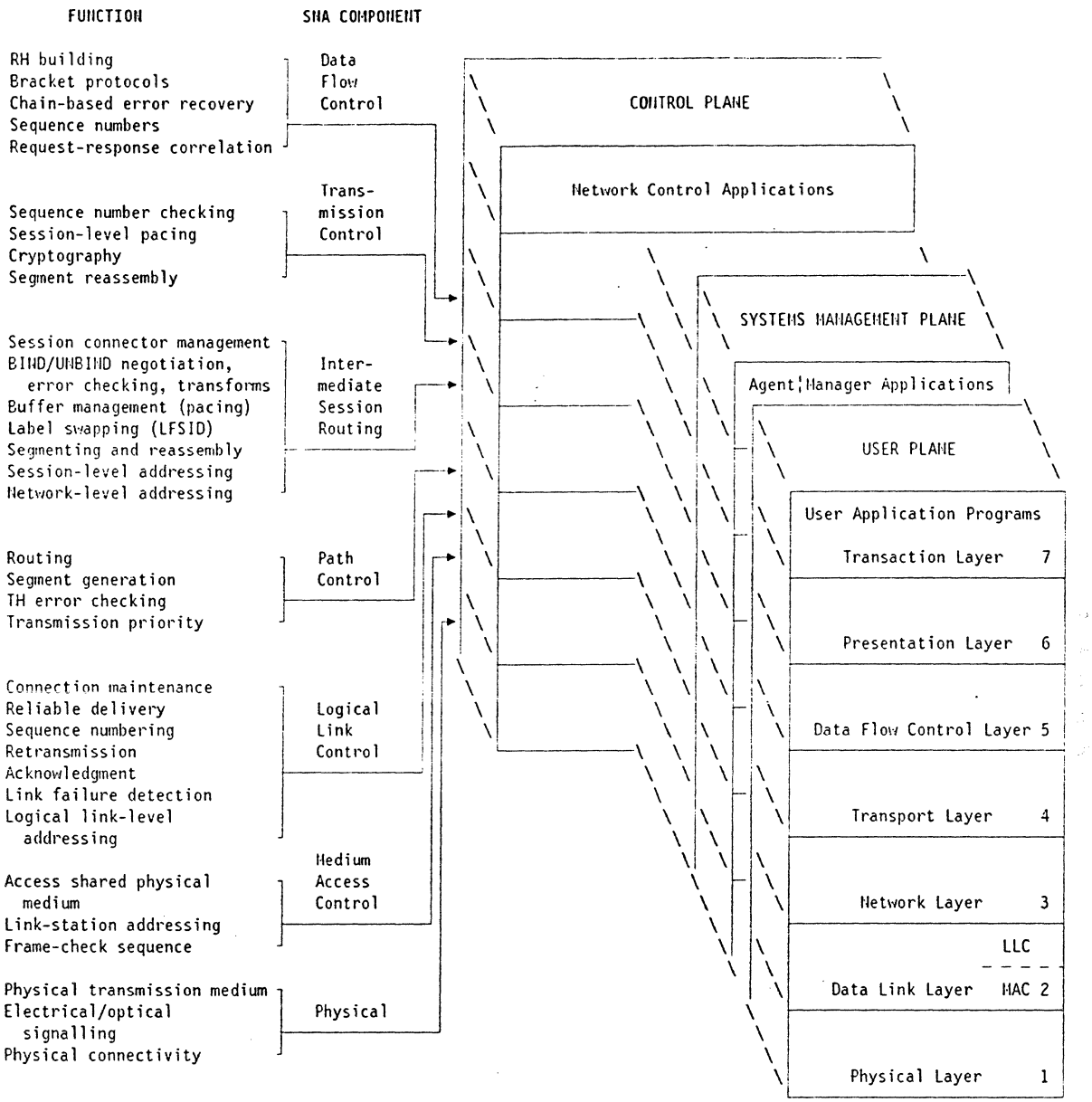


Figure 1. Layered Model Showing SNA Function Placement in User, Systems Management, and Control Planes

Figure 1 shows APPN's functions divided approximately into horizontal strata, according to the OSI networking model, like a layer cake. [18], [29] In addition, one can imagine vertical slices through these layers, like a slice of cake, containing some cake from each of the layers. There may be chunks of fruit in the cake, so that each slice is a little different. One of these slices is the *user plane*, the piece of system software on a computer directly supporting end user application programs running on that computer. APPC—Advanced Program-to-Program Communication—also called LU 6.2 or Logical Unit type 6.2— is an

example of a user plane. Another slice is the *systems management plane* (shown behind the user plane). Systems management is often structured according to client-server principles, with a client component in an end-user's computer being called an *agent* or *entry point*, and a server component in a specialized (but not necessarily centralized) computer being called a *manager* or a *focal point*.[2] The slice that this paper focuses on is the *control plane*, sometimes called the *control point* in an SNA node. Its job is to support the user and management planes, automating such chores as the distribution of routing information and directories. Bear in mind that while some networking architectures (LAN architectures, in particular) combine the control and management planes, in APPN they are distinct. Discussion of the management plane is outside the scope of this paper.

APPN NODE TYPES

This section introduces the APPN node types.

LEN End Node

A LEN end node (Figure 2) has no network control transaction programs at all, and no client support for requesting the services of a network node: it has no APPN control plane.

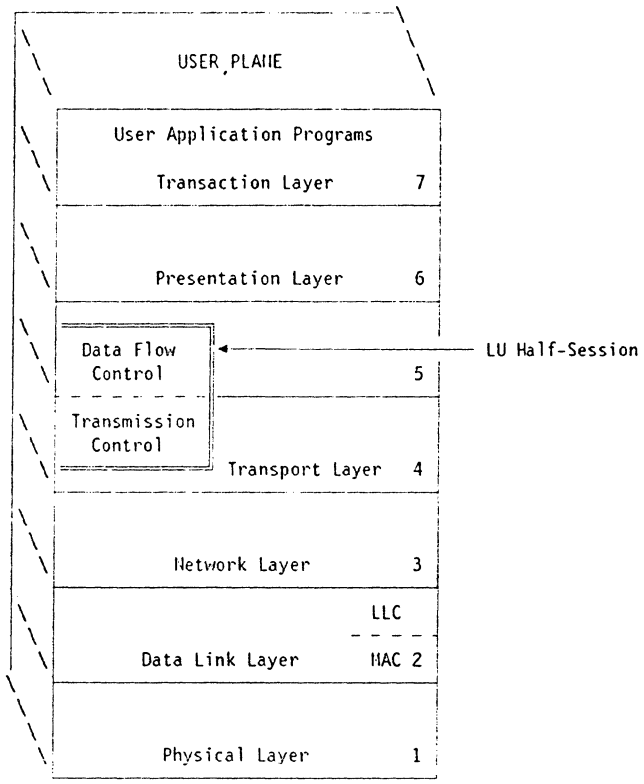
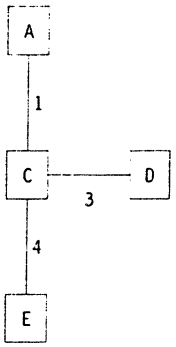


Figure 2. LEN End Node's Lack of an APPN Control Plane

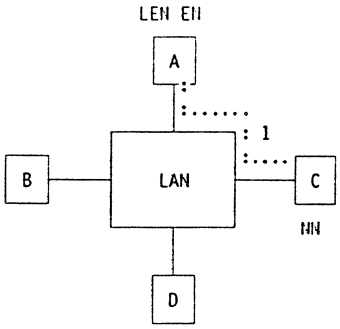
The routing function in a LEN end node is simply to transmit to an adjacent node using a predefined link and any required link signalling information (such as the link station address on an SDLC link, the MAC and SAP addresses on a LAN, or a selected adapter and telephone dial digits for a switched connection). Without default routing (explained in Figure 4 through Figure 5), a LEN end node must have definitions for the locations of all partner applications. One consequence is that if a network administrator decides to relocate an application to a different computer, she needs to distribute updated definitions to **every** LEN EN that accesses that application. (This is one of the things APPN was invented to fix!) Figure 3 illustrates such a configuration and the definitions required at LEN EN C.



C's definitions	Resulting route
For A, use link 1	C-1-A
for D, use link 3	C-3-D
For E, use link 4	C-4-E

Figure 3. LEN EN Explicit Routing

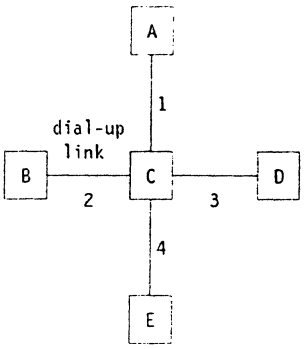
Alternatively, a LEN EN may define all partner applications as residing on an adjacent NN (whether they actually reside there or not). In this case the LEN EN acts as a passive client, accessing the services of a network node server indirectly, by simply attempting to route data to it (sending it a BIND, in SNA parlance). While this method relieves the LEN EN of predefining individual network addresses for all its partner applications (a single definition "all partners reside on the NN" will suffice), the resulting sessions always traverse the network node, not always an efficient route (especially where direct mesh connectivity exists between every pair of computers, as on a LAN). Figure 4 illustrates this possibility.



A's definitions	Resulting route
For B, use link 1	A-1-C-LAN-B
for C, use link 1	A-1-C-
For D, use link 1	A-1-C-LAN-D

Figure 4. LEN EN Default Routing on a LAN. *LEN EN A minimizes definitions by defining all its partners as residing on an NN. For the LAN mesh topology shown, this results in inefficient routes that traverse NN C's intermediate session routing at layer 4 even where direct routes are available.*

On the other hand, defining all partners to reside on an adjacent NN may be quite acceptable if the LEN EN is a portable computer with a single dial-up line to access network computing resources. In such a configuration, all connections would traverse the NN anyway, as Figure 5 illustrates:



B's definitions	Resulting route
For A, use link 2	B-2-C-1-A
For C, use link 2	B-2-C
for D, use link 2	B-2-C-3-D
For E, use link 2	B-2-C-4-E

Figure 5. LEN EN Default Routing Over a Switched Link. *This LEN EN minimizes definitions by defining all its partners to reside on a NN. Because of the topology, the resulting routes are acceptable—as good as they would be for an APPN EN.*

APPN End Node

An APPN end node adds a small control plane, with client transaction programs (at the transaction layer—layer 7) to register its applications (Logical Units, or LUs, in SNA lingo) to a network node and request services like locating destination applications and selecting routes.[24]

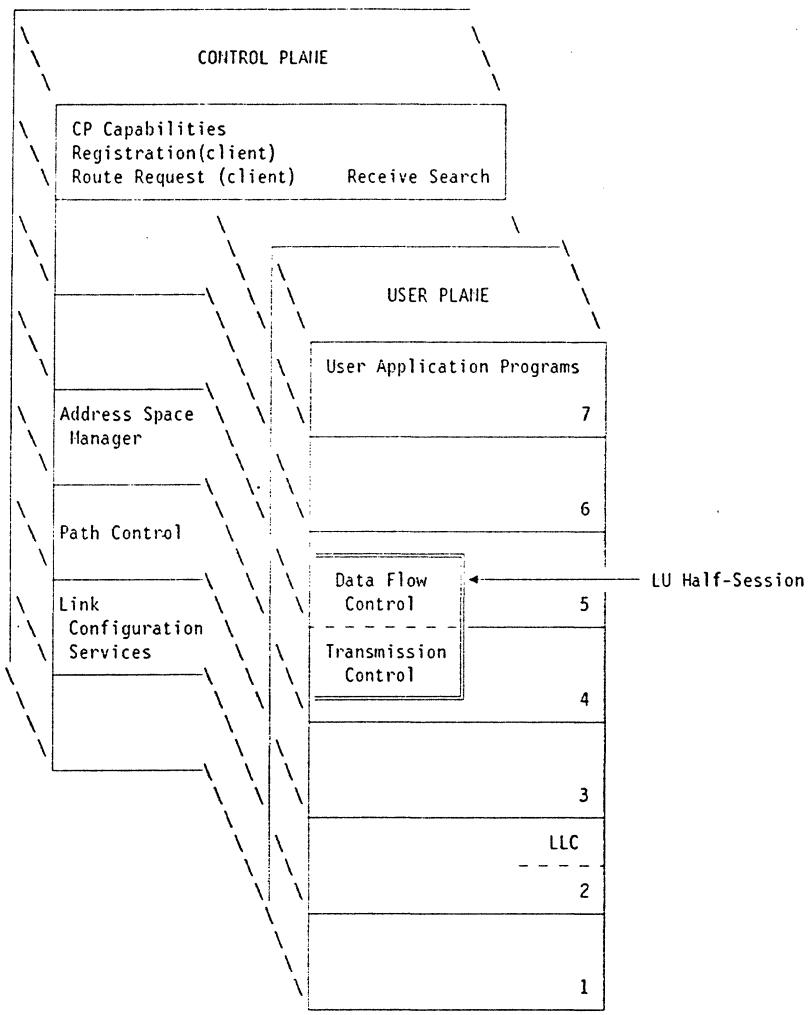


Figure 6. An APPN End Node's Small Control Plane Supporting Networking Client Functions

Unlike the LEN EN which interacts with a network node server passively if at all, an APPN EN actively requests NN services. Some of the benefits over the LEN EN are better dynamics, less definition, and better routes. An APPN EN uses the route provided by its network node server. A different route may be provided every time a new session is set

up, and the route provided does not necessarily traverse the NN. This point is illustrated below. In Figure 4 *A* was a LEN EN; in Figure 7 *A* is an APPN EN.

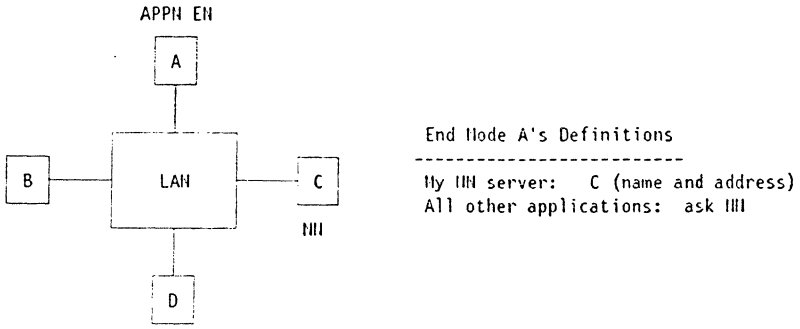


Figure 7. An APPN EN Knowing Only its Network Node Server

The routing function in an APPN EN is still minimal: an EN can only be the endpoint of a session, never an intermediate node of someone else's session. The transport layer of an APPN EN is enhanced by its support for adaptive pacing. The network layer is the same as a LEN EN.

APPN Network Node

A network node adds specialized network control transaction programs at the transaction layer in the control plane, to manage distributed directories and maintain the replicated topology database used for route computation, as well as server support for end node clients.[25]

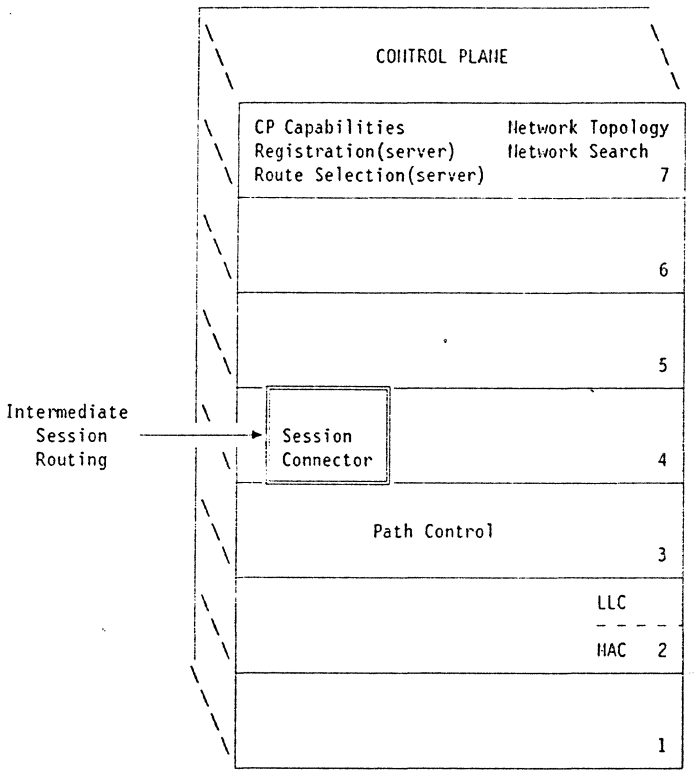


Figure 8. An APPN Network Node's Extended Control Plane and Optional User Plane

It also adds *intermediate session routing*—ISR—the ability to forward packets for applications that do not reside on the NN itself. ISR consists of enhancements at the transport and network layers. One part of ISR is a component called a *session connector*, occupying a similar position and performing similar functions in the protocol stack as the LU half-session in a session endpoint node. ISR functions include error recovery, adaptive pacing, and the adjustment of packet sizes via segmentation and reassembly.

As Figure 8 shows, not every network node has a user plane. A router that does not host any end-user applications is an example of a specialized NN that does not need a user plane.[1]

---

1 Many people—even some of IBM's marketing literature— describe SNA as "non-routable." This is not strictly true. The capability for intermediate session routing previously existed in SNA Type 5 and Type 4 nodes such as VTAM and NCP,

Let's examine intermediate session routing in more detail, in order to compare it with High Performance Routing.

using FID4 transport (layer 4) over explicit routes and virtual routes (ERs and VRs—the SNA path control network—at layer 3). The backbone was physically structured as a mesh, and the peripheral network, using FID2 transport, was strictly hierarchical. Setting up the SNA "routing tables"—ER and VR path definitions—was a laborious manual process or required the use of a tool like NetDA (Network Design Aid). APPN enhances the FID2 transport that first emerged in subarea SNA's peripheral network, and automates the maintenance of routing information and directories. **APPN is native routing technology for SNA.**

EXISTING ROUTING TECHNIQUES

The literature—and vendors' product lines—are filled with a variety of routing techniques, including routing by network address, label swapping, and source routing. And these routing techniques are often supplemented by network control algorithms to dynamically distribute routing tables or maintain a topology database. They may also be supplemented by discovery or address resolution protocols to dynamically map the name of a desired communication partner to a network-layer address or routing information. All the routing techniques discussed below are suitable, in general, for implementation in either hardware or software, while network control algorithms and address resolution protocols are frequently implemented in software.

Routing by network address is the technique used in Internet Protocol (IP). A single 32-bit routing label that must be unique within the scope of an entire internetwork represents the final destination, and serves as an index into a routing table specifying the next hop. The next hop taken depends on the current state of the routing table at the node processing the packet [16]. Several algorithms exist to distribute IP routing tables, some standard and some proprietary. One of the mostly widely used standards, the Routing Information Protocol (RIP), distributes the entire IP routing table periodically at timed intervals. This type of table distribution is called a *path status* algorithm. As individual IP subnetworks become larger, the amount of administrative traffic generated by these regular routing table updates grows exponentially, placing an upper bound on the size of an individual IP subnetwork. IGRP, Cisco's proprietary routing algorithm, also uses a path status algorithm to distribute its routing table updates and, consequently, also places an upper bound on the size of a subnetwork. Within a single IP subnet, it is necessary that all routers support the same algorithm. Hence the focus on standards rather than on proprietary techniques. A relatively new standard algorithm for TCP/IP, Open Shortest Path First (OSPF), is gaining in support among router vendors [16] and uses a more efficient *link-state* type of algorithm (defined below).

Label swapping is a technique used in current APPN (APPN/ISR) and, interestingly, also in the CCITT high-speed recommendation for Asynchronous Transfer Mode (ATM). A packet bears a single network-layer routing label, representing the next hop. A router or high-speed switch substitutes a new label before transmitting the packet. In connection-oriented protocols, the label-swap tables are generally set up in intermediate nodes when the connection is established, based either on predefined information or on a topology database reflecting the state of the network at the time of connection establishment. The APPN topology database updates are only distributed when information changes. This type of algorithm is called a *link state* algorithm. Link-state algorithms generate much less administrative traffic than path status algorithms, removing one barrier to the growth of larger individual subnetworks. TCP/IP's OSPF is also a link state algorithm.

Source routing is a third routing technique, commonly seen in LAN bridging, but also being currently applied in high-speed trials such as the Aurora test bed. In source routing a list of routing labels, representing the entire route, prefixes the packet. The route is determined in advance, usually based on a discovery protocol or a topology database. Some argue that source routing is the most efficient of these three techniques, requiring the least processing at intermediate nodes, yielding the maximum throughput.

LINK-SHARING—A FACT OF LIFE

Whatever routing technique a protocol uses, most state-of-the-art protocols (APPN included) provide a means for different routing stacks to share the transmission medium. A medium access control (MAC) sublayer provides a graceful and standard way to share the link. Examples of media with a distinct MAC sublayer are token ring (802.5) [23], Ethernet, 802.3. and FDDI. The Point to Point Protocol for HDLC (PPP—RFC 1330 [4], [35]); and Frame Relay (Multiprotocol Encapsulation—RFC 1294 [5] and the Frame Relay Forum Implementation Agreements) are not strictly MAC-layer technologies but permit similar link sharing. If the medium has a MAC sublayer, logical channels can be established between paired adjacent link stations on the basis of predefinition, discovery, or a capabilities exchange protocol peculiar to that medium. It is likely that a MAC sublayer or similar standard will also be defined for any new transmission technologies that may emerge in the future.

TRADITIONAL SNA APPROACHES TO ROUTING AND ERROR RECOVERY

Traditional SNA—subarea SNA and APPN/ISR—has been fundamentally connection-oriented in its approach to routing. In general, traditional SNA attempts to provide high reliability over a variety of links (including error-prone links with poor characteristics). Error recovery is performed at the data link layer (layer 2) with connection-oriented data link controls like SDLC, X.25 ELLC, or LAN logical link control type 2 (IEEE standard 802.2). [27] [33] Recovery is also performed at the transport layer—layer 4 (in reassembling segmented data, the transmission control component of the LU's half-session ensures that no packets are missing or out of order) and at the data flow control layer—layer 5 (the half-session enforces chains as the unit of application-level error recovery). Figure 9 illustrates these three levels of error recovery. If any of these protocols are violated, due to failure of the underlying transmission facilities, unrecovered packet losses in the network due to congestion at intermediate nodes, inability of the receiving application to buffer all the received data, or transparent rerouting by a subnetwork that causes some packets to arrive out of order, traditional SNA deactivates the session. With its key design point to support business-critical applications like finance, order entry, inventory control, and credit authorization, traditional SNA has industrial-strength algorithms to detect and prevent the occurrence of these error conditions. In addition, sync point (an APPC checkpointing service) ensures the integrity of distributed databases. In case of session, database, or processor

failure, all applications and databases assume a known state and a distributed transaction can resume exactly where it left off once communication is reestablished.

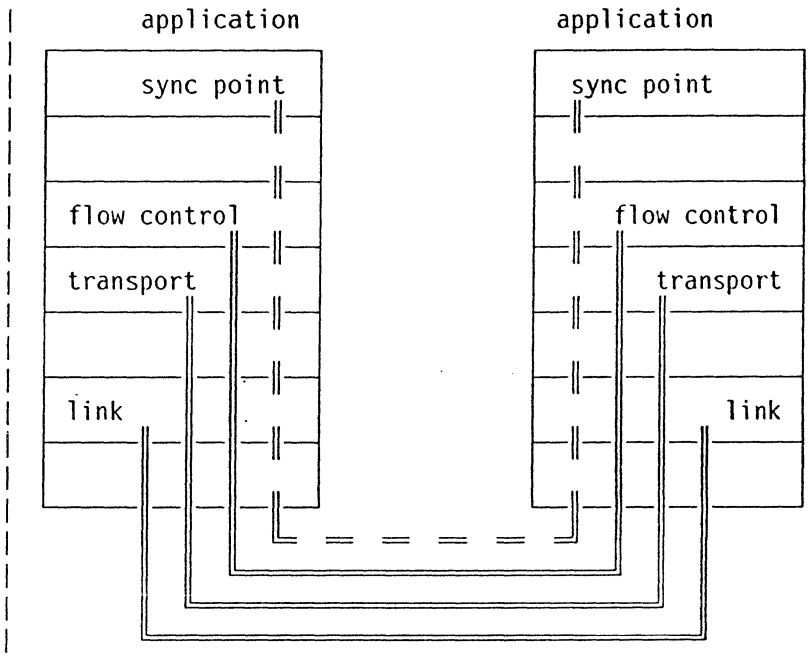


Figure 9. Four Levels of Error Recovery in Traditional SNA

ROUTING IN APPN/ISR

APPN/ISR determines the route for a session when the session is set up; the route remains in effect for its duration. The route is chosen based on a user-specified class of service, a transmission priority, the destination, and the available routes, with randomization if more than one route is acceptable. Every session has a unique identifier (an "FQPCID"), assigned by the origin node, that refers to the session at every node it traverses, throughout its lifetime. This identifier is used for network management and by the transaction programs in the control plane during session setup, but not for routing.

The routing field or network-layer header in APPN/ISR has a single routing label 17 bits long called a *Local-Form Session Identifier* (LFSID), defined by the SNA FID2 transmission header format. It needs to be unique only on a given link. Because it uses FID2, APPN/ISR routing can coexist with pre-APPN traffic, such as 3270 terminal traffic, using the same link. An APPN route is a series of *session stages*, end to end, each with its own LFSID routing label.

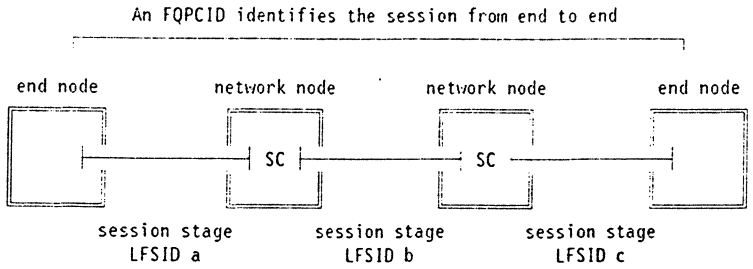


Figure 10. Session Stages Interconnected by Session Connectors

In a network node, the LFSID indexes a "routing table." This table is distinct from the topology database. Each entry in this table is called a *session connector* and is created at the time of session establishment. As an NN forwards a packet from an inbound link to an outbound link, it replaces the LFSID in the packet header with the LFSID from the session connector. APPN routing can therefore be classified as *label-swap routing*. There can be many SNA sessions at once on a logical link, each with a distinct LFSID. APPN nodes usually select LFSIDs dynamically, during session set-up. (Pre-APPN nodes may have LFSIDs preassigned.)

Because of APPN's original design point to support business-critical applications over good-to-poor links, in additional to label-swapping, ISR performs additional, transport-layer functions at intermediate nodes. These other functions include segmenting and reassembly, pacing, and priority queuing for transmission. Figure 11 illustrates function placement in APPN intermediate session routing. Let's examine each of these functions more detail.

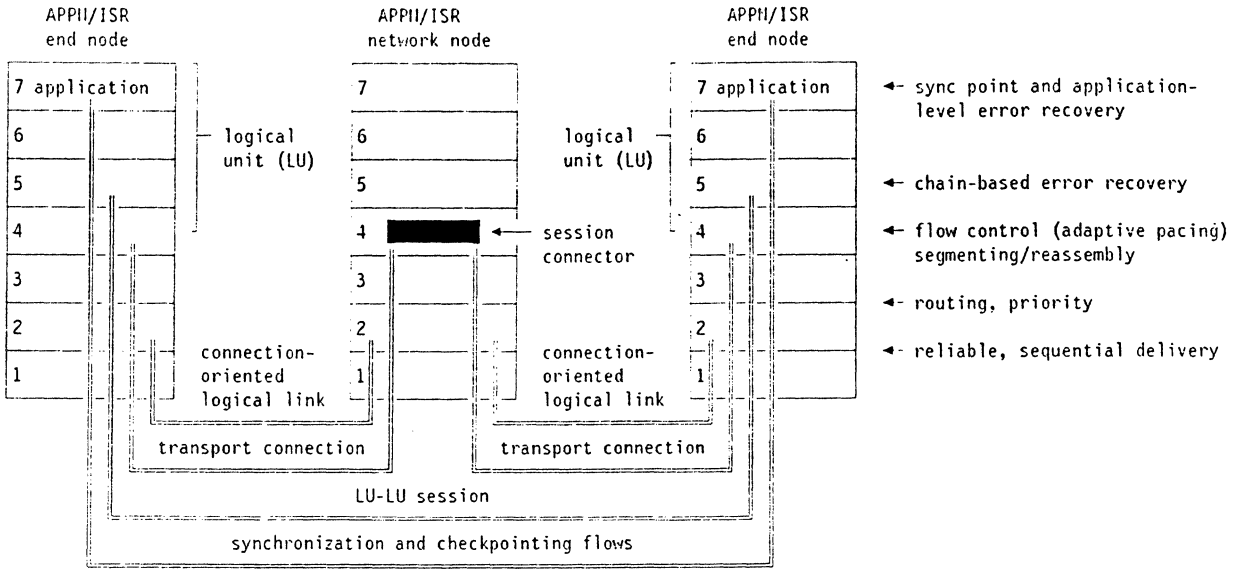


Figure 11. APPN Intermediate Session Routing

Segmenting and Reassembly

Different links in a network may support different maximum packet sizes, for reasons of link speed, transmission delay, data link control timing requirements, fairness, and node buffer capacities[38] . This point is illustrated by one of the graphs from "Data Link Control and Contemporary Data Links" by Traynham and Steen (IBM technical report 29.0168, June 1977), reproduced in Figure 12.

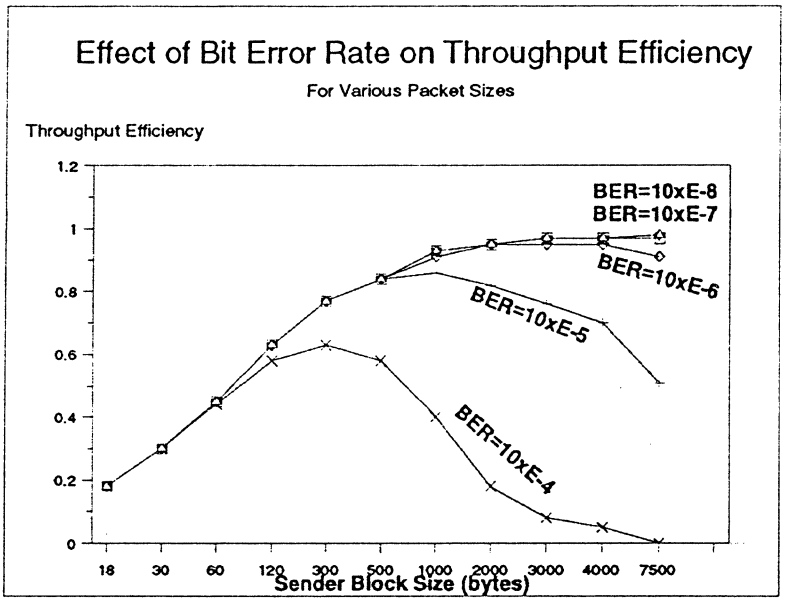


Figure 12. Throughput Efficiency as a Function of Bit Error Rate and Packet Size

In an architecture like APPN that embraces a variety of LAN and WAN transmission media of varying speeds and characteristics, it is not reasonable to expect every node to agree on the same "best" packet size. As a general strategy to maximize performance, APPN/ISR sends the largest packet size allowable on each link. When necessary, segmentation and reassembly are done at intermediate nodes. This is one of the functions performed by APPN intermediate session routing. Any changes to ISR must address the issue of packet size in some manner.

Adaptive Pacing

Pacing is a flow control and congestion control technique to adjust the sender's transmission rate according to the capacity of the receiving node's buffers. Pacing is another function performed by APPN intermediate session routing. In APPN/ISR, pacing occurs independently on each session stage or BIND hop. APPN nodes support both fixed and adaptive pacing. Adaptive pacing is preferred, while fixed pacing permits interoperation with older SNA nodes. Because each APPN session stage is independently paced, every node (nodes supporting applications, as well as routers) can adapt the pacing for the traffic it handles in accordance with its own local congestion conditions. This is the basis for global flow control and congestion management in APPN/ISR. Any changes to ISR must improve upon this already-superior existing function.

With fixed pacing, a predetermined number of packets can be sent before the sender has to wait for the receiver to give permission to send more. Adaptive pacing is a more powerful and flexible scheme wherein a sender can send only a limited number, or *window*, of packets per explicit grant of permission to proceed. The window size is changed dynamically, based on conditions at the receiver. This lets a receiving node manage the rate at which it receives data into its buffers. Adaptive pacing provides a node supporting many sessions, or unpredictable bursts of traffic, a dynamic way to allocate resources to a session that has a burst of activity, and to reclaim unused resources from sessions that have no activity (rather than predefining a buffer pool of a particular size for every active session). Thus we see that adaptive pacing allows the receiving node to use its available buffer resources efficiently. It can also prevent potential protocol deadlocks.

If a node is running low on buffer resources, it uses pacing to tell the upstream node to slow down. If that node becomes congested, it in turn may tell its upstream node to slow down. When a node is not congested, it gives the upstream node permission to send faster. When a receiving node gives a sender permission to send a certain window size, the receiver has reserved sufficient buffers in advance, guaranteeing that data, once sent, will not be lost due to congestion. In practice, many products use statistical or demand buffering schemes, which are acceptable as long as confirmed buffers are available when needed.

A separate instance of adaptive session-level pacing exists for each session running to or through a node, to manage the flow of data on one LU-LU session. Adaptive session-level pacing also applies to the sessions between APPN control points. Adaptive session-level pacing occurs independently on each session stage. Pacing is done by the half-session component of the LU in a node containing a session endpoint, and by the session connector component in an intermediate node. This will become important when we examine how HPR can replace the ISR function (including the session connector) in network nodes, and how HPR can supplement the equivalent component—the LU half-session—in session endpoint nodes.

Priority Queuing for Transmission

Transmission priority permits more important data to pass less important data at queuing points in the network. Priority is another function performed by APPN intermediate session routing. Any changes to ISR must also be equal to or better than the existing support in this area. APPN has four priority levels: a *network* priority, and three session-level priorities: *high*, *medium*, and *low*. Network priority is the highest and is reserved for network control traffic such as pacing messages, topology database distribution, and session establishment. The other three priority levels are for user traffic. A user selects a priority level indirectly, by specifying a mode name defining a session's characteristics. The mode name maps to a class of service definition, which in turn specifies the priority level associated with that class of service. The transmission priority selected for a session is carried in the session activation request (BIND) at session establishment, allowing every node along the path to assign the same priority value, to be used in routing. A transmission priority applies to the session for its lifetime, at every node it traverses. Both ENs and NNs support transmission priority.

Transmission priority is implemented by the path control component in APPN. One function of path control is to direct traffic to the right outbound link. Path control can also multiplex different sessions on a single link. Another function of path control is to ensure that higher-priority data is transmitted before lower-priority data. This is generally implemented as four different queues into which message units are placed, depending on the priority associated with the corresponding session. After the DLC finishes transmitting the current message, path control picks the next message for transmission, selecting from the highest priority queue having a message unit waiting. To ensure that lower-priority data is not preempted indefinitely, an aging mechanism is also used.

A NEW SNA APPROACH TO ROUTING

Many people have observed that some protocols duplicate certain functions at layers 2 (data link control) and 4 (transport), leading to difficulties and ambivalence in discussions of the subject (especially at meetings of international standards bodies!). Furthermore, the current 7-layer model (see Figure 1), mirrored in the organization of standards bodies, does not adequately describe a new class of protocols that are so versatile they can act either as a transport (layer 4) or as a virtual link (layer 2). [19][28][36] The current paradigm is ripe for an overhaul.

Logical link control (LLC—the upper half of layer 2) was originally created to permit the coexistence of both connection-oriented and connectionless service, between multiple link stations, on the same LAN segment. With the extension of this technology by bridging, and then remote bridging which introduced longer and variable end-to-end delays, came sensitivities to the LLC timeout values which are used to detect link or link-station failure. Nevertheless, the principle of extending a data link layer connection across multiple hops is now firmly established. The advantage of data forwarding at a low layer is performance. The disadvantage is that higher layer protocols (like APPN) that select routes based on APPN link characteristics can't see the actual characteristics of the links interconnecting remote bridges. A bridged link appears as a single link in the APPN topology database, with a single set of link characteristics hiding the complexity of multiple hops. The inability to know the true hop count, or to distinguish between a slow link and a fast one, can lead to poor route choices.

One solution to this problem is to replace pairs of remote bridges by pairs of APPN/ISR network nodes. However, this only solves part of the problem, since the ISR functions of pacing and segmentation/reassembly, absent from bridges, would reintroduce delays, and current bridging standards lack support for priority.

A solution was needed to better integrate the bridging concept into APPN. With good links, some of traditional SNA's error recovery is redundant. One possible way to reduce overhead is to omit error recovery at the data link layer, replacing the usual connection-oriented LLC with a connectionless LLC. The transport is replaced by APPN's versatile new transport protocol, RTP—Rapid Transport Protocol, which efficiently provides any needed error recovery over multiple hops.Figure 13 illustrates this approach with a simple two-node network (ignoring for the time being any changes needed in intermediate nodes).

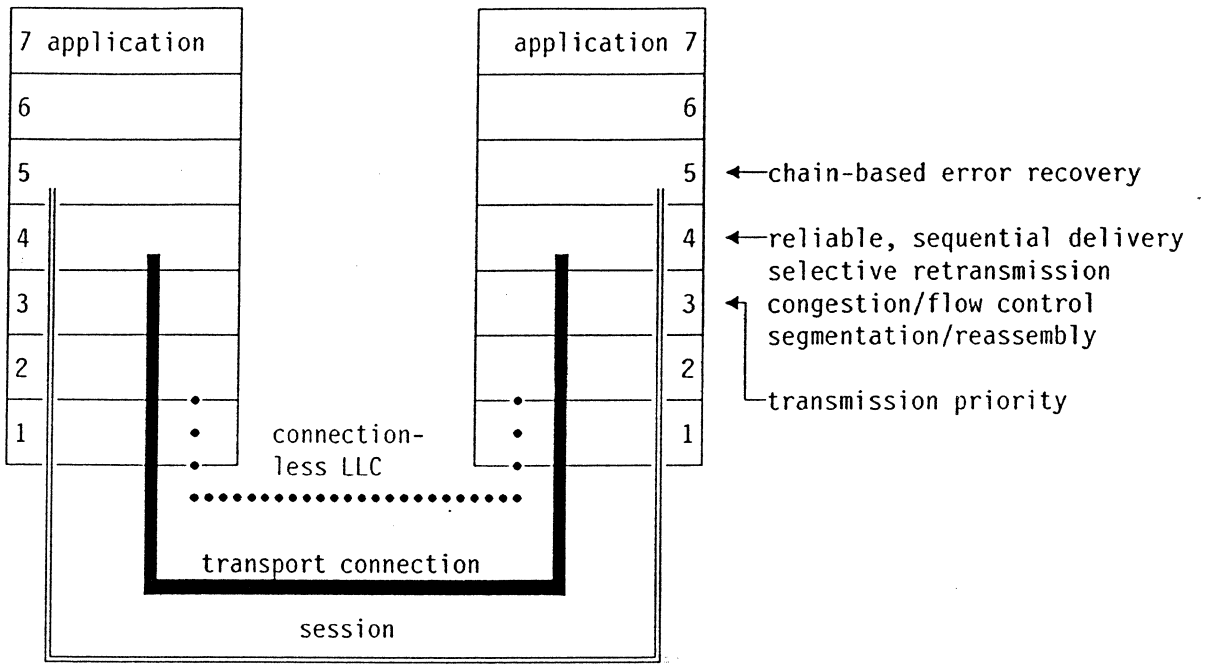


Figure 13. A New Transport with a Connectionless Logical Link Control

One drawback to this approach is the placement of transmission priority (path control) **below** the transport.

An alternative function placement is shown in Figure 14. The new transport becomes part of the LLC sublayer in layer 2, acting as an enhanced logical link. This is true when the logical link comprises not only a single hop, but multiple hops. In this paper we'll call this versatile new class of protocols *transport-oriented logical link controls*. Thus, a transport-oriented LLC like RTP, spanning multiple links and nodes, if it meets the needs of upper layer components to which it provides service, can replace one or more APPN/ISR session stages, acting as a "virtual link." APPN+ takes advantage of this principle (see Figure 15).

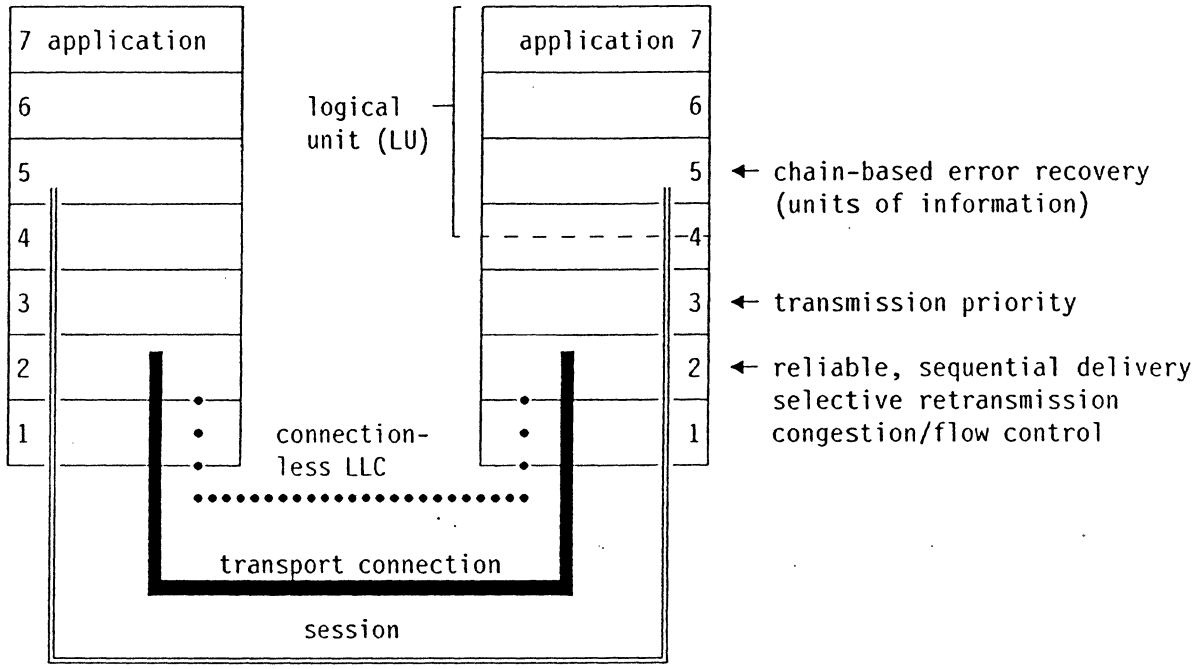


Figure 14. A Transport-Oriented Logical Link Control

The function placement of Figure 14 is better than that of Figure 13 for several reasons. First, it places transmission priority **above** the new transport—important for a node with more than one link. Second, it permits the existing transport (such as an LU half-session—see Figure 6) to be kept intact for the other key services it provides, and because of its integration and packaging into system software (like APPC).

The first property, the coupling of priority scheduling to the rest of the protocol stack, can be understood as follows. In subarea and APPN SNA, the users declare the class of service (COS) needed for their traffic at the transaction program API or user logon API by giving a mode name. The mode name is mapped to a class of service, which in turn specifies both the route through the network (either by listing valid choices in the subarea case, or by defining the parameters of the route selection algorithm in the APPN case) and the transmission priority to be used for the traffic. In subarea, the route is fixed by the ER routing tables and the VR-to-ER mapping, while the priority is carried in the transmission header of each packet. In APPN, the route is fixed by the route selection control vector on the session initiation request (BIND) and the priority is carried in the BIND and saved at each intermediate session routing point, where it is used on the fly for each packet. In both subarea and APPN, the actual priority queueing is done at the top of the DLC component: when the line is finished transmitting the current frame (for example, at the end-of-frame interrupt from the hardware), the highest priority message is taken off the DLC transmit queue (or, in the case of multilink transmission groups, the MLTG transmit queue). An aging algorithm ensures that even low priority traffic gets through under heavy loads.

HPR needs to preserve the relations above, so needs to have priority scheduling queues at the end-of-frame events even on connectionless DLCs. HPR ties the session class-of-service to these queues by encoding priority bits into the HPR headers. If this were not done (e.g., as it cannot be done in networks that lack priority link queuing, or adequate coupling of it to user COS) then HPR would not have preserved the COS semantics at the user APIs.

One may well ask why, if the new transport is so much *like* an LLC, IBM did not choose an existing standard (on a LAN), with traditional bridging (to get across the WAN). The answer is that traditional LLCs are not up to the task. They are restricted to particular media and are not optimized for multiple hops. A new class of transport protocols was needed. APPN/HPR is not limited to LANs: it can run over any transmission medium that supports an unacknowledged (or connectionless) type of service, for example: LAN LLC, LAP-D, LAP-E, or SDLC (using unnumbered information—UI—frames) or X.25 (using QLLC). Furthermore, RTP provides advanced functions like selective retransmission and adaptive rate-based congestion control that no existing standard supports. Another reason is that even when a particular bridging technology supports non-disruptive rerouting at layer 2 (and many new ones, like frame relay or Data Link Switching [34], do), its selection of a new route is not integrated into APPN and is not based on class of service. An advantage for APPN/HPR, as compared with bridging or TCP/IP to span the wide area network, is that its route selection includes awareness of link characteristics (speed, delay, cost, security) and node characteristics (route addition resistance, congestion).

INTRODUCTION TO APPN HIGH PERFORMANCE ROUTING

APPN/HPR augments APPN/ISR's layer 3-4 transport and network functions with two new elements: Rapid Transport Protocol (RTP) and Automatic Network Routing (ANR), shifting the locus of APPN routing from layer 4 down to layer 2. [3] [15] [14]. Each is described more fully below.

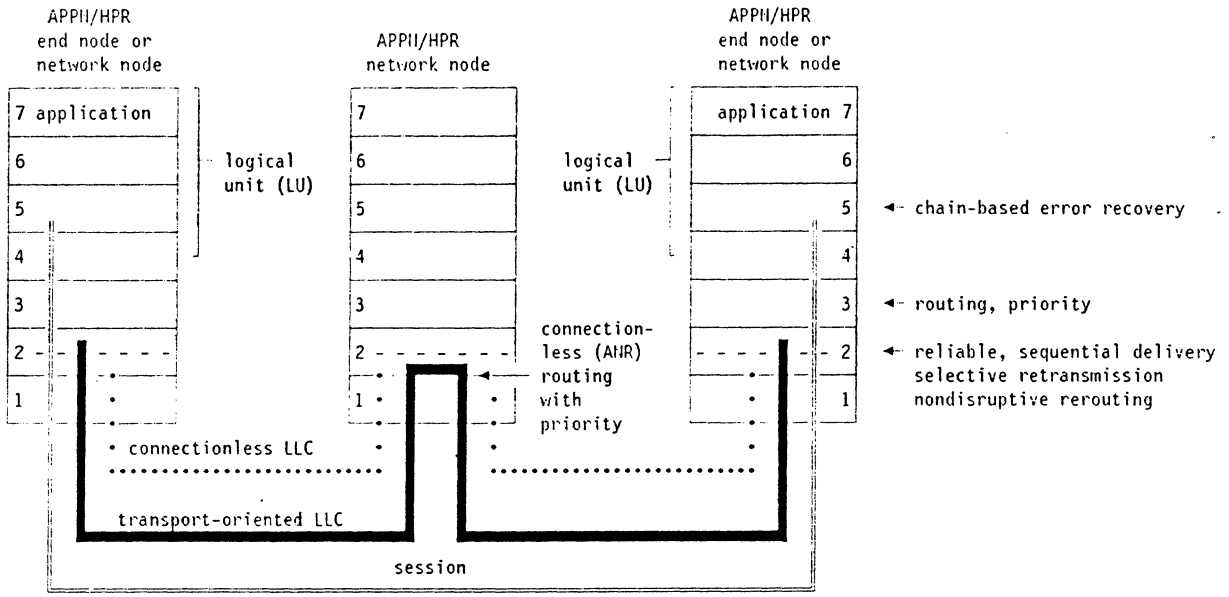


Figure 15. Multiple-Hop Transport-Oriented Logical Link Control. *The data link layer (2) is split into multiple sublayers. Above the standard logical link and MAC sublayers resides the ANR sublayer in all three nodes. In the session endpoint nodes, layer 2 is topped off by an RTP sublayer.*

APPN/HPR also includes an ISR/HPR boundary function (not shown) to adapt an HPR-capable part of the network to an ISR-only part of the network, plugging one side of an ISR session connector to the end of an RTP transport-oriented logical link.

Because APPN/HPR is completely integrated into SNA and does not change APPN's control plane at all, any node can be upgraded to the HPR level of function transparently, continuing to interoperate with adjacent nodes still at the ISR level of function. As soon as two or more adjacent nodes are HPR-capable, initial benefits of HPR—non-disruptive rerouting, adaptive rate-based congestion control, selective retransmission, fair multiprotocol transport—start to be realized. As soon as two or more HPR-capable links exist back-to-back, such that high performance routing replaces intermediate session routing in at least one intermediate node (shown in Figure 15), further HPR benefits are experienced—fast routing *with* priority and reduced intermediate node storage.

HPR provides a connection-oriented transport (RTP), end to end, over connectionless source routing (ANR), over the minimal data link control. RTP acts as a "virtual link." The amount of function that RTP demands of the underlying DLC depends on the quality of the transmission medium. On high quality links, the DLC is not asked to provide reliable delivery, sequence numbering, retransmission, or acknowledgment. It merely provides a frame check sequence: errored packets are simply discarded. Links with bit error rates on the order of 10^{-7} or better are good candidates to use a connectionless DLC under RTP. Such links typically use digital transmission over fiber media. On high-error-rate links a

connection-oriented DLC with error recovery may be used under RTP. In either case the benefits of HPR are significant.

An HPR path can also include multilink transmission groups—essentially, a bundle of links between adjacent nodes that are treated as a single "fat" pipe. Benefits of MLTG include high availability (if one link of a multilink group fails, the MLTG remains operational) and bandwidth on demand (additional switched links can be dialled up to augment the bandwidth an existing link). Long a feature of subarea SNA networking, MLTG can easily be added to HPR, without the performance and storage penalty of refifoing disordered packets at each MLTG hop. Reordering only needs to be performed at the endpoints of the RTP logical link, a task RTP was designed to accomodate.

RTP insulates the upper layers—the LU—and the user from any awareness of path switching, multipath routing, network-related congestion control activities, retransmissions, acknowledgments, packet resequencing, multiplexing, and so forth. Thus a user's investment in existing SNA applications is completely preserved.

ANR—Automatic Network Routing

The functions of ANR used by APPN/HPR are the following:

- Source routing with locally specified labels
- Connectionless, stateless, fast routing
- Discarding incoming packets in the event of congestion
- Servicing the outbound transmission link based on priority.

There's not much more to say—ANR is elegant and simple. [3], [15], [14]. ANR functions are done at every node along the path of an RTP transport-oriented logical link.

RTP—Rapid Transport Protocol

The functions of RTP used by APPN/HPR are the following:

- Connection awareness (of each individual session using the RTP logical link, the session partners of that session, the transmission priority, the current ANR route for the network connection, and if a path switch has occurred, all previous routes used by the logical link over its lifetime)
- Optional reliable delivery (sequence numbering, acknowledgment, and selective retransmission)
- Reordering, if needed (may be needed after a path switch, or if the route contains one or more multilink transmission groups)

- Determining the smallest maximum packet size along the path of the RTP logical link and ensuring (through segmentation) that all message units offered to the link are the proper size (this function eliminates the need for segmentation and reassembly at intermediate nodes)
- Flow control and congestion control/avoidance (Adaptive Rate-Based—ARB—explained in a subsequent section)
- Providing an interface to the ANR sublayer (below)
- Providing an interface to SNA path control (above)
- Non-disruptive route switching
- Multiplexing (more than one session having the same class of service and transmission priority requirements can share a single transport-oriented logical link).

RTP is connection oriented. These RTP functions are done only at the endpoints of an transport-oriented logical link, not at any intermediate nodes (as shown in Figure 15).

At this point, the reader may be struck by apparent similarities between HPR and TCP/IP. One of the features that sets them apart, however, is HPR's congestion control mechanism, described in the following section.

ARB—Adaptive Rate-Based Congestion Control

A new technique for flow control and congestion control was needed for APPN+, to compensate for the loss of adaptive pacing function in intermediate nodes. This new function is called ARB—adaptive rate-based congestion control.

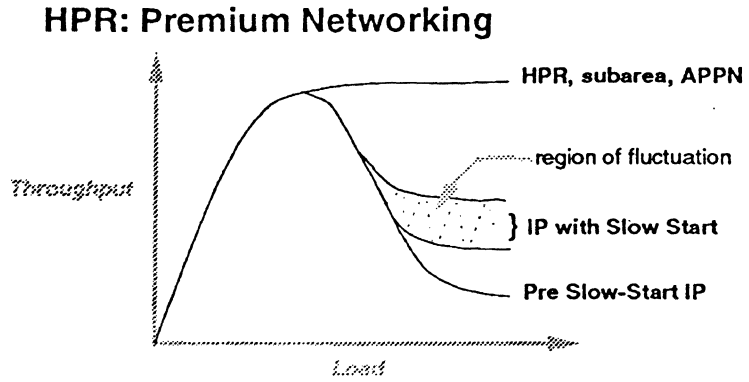
The function of ARB is to regulate the input traffic (load) offered to the RTP logical link in the face of changing network conditions. It is preventive, rather than reactive. When the network approaches congestion (increasing delay, decreasing throughput), ARB reduces the input traffic rate until the network's capacity is restored. When possible, ARB increases the sending rate without exceeding the rate that the receiving endpoint can handle.

ARB uses a closed-loop feedback mechanism based on information exchanged periodically between RTP components at the **endpoints** of an RTP logical link. (No ARB function is performed in intermediate nodes.) The feedback consists of information about two rates: the rate at which RTP accepts data arriving from the network, and the rate at which the RTP hands off the data to a recipient (such as the SNA path control component). Based on this feedback, the sender predicts when congestion is likely to develop, and takes steps to prevent it. If congestion does occur, the sender takes stronger measures to bring the network back to normal.

Because ARB addresses fairness at the data link layer, among different RTP logical links (which may be carrying multiple SNA sessions), APPN's existing adaptive session-level pacing algorithm continues to be used in networks with APPN/HPR, to provide fairness among SNA sessions at the data flow control layer (layer 5) in the LU half-session.

Analysis and simulation convinced researchers Rong-Feng Chang et al. [13] that ARB is superior to the "slow-start" congestion control algorithm of standard TCP/IP architecture [1], introduced after the Internet experienced a series of congestion collapses in October of 1986 [30]. In particular, ARB allows high link utilization rates (on the order of 80—90%) and is preventive, rather than reactive. The same study suggested that "slow-start" causes expensive link under-utilization, with lower design loadings (which should be a matter of concern to network administrators and people with budgets). The study also concluded that TCP slow-start exhibited unfairness and bias against certain kinds of traffic due to wide oscillations in packet delay and throughput (these should be of great concern to network users). Additional deficiencies of TCP slow-start cited in [13] included periodic packet losses, systematic discrimination against particular connections, bias against connections with long round-trip times, and bias against bursty traffic. It is worthwhile to note that many IP router vendors address these *TCP architecture* defiencies with proprietary extensions or product-internal enhancements. One informal mechanism used by several vendors is to give routing priority to short packets, which are assumed to be acknowledgments.

Figure 16 illustrates the relationship of effective congestion control algorithms to throughput and response time, which translate directly into cost savings and user satisfaction.



	HPR	Standard TCP/IP (*RFC 1122)
Sensing	Predictive	Reactive
Control	Rate	Windows
Line Loading	High	Lower
Response Time	Stable	Fluctuating

| Figure 16. Relationship of Effective Congestion Control to Throughput and Response Time

HIGH PERFORMANCE ROUTING VERSUS INTERMEDIATE SESSION ROUTING

Like APPN/ISR, APPN/HPR is connection-oriented, with the entire route for a session determined in advance.

Unlike ISR, which uses label-swapping, the entire route prefixes each HPR packet. The route is encoded as an arbitrary-length string of routing labels. This technique is classified as *source routing*. (The general concept of source routing may be familiar, being used in some LAN MAC-layer bridging protocols [23]). The particular source routing technique used for APPN/HPR is ANR—automatic network routing. The labels vary in length, typically 1-2 bytes. Each routing label has local significance only: it is meaningful relative to the node processing the label. The first label always represents the next hop (or, at the last
| node, the terminus of the RTP logical link). Routing consists of stripping off the first routing label and transmitting the packet on the link indicated by the stripped label.

| Because of ANR's simplicity, existing platforms that implement it may well realize significant performance gains of 3—10 times on their existing hardware. That is, on many platforms, HPR function might be deliverable in the form of a software upgrade. (By contrast, other advanced routing techniques such as ATM often require expensive and specialized

new hardware.) Performance gains at the upper end of this range can be expected on hardware optimized for ANR.

A single RTP logical link between HPR-capable endpoints replaces two or more back-to-back ISR session stages. There can be many RTP logical links at a time on a transmission link; intermediate nodes are unaware of, and unconcerned with, these individual connections: they only see a stream of packets prefixed with routing labels representing their outbound links.

Many SNA sessions can share a single RTP logical link, provided they have the same class of service and transmission priority.

If a path fails, the RTP component at the endpoint of the logical link obtains a new path, still based on the desired class of service and transmission priority, and keeps going—all without the session's awareness. If packets get lost or out of sequence during the switch, RTP takes care of it transparently. As a result of path switch, it's even possible for the forward and reverse directions of data flow to follow different routes through the network.

FRAME RELAY—A HIGH SPEED "SDLC" FOR HPR

While SNA absorbs and runs over many different types of links, from S/390 channels to the synchronous framing mode of SDLC, SDLC has always played an important role in both subarea SNA and in APPN. In some sense, it has been the template DLC, the one on which the others were modeled. This shows up in the presence of XID exchanges on other DLCs, when XID is the name of a control frame within SDLC. SDLC, however, has the disadvantage of being a single access link, as contrasted with 802.x LANs and frame relay, both of which support link connections to multiple partners through a single hardware connection. Multiple access helps to reduce costs of ports and access lines into private or public carrier networks. And, while 802.x MACs have difficulty in working directly over WAN distances and at commonly available carrier line speeds, frame relay is well-adapted to use in WAN configurations.

In light of its benefits, frame relay has been adopted as one of the two preferred DLCs for HPR: preferred in the sense that products are encouraged to implement both frame relay and 802.x LAN connections for HPR support. This encouragement stops short of a mandatory architecture requirement because of the wide diversity of products and market niches in the world, but certainly the general-purpose networking products such as the IBM 6611, 3745, 3174, 3172, AS/400, and CM/2 products are expected to support both FR and 802.2 LANs for HPR connections.

HPR's support of FR includes both "through a carrier" and "null carrier" configurations. Since FR uses HDLC framing, it runs on existing SDLC adapters as a software or microcode upgrade; being configurable as null network connections allows it to be used wherever point-point full-duplex SDLC lines are now used, without changing line provisioning in

| any way. Multidrop SDLC lines pose another problem: some can be converted to multiple-
| connection FR services through a carrier network (the carrier's network becomes the
| "multidrop" in a certain sense, but with added function as compared to multidrop since the
| multidrop polling is removed); some can be reconfigured to use APPN/HPR routing ser-
| vices, perhaps even with line cost savings; some will have to remain as SDLC multidrop
| until we can support multidrop FR configurations (while multidrop is not part of the FR
| standard, it is easy enough to add, and we intend to do so).

| AREAS FOR FUTURE STUDY

| It should also be possible to have multi-path RTP logical links—so that, by choice, a
| session uses two or more different ANR paths at once. This is sometimes called *bifurcated*
| routing. This would be advantageous in several ways. It could further reduce the impact
| of network failures (provided that at least one path remains viable). It could provide a way
| to aggregate the physical capacity of several links, each with inadequate bandwidth by
| itself, into a trunk with sufficient bandwidth. It could allow network providers to take advan-
| tage of costly parallel capacity, installed for reasons of high availability.

| While such an option adds complexity to HPR's non-disruptive rerouting logic and to
| network management, it appears to be a fruitful avenue for further research. Another pos-
| sibility is to couple this function with awareness of actual bandwidth requirements of con-
| nections and the knowledge of link utilization in a sophisticated high-speed network.

| Also for a future study is a thorough survey of the existing literature on congestion control
| and analysis of trends concerning connection-oriented and connectionless transports.

CONCLUSIONS AND IMPLEMENTATION RECOMMENDATIONS

We've shown how APPN/ISR and APPN/HPR can work together and discussed HPR routing in some detail. What sort of a product can benefit from implementing HPR? An obvious candidate is a router. A router typically supports several high-speed WAN link attachments and has the capacity to switch large numbers of packets. This appears to be natural fit with HPR's design points.

Can a computer that supports user applications (typically, an end node) also benefit from implementing HPR? We believe the answer is a resounding **yes.** Consider a typical installation with many individual workstations attached to a LAN—possibly quite a large, bridged LAN, shown in Figure 17. Several routers (3 and 4, 10 and 11) provide connectivity between stations on the LAN (1 and 13) and a WAN backbone made up of additional routers and packet switches (not shown). So far, we have shown HPR's benefit when a WAN link fails (if any of links 5—9 fails, the HPR-enabled routers can switch to a new route across the WAN). But one class of failures for which few solutions exist today is failure of a WAN access node such as routers 3, 4, 10 or 11. The typical user (1) is in session with applications both on the local LAN (2) and across the WAN, perhaps to a remote LAN (12, 13). Today, if the router or LAN segment or bridge through which a session is routed goes down or experiences problems, the session often fails. If the user's computer (1 or 13) includes HPR function, path switching will likely be successful if a local path to an alternate NN exists. Performance and storage benefits are also realized in network nodes 3, 4, 10, and 10 if end nodes 1, 3 support HPR.

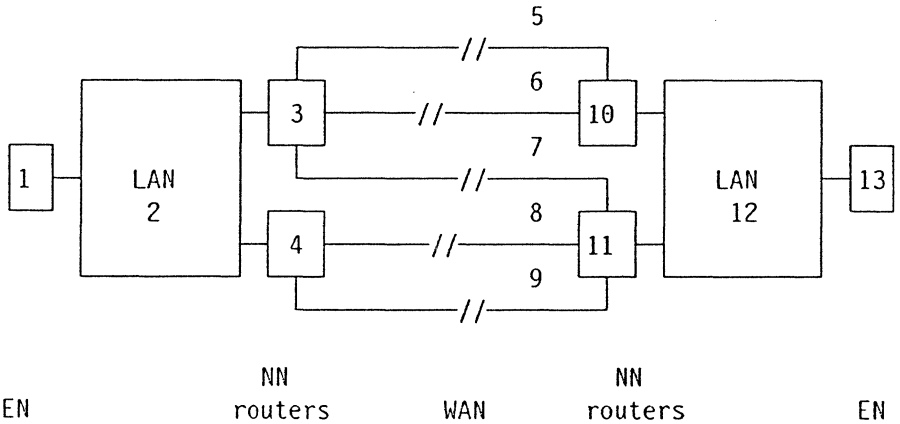


Figure 17. Benefits of HPR for Both End Nodes and Network Nodes

SUMMARY

APPN/HPR, also known as APPN+, is a promising new technology for network nodes and end nodes that transparently extends SNA, replacing the intermediate session routing function in selected nodes with the faster high performance routing. APPN/HPR includes a new connection-oriented transport layer protocol, Rapid Transport Protocol (RTP), one of a new class of transport protocols that can also serve as a logical link (with priority) over multiple hops. APPN/HPR also includes a new type of connectionless source routing called Automatic Network Routing (ANR). APPN/HPR provides nondisruptive rerouting based on class of service, fast packet switching, minimal intermediate node storage, a new adaptive rate-based congestion prevention algorithm (ARB), and a drop-in migration to existing SNA networks based on an ISR/HPR boundary function, for seamless interoperation with current SNA products and protocols.

ACKNOWLEDGMENTS

The authors are indebted to their colleagues at IBM Networking Systems, Ray Bird and Lap Huynh, for creating the HPR architecture specification, and to Dr. Raif Onvural for his kind encouragement, without which this article would not have been written. Thanks are also due to Barry Groner, Jane Munn, and Rick McGee, managers who supported early publication of this material.

REFERENCES

[1] R. Braden (editor), Requirements for Internet Hosts Communication Layer RFC 1122, 1989.

[2] Michael O. Allen and Sandra L. Benedict, "SNA Management Services Architecture for APPN Networks," *IBM Systems Journal*, vol. 31, no. 2, pp. 336-352, 1992. Describes network management architecture for APPN.

[3] B. Awerbuch, Israel Cidon, Inder S. Gopal, Marc Kaplan, and Shay Kutten, "Distributed Control for PARIS," *Proceedings of 9th ACM Symposium on Principles of Distributed Computing*, Quebec, Canada: ACM, August 1990.

[4] F. Baker, The Point-to-Point Protocol Extensions for Bridging RFC 1220, 1991.

[5] T. Bradley, C. Brown, and A. Malis, Multiprotocol Interconnect Over Frame Relay, 1992.

[6] IBM Corp., AS/400 Distributed Systems Implementation Guide, GG22-9458. Discusses the decision criteria that must be considered when choosing a topology for an AS/400 APPN network

[7] IBM Corp., S3/X and AS/400 APPN Nodes Using the SNA/LEN Subarea, GG24-3288. Describes the incorporation of a S/370 SNA subarea into a network comprising APPN network nodes. Intended for systems programmers and systems engineers in the intermediate systems and VTAM/NCP areas.

[8] IBM Corp., APPN/Subarea Networking Design and Interconnection, GG24-3364. A guide for planning interconnection of APPN and SNA subarea networks.

[9] IBM Corp., Networking Services/2 Installation, Customization, and Operation, GG24-3662. Provides planning information for IBM SAA Networking Services/2. Contains an extended example on connecting Networking Services/2 and AS/400, with their respective configurations.

[10] IBM Corp., APPN Architecture and Product Implementations Tutorial, GG24-3669. Tutorial on APPN, with an overview of various product implementations.

[11] IBM Corp., 3174 APPN Implementation Guide, GG24-3702. Provides guidance on implementing the 3174 APPN functions in various scenarios.

[12] IBM Corp., AS/400 APPN with PS/2 APPN, 3174 APPN, 5394 and Subarea Networking, GG24-3717. Provides several scenarios of interaction of these nodes including sample definitions and traces.

[13] Rong-Feng Chang, James P. Gray, and Lap Huynh, Comparison of Congestion Control Performance of APPN+ and TCP, IBM Corp. Unclassified, Technical report 29.1490, December 1992. Request ARBTCPN PACKAGE from MPETERS at RALVM6. Describes dynamic problems of TCP and shows that APPN/HPR with ARE significantly outperforms TCP congestion control. Appendix B contrasts APPN and TCP/IP.

[14] Israel Cidon and Inder Gopal, "Paris: An Approach to Integrated High-Speed Private Networks," *Int. Jour. of Digital and Analog Cabled Sys. 1*, pp. 77-85, 1988.

[15] Israel Cidon, Inder Gopal, and Shay Kutten, "New models and algorithms for future networks," *0-89791-277-2/88/0007/0075*, ACM, 1988.

[16] Douglas E. Comer, *Internetworking with TCP/IP: Principles, Protocols, and Architecture*, Prentice Hall, 1991.

[17] IBM Corp., Client/Server Computing: The New Model for Business, IBM Corp., February 1992.

[18] Rudy K. Cypser, *Communications for Cooperating Systems: OSI, SNA, and TCP/IP*, Addison-Wesley Publishing Co., 1991.

[19] W. Doeringer, D. Dykeman, M. Kaiserswerth, B. Meister, H. Rudin, and R. Williamson, "A Survey of Light-Weight Transport Protocols for High-Speed Networks," *IEEE Transactions on Communications*, vol. 38, no. 11, pp. 2025-2039, November 1990. Surveys and classifies high-speed transport protocols

[20] J. P. Gray, The Future of Networking,. presentation at Duke University, March 1993

[21] P. E. Green, R. J. Chappuis, J. D. Fisher, P. S. Frosch, and C. E. Wood, "A Perspective on Advanced Peer to Peer Networking," *IBM Systems Journal*, vol. 26, no. 4, pp. 414-428, 1987.

[22] IBM Corp., AS/400 Communications: APPN Network User's Guide, Publication number SC41-8188. Describes the APPN support provided by the AS/400 system. Also describes APPN concepts and provides information for configuring an APPN network. APPN advanced considerations and configuration examples are included.

[23] IBM Corp., IBM Local Area Network Technical Reference, SC30-3383. Describes token ring LANs, including source routing bridging.

[24] IBM Corp., Systems Network Architecture Type 2.1 Node Reference, SC30-3422-2 (March 1991). Defines the architecture for APPN end node and LEN end node at the pre-HPR level of function.

[25] IBM Corp., Systems Network Architecture APPN Architecture Reference, SC30-3422-3 (March 1993). Defines the architecture for APPN network node, APPN end node, and LEN end node at the APPN/ISR level of function.

[26] IBM Corp., Networking Services/2 Installation and Network Administrator's Guide, SC52-1110. Describes the APPN support provided by Networking Services/2 for OS/2 Extended Edition. Also describes APPN concepts and provides information for configuring an APPN network.

[27] IEEE, Project 802—Logical Link Control. This standard defines the 802.2 Logical Link Control protocol.

[28] Protocol Engines, Inc., XTP Protocol Definition, 1989. This defines the Express Transfer Protocol, one of a new class of optimistic transport layer protocols.

[29] ISO, Information Processing Systems—Open System Interconnection—Basic Reference Model IEEE 7498, 1984. Defines the ISO 7-layer reference model.

[30] V. Jacobson, "Congestion Avoidance and Control," *ACM SIGCOMM*, vol. '88, pp. 314-329, August 1988. Describes the "collapse of the internet" before the invention of TCP slow-start.

[31] Steven T. Joyce and John Q. Walker II, "Advanced Peer-to-Peer Networking (APPN): An Overview," *ConneXions--The Interoperability Report*, vol. 6, no. 10, pp. 2-9, October 1992.

[32] Steven T. Joyce and John Q. Walker II, "Advanced Peer-to-Peer Networking (APPN): An Overview," *IBM Personal Systems Technical Solutions*, no. G325-5014-00, pp. 67-72, January 1992.

[33] Matthias Keiserswerth, "A Parallel Implementation of the ISO 8802-2.2 Protocol," *IEEE Tricomm '91*, April 1991.

[34] David Kushi and Roy C. Dixon, Data Link Switching: Switch-to-Switch Protocol RFC 1434, 1993.

[35] D. Perkins and R. Hobby, The Point-to-Point Protocol (PPP) Initial Configuration Options RFC 1172, 1990.

[36] Robert M. Sanders and Alfred C. Weaver, The Xpress Transfer Protocol (XTP) - A Tutorial, Computer Networks Laboratory, Dept. of Computer Science, Univ. of Virginia, TR-89-10, January 1990.

[37] Robert A. Sultan, Parviz Kermani, George A. Grover, Tsippi P. Barzilai, and Alan E. Baratz, "Implementing System/36 Advanced Peer to Peer Networking," *IBM Systems Journal*, vol. 26, no. 4, pp. 429-452, 1987.

[38] Kenneth C. Traynham and Robert F. Steen, Data Link Control and Contemporary Data Links, IBM Corp. Unclassified, Technical report 29.0168, June 1977. Analyzes the relationship between bit error rate, packet size, and number of frames outstanding on throughput efficiency. Based on mathematical models, with many graphs.

A PREVIEW OF APPN HIGH PERFORMANCE ROUTING

James P. Gray, Marcia L. Peters[1]

IBM Corporation
SNA Studies
[1]Networking Systems Architecture
Internal Mail Address C74/673
P.O. Box 12195
Research Triangle Park, NC 27709

Abstract: After a brief review of APPN—Advanced Peer-to-Peer Networking—and a survey of existing routing techniques, a new SNA approach to routing called HPR—APPN High Performance Routing—is introduced. Topics covered in this overview include HPR function placement within the OSI layered model, priority scheduling for multilink transmission groups, Automatic Network Routing, Rapid Transport Protocol, Adaptive Rate-Based congestion control, the relationship of effective congestion control algorithms to throughput and response time, and HPR's selection of frame relay as a preferred data link control.

INTRODUCTION

What is APPN?: APPN—Advanced Peer-to-Peer Networking—is an extension of SNA—Systems Network Architecture. APPN was first announced in 1987. At the time of writing, APPN was available on the following IBM products:

- AS/400 [21] [6] [7]
- 3174 Establishment Controller [11]
- OS/2 Communications Manager [25] [9]
- System/36 [7]
- DPPX/370
- System/390 mainframe and 3745 front end processor (VTAM 4.1, NCP 6.2)
- 6611 Network Processor router

Availability on the RISC System/6000 via AIX SNA Services is planned for 1993. A number of other companies also offer APPN in their products, including Brixton, InSession Inc., and Systems Strategies Inc. Additional vendors have demonstrated APPN prototypes or are expected to offer APPN in their products, including: Advanced Computer Communications, Apple Computers, 3Com, Cabletron Systems, Cisco Systems, CrossComm, Data Connection Ltd., Network Equip-

ment Technologies, Network Systems Corporation, Novell, Retix, Siemens-Nixdorf, Ungermann-Bass Access-One, Unisys, and Wellfleet Communications. IBM opened the APPN End Node protocols in 1991 by publication of the SNA Type 2.1 Node Reference [23]. In March 1993 IBM opened the APPN Network Node protocols by source code licensing, technology licensing, and publication [24].

What is APPN/HPR?: APPN/High Performance Routing, which IBM called *APPN+* when revealing its future networking directions in March, 1992, is a further extension of SNA to take advantage of fast links with low error rates. It replaces ISR (intermediate session routing)—the routing technique in current APPN products—with HPR (high performance routing). One of its key benefits is the ability to nondisruptively reroute sessions around failed nodes and links. APPN/HPR can improve intermediate routing performance by 3 to 10 times, greatly reduce network node storage requirements, and augment existing LU-based flow control with an advanced congestion avoidance algorithm called ARB—Adaptive Rate-Based congestion control. In late 1992, IBM said it intends to make APPN/HPR technology available in a future release of its network node source code license, then in March 1993, IBM announced plans to publish a draft specification of the HPR formats and protocols in the third quarter.

APPN'S CLIENT-SERVER MODEL

The term *peer-to-peer* is misleading because it captures only one of two equally important aspects of APPN. Equally descriptive terms could have been *distributed* or *client/server. Distributed* means decentralized. APPN is not locked into the strict hierarchical topology required by its predecessor, SNA subarea architecture: APPN architecture supports a variety of physical topologies such as star, hub, mesh, and hierarchical, over a variety of transmission media including most popular LAN and WAN media. *Distributed* says that important data and important applications may reside on any computer in the network. In other words, computers other than mainframes are running significant applications. Such applications are network-centric rather that mainframe-centric. The term *peer* accurately characterizes this aspect of APPN. *Client-server* says that in application design, function placement is flexible [17]. It recognizes that computers differ in terms of physical location and physical security, the level of attended support (e.g. operator intervention, regular backup, and so forth), connectivity to other computers, permanent storage media (disk, tape, and the like), the amount of RAM for running large applications, support for multitasking, and computing capacity (MIPs, FLOPs). Client-server principles encourage flexible application design so that smaller computers can take advantage of the capabilities of larger computers. It is the client-server aspect of APPN that the term *peer* does not suggest.

With client-server concepts in mind, and to support emerging networked applications for distributed computing, APPN defines two main node types: the end node (EN) and the network node (NN). APPN also interoperates well with a pre-APPN node type called the Low-Entry Networking end node (LEN EN), a subset of the APPN EN. LEN was a precursor to APPN; APPN is the strategic base for enhancements to SNA. (All three of these node types are classified as SNA Type 2.1 nodes.) This general architectural structure makes sense. The control plane in an NN requires more physical resources (CPU, RAM, disk) to support the network control applications and services that it provides. By offloading most of these functions to an NN, an APPN EN can be relatively small and inexpensive, and/or have more of its resources left for applications, and still partake of APPN's automation and dynamics. An NN may also have specialized routing hardware and attachments to many WAN links.

A network node is a **server**, in the sense of providing network services to its end node clients. The services are directory services and route selection services. An brief overview will illustrate the respective client and server roles of ENs and NNs.

Directory services means locating a partner application: determining what computer the application currently resides on. This avoids the EN having that information predefined. This also gives network administrators flexibility in moving an application to the computer best able to support it. With APPN, moving an application requires only local definition changes at the computer where the application used to be, and at the computer it moves to. All other computers learn this information dynamically, as needed.

Route selection services means picking the best route for the session, based on a user-specified class of service and transmission priority, what possible routes are currently available through the network, and the destination computer's location. Route selection also avoids congested nodes, and randomizes among equivalent routes, in order to distribute the load.

These network control applications of APPN have been described extensively in the literature [30], [31], [10], [8], [12], [36], [20]. Less often discussed are its lower layers.

A network node is also a **router**: it forwards traffic that crosses it on the way to somewhere else. The second half of this article discusses APPN routing techniques in more detail.

First though, let's examine the differences in these APPN node types using a layered architecture model. We will cover differences in network control applications (in the uppermost, or transaction, layer) and differences in the lower layers (Layers 2, 3 and 4—the data link, network and transport layers). With this foundation we can examine APPN high performance routing, describe its benefits, show how it complements existing APPN intermediate session routing functions, and consider what kinds of networking products it is appropriate for.

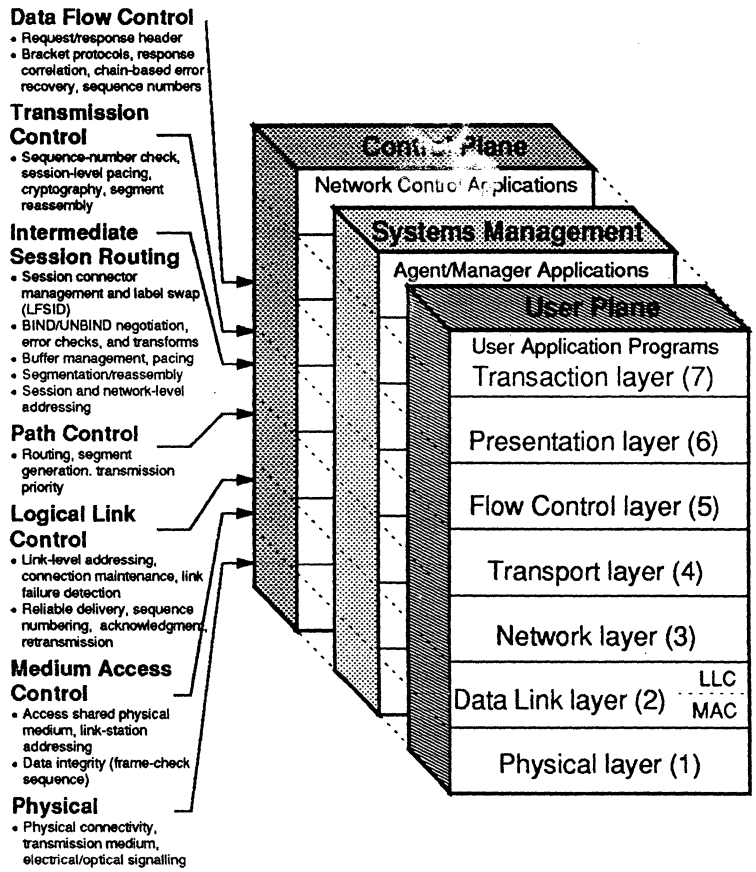


Figure 1. Layered Model Showing SNA Function Placement

Figure 1 shows APPN's functions divided approximately into horizontal strata, according to the OSI networking model, like a layer cake. [18], [28] In addition, one can imagine vertical slices through these layers, like a slice of cake, containing some cake from each of the layers. There may be chunks of fruit in the cake, so that each slice is a little different. One of these slices is the *user plane*, the piece of system software on a computer directly supporting end user application programs running on that computer. APPC—Advanced Program to Program Communication—also called LU 6.2 or Logical Unit type 6.2—is an example of a user plane. Another slice is the *systems management plane* (shown behind the user plane). Systems management is often structured according to client-server principles, with a client component in an end-user's computer being called an *agent* or *entry point*, and a server component in a specialized (but not necessarily centralized) computer being called a *manager* or a *focal point*.[2] The slice that this paper focuses on is the *control plane*, sometimes called the *control point* in an SNA node. Its job is to support the user and management planes, automating such chores as the distribution of routing information and directories. Bear in mind that while some networking architectures (LAN architectures, in particular) combine the control and management planes, in APPN they are distinct. Discussion of the management plane is outside the scope of this paper.

The next section introduces the APPN node types.

LEN End Node: A LEN end node has no network control transaction programs at all, and no client support for requesting the services of a network node: it has no APPN control plane. The routing function in a LEN end node is simply to transmit to an adjacent node using a predefined link and any required link signalling information (such as the link station address on an SDLC link, the MAC and SAP addresses on a LAN, or a selected adapter and telephone dial digits for a switched connection). Without default routing (explained in Figure 2), a LEN end node must have definitions for the locations of all partner applications. One consequence is that if a network administrator decides to relocate an application to a different computer, she needs to distribute updated definitions to **every** LEN EN that accesses that application. (This is one of the things APPN was invented to fix!) The left panel of Figure 2 illustrates such a configuration and the definitions required at LEN EN *A*.

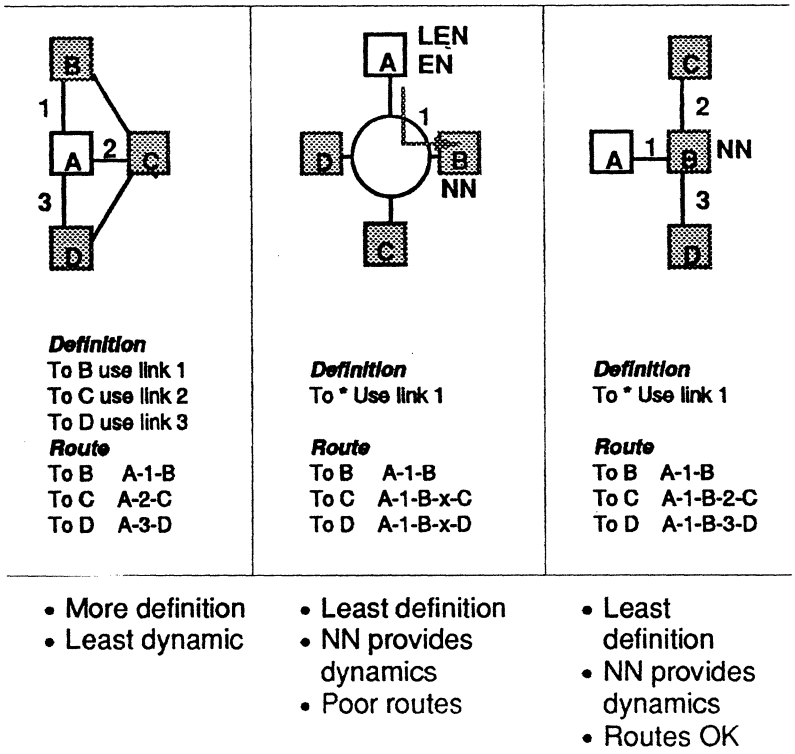


Figure 2. LEN End Node Explicit and Default Routing

Alternatively, a LEN EN may define all partner applications as residing on an adjacent NN (whether they actually reside there or not). This is termed "default" routing and is shown in the center and right panels of Figure 2. In this case the LEN EN acts as a passive client, accessing the services of a network node server indirectly, by simply attempting to route data to it (sending it a BIND, in SNA parlance). While this method relieves the LEN EN of predefining individual network addresses for all its partner applications (a single definition "all partners reside on the NN" will suffice), the resulting sessions always traverse the network node, not always an efficient route (especially where direct mesh connectivity exists between every pair of computers, as on a LAN). On the other hand, default routing may be quite acceptable if the LEN EN is a portable computer with a single dial-up line to access network computing resources. In such a configuration, all connections would traverse the NN anyway, as the right panel of Figure 2 illustrates.

APPN End Node: An APPN end node adds a small control plane, with client transaction programs (at the transaction layer—layer 7) to register its applications (Logical Units, or LUs, in SNA lingo) to a network node and request services like locating destination applications and selecting routes. [23] Figure 3 illustrates both APPN and LEN end nodes (the latter being a user plane without an APPN control plane).

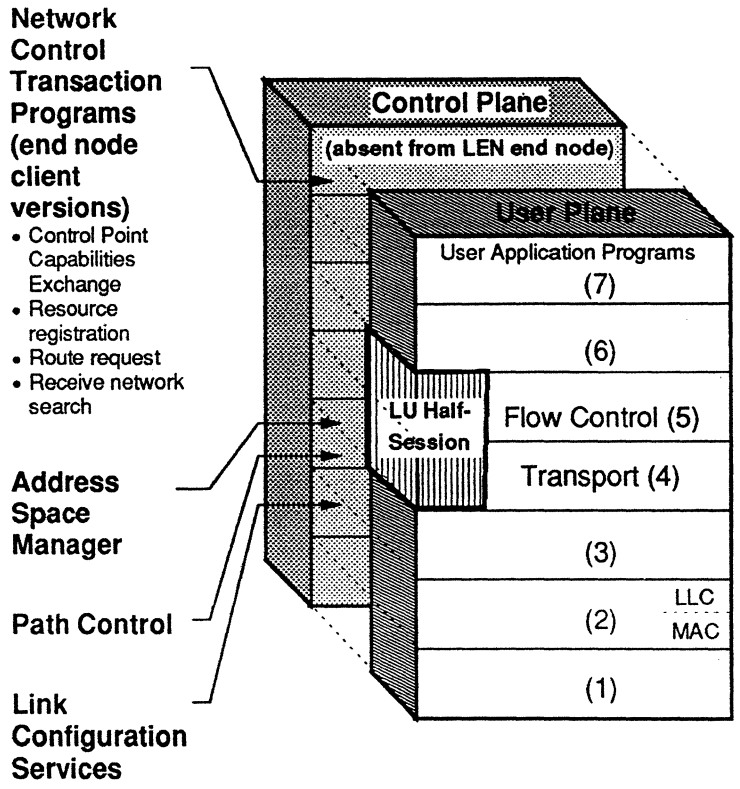
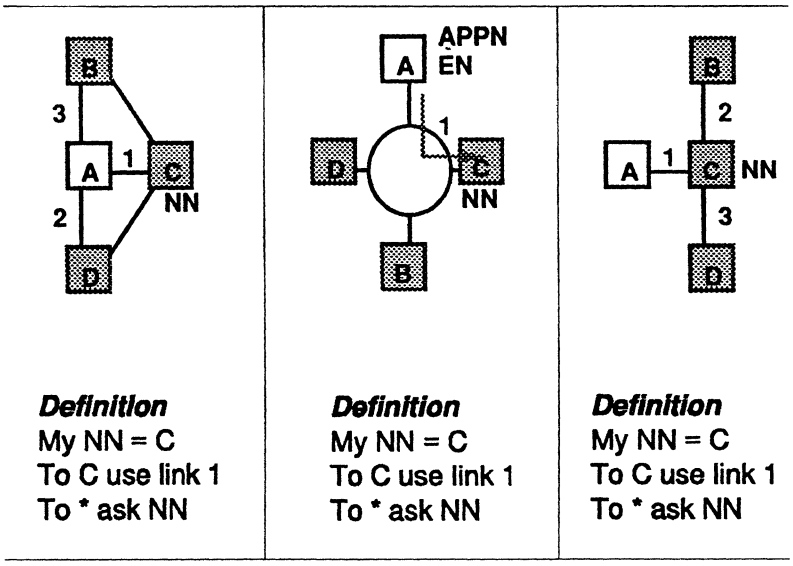


Figure 3. An APPN End Node

Unlike the LEN EN which interacts with a network node server passively if at all, an APPN EN actively requests NN services. Some of the benefits over the LEN EN are better dynamics, less definition, and better routes. An APPN EN uses the route provided by its network node server. A different route may be provided every time a new session is set up, and the route provided does not necessarily traverse the NN. This point is illustrated below. In Figure 2, *A* was a LEN EN; in Figure 4, *A* is an APPN EN.



- One definition: network node server
- NN server provides dynamics

Figure 4. An APPN EN Knowing Only its Network Node Server

The routing function in an APPN EN is still minimal: an EN can only be the endpoint of a session, never an intermediate node of someone else's session. The transport layer of an APPN EN is enhanced by its support for adaptive pacing. The network layer is the same as a LEN EN.

APPN Network Node: A network node adds specialized network control transaction programs at the transaction layer in the control plane, to manage distributed directories and maintain the replicated topology database used for route computation, as well as server support for end node clients. [24]

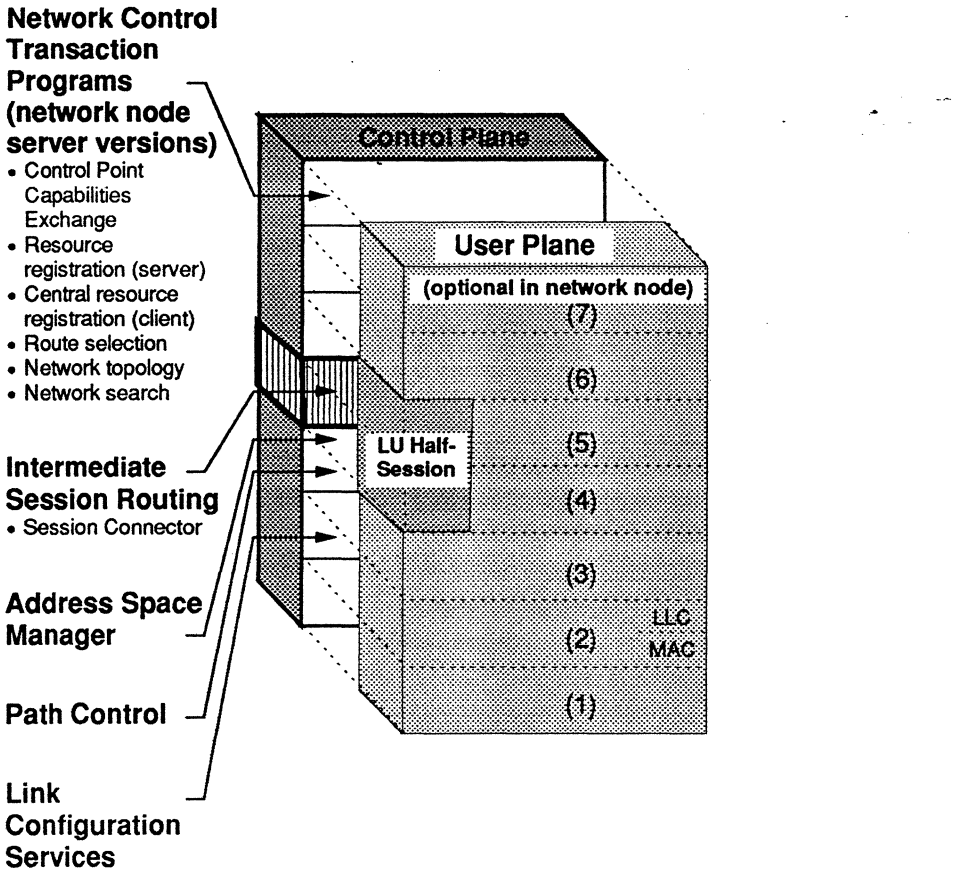


Figure 5. An APPN Network Node's Extended Control Plane

It also adds *intermediate session routing*—ISR—the ability to forward packets for applications that do not reside on the NN itself. ISR consists of enhancements at the transport and network layers. One part of ISR is a component called a *session connector*, occupying a similar position and performing similar functions in the protocol stack as the LU half-session in a session endpoint node. (Compare the shaded components in the user and control planes of Figure 5.) ISR functions include error recovery, adaptive pacing, and the adjustment of packet sizes via segmentation and reassembly.

As Figure 5 shows, not every network node has a user plane. A router that does not host any end-user applications is an example of a specialized NN that does not need a user plane.

Many people—even some of IBM's marketing literature—describe SNA as "non-routable." This is not strictly true. The capability for intermediate session routing previously existed in SNA Type 5 and Type 4 nodes such as VTAM and NCP, using FID4 transport (layer 4) over explicit routes and virtual routes (ERs and VRs—the SNA path control network—at layer 3). The backbone was physically structured as a mesh, and the peripheral network, using FID2 transport, was strictly hierarchical. Setting up the SNA "routing tables"—ER and VR path definitions—was a

laborious manual process or required the use of a tool like NetDA (Network Design Aid). APPN enhances the FID2 transport that first emerged in subarea SNA's peripheral network, and automates the maintenance of routing information and directories. **APPN is native routing technology for SNA.**

Let's examine intermediate session routing in more detail, to understand how High Performance Routing differs from it.

EXISTING ROUTING TECHNIQUES

The literature—and vendors' product lines—are filled with a variety of routing techniques, including routing by network address, label swapping, and source routing. And these routing techniques are often supplemented by network control algorithms to dynamically distribute routing tables or maintain a topology database. They may also be supplemented by discovery or address resolution protocols to dynamically map the name of a desired communication partner to a network-layer address or routing information. All the routing techniques discussed below are suitable, in general, for implementation in either hardware or software, while network control algorithms and address resolution protocols are frequently implemented in software.

Routing by network address is the technique used in Internet Protocol (IP). A single 32-bit routing label that must be unique within the scope of an entire internetwork represents the final destination, and serves as an index into a routing table specifying the next hop. The next hop taken depends on the current state of the routing table at the node processing the packet [16]. Several algorithms exist to distribute IP routing tables, some standard and some proprietary. One of the mostly widely used standards, the Routing Information Protocol (RIP), distributes the entire IP routing table periodically at timed intervals. This type of table distribution is called a *path status* algorithm. As individual IP subnetworks become larger, the amount of administrative traffic generated by these regular routing table updates grows exponentially, placing an upper bound on the size of an individual IP subnetwork. IGRP, Cisco's proprietary routing algorithm, also uses a path status algorithm to distribute its routing table updates and, consequently, also places an upper bound on the size of a subnetwork. Within a single IP subnet, it is necessary that all routers support the same algorithm. Hence the focus on standards rather than on proprietary techniques. A relatively new standard algorithm for TCP/IP, Open Shortest Path First (OSPF), is gaining in support among router vendors [16] and uses a more efficient *link-state* type of algorithm (defined below).

Label swapping is a technique used in current APPN (APPN/ISR) and, interestingly, also in the CCITT high-speed recommendation for Asynchronous Transfer Mode (ATM). A packet bears a single network-layer routing label, representing the next hop. A router or high-speed switch substitutes a new label before transmitting the packet, In connection-oriented protocols, the label-swap tables are generally set up in intermediate nodes when the connection is established, based either on predefined information or on a topology database reflecting the state of the network at the time of connection establishment. The APPN topology database updates are only distributed when information changes. This type of algorithm is called a *link state* algorithm. Link-state algorithms generate much less administrative traffic than path status algorithms, removing one barrier to the growth of larger individual subnetworks. TCP/IP's OSPF is also a link state algorithm.

Source routing is a third routing technique, commonly seen in LAN bridging. A source-routing variant called Packet Transfer Mode (PTM) is currently being applied in high-speed trials such as the Aurora test bed. In source routing a list of routing labels, representing the entire route, prefixes the packet. The route is determined in advance, usually based on a discovery protocol or a topology database. Some argue that source routing is the most efficient of these three techniques, requiring the least processing at intermediate nodes, yielding the maximum throughput.

Link-Sharing—a Fact of Life: Whatever routing technique a protocol uses, most state-of-the-art protocols (APPN included) provide a means for different routing stacks to share the transmission medium. A medium access control (MAC) sublayer provides a graceful and standard way to share the link. Examples of media with a distinct MAC sublayer are token ring (802.5) [22], Ethernet, 802.3. and FDDI. The Point to Point Protocol for HDLC (PPP—RFC 1330 [4], [34]); and Frame Relay (Multiprotocol Encapsulation—RFC 1294 [5] and the Frame Relay Forum Implementation Agreements) are not strictly MAC-layer technologies but permit similar link sharing. If the medium has a MAC sublayer, logical channels can be established between paired adjacent link stations on the basis of predefinition, discovery, or a capabilities exchange protocol peculiar to that medium. It is likely that a MAC sublayer or similar standard will also be defined for any new transmission technologies that may emerge in the future.

Traditional SNA Approaches to Routing and Error Recovery: Traditional SNA—subarea SNA and APPN/ISR—has been fundamentally connection-oriented in its approach to routing. In general, traditional SNA attempts to provide high reliability over a variety of links (including error-prone links with poor characteristics). Error recovery is performed at the data link layer (layer 2) with connection-oriented data link controls like SDLC, X.25 ELLC, or LAN logical link control type 2 (IEEE standard 802.2). [26] [32] Recovery is also performed at the transport layer—layer 4 (in reassembling segmented data, the transmission control component of the LU's half-session ensures that no packets are missing or out of order) and at the data flow control layer—layer 5 (the half-session enforces chains as the unit of application-level error recovery). Figure 6 illustrates these three levels of error recovery. If any of these protocols are violated, due to failure of the underlying transmission facilities, unrecovered packet losses in the network due to congestion at intermediate nodes, inability of the receiving application to buffer all the received data, or transparent rerouting by a subnetwork that causes some packets to arrive out of order, traditional SNA deactivates the session. With its key design point to support business-critical applications like finance, order entry, inventory control, and credit authorization, traditional SNA has industrial-strength algorithms to detect and prevent the occurrence of these error conditions. In addition, sync point (an APPC checkpointing service) ensures the integrity of distributed databases. In case of session, database, or processor failure, all applications and databases assume a known state and a distributed transaction can resume exactly where it left off once communication is reestablished.

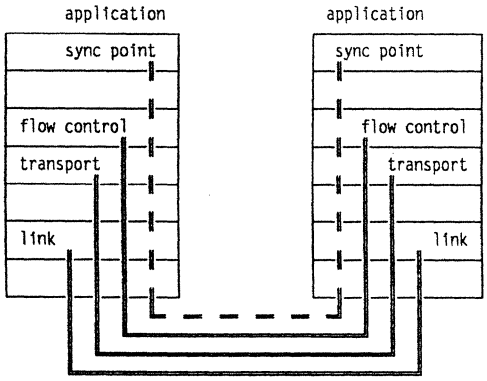


Figure 6. Four Levels of Error Recovery in Traditional SNA

Routing in APPN/ISR: APPN/ISR determines the route for a session when the session is set up; the route remains in effect for its duration. The route is chosen based on a user-specified class of service, a transmission priority, the destination, and the available routes, with randomization if more than one route is acceptable. Thus different sessions between the same pair of LUs can have different routes based on user-specified parameters. Every session has a unique identifier (an "FQPCID"), assigned by the origin node, that refers to the session at every node it traverses, throughout its lifetime. This identifier is used for network management and by the transaction programs in the control plane during session setup, but not for routing.

The routing field or network-layer header in APPN/ISR has a single routing label 17 bits long called a *Local-Form Session Identifier* (LFSID), defined by the SNA FID2 transmission header format. It needs to be unique only on a given link. Because it uses FID2, APPN/ISR routing can coexist with pre-APPN traffic, such as 3270 terminal traffic, using the same link. An APPN route is a series of *session stages*, end to end, each with its own LFSID routing label.

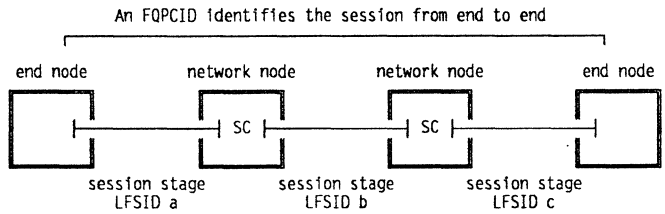


Figure 7. Session Stages Interconnected by Session Connectors

In a network node, the LFSID indexes a "routing table." This table is distinct from the topology database. Each entry in this table is called a *session connector* and is created at the time of session establishment. As an NN forwards a packet from an inbound link to an outbound link, it replaces the LFSID in the packet header with the LFSID from the session connector. APPN routing can therefore be classified as *label-swap routing.* There can be many SNA sessions at once on a logical link, each with a distinct LFSID. APPN nodes usually select LFSIDs dynamically, during session set-up. (Pre-APPN nodes may have LFSIDs preassigned.)

Because of APPN's original design point to support business-critical applications over good-to-poor links, in additional to label-swapping, ISR performs additional, transport-layer functions at intermediate nodes. These other functions include segmenting and reassembly, pacing, and priority queuing for transmission. Figure 8 illustrates function placement in APPN intermediate session routing. Let's examine each of these functions more detail.

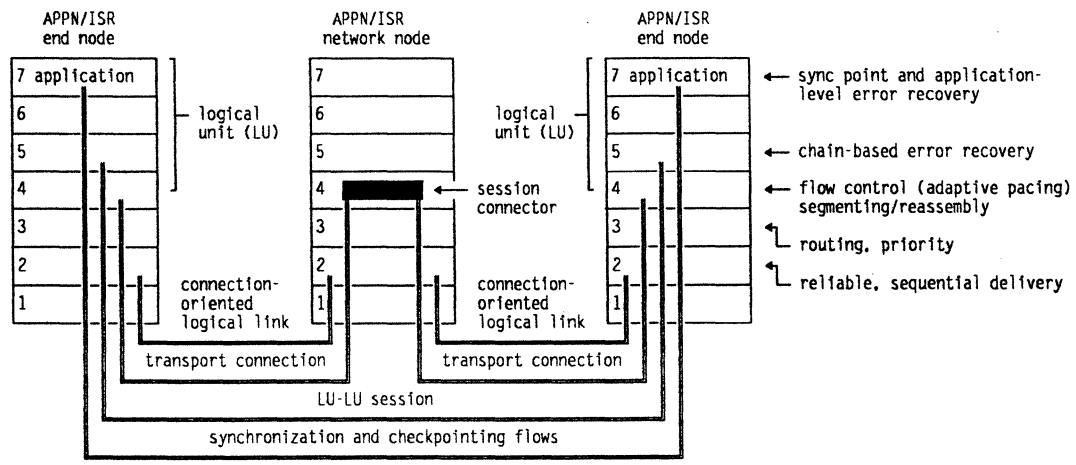


Figure 8. APPN Intermediate Session Routing

Segmenting and Reassembly: Different links in a network may support different maximum packet sizes, for reasons of link speed, transmission delay, data link control timing requirements, fairness, and node buffer capacities. [37] This point is illustrated by one of the graphs from "Data Link Control and Contemporary Data Links" by Traynham and Steen (IBM technical report 29.0168, June 1977), reproduced in Figure 9.

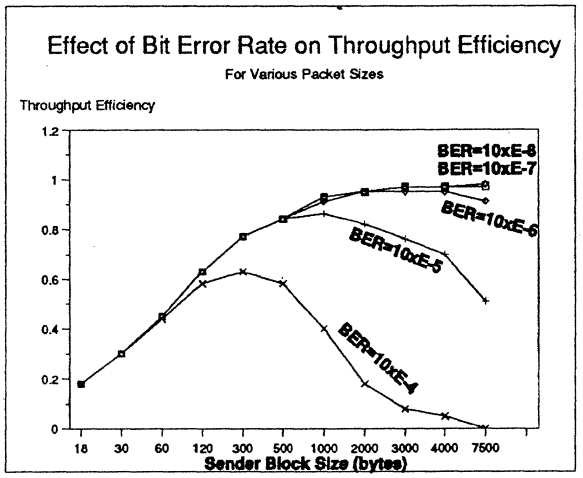


Figure 9. Throughput Efficiency as a Function of Bit Error Rate and Packet Size

In an architecture like APPN that embraces a variety of LAN and WAN transmission media of varying speeds and characteristics, it is not reasonable to expect every node to agree on the same "best" packet size. As a general strategy to maximize performance, APPN/ISR sends the largest packet size allowable on each link. When necessary, segmentation and reassembly are done at intermediate nodes. This is one of the functions performed by APPN intermediate session routing. Any changes to ISR must address the issue of packet size in some manner.

Adaptive Pacing: Pacing is a flow control and congestion control technique to adjust the sender's transmission rate according to the capacity of the receiving node's buffers. Pacing is another function performed by APPN intermediate session routing. In APPN/ISR, pacing occurs independently on each session stage or BIND hop. APPN nodes support both fixed and adaptive pacing. Adaptive pacing is preferred, while fixed pacing permits interoperation with older SNA nodes. Because each APPN session stage is independently paced, every node (nodes supporting applications, as well as routers) can adapt the pacing for the traffic it handles in accordance with its own local congestion conditions. This is the basis for global flow control and congestion management in APPN/ISR. Any changes to ISR must improve upon this already-superior existing function.

With fixed pacing, a predetermined number of packets can be sent before the sender has to wait for the receiver to give permission to send more. Adaptive pacing is a more powerful and flexible scheme wherein a sender can send only a limited number, or *window*, of packets per explicit grant of permission to proceed. The window size is changed dynamically, based on conditions at the receiver. This lets a receiving node manage the rate at which it receives data into its

buffers. Adaptive pacing provides a node supporting many sessions, or unpredictable bursts of traffic, a dynamic way to allocate resources to a session that has a burst of activity, and to reclaim unused resources from sessions that have no activity (rather than predefining a buffer pool of a particular size for every active session). Thus we see that adaptive pacing allows the receiving node to use its available buffer resources efficiently. It can also prevent potential protocol deadlocks.

If a node is running low on buffer resources, it uses pacing to tell the upstream node to slow down. If that node becomes congested, it in turn may tell its upstream node to slow down. When a node is not congested, it gives the upstream node permission to send faster. When a receiving node gives a sender permission to send a certain window size, the receiver has reserved sufficient buffers in advance, guaranteeing that data, once sent, will not be lost due to congestion. In practice, many products use statistical or demand buffering schemes, which are acceptable as long as confirmed buffers are available when needed.

A separate instance of adaptive session-level pacing exists for each session running to or through a node, to manage the flow of data on one LU-LU session. Adaptive session-level pacing also applies to the sessions between APPN control points. Adaptive session-level pacing occurs independently on each session stage. Pacing is done by the half-session component of the LU in a node containing a session endpoint, and by the session connector component in an intermediate node. This will become important when we examine how HPR can replace the ISR function (including the session connector) in network nodes, and how HPR can supplement the equivalent component—the LU half-session—in session endpoint nodes.

Priority Queuing for Transmission: Transmission priority permits more important data to pass less important data at queuing points in the network. Priority is another function performed by APPN intermediate session routing. Any changes to ISR must also be equal to or better than the existing support in this area. APPN has four priority levels: a *network* priority, and three session-level priorities: *high*, *medium*, and *low*. Network priority is the highest and is reserved for network control traffic such as pacing messages, topology database distribution, and session establishment. The other three priority levels are for user traffic. A user selects a priority level indirectly, by specifying a mode name defining a session's characteristics. The mode name maps to a class of service definition, which in turn specifies the priority level associated with that class of service. The transmission priority selected for a session is carried in the session activation request (BIND) at session establishment, allowing every node along the path to assign the same priority value, to be used in routing. A transmission priority applies to the session for its lifetime, at every node it traverses. Both ENs and NNs support transmission priority.

Transmission priority is implemented by the path control component in APPN. One function of path control is to direct traffic to the right outbound link. Path control can also multiplex different sessions on a single link. Another function of path control is to ensure that higher-priority data is transmitted before lower-priority data. This is generally implemented as four different queues into which message units are placed, depending on the priority associated with the corresponding session. After the DLC finishes transmitting the current message, path control picks the next message for transmission, selecting from the highest priority queue having a message unit waiting. To ensure that lower-priority data is not preempted indefinitely, an aging mechanism is also used.

A NEW SNA APPROACH TO ROUTING

Many people have observed that some protocols duplicate certain functions at layers 2 (data link control) and 4 (transport), leading to difficulties and ambivalence in discussions of the subject (especially at meetings of international standards bodies!). Furthermore, the current 7-layer model (see Figure 1), mirrored in the organization of standards bodies, does not adequately describe a new class of protocols that are so versatile they can act either as a transport (layer 4) or as a virtual link (layer 2). [19][27][35] The current paradigm is ripe for an overhaul.

Logical link control (LLC—the upper half of layer 2) was originally created to permit the coexistence of both connection-oriented and connectionless service, between multiple link stations, on

the same LAN segment. With the extension of this technology by bridging, and then remote bridging which introduced longer and variable end-to-end delays, came sensitivities to the LLC timeout values which are used to detect link or link-station failure. Nevertheless, the principle of extending a data link layer connection across multiple hops is now firmly established. The advantage of data forwarding at a low layer is performance. The disadvantage is that higher layer protocols (like APPN) that select routes based on APPN link characteristics can't see the actual characteristics of the links interconnecting remote bridges. A bridged link appears as a single link in the APPN topology database, with a single set of link characteristics hiding the complexity of multiple hops. The inability to know the true hop count, or to distinguish between a slow link and a fast one, can lead to poor route choices.

One solution to this problem is to replace pairs of remote bridges by pairs of APPN/ISR network nodes. However, this only solves part of the problem, since the ISR functions of pacing and segmentation/reassembly, absent from bridges, would reintroduce delays, and current bridging standards lack support for priority.

A solution was needed to better integrate the bridging concept into APPN. With good links, some of traditional SNA's error recovery is redundant. One possible way to reduce overhead is to omit error recovery at the data link layer, replacing the usual connection-oriented LLC with a connectionless LLC. The transport is replaced by APPN's versatile new transport protocol, RTP—Rapid Transport Protocol, which efficiently provides any needed error recovery over multiple hops. One drawback to this approach is the placement of transmission priority (path control) **below** the transport.

An better function placement is shown in Figure 10. The new transport becomes part of the LLC sublayer in layer 2, acting as an enhanced logical link. This is true when the logical link comprises not only a single hop, but multiple hops. In this paper we'll call this versatile new class of protocols *transport-oriented logical link controls.* Thus, a transport-oriented LLC like RTP, spanning multiple links and nodes, if it meets the needs of upper layer components to which it provides service, can replace one or more APPN/ISR session stages, acting as a "virtual link." APPN+ takes advantage of this principle (see Figure 11).

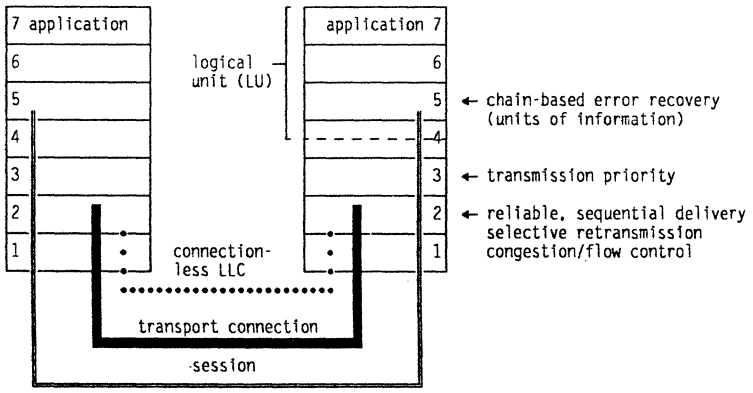


Figure 10. A Transport-Oriented Logical Link Control

The function placement of Figure 10 is better for several reasons. First, it places transmission priority **above** the new transport—important for a node with more than one link. Second, it permits the existing transport (such as an LU half-session—see Figure 3) to be kept intact for the other key services it provides, and because of its integration and packaging into system software (like APPC).

The first property, the coupling of priority scheduling to the rest of the protocol stack, can be understood as follows. In subarea and APPN SNA, the users declare the class of service (COS) needed for their traffic at the transaction program API or user logon API by giving a mode name. The mode name is mapped to a class of service, which in turn specifies both the route through the network (either by listing valid choices in the subarea case, or by defining the parameters of the

route selection algorithm in the APPN case) and the transmission priority to be used for the traffic. In subarea, the route is fixed by the ER routing tables and the VR-to-ER mapping, while the priority is carried in the transmission header of each packet. In APPN, the route is fixed by the route selection control vector on the session initiation request (BIND) and the priority is carried in the BIND and saved at each intermediate session routing point, where it is used on the fly for each packet. In both subarea and APPN, the actual priority queueing is done at the top of the DLC component: when the line is finished transmitting the current frame (for example, at the end-of-frame interrupt from the hardware), the highest priority message is taken off the DLC transmit queue (or, in the case of multilink transmission groups, the MLTG transmit queue). An aging algorithm ensures that even low priority traffic gets through under heavy loads.

HPR needs to preserve the relations above, so needs to have priority scheduling queues at the end-of-frame events even on connectionless DLCs. HPR ties the session class-of-service to these queues by encoding priority bits into the HPR headers. If this were not done (e.g., as it cannot be done in networks that lack priority link queuing, or adequate coupling of it to user COS) then HPR would not have preserved the COS semantics at the user APIs.

One may well ask why, if the new transport is so much *like* an LLC, IBM did not choose an existing standard (on a LAN), with traditional bridging (to get across the WAN). The answer is that traditional LLCs are not up to the task. They are restricted to particular media and are not optimized for multiple hops. A new class of transport protocols was needed. APPN/HPR is not limited to LANs: it can run over any transmission medium that supports an unacknowledged (or connectionless) type of service, for example: LAN LLC, LAP-D, LAP-E, or SDLC (using unnumbered information—UI—frames) or X.25 (using QLLC). Furthermore, RTP provides advanced functions like selective retransmission and adaptive rate-based congestion control that no existing standard supports. Another reason is that even when a particular bridging technology supports non-disruptive rerouting at layer 2 (and many new ones, like frame relay or Data Link Switching [33], do), its selection of a new route is not integrated into APPN and is not based on class of service. An advantage for APPN/HPR, as compared with bridging or TCP/IP to span the wide area network, is that its route selection includes awareness of link characteristics (speed, delay, cost, security) and node characteristics (route addition resistance, congestion).

Introduction to APPN High Performance Routing: APPN/HPR augments APPN/ISR's layer 3-4 transport and network functions with two new elements: Rapid Transport Protocol (RTP) and Automatic Network Routing (ANR), shifting the locus of APPN routing from layer 4 down to layer 2. [3] [15] [14]. Each is described more fully below.

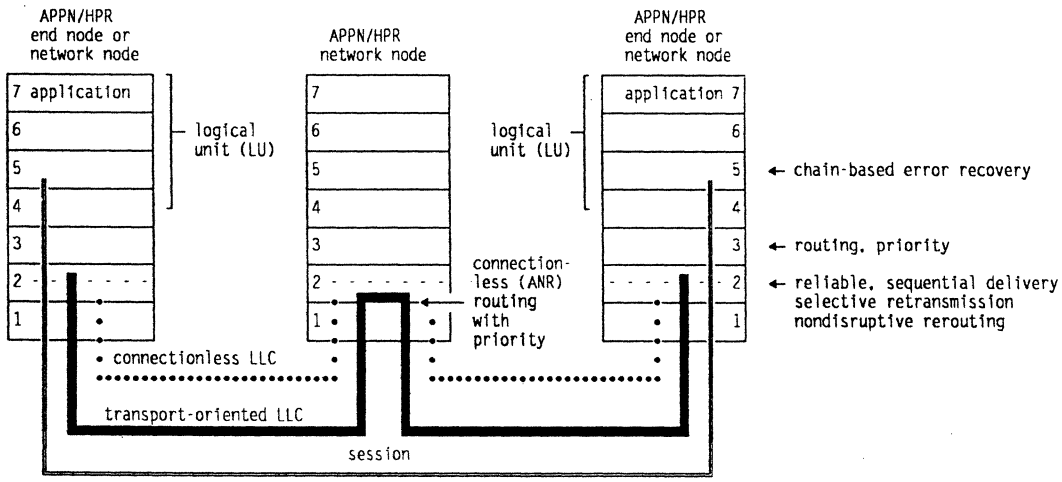


Figure 11. A Multiple-Hop Transport-Oriented Logical Link Control

APPN/HPR also includes an ISR/HPR boundary function (not shown) to adapt an HPR-capable part of the network to an ISR-only part of the network, plugging one side of an ISR session connector to the end of an RTP transport-oriented logical link.

Because APPN/HPR is completely integrated into SNA and does not change APPN's control plane at all, any node can be upgraded to the HPR level of function transparently, continuing to interoperate with adjacent nodes still at the ISR level of function. As soon as two or more adjacent nodes are HPR-capable, initial benefits of HPR—non-disruptive rerouting, adaptive rate-based congestion control, selective retransmission, fair multiprotocol transport—start to be realized. As soon as two or more HPR-capable links exist back-to-back, such that high performance routing replaces intermediate session routing in at least one intermediate node (shown in Figure 11), further HPR benefits are experienced—fast routing *with* priority and reduced intermediate node storage.

HPR provides a connection-oriented transport (RTP), end to end, over connectionless source routing (ANR), over the minimal data link control. RTP acts as a "virtual link." The amount of function that RTP demands of the underlying DLC depends on the quality of the transmission medium. On high quality links, the DLC is not asked to provide reliable delivery, sequence numbering, retransmission, or acknowledgment. It merely provides a frame check sequence: errored packets are simply discarded. Links with bit error rates on the order of 10^{-7} or better are good candidates to use a connectionless DLC under RTP. Such links typically use digital transmission over fiber media. On high-error-rate links a connection-oriented DLC with error recovery may be used under RTP. In either case the benefits of HPR are significant.

An HPR path can also include multilink transmission groups—essentially, a bundle of links between adjacent nodes that are treated as a single "fat" pipe. Benefits of MLTG include high availability (if one link of a multilink group fails, the MLTG remains operational) and bandwidth on demand (additional switched links can be dialled up to augment the bandwidth an existing link). Long a feature of subarea SNA networking, MLTG can easily be added to HPR, without the performance and storage penalty of refifoing disordered packets at each MLTG hop. Reordering only needs to be performed at the endpoints of the RTP logical link, a task RTP was designed to accomodate.

RTP insulates the upper layers—the LU—and the user from any awareness of path switching, multipath routing, network-related congestion control activities, retransmissions, acknowledgments, packet resequencing, multiplexing, and so forth. Thus a user's investment in existing SNA applications is completely preserved.

ANR—Automatic Network Routing: The functions of ANR used by APPN/HPR are the following:

- Source routing with locally specified labels
- Connectionless, stateless, fast routing
- Discarding incoming packets in the event of congestion
- Servicing the outbound transmission link based on priority.

There's not much more to say—ANR is elegant and simple. [3], [15], [14]. ANR functions are done at every node along the path of an RTP transport-oriented logical link.

RTP—Rapid Transport Protocol: The functions of RTP used by APPN/HPR are the following:

- Connection awareness (of each individual session using the RTP logical link, the session partners of that session, the transmission priority, the current ANR route for the network connection, and if a path switch has occurred, all previous routes used by the logical link over its lifetime)
- Optional reliable delivery (sequence numbering, acknowledgment, and selective retransmission)
- Reordering, if needed (may be needed after a path switch, or if the route contains one or more multilink transmission groups)

- Determining the smallest maximum packet size along the path of the RTP logical link and ensuring (through segmentation) that all message units offered to the link are the proper size (this function eliminates the need for segmentation and reassembly at intermediate nodes)
- Flow control and congestion control/avoidance (Adaptive Rate-Based — ARB)
- Providing an interface to the ANR sublayer (below)
- Providing an interface to SNA path control (above)
- Non-disruptive rerouting
- Multiplexing (more than one session having the same class of service and transmission priority requirements can share a single transport-oriented logical link).

RTP is connection oriented. These RTP functions are done only at the endpoints of an transport-oriented logical link, not at any intermediate nodes (as shown in Figure 11).

At this point, the reader may be struck by apparent similarities between HPR and TCP/IP. One of the features that sets them apart, however, is HPR's congestion control mechanism, described in the following section.

ARB—Adaptive Rate-Based Congestion Control: A new technique for flow control and congestion control was needed for APPN+, to compensate for the loss of adaptive pacing function in intermediate nodes. This new function is called ARB—adaptive rate-based congestion control.

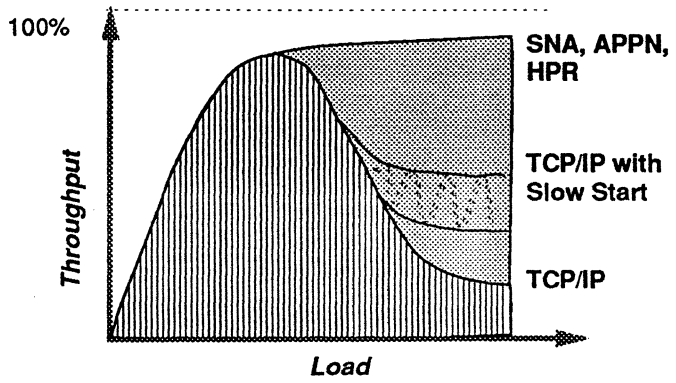
The function of ARB is to regulate the input traffic (load) offered to the RTP logical link in the face of changing network conditions. It is preventive, rather than reactive. When the network approaches congestion (increasing delay, decreasing throughput), ARB reduces the input traffic rate until the network's capacity is restored. When possible, ARB increases the sending rate without exceeding the rate that the receiving endpoint can handle.

ARB uses a closed-loop feedback mechanism based on information exchanged periodically between RTP components at the **endpoints** of an RTP logical link. (No ARB function is performed in intermediate nodes.) The feedback consists of information about two rates: the rate at which RTP accepts data arriving from the network, and the rate at which the RTP hands off the data to a recipient (such as the SNA path control component). Based on this feedback, the sender predicts when congestion is likely to develop, and takes steps to prevent it. If congestion does occur, the sender takes stronger measures to bring the network back to normal.

Because ARB addresses fairness at the data link layer, among different RTP logical links (which may be carrying multiple SNA sessions), APPN's existing adaptive session-level pacing algorithm continues to be used in networks with APPN/HPR, to provide fairness among SNA sessions at the data flow control layer (layer 5) in the LU half-session.

Analysis and simulation convinced researchers Rong-Feng Chang et al. [13] that ARB is superior to the "slow-start" congestion control algorithm of standard TCP/IP architecture [1], introduced after the Internet experienced a series of congestion collapses in October of 1986 [29]. In particular, ARB allows high link utilization rates (on the order of 80—90%) and is preventive, rather than reactive. The same study suggested that "slow-start" causes expensive link under-utilization, with lower design loadings (which should be a matter of concern to network administrators and people with budgets). The study also concluded that TCP slow-start exhibited unfairness and bias against certain kinds of traffic due to wide oscillations in packet delay and throughput (these should be of great concern to network users). Additional deficiencies of TCP slow-start cited in [13] included periodic packet losses, systematic discrimination against particular connections, bias against connections with long round-trip times, and bias against bursty traffic. It is worthwhile to note that many IP router vendors address these *TCP architecture* defiencies with proprietary extensions or product-internal enhancements. One informal mechanism used by several vendors is to give routing priority to short packets, which are assumed to be acknowledgments.

Figure 12 illustrates the relationship of effective congestion control algorithms to throughput and response time, which translate directly into cost savings and user satisfaction.



	HPR	Standard TCP/IP (*RFC 1122)
Sensing	Predictive	Reactive
Control	Rate	Windows
Line Loading	High	Lower
Response Time	Stable	Fluctuating

Figure 12. Relationship of Effective Congestion Control to Throughput and Response Time

HIGH PERFORMANCE ROUTING VS. INTERMEDIATE SESSION ROUTING

Like APPN/ISR, APPN/HPR is connection-oriented, with the entire route for a session determined in advance.

Unlike ISR, which uses label-swapping, the entire route prefixes each HPR packet. The route is encoded as an arbitrary-length string of routing labels. This technique is classified as *source routing*. (The general concept of source routing may be familiar, being used in some LAN MAC-layer bridging protocols [22]). The particular source routing technique used for APPN/HPR is ANR—automatic network routing. The labels vary in length, typically 1-2 bytes. Each routing label has local significance only: it is meaningful relative to the node processing the label. The first label always represents the next hop (or, at the last node, the terminus of the RTP logical

link). Routing consists of stripping off the first routing label and transmitting the packet on the link indicated by the stripped label.

Because of ANR's simplicity, existing platforms that implement it may well realize significant performance gains of 3—10 times on their existing hardware. That is, on many platforms, HPR function might be deliverable in the form of a software upgrade. (By contrast, other advanced routing techniques such as ATM often require expensive and specialized new hardware.) Performance gains at the upper end of this range can be expected on hardware optimized for ANR.

A single RTP logical link between HPR-capable endpoints replaces two or more back-to-back ISR session stages. There can be many RTP logical links at a time on a transmission link; intermediate nodes are unaware of, and unconcerned with, these individual connections: they only see a stream of packets prefixed with routing labels representing their outbound links.

Many SNA sessions can share a single RTP logical link, provided they have the same class of service and transmission priority.

If a path fails, the RTP component at the endpoint of the logical link obtains a new path, still based on the desired class of service and transmission priority, and keeps going—all without the session's awareness. If packets get lost or out of sequence during the switch, RTP takes care of it transparently. As a result of path switch, it's even possible for the forward and reverse directions of data flow to follow different routes through the network.

Frame Relay—A High Speed "SDLC" for HPR: While SNA absorbs and runs over many different types of links, from S/390 channels to the synchronous framing mode of SDLC, SDLC has always played an important role in both subarea SNA and in APPN. In some sense, it has been the template DLC, the one on which the others were modeled. This shows up in the presence of XID exchanges on other DLCs, when XID is the name of a control frame within SDLC. SDLC, however, has the disadvantage of being a single access link, as contrasted with 802.x LANs and frame relay, both of which support link connections to multiple partners through a single hardware connection. Multiple access helps to reduce costs of ports and access lines into private or public carrier networks. And, while 802.x MACs have difficulty in working directly over WAN distances and at commonly available carrier line speeds, frame relay is well-adapted to use in WAN configurations.

In light of its benefits, frame relay has been adopted as one of the two preferred DLCs for HPR: preferred in the sense that products are encouraged to implement both frame relay and 802.x LAN connections for HPR support. This encouragement stops short of a mandatory architecture requirement because of the wide diversity of products and market niches in the world, but certainly the general-purpose networking products such as the IBM 6611, 3745, 3174, 3172, AS/400, and CM/2 products are expected to support both FR and 802.2 LANs for HPR connections.

HPR's support of FR includes both "through a carrier" and "null carrier" configurations. Since FR uses HDLC framing, it runs on existing SDLC adapters as a software or microcode upgrade; being configurable as null network connections allows it to be used wherever point-point full-duplex SDLC lines are now used, without changing line provisioning in any way. Multidrop SDLC lines pose another problem: some can be converted to multiple-connection FR services through a carrier network (the carrier's network becomes the "multidrop" in a certain sense, but with added function as compared to multidrop since the multidrop polling is removed); some can be reconfigured to use APPN/HPR routing services, perhaps even with line cost savings; some will have to remain as SDLC multidrop until we can support multidrop FR configurations (while multidrop is not part of the FR standard, it is easy enough to add, and we intend to do so).

Areas for Future Study: It should also be possible to have multi-path RTP logical links—so that, by choice, a session uses two or more different ANR paths at once. This is sometimes called *bifurcated* routing. This would be advantageous in several ways. It could further reduce the impact of network failures (provided that at least one path remains viable). It could provide a way to aggregate the physical capacity of several links, each with inadequate bandwidth by itself, into a trunk with sufficient bandwidth. It could allow network providers to take advantage of costly parallel capacity, installed for reasons of high availability.

While such an option adds complexity to HPR's non-disruptive rerouting logic and to network management, it appears to be a fruitful avenue for further research. Another possibility is to couple this function with awareness of actual bandwidth requirements of connections and the knowledge of link utilization in a sophisticated high-speed network.

Also for a future study is a thorough survey of the existing literature on congestion control and analysis of trends concerning connection-oriented and connectionless transports.

Conclusions and Implementation Recommendations: We've shown how APPN/ISR and APPN/HPR can work together and discussed HPR routing in some detail. What sort of a product can benefit from implementing HPR? An obvious candidate is a router. A router typically supports several high-speed WAN link attachments and has the capacity to switch large numbers of packets. This appears to be natural fit with HPR's design points.

Can a computer that supports user applications (typically, an end node) also benefit from implementing HPR? We believe the answer is a resounding **yes**. Consider a typical installation with many individual workstations attached to a LAN—possibly quite a large, bridged LAN, shown in Figure 13. Several routers (3 and 4, 10 and 11) provide connectivity between stations on the LAN (1 and 13) and a WAN backbone made up of additional routers and packet switches (not shown). So far, we have shown HPR's benefit when a WAN link fails (if any of links 5—9 fails, the HPR-enabled routers can switch to a new route across the WAN). But one class of failures for which few solutions exist today is failure of a WAN access node such as routers 3, 4, 10 or 11. The typical user (1) is in session with applications both on the local LAN (2) and across the WAN, perhaps to a remote LAN (12, 13). Today, if the router or LAN segment or bridge through which a session is routed goes down or experiences problems, the session often fails. If the user's computer (1 or 13) includes HPR function, path switching will likely be successful if a local path to an alternate NN exists. Performance and storage benefits are also realized in network nodes 3, 4, 10, and 11 if end nodes 1, 13 support HPR.

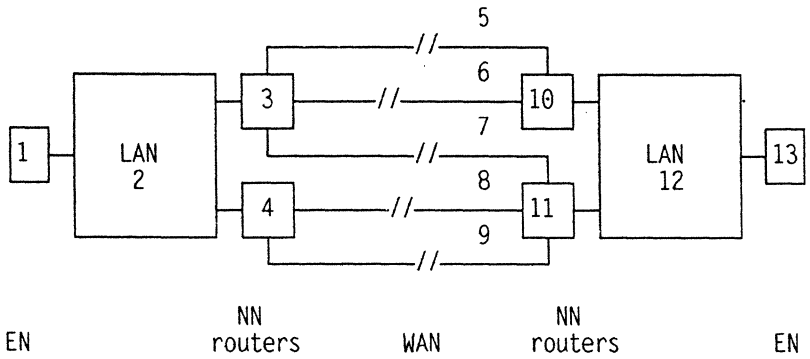


Figure 13. Benefits of HPR for Both End Nodes and Network Nodes

SUMMARY

APPN/HPR, also known as APPN+, is a promising new technology for network nodes and end nodes that transparently extends SNA, replacing the intermediate session routing function in selected nodes with the faster high performance routing. APPN/HPR includes a new connection-oriented transport layer protocol, Rapid Transport Protocol (RTP), one of a new class of transport protocols that can also serve as a logical link (with priority) over multiple hops. APPN/HPR also includes a new type of connectionless source routing called Automatic Network Routing (ANR). APPN/HPR provides nondisruptive rerouting based on class of service, fast packet switching, minimal intermediate node storage, a new adaptive-rate-based congestion prevention algorithm (ARB), and a drop-in software migration from existing SNA networks based on an ISR/HPR boundary function, for seamless interoperation with current SNA products and protocols.

ACKNOWLEDGMENTS: The authors are indebted to their colleagues at IBM Networking Systems, Ray Bird and Lap Huynh, for creating the HPR architecture specification, and to Dr. Raif Onvural for his kind encouragement, without which this article would not have been written. Thanks are also due to Barry Groner, Jane Munn, and Rick McGee, managers who supported early publication of this material.

References

[1] R. Braden (editor), Requirements for Internet Hosts Communication Layer RFC 1122, 1989.

[2] Michael O. Allen and Sandra L. Benedict, "SNA Management Services Architecture for APPN Networks," *IBM Systems Journal*, vol. 31, no. 2, pp. 336-352, 1992. Describes network management architecture for APPN.

[3] B. Awerbuch, Israel Cidon, Inder S. Gopal, Marc Kaplan, and Shay Kutten, "Distributed Control for PARIS," *Proceedings of 9th ACM Symposium on Principles of Distributed Computing*, Quebec, Canada: ACM, August 1990.

[4] F. Baker, The Point-to-Point Protocol Extensions for Bridging RFC 1220, 1991.

[5] T. Bradley, C. Brown, and A. Malis, Multiprotocol Interconnect Over Frame Relay, 1992.

[6] IBM Corp., AS/400 Distributed Systems Implementation Guide, GG22-9458. Discusses the decision criteria that must be considered when choosing a topology for an AS/400 APPN network

[7] IBM Corp., S3/X and AS/400 APPN Nodes Using the SNA/LEN Subarea, GG24-3288. Describes the incorporation of a S/370 SNA subarea into a network comprising APPN network nodes. Intended for systems programmers and systems engineers in the intermediate systems and VTAM/NCP areas.

[8] IBM Corp., APPN/Subarea Networking Design and Interconnection, GG24-3364. A guide for planning interconnection of APPN and SNA subarea networks.

[9] IBM Corp., Networking Services/2 Installation, Customization, and Operation, GG24-3662. Provides planning information for IBM SAA Networking Services/2. Contains an extended example on connecting Networking Services/2 and AS/400, with their respective configurations.

[10] IBM Corp., APPN Architecture and Product Implementations Tutorial, GG24-3669. Tutorial on APPN, with an overview of various product implementations.

[11] IBM Corp., 3174 APPN Implementation Guide, GG24-3702. Provides guidance on implementing the 3174 APPN functions in various scenarios.

[12] IBM Corp., AS/400 APPN with PS/2 APPN, 3174 APPN, 5394 and Subarea Networking, GG24-3717. Provides several scenarios of interaction of these nodes including sample definitions and traces.

[13] Rong-Feng Chang, James P. Gray, and Lap Huynh, Comparison of Congestion Control Performance of APPN+ and TCP, IBM Corp. Unclassified, Technical report 29.1490, December 1992. Describes dynamic problems of TCP and shows that APPN/HPR with ARB significantly outperforms TCP congestion control. Appendix B contrasts APPN and TCP/IP.

[14] Israel Cidon and Inder Gopal, "Paris: An Approach to Integrated High-Speed Private Networks," *Int. Jour. of Digital and Analog Cabled Sys. 1*, pp. 77-85, 1988.

[15] Israel Cidon, Inder Gopal, and Shay Kutten, "New models and algorithms for future networks," *0-89791-277-2/88/0007/0075*, ACM, 1988.

[16] Douglas E. Comer, *Internetworking with TCP/IP: Principles, Protocols, and Architecture*, Prentice Hall, 1991.

[17] IBM Corp., Client/Server Computing: The New Model for Business, IBM Corp., February 1992.

[18] Rudy K. Cypser, *Communications for Cooperating Systems: OSI, SNA, and TCP/IP*, Addison-Wesley Publishing Co., 1991.

[19] W. Doeringer, D. Dykeman, M. Kaiserswerth, B. Meister, H. Rudin, and R. Williamson, "A Survey of Light-Weight Transport Protocols for High-Speed Networks," *IEEE Transactions on Communications*, vol. 38, no. 11, pp. 2025-2039, November 1990. Surveys and classifies high-speed transport protocols

[20] P. E. Green, R. J. Chappuis, J. D. Fisher, P. S. Frosch, and C. E. Wood, "A Perspective on Advanced Peer to Peer Networking," *IBM Systems Journal*, vol. 26, no. 4, pp. 414-428, 1987.

[21] IBM Corp., AS/400 Communications: APPN Network User's Guide, Publication number SC41-8188. Describes the APPN support provided by the AS/400 system. Also describes APPN concepts and provides information for configuring an APPN network. APPN advanced considerations and configuration examples are included.

[22] IBM Corp., IBM Local Area Network Technical Reference, SC30-3383. Describes token ring LANs, including source routing bridging.

[23] IBM Corp., Systems Network Architecture Type 2.1 Node Reference, SC30-3422-2 (March 1991). Defines the architecture for APPN end node and LEN end node at the pre-HPR level of function.

[24] IBM Corp., Systems Network Architecture APPN Architecture Reference, SC30-3422-3 (March 1993). Defines the architecture for APPN network node, APPN end node, and LEN end node at the APPN/ISR level of function.

[25] IBM Corp., Networking Services/2 Installation and Network Administrator's Guide, SC52-1110. Describes the APPN support provided by Networking Services/2 for OS/2 Extended Edition. Also describes APPN concepts and provides information for configuring an APPN network.

[26] IEEE, Project 802—Logical Link Control. This standard defines the 802.2 Logical Link Control protocol.

[27] Protocol Engines, Inc., XTP Protocol Definition, 1989. This defines the Express Transfer Protocol, one of a new class of optimistic transport layer protocols.

[28] ISO, Information Processing Systems—Open System Interconnection—Basic Reference Model IEEE 7498, 1984. Defines the ISO 7-layer reference model.

[29] V. Jacobson, "Congestion Avoidance and Control," *ACM SIGCOMM*, vol. '88, pp. 314-329, August 1988. Describes the "collapse of the internet" before the invention of TCP slow-start.

[30] Steven T. Joyce and John Q. Walker II, "Advanced Peer-to-Peer Networking (APPN): An Overview," *ConneXions--The Interoperability Report*, vol. 6, no. 10, pp. 2-9, October 1992.

[31] Steven T. Joyce and John Q. Walker II, "Advanced Peer-to-Peer Networking (APPN): An Overview," *IBM Personal Systems Technical Solutions*, no. G325-5014-00, pp. 67-72, January 1992.

[32] Matthias Keiserswerth, "A Parallel Implementation of the ISO 8802-2.2 Protocol," *IEEE Tricomm '91*, April 1991.

[33] David Kushi and Roy C. Dixon, Data Link Switching: Switch-to-Switch Protocol RFC 1434, 1993.

[34] D. Perkins and R. Hobby, The Point-to-Point Protocol (PPP) Initial Configuration Options RFC 1172, 1990.

[35] Robert M. Sanders and Alfred C. Weaver, The Xpress Transfer Protocol (XTP) - A Tutorial, Computer Networks Laboratory, Dept. of Computer Science, Univ. of Virginia, TR-89-10, January 1990.

[36] Robert A. Sultan, Parviz Kermani, George A. Grover, Tsippi P. Barzilai, and Alan E. Baratz, "Implementing System/36 Advanced Peer to Peer Networking," *IBM Systems Journal*, vol. 26, no. 4, pp. 429-452, 1987.

[37] Kenneth C. Traynham and Robert F. Steen, Data Link Control and Contemporary Data Links, IBM Corp. Unclassified, Technical report 29.0168, June 1977. Analyzes the relationship between bit error rate, packet size, and number of frames outstanding on throughput efficiency. Based on mathematical models, with many graphs.

ABOUT THE AUTHORS: Marcia Peters leads the APPN architecture group at IBM's Networking Systems line of business in Research Triangle Park, North Carolina where she develops architecture and product strategies to commercialize new technologies. She previously worked on a new broadband networking architecture, specializing in network control algorithms for multicast and distributed directories for multiprotocol routing. She also contributed enhancements for the seamless interconnection of APPN and subarea SNA architectures. Before joining IBM in 1988, she was lead programmer at Decision Data Computer Corp., a vendor of plug-compatible equipment for the AS/400 and System/36 family, and a development programmer at Telex Computer Products and Raytheon Data Systems, producing SNA networking products. She received a B.A. in music from Swarthmore College in Pennsylvania in 1975. She is a senior member of the IEEE. She has filed applications for over 5 US patents and published over 40 technical disclosures and articles.

Dr. James P. Gray joined IBM in 1970 in Raleigh, North Carolina. After contributing to an IBM microprocessor architecture, from 1972 to 1984 he contributed to various aspects of SNA, including APPC, syncpoint, and APPN. In 1984 he was named an IBM Fellow and manager of SNA Studies, a group that explores issues in networking and distributed processing. Dr. Gray earned a B.E. in Electrical Engineering from Yale College in 1965 and a Ph.D. in communication theory from Yale's department of Engineering and Applied Science in 1970. He is a fellow of the IEEE and a member of ACM.