APPN and Extensions: The New Industry Standard for SNA Internetworking

Marcia L. Peters

Lead Architect — Advanced Peer-to-Peer Networking
IBM Corp.
Networking Systems Division
Dept. C74, Bldg. 673
P.O. Box 12195
Research Triangle Park, NC 27709, U.S.A.
MPETERS at RALVM6
mpeters @ ralvm6.vnet.ibm.com

Abstract

Among many alternatives for SNA internetworking, the most natural and direct approach is SNA's native routing protocol, Advanced Peer-to-Peer Networking. Until recently APPN has not enjoyed as much growth as its reputation as a powerful, flexible, and easy-to-use networking solution for client/server and distributed applications warrants. We examine APPN's remaining shortcomings:

- it's proprietary
- it lacks TCP/IP's dynamic rerouting, and
- it has configuration restrictions on support for 3270 traffic

and show how they are being addressed by the newly-published High Performance Routing and Dependent LU Requester functions. Also discussed is the role that frame relay plays in the short-term to leverage current investments and position for emerging technologies like Asynchronous Transfer Mode (ATM), making APPN a vital first step on the path to ATM. With the removal of these last remaining concerns, APPN+ appears well positioned to meet the needs of the expanding enterprise LAN internet.

Trademarks IBM, Operating System/2, OS/2, VTAM, AS/400, AIX, AnyNet, and APPN are trademarks of International Business Machines Corporation.

Challenges Faced by Network Managers Today

Network managers face a series of difficult challenges today. While a small fraction are "bleeding edge" early adopters of new technology, viewing the network as a fundamental strategic asset and deploying the latest — and least mature — technological advances for the sake of the new possibilities they may create, more often the pressures on network managers are fiscal, and network decisions are driven by margins and bottom-line measurements. Among these are the need to reduce network costs and ensure that networking expenditures are justified by their contribution to the business. Other challenges include the need to leverage and exploit current network investments, manage migration and growth, move to open standards-based products and protocols, and position for emerging technologies. With the blizzard of announcements about ATM and other emerging technologies, it's hard to sort out fact from fantasy.

Figure 1 shows where Systems Network Architecture (SNA) fits in the transition to ATM by tracing its evolution through five stages, a process of constant reinvigoration to meet new challenges.

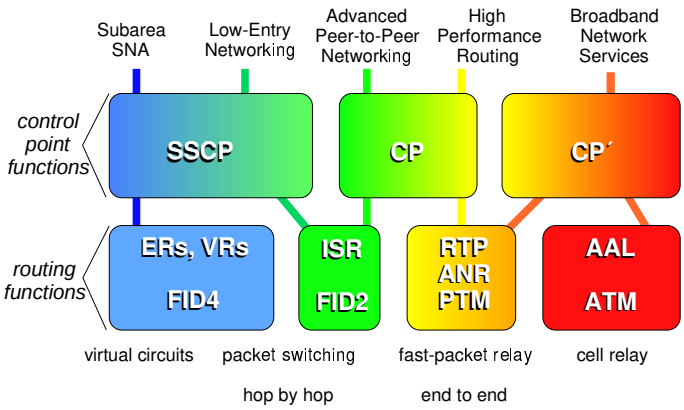

Figure 1. SNA Evolution from Proprietary to Open Standards

Traditional "subarea" SNA uses a System Services Control Point (SSCP), implemented exclusively on mainframes, to manage and control hierarchical SNA networks. Subarea routing uses Virtual Routes and Explicit Routes, defined by a system programmer, and a FID4 transmission header packet format. ("FID" means "format identifier").

In the early '80s Low Entry Networking (LEN), first seen on mid-range processors to supplement the new Advanced Program-to-Program Communications (APPC), handled routing to adjacent nodes only. For the first time it allowed two non-mainframe SNA nodes to act as peers, with reduced system-definition requirements and improved ease-of-use. LEN still relied on a hierarchical SSCP for any routing more distant than one hop. LEN used a connection-oriented packet-switching technique based on the FID2 transmission header format which first arose in subarea SNA for transmissions between a PU type 2 peripheral node (like a 3274 control unit) and a PU type 4 (like a 3725 communications controller).

In 1986 APPN kept the FID2 packet format but added intermediate session routing capability (in network nodes) and a sophisticated new distributed control point (CP). This new CP permitted arbitrary topologies including both peer-to-peer and hierarchical. The majority of system-programmer definitions were no longer needed: APPN figures out where things are located and the best way to get there dynamically, letting users adapt the network to the needs of a changing organization, rather than the other way around.

Now, in 1993-'94, High Performance Routing fulfills the promise of APPN, already acknowledged to be superior to TCP/IP in ease of use and throughput efficiency, by adding to existing APPN new routing functions borrowed from IBM's BroadBand Network Services (more on BBNS later). The new packet format is called Packet Transfer Mode (PTM). PTM combines a new source-routing technique called Automatic Network Routing (ANR) with Rapid Transport Protocol, an end-to-end connection-awareness component that provides a "pipe" on which various data can be multiplexed. ANR permits very fast routing, with priorities, through intermediate nodes and is readily implementable as "outboard routing" on a multiprotocol router. RTP provides effective congestion control in a fast-packet network via its Adaptive Rate

Based congestion control algorithm. RTP also provides any needed error recovery, segmentation/reassembly, and rerouting, end to end, making optimum use of new high-speed links with better bit-error rates than the mix of link speeds and link qualities for which SNA was originally designed.

Finally, BroadBand Network Services, announced in July 1993, has a new control point and a new packet (cell) format, Asynchronous Transfer Mode (ATM), plus PTM for efficient transport of non-isochronous data. The ATM standard defines adaptation layers that map several kinds of traffic to 53-byte fixed-length ATM cells. A new control point was needed defining the new inter-node flows required to control and manage a broadband network with guaranteed Quality-of-Service, perform bandwidth reservation and adaptation, perform multiprotocol directory services, manage multicast groups, and so forth. As none of these functions had yet been addressed by ATM standards, IBM invented the necessary control flows and announced it would make BBNS open [4]. As a new subnetwork, BBNS represents not merely the flowering of the evolutionary tree for SNA, but for other protocols as well. Based on ATM cells plus PTM packets, it carries not only traditional SNA data and that of other protocols, without discrimination, but new real-time data like voice, video, even new complex yet-to-be-invented hybrid multimedia information types.

The Network from Hell — or Coping With Diversity

Multiprotocol networks, or rather several parallel uniprotocol networks, are the rule rather than the exception these days [19]. How did this come about? While the glass house was conducting business as usual in its host-centric network of S/390 mainframe hosts, 3745 NCPs and 3174 control units, out in end-user departments and branch offices, PCs and workstations were achieving new levels of price-performance, and LANs proliferated. The client-server computing trend was born. End users selected applications software based on their business needs, each dragging along a particular network operating system (such as LAN Requester, Novell NetWare, or Banyan Vines) and its associated protocols by default. Later, additional user groups chose different applications, network and LAN protocols burgeoned, and soon routers were the only answer for interconnecting diverse user communities.

Soon the network operations group, based on its WAN operational experience, assumes new responsibility to manage the LAN-based internet along with the traditional SNA network, which must remain stable, reliable, predictable, and fault-tolerant to support such key business processes as order entry, credit authorization, funds transfer, on-line transaction processing, and inventory control. At first the network manager struggles simply to cope with diversity. Getting all the applications simply to coexist can be a daunting task. After developing the skills and know-how to surmount this initial obstacle, the savvy network manager may start to reflect on opportunities to save line costs and reduce administrative expenses.

Network Consolidation

An obvious first step is to consolidate separate physical networks. Just a single LAN can host different network protocols such as TCP/IP, SNA, APPN, NetBIOS, and IPX simultaneously, a single frame Relay link provides the same type of multiplexed shared access in a WAN. Frame relay is exciting because it can be deployed today on existing hardware, bringing added value and possibly years of added life to existing equipment. Frame relay is an open CCITT standard, widely supported by many vendors including most of IBM's networking product line. With the addition of RFC 1294 (multiprotocol encapsulation), a single frame relay virtual circuit can be shared among multiple routing stacks, minimizing the cost of the single most expensive resource in the network — WAN bandwidth [33]. The price of public frame relay is typically not distance-sensitive, either. Where does ATM fit in? Besides its much-touted new capability to carry voice, video, and other types of real-time multimedia along with data, physical network consolidation is one of ATM's promises. And major frame relay vendors — IBM included — say that their ATM offerings will sport the ability to access an ATM net via a standard frame-relay interface [6]. Frame relay is clearly an interim technology, but one that is eminently useful in the short-term to leverage current investments and position for emerging technologies.

With a shared physical network, the manager can economize by purchasing bandwidth in bulk. Is this the final step? No, because multiple separately-administered logical networks still exist. Interim solutions like the new industry standard for encapsulating SNA inside TCP/IP — Data Link Switching (RFC 1434) — do not fully address this problem either.

IBM's AnyNet™ family of products, and the corresponding Multiprotocol Transport Networking architecture [9] [17], an X/Open candidate interface for transport-layer independence, provide an intriguing — and open — answer to this latter problem.

SNA — Out in the Cold?

Will all these changes leave traditional SNA networks out in the cold? The August 1993 Supplement to **Business Communications Review** surveyed several techniques for SNA internetworking, among them SRB bridging, IP encapsulation, and LLC2-to-SDLC conversion [14] [16]. All these approaches tackle the problem indirectly and have tradeoffs. The most natural and direct approach to SNA internetworking is SNA's native routing protocol, Advanced Peer-to-Peer Networking (APPN™). APPN has been available since 1986 on IBM's midrange computers — the AS/400 and System/36. More recent APPN offerings on OS/2 Communications Manager, 3174, VTAM/NCP, the 6611 router, and soon the 3172 and AIX SNA Service/6000 complete the product line. Widespread industry endorsement of APPN makes this approach look increasingly appealing. Still, while widely recognized as a powerful, flexible, and easy-to-use networking solution for client/server [11], analysts continue to point out several APPN shortcomings [26]: (1) it's proprietary, (2) it lacks TCP/IP's dynamic rerouting, and (3) it has restrictions on the support for 3270 sessions, still the bulk of SNA traffic. Let's examine each of these concerns in turn.

But It's Proprietary...Right?

Proprietary? This is an often-heard accusation leveled at IBM. Consider the following:

The Network Node (NN) architecture was published in March 1993 [21]. (Publications in '86 and '91 opened the LEN and End Node (EN) portions of APPN architecture.) For $300, a platform vendor can purchase the APPN Architecture Reference.

IBM offers its own APPN source code to vendors who want rapid time-to-market. The cost is $400,000 plus royalties — the cost of hiring four good programmers for a year. Cheap, considering the development time saved. 3Com, CrossComm, NET, and Cisco have all taken advantage of this offering.

IBM offers a Technology Package giving vendors who develop their own code easy and affordable access to APPN intellectual property including patents, trademark, and copyright permission. The package for NN costs $25,000 per year and the EN package is a $5,000 one-time charge. Predatory pricing? I don't think so. By comparison, consider how broadly other vendors open their proprietary protocols to industry competitors. Who is more open now?

That's not all.

Data Connection Ltd., a U.K. source-code vendor, offers its own third-party APPN source code and has recently garnered a number of OEM customers, including Cabletron, Hewlett-Packard, Hitachi Data Systems, Memorex-Telex, Northern Telecom, Wellfleet, and Unisys [34]. With the list of existing APPN products from System Strategies, Brixton, and InSession, those expected from Advanced Computer Communications, Apple, Ascom Timeplex, Bytex, Eicon Technology, Network Systems Corp,. Novell, Proteon, Retix, Siemens-Nixdorf, Sync Research, and Ungermann-Bass [27], and the 52-or-more APPC LEN (low-entry networking) platforms [8], this makes a pretty impressive showing of industry-wide support.

The APPN Implementers' Workshop, established by IBM to foster high-quality APPN implementations from a variety of vendors, has met quarterly since 1Q93 and has attracted over 200 people from more than 50 companies. The AIW sponsors special interest groups on topics such as SNMP management of APPN, Dependent LU Requester, High Performance Routing, and Data Link Switching. Technical discussions continue between meetings through electronic conferencing and e-mail.

IBM published a draft specification for Dependent LU Requester, a technique to route 3270 traffic natively across unrestricted APPN configurations, at the AIW in 3/93 and since then has been conducting a review of the specification with full participation of all major SNA routing vendors. AIW approval is anticipated in early 1994.

IBM published a draft specification for High Performance Routing in 9/93; the AIW immediately

formed a Special Interest Group (SIG) tasked with completing the spec and ensuring it meets requirements for SNA routing. AIW approval is anticipated on or before early 1995.

It's fair to say that this avalanche of recent announcements from most major networking vendors is a strong endorsement of APPN [1], with its new HPR and DLUR enhancements, as SNA's true heir, offering the kind of plug-and-work dynamics, fairness, class of service, priorization, and ease-of-use, as well as economical use of links required for modern client-server networking, next-generation functions acknowledged to be lacking in today's TCP/IP bridge/router neworks [18].

For all practial purposes, APPN's destiny in the hands of the APPN Implementers' Workshop. APPN meets **my** working definition of open: high-quality, interoperable products from a variety of vendors, competing within a free market to bring you the very best product for your money.

Peer-to-peer networking has come of age. With the addition of High Performance Routing and Dependent LU Requester, APPN will prove itself to be the true industry standard for SNA internetworking and a vital first step on the path to ATM.

What does APPN Do for My Network?

Advanced Peer-to-Peer Networking (APPN™) was designed to make networking truly a democratic affair in which computers of every type can participate on equal terms, as either an application processor/client (End Node), or a router/server (Network Node). Since 1986 APPN has been applying — to the old problem of networking — the "newly-discovered" client/server principles of flexibile application design and the ability to distribute parts of an application across different platforms based on functional requirements. In this case the "application" is the transport of connection-oriented SNA data, including not only trendy new APPC client/server applications (also called LU type 6.2), but the traditional 3270 terminal and printer data often associated with legacy applications. For APPN this means offloading CPU-intensive chores like route selection, maintenance of current network topology data, and distributed directory services to the platform best suited to perform them, a network node, thus freeing up CPU cycles and storage for the applications that users run on end systems from desktops to mainframes [21].

What is a Legacy Application?

I always felt the term "legacy application" was derogatory until a customer explained it to me. To him, a legacy application is one that has been bought and paid for. Its cost was justified to the business at the time. He doesn't lose sleep over it and it's not giving him any new gray hairs because all the bugs were driven out long ago. It's a sunk investment that is now in the payback period and the current return-on-investment is profitable. It's trouble-free and gets the job done.

But APPN is not merely a transport. APPN as a whole meets a series of technical objectives intended to reduce costs, increase user productivity, and ensure reliability and performance. While one can quibble over traditional SNA's ability to route and be routed [15], High Performance Routing (HPR) clearly moots that argument by making SNA data "routable" according to definitions that would satisfy the most critical router advocate.

APPN's goal is to create a highly efficient, scalable "plug and work" network that is easy to use, administer, and operate. At its heart are several distributed network control applications — topology, directory, and route selection — that bring sophisticated automation and dynamics to SNA.

Like other connection-oriented protocols, APPN divides connection setup into three phases. First, it discovers where the remote partner resides, based on the partner's portable NETID.LUNAME address, and obtains all necessary connectivity information. Second, it selects the best route for the particular session, based on the application's class-of-service and priority needs (encoded as a "mode" by an administrator or programmer). Third, a connection (session) is actually established, locking in a chain of locally-unique but dynamically-assigned network-layer addresses, LFSIDs, for the duration of the session, ensuring the presence and availability of all network resources necessary to sustain the connection.

The presence of a connection simplifies problem determination, enables effective congestion control and flow control to maximize throughput efficiency, and allows prioritization of different kinds of traffic, permitting throughput for critical data during times of network

overload, without doubling or triping bandwidth charges.

Should I Wait for HPR?

Should you wait for High Performance Routing? No — why wait when you can realize many of APPN's benefits now. While HPR brings several awaited improvements to APPN, recent independent tests show that **current** APPN performance surpasses that of TCP/IP in all configurations tested.

Current APPN Performance

Recent performance tests by Kevin Tolley, independent tester at InterLab, in SeaGirt, N.J. [7], defied conventional wisdom and showed **standard** (i.e. pre-HPR) APPN outperforming TCP/IP by factors of 20-300%. The test was commissioned by Tom Routt of Vedacom (a Seattle consultancy) as part of a comprehensive 375-page study comparing APPN and TCP/IP to which consultants Dr. John Pickens and Louise Herndon Wells (both in Milpitas, Cal.) also contributed [28].

Using closely matched PS/2 80486 workstations, adapters, and operating systems (OS/2 Version 2), APPN consistently outperformed TCP/IP on a variety of tests, including file transfers and inquiry/response tranactions, even after optimizing packet sizes and window parameters for each protocol under test. In the best case, a single-segment LAN, APPN outpaced TCP/IP by 300%, achieving a steady-state throughput of nearly 15 Mbps on a 16 Mbps token ring. This compared to 4 Mbps through the TCP/IP stack. In the case where the two protocols showed the least difference, routing between two LANs interconnected by a T1 bottleneck link, APPN still surpassed TCP by 20%. The APPN software was IBM's Communications Manager/2 and the TCP/IP software for OS/2 came from FTP Software, Inc.

Topology Database

Like TCP/IP and other popular routing schemes, APPN distributes routing information dynamically, freeing administrators from the task of predefining routes. In APPN this serves users' need to easily add, delete or move any node. APPN calls this information the "topology database". Beyond this superficial similarity, there are many differences between how TCP/IP uses and distributes routing tables and how APPN uses and distributes its topology database (TDB).

APPN excludes end nodes, the majority of computers in a network, from the TDB. The TDB is, therefore, a map of current connectivity between backbone nodes only. The connectivity information includes link characteristics, such as speed, propagation delay, security level, and cost, which may be factored in during route selection. For efficiency, the TDB is updated only on an event-driven basis, for example when the status of a link changes (up or down). Updates, initiated where the change occurs, are rapidly propagated to all other NNs on a long-lived network control infrastructure of CP-to-CP sessions. When a link failure partitions an NN from its network, the NN retains TDB data so when the link comes back up it can checkpoint with its adjacent neighbors, minimizing the flow of topology information at times when network stability is especially critical. (How important is network stability? Just ask a network manager whose job is on the line every time the network is down, where each second down means lost dollars — and orders and customers.)

Information on how ENs connect to one another and to the backbone is acquired through other means. An EN may have any number of links both to the network and to other ENs. Omitting ENs prevents the TDB itself from being a premature barrier to the growth of individual APPN topology domains.

By comparison, most TCP/IP routing protocols are excessively "chatty", some sending full routing table updates at 30-second intervals. IP routing tables include all computers ("hosts") and require preassignment of network addresses. Moving an IP host requires assigning a new address and updating one or more Domain Name Servers. Frequently, although multiple physical paths may exist, only a single logical path is permitted between any two communicating TCP hosts.

Distributed Directory

APPN's directory service lets users add, move, or delete network resources without any concern for address assignments or coordinated definitions. The directory has two basic components: the automatic registration of EN resources (LUs) to an NN server (with an optional second-level registration to a central directory server such as VTAM), and a search function, performed primarily by NNs with occasional

cooperation from adjacent ENs, to discover and cache the locations of remote resources. During the search the NN also learns any additional EN connectivity information not in the TDB, to be factored in during route computation. TCP/IP's Domain Name Server serves a similar function, but requires considerable coordinated definition at the DNS and elsewhere.

Route Selection

APPN's route selection, described in preceding sections, satisfies customers' need to distribute different kinds of traffic onto links with varied characteristics, resulting in a network that can consistently meet response-time and delay requirements without going over budget. Key to APPN's route selection — and perhaps unique among today's internetworking protocols — is the separation of the search phase from the route selection phase. Unlike source-route bridging, which combines the discovery of network entities with route discovery, APPN completely separates discovery from route selection. The path chosen for a session is based exclusively on location and topology information plus class of service, and is unrelated to the path of the search that located the remote resource. Class of service even lets different sessions between the same pair of LUs traverse diverse routes. When several alternate routes of equal weight exist, APPN load-balances effectively among them. APPN also features "multi-tail" connectivity of end nodes to the backbone network, important for high-volume and high-availability configurations.

APPN Benefits Confirmed

Confirmation comes from Gerry Williamson, senior systems programmer at Provident Life and Accident Insurance (Chatanooga, Tenn.), whose firm evaluated both TCP/IP and APPC/APPN for a new client/server application before choosing APPN [3]. This 4200-employee company with $17 billion in assets was one of the early users of VTAM V4R1 APPN.

Provident prototyped a client/server file transfer application between MVS and OS/2. TCP/IP on MVS with 3172 and standard ftp achieved file transfer rates of 500 kilobytes per minute. A comparable assembly-language MVS APPC program, talking to an OS/2 APPC prototype application written in IBM's batch language, REXX, initially attained 6 megabytes per minute, 12 times faster. With the REXX client rewritten in C, throughput surged to 10-12 megabytes per minute, outperforming TCP/IP more than 20-fold.

Beyond performance, Provident found other benefits of APPN, including simplified network definition and immediate availability after moves/changes. This in turn reduced administration costs and increased productivity. Before implementing APPN, OS/2 network definition files (.NDFs) were 266 lines long, due to predefining each remote application. Each move or change to any application required distributing an updated .NDF file to every PC. With APPN, .NDF files shrunk to 66 lines, but more importantly no longer required maintenance with every new application.

A key strategy to minimize the size of their support staff was to minimize the choices in their network. This is an unusual choice, but it paid off handsomely: their normalized total networking expense is only 58% of the average. The Provident network handles two million transactions per day with a total end-to-end response time of 1.5-1.75 sec. (of which 0.8 sec. is spent in host processing) over 14.4 and 56 kbps lines.

High Performance Routing — APPN+

HPR inherits all the ease-of-use and dynamics inherent in the APPN control point. Beyond that, as a next-generation routing protocol, HPR combines the best features of connection-oriented and connectionless networking. (It's interesting to note that the Internet Engineering Task Force in RFC 1290 acknowledges the need for TCP/IP to become more connection-oriented if it is to keep up with a changing user community.) Best of all, HPR runs on existing hardware (as ATM does not), and is deliverable as a small software upgrade, adding only about 10% more code to an existing APPN EN or NN product. Because HPR is "APPN+", it interoperates seamlessly with existing APPN products, permitting an easy and gradual migration. SNA sessions will run over any combination of APPN and APPN+ nodes.

From IP, HPR borrows (and improves upon) the concept of dynamic rerouting — letting sessions survive link failures. From TCP, HPR borrows (and improves upon) the concept of an end-to-end connection, removing connection awareness from intermediate nodes. From SNA, HPR borrows the deterministic stability and support for priority and class of service associated with having a predetermined route. And HPR adds some innovations that it has in common with IBM's new ATM control point, BroadBand Network Services: ANR, RTP, and ARB. APPN also uses existing data links, such as Frame Relay, LANs, or PPP, in a connectionless manner when warranted by low bit-error rates.

HPR Data Links

Error recovery at the data link layer (i.e., DLC "elements of procedure") is optional in HPR. If the link is good (has few bit errors), DLC recovery is omitted. In this case the DLC's only job is to check data integrity using a standard frame check sequence, and discard any packets in error. On a LAN, for example, HPR uses the connection-oriented LLC2 (802.2), but operates it in a connectionless manner by sending unnumbered information (UI) frames. Lost or discarded packets are recovered by the RTP endpoints.

Automatic Network Routing

Automatic Network Routing (ANR) routes packets on a predetermined path, omitting connection awareness, software overhead, and control block storage in intermediate nodes. It readily supports router hardware architectures, provides faster routing, and increases the number of connections a given router can carry.

Similar in concept to token ring Source-Route Bridging, the full route prefixes each packet as a list of ANR labels. At link activation, individual nodes assign ANR labels to their own links. Labels are typically 1-2 bytes, but may be longer. Label assignment requires neither negotiation or system definition. An HPR node selects a route via the standard APPN route selection algorithm (class of service, destination, topology database). During route setup, the origin node learns the ANR labels representing each link specified in a standard APPN Route Selection Control Vector (RSCV). The ANR string may be any length. The ANR string is then used for all subsequent packets, until a link fails and the RTP endpoints reroute to a new path. During a path switch, a new ANR string is generated (and a new Route Setup performed if needed) and used for subsequent packets. Multiple sessions with the same priority can share a single RTP connection, reducing storage needs and expediting re-routing.

Routing a packet is as simple as considerating the priorities of all packets awaiting transmission, then (for the packet chosen for transmission) inspecting and deleting the first label — always at a fixed position in the header — and sending the packet on the link identified by the stripped label. Although packets shrink during travel, the route's endpoints retain end-to-end route information for management purposes. The labels for the reverse path may differ from the forward path.

ANR reduces the number of instructions required to route a packet in a manner that can improve internal performance by 3-10 times over basic APPN intermediate session routing (ISR). It also lets a given intermediate node support a greater number of intermediate sessions by eliminating NN control block storage.

Rapid Transport Protocol

RTP is a next-generation "lightweight" connection-oriented transport with an interesting pedigree: TCP evolved into Protocol Engines, Inc.'s "Express Transfer Protocol" (once proposed to IEEE 802.2 as an LLC type 4 [30]), which with the removal of dependencies on particular topologies and underlying network protocols, and the addition of certain new features, in turn evolved into RTP.

RTP was designed for fast-packet networks with good links. It optimistically assumes that in the normal case, everything goes right [10]. (Previous protocols assumed everything would go wrong, optimizing for error cases instead of steady-state operations.) RTP connection setup and takedown protocols are very fast: the designers benefitted from the lessons of TCP and XTP [22] [29]. Using data streaming and piggybacking, RTP exploits the fact that new computers often have larger buffer capacities and can send at high speed. The number of acknowledgments, status, flow control, and other protocol handshakes is reduced, and those still needed often ride "piggyback" on data messages to further reduce overhead.

For earlier, error-prone links, segmentation to an optimum packet size per link was the best strategy to optimize throughput [31]. Thus APPN ISR does segmentation and reassembly on a hop-by-hop basis as needed for packet-size mismatches, requiring buffering and session awareness in each intermediate node. In fast networks this can reduce performance. RTP determines the largest packet size supported across an entire route, and segments packets exceeding this maximum before the first ANR hop. No processing is done at intermediate nodes; the receiving RTP reassembles the segments.

RTP provides reliability by byte-sequence-numbering the data, identifying missing sequence numbers, and selectively retransmitting missing data. Unlike TCP or SDLC which use "go-back-n" recovery protocols, RTP uses a more efficient "selective retransmission" algorithm. "Go-back-n" means if I send packets 1,2,3...10 and you missed number 3, then I must "go back" and retransmit numbers 3,4,5,6...10. Selective retransmission would only retransmit number 3. This same logic is a foundation for MLTG support in HPR, enabling the software to rearrange packets that arrive out of sequence. It is likewise fundamental to preserving the packet order expected by SNA's connection-oriented upper layers during a path-switch.

RTP implements non-disruptive path switching using a timer to detect link or node failures along the path. After several retransmission attempts, RTP asks the APPN control point to compute a new path: the sessions stay up.

Adaptive Rate-Based Congestion Control

ARB is based on a rate-control and feedback mechanism conducted between the two endpoints of an RTP connection. Each RTP node periodically samples the rate at which it is sending, receiving, and delivering received data to an application (thus freeing buffers to receive new data). Rate messages are periodically piggybacked on data. Based on the rate that is getting through the network, the endpoints either speed up or slow down. By measuring throughput, they are able to determine their location on the throughput efficiency curve (see “How Link Costs Will Shape Future Networks” and Figure 2) in time to return the network to its peak operating state before response-time delays occur or throughput is reduced.

ARB Efficiency Confirmed

In one IBM experiment designed to single out the effects of slow-start versus ARB, ARB's throughput efficiency exceeded TCP's by 28%. Although only around 792 packets were lost during a run of 163,515 packets, TCP retransmitted a total of 2446 times. 1654 of those retransmissions were unnecessary, since the packets were not actually lost, but merely delayed due to excessive congestion. In contrast, a similar number of packets (675) were lost during the ARB trial, but only 795 were retransmitted, and of that number, only 120 were retransmitted unnecessarily. TCP's unnecessary retransmissions exceeded those of HPR's ARB by 1300% [20].

If two or more RTP connections have common links, the contending ARB mechanisms operate to share each link's available bandwidth fairly among competing traffic streams. This mechanism shows promise to provide fairness not only among SNA traffic but possibly for multiprotocol traffic carried over HPR as well.

IBM Product Directions

HPR is clearly a strategic direction for IBM and one to which it is committed long-term. IBM has signalled that 3172, 6611, VTAM/NCP/3745, 3174, Communications Manager/2, and AS/400 are all candidates to receive the new function in a short- to medium-term timeframe. Considering the small size of HPR and its significant "bang for the buck", and that a draft of the architecture has already been published at the APPN Implementers' Workshop, rapid deployment of HPR appears inevitable. With the removal of configuration restrictions for 3270 traffic in APPN via the Dependent LU Requester architecture, also published at the AIW and also slated for product availability in the short- to medium-term, APPN+ appears to meet the needs of the expanding enterprise LAN internet.

How Link Costs Will Shape Future Networks

Or, Just How Important is Throughput Efficiency?

No one doubts that a network's ability to perform effective congestion control translates directly into line cost savings. This is a key differentiator between the SNA protocol family — subarea, APPN, and HPR — and others such as LAN- and TCP/IP-based protocols. Choosing private-network equipment based on its ability to yield anticipated phone-bill savings, however, depends on your projections about the future of telephone charges.

Line costs typically account for 25-40% of a networking customer's costs [12]. Will this cost remain significant over the projected lifetimes of networking equipment, or are line costs falling? A U.S. government General Accounting Office study showed that from 1990-1992, long-distance telephone costs (from interexchange carriers such as AT&T, MCI, WilTel or Sprint) for a T1 (1.5 Mbps) link declined an average of 12% per year. Meanwhile, local T1 links (from a Regional Bell Operating Company such as NYNEX or Southern Bell) became more expensive at a rate of 7% annually. Local telephone accounts for about 50% of the average networking telephone bill. As LAN-interconnect traffic increases, many customers find that their networking costs are gradually shifting from interexchange carriers to local exchange carriers, resulting in a zero net change in the phone bill. There continues to be a good opportunity for private network vendors to provide value-add over public carrier services.

Will Line Costs drop to Near Zero?

A low incremental cost to add additional DS3 circuits in metropolitan areas is often cited to predict the rapid penetration of broadband technology [25]. The same data are sometimes cited to support the view that the price of bandwidth will soon drop to a level where it becomes an insignificant factor in network planning. But if you believe line costs will drop to near zero, then you believe that all carriers will go out of business, and frankly I don't think their stockholders will let that happen. On the contrary. Because of inter-carrier competition, the continued high cost of infrastructure maintenance, inflexible cost structures related to long planning horizons, and other factors, the carriers' "wholesale prices" for bandwidth in bulk will always be cheaper for those consumers who require large volumes, and this relationship will hold even as continued fiber deployment pushes down the raw price-per-bit. I predict a rosy future for private networking.

Internationally, telephone rates are still sky-high in many locations, and the ability to deploy new service in a timely or cost-effective manner is limited by the inefficiencies of government PTT monopolies.

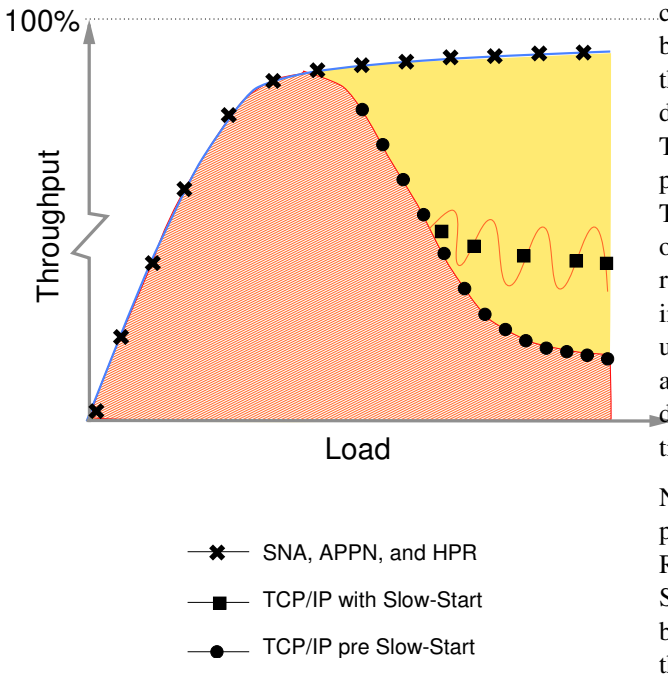

Figure 2. Throughput Efficiency

In Figure 2, the horizontal axis represents the load on a network and the vertical axis represents the network's throughput. For equal values of load and throughput, output equals input: the network is keeping up with the load and no packets are being queued for long. Under light loads, protocol differences are not always apparent; under peak loads, dramatic differences in throughput effiency can be observed. (See also [13] and [2].)

Examine the bottom curve. The TCP/IP protocol suite has no effective congestion control. At best, it reacts after congestion develops, but without a consistent path, congested nodes have no certain way to notify senders to slow down.[1] Originally TCP had no flow control either. Around 1988, portions of the Internet experienced a series of congestion collapses as more traffic entered the network than network designers had planned for. As queuing delays got longer or packets were lost due to congestion, the TCP protocol timed out, causing retransmissions [5].

Look at the middle curve. Van Jacobson invented "slow-start" flow control for TCP to help address the congestion problem. A lightly loaded network is fine, but if load exceeds some critical point, the average throughput of the network declines. Not only does it decline, but response time becomes highly variable. These protocols are also very sensitive to delay, timing parameters, and cross-traffic. Even with slow-start, a TCP/IP network resembles a city with no speed limit or traffic lights. All the drivers try their hardest to reach their destinations on time, but because each intersection is unregulated and overall routing is unplanned, less traffic gets through, long tie-ups occur at busy intersections, even the alternate routes bog down as all the drivers decide to reroute at the same time, and some collisions and other mishaps occur.

Now examine the top curve representing SNA fixed pacing, APPN adaptive pacing, and HPR Adaptive Rate-Based congestion control. All three of these SNA protocols are designed to prevent congestion before it occurs, and to maintain high levels of throughput in a steady state, under all sorts of loads. In fact, a properly-tuned SNA network can run at 95% utilization with adequate response time [32].

One way to Improve Competitiveness

The option to design a network with excess unused capacity is not generally available to private businesses. Said another way, those who minimize line costs can improve their profitability as compared with less-efficient competitors.

SNA congestion control operates in concert with SNA prioritization. SNA marks different types of traffic with four priority levels. For example sending the Webster's Dictionary might be low priority but processing a credit card authorization might be high priority. A read/write transaction that temporarily locks a shared database must certainly be considered high priority, due to its potential to disrupt the entire operation if delayed. When the load on the network exceeds capacity, high-priority traffic gets through, while low-

[1] It's worthwhile to note that during the development of the TCP/IP protocol in the original government-funded research networks, ArpaNet and DoDNet, developers had no real motivation to develop effective congestion controls: network managers dealt with the problem by adding new links, at taxpayers' expense.

priority traffic is delayed through back-pressure (flow control). SNA network designers know that you can design an SNA (or APPN or HPR) network based on the peak anticipated interactive traffic load, and batch traffic gets the "leftovers" at no extra cost. Being connection-oriented gives SNA a means to apply this back-pressure effectively. SNA congestion control is like a well-regulated city street grid or highway. By admitting traffic gradually and spacing the cars apart, speeds are maintained, few accidents occur, and ultimately everyone gets to their destination faster. During overloads, prioritization allows "high occupancy vehicles" such as buses and commuter vans first access to the highway, maximizing the number of commuters who get to work on time.

The Last Puzzle-Piece: Dependent LU Requester

It is widely, although incorrectly, believed that APPN does not support 3270 traffic. This view is probably at the heart of a corresponding and likewise mistaken belief that SNA is non-routable. Truth can be stranger than fiction, so bear with this explanation for a moment.

3270 data, also called dependent LU traffic, is generated by logical units type 0,1,2,3 (older printers and terminals) and their control units. Built in the days before personal computers and having little or no distributed intelligence, dependent devices need control sessions with a host System Services Control Point: SSCP-PU and SSCP-LU sessions. Many older 3270 devices are still functioning well and are not slated for replacement anytime soon.

As discussed in Figure 1, the packet format or SNA transmission header (TH) used by APPN evolved from the formats and protocols for PU type 2.0 peripheral nodes. For PU type 2.0, certain fields in the TH — the Origin Address Field or OAF and the Destination Address Field or DAF — identify transmissions **from** and **to** the host, and distinguish data bound for one dependent LU from another.

When Low-Entry Networking came on the scene, the field containing the OAF and DAF expanded by one bit to let the new T2.1 peer nodes dynamically assign session addresses, now called Local Form Session Identifiers (LFSIDs). For migration, the new LFSID encompassed all existing OAF/DAF address pairs, enabling APPC traffic to coexist on the same links with older 3270 data. APPN's Intermediate Session Routing uses LFSIDs to identify not only sessions terminating locally, but those routed across the network too. Thus from a packet-format perspective, APPN could always accomodate 3270 traffic natively. The problem lies elsewhere, in superimposing the hierarchical control point structure required by dependent devices on top of the flat APPN infrastructure without disturbing APPN's ease-of-use advances. The good news is that existing APPN NNs, for example AS/400, **can** route 3270 data natively.[2]

One challenge came in migrating the older SNA session-setup flows required for dependent devices to a pure APPN environment. These cross-domain search flows, once conducted only between mainframes on SSCP-SSCP sessions, support the widely-used ability to write sophisticated VTAM applications. Functions include secondary-LU-initiated sessions (as when a dependent terminal powers on and automatically receives a host logon screen), third-party-initiated sessions (as when an application signon request is routed from a logon-manager to a different host where the application resides), queuing for a single-session-capable device to become available for a new session, automatic logon, and so forth. This challenge was addressed in VTAM V4R1. VTAM's APPN support includes the ability to perform all these functions in pure APPN and mixed APPN-subarea environments.

The second challenge, now being solved by the Dependent LU Requester architecture, is to remove configuration restrictions on the placement of dependent devices in an APPN network. Even with V4R1 — barring gateways or bridging — dependent devices still must be logically adjacent to the host pro-

---

[2] Contrast "natively" with "encapsulation". You can tell if a protocol is encapsulated by examining the packet structure, categorizing the various headers and counting them. If you count more than one instance of a particular layer's header (for example, two Data Link Layer headers or two Transport Headers), then encsapsulation is being used. So, while Data Link Switching is a standard for carrying SNA traffic over TCP/IP, it is **not** native APPN routing: it is simply an efficient encapsulation method.

viding them SSCP services. Yes, there are other solutions. In IBM's view, the number of technical solutions proposed, and implemented, reflects the diversity of customer requirements (see, for example, [24] [14] [16]).

Dependent LU Requester (DLUR) solves this final problem in the most generic and flexible manner. By adding a small piece of client logic (the DLUR) to existing APPN nodes that support the attachment of older devices, SSCP-PU and SSCP-LU support can be "piped" over an arbitrary number of APPN hops, from an existing SSCP to wherever it is needed. A corresponding piece of server logic (the Dependent LU Server — DLUS) in the host, expected on VTAM in the very near-term, completes the picture.

DLUR Benefits

What are some of the benefits? Dependent sessions can now benefit from advanced APPN route selection, gaining the benefits of dynamic alternate routes, no longer consuming valuable mainframe MIPS or 3745 control block storage for pure routing functions, saving money or relieving some of the burden on a "maxed-out" 3745. LU-LU sessions can traverse different — possibly more direct — paths through the network than SSCP control sessions, leading to improved configuration flexibility and better performance. Dependent sessions over APPN will benefit from HPR's nondisruptive rerouting. With or without HPR, if a DLUR control session is lost, LU-LU sessions will not be disrupted, resulting in higher availability. Potentially, DLURs could even be programmed to actively seek out any available DLUS anywhere in the network, further improving availability. DLUR greatly benefits network management, since gateways, bridges, and routers between a dependent device and a host that were formerly invisible to network management now become visible and manageable when replaced by DLURs. DLUR also offers simple migration, since it can be installed virtually anywhere in an APPN network. Of course, existing devices such as 3279 terminals, 3270 emulators, and 3274 control units, and existing applications, are unchanged.

A more fundamental benefit is the APPN flexibility DLUR confers on traditional SNA applications. DLUR lets users adapt the network to the needs of a changing organization, rather than the other way around.

Proprietary? No. A draft DLUR specification, published by IBM at the APPN Implementers' Workshop in early 1993, is slated for approval early in 1994, and numerous products — such as IBM's 3174 — are planning for early releases.

Summary

Among many alternatives for SNA internetworking, the most natural and direct approach is SNA's native routing protocol, Advanced Peer-to-Peer Networking. Until recently APPN has not enjoyed as much growth as its reputation as a powerful, flexible, and easy-to-use networking solution for client/server and distributed applications warrants. This report has examined APPN's remaining shortcomings and showed how each of them is being addressed. With the removal of these last remaining concerns, APPN+ appears well positioned to meet the needs of the expanding enterprise LAN internet.

About the Author

Marcia Peters leads the outreach effort for IBM Networking Systems' APPN architecture team in Research Triangle Park, North Carolina. She also contributed to Broadband Network Services, IBM's strategic architecture for controlling and managing ATM networks, and developed APPN enhancements for seamlessly interconnecting APPN and and subarea SNA. Before joining IBM in 1988, she developed plug-compatible 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 six U.S. patent applications and published over 40 technical disclosures and articles.

References

[1] anonymous, "InSession Moves Faster," *Communications Week*, p. 2, October 25 1993. Plans to add High Performance Routing to a non-IBM APPN end node product.

[2] 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. Hardcopy available by mail. Contact mpeters @ ralvm6.vnet.ibm.com. Describes dynamic problems of TCP and shows that APPN/HPR with ARB significantly outperforms TCP congestion control. Appendix B contrasts APPN and TCP/IP.

[3] Gerry Williamson of Provident Life and Accident Insurance Co., APPN at Provident, APPN Implementers' Workshop, 3, September 1993. Presentation on Provident's experiences as an early user of VTAM V4R1 APPN. Discusses APPN vs. TCP/IP performance trials and other APPN benefits.

[4] Andrew Collier, "IBM to Open Parts of Autobahn Architecture to Fellow Vendors," *Communications Week*, pp. 27-28, June 1993. Discusses IBM's commitment to make its ATM control point an open standard.

[5] Douglas E. Comer, *Internetworking with TCP/IP*, Prentice Hall, pp. 190-193, 1991. Discusses problems of TCP/IP flow control and the collapse of portions of the Internet.

[6] Michael Cooney, "The New IBM Makes Bold ATM Pitch," *Network World*, pp. 1,47, July 19 1993. Covers IBM's ATM announcement.

[7] Michael Cooney, "APPN tops TCP/IP in performance face-off," *Network World*, vol. 10, no. 41, October 11 1993. Summarizes InterLab independent tests of APPN vs. TCP/IP. APPN surpassed TCP/IP in all tests, and in some by as much as 300%.

[8] IBM Corp., The APPC Resource Book, IBM Corp., 1993. Courses, books, development tools, applications, and platforms supporting APPC.

[9] IBM Corp., Multiprotocol Transport Networking (MPTN) Architecture: Technical Overview, IBM Corp., Publication order number GC31-7073, 1993. Introduces principles and concepts of multiprotocol transport networking.

[10] 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

[11] Norman Friedman, "APPN Rises to the Enterprise Architecture Challenge," *Data Communications*, pp. 87-98, February 1993. Review APPN's key features and strengths as an enterprise architecture.

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

[13] James P. Gray and Marcia L. Peters, "A Preview of APPN High Performance Routing," *Local Area Network Interconnection*, pp. 139-160., New York and London: Plenum Press, 1993. Describes basic issues of congestion control. Positions HPR with respect to OSI 7-layer model. Surveys existing routing techniques.

[14] Guruge, Anura, "Bridging the gap between SNA and multiprotocol internets," *Business Communications Review Supplement on Internetworking*, pp. 2-4, August 1993. Reviews 2 dominant techniques for integrating SNA and non-SNA LAN traffic: source route bridging and IP encapsulation.

[15] Guruge, Anura, "SNA Routing," *Business Communications Review Supplement on Internetworking*, p. 15, August 1993. Argues in favor of the proposition that SNA is and has always been routable, via PU type 4.

[16] Guruge, Anura, "Merging SNA link traffic into LAN/WAN internets," *Business Communications Review Supplement on Internetworking*, pp. 18-23, August 1993. Reviews 4 techniques to integrate SNA/SDLC link traffic with LAN traffic: 1. Synchronous Passthrough, 2. remote polling, 3. SDLC to Token Ring LLC Transformation, and 4. remote SNA switching with host passthrough.

[17] Guruge, Anura, "IBM's Networking Blueprint and AnyNET," *Business Communications Review*, pp. 43-47, August 1993.

[18] Anura Guruge, "The Blue-Brick Road to ATM," *Business Communications Review*, pp. 31-35, October 1993. Analyzes IBM's strategy to help users migrate to ATM from existing SNA and multiprotocol networks. Discusses the role of APPN, HPR, DLUR, Frame Relay, and AnyNET in the migration.

[19] Guruge, Anura, "Integrating SNA and LAN traffic: It's wait and see," *Network World*, vol. 10, no. 33, p. 36, Aug. 93. Discusses the dreaded "parallel network syndrome".

[20] Lap Huynh, Private communication, to be published. Results of experiment isolating effects of TCP slow-start and HPR's adaptive rate-based congestion control. ARB outperformed slow-start by 28%.

[21] 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.

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

[23] 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.

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

[25] John M. McQuillan, Next Generation Network Architectures, 1991. Marginal cost of additional DS3 circuits in metropolitan areas.

[26] Maureen Molloy, "APPN May Hold Key to SNA/LAN Integration," *Network World*, vol. 10, no. 34, pp. 1, 118, August 23 1993. Routable APPN may emerge as a solution to support virtually any kind of traffic, but users are still divided about the APPN question.

[27] Kim Murray, APPC/APPN: Open, Advanced, and Available, IBM Corp., October 1993. Highlights current status of APPC and APPN in a wide variety of products. Available electronically on Compuserve via "GO APPC".

[28] Thomas Routt, John R. Pickens, and Louise Herndon Wells, APPN & TCP/IP: Comparisons, Contrasts, Integration, Decision Drivers, Vedacom Corp., Seattle, Washington, phone 206-527-3434., September 1993. 375-page consultant analysis addressing all facets of the APPN and TCP/IP comparison. Includes fuller discussion of InterLab performance study in which APPN outperformed TCP/IP by as much as 300% in similar configurations.

[29] 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.

[30] Stallings, William, "Streamlining High Speed Internetworking Protocols," *Telecommunications*, no. 91-14569 V25 N2(North American Edition), pp. 43-47, Feb. 1991. Projects the acceptance of XTP as a standard.

[31] 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.

[32] John G. Waclawsky, TCP/IP and SNA Performance and Stability, IBM Advanced Enterprise Networking Briefing, September 1993. Reviews APPN and TCP/IP performance and stability characteristics and highlights shortcomings of TCP/IP flow control.

[33] Joanie Wexler, "Frame relay upstages ATM," *Computerworld*, p. 12, September 6 1993. Discusses frame relay's tactical advantages for customers who need to consolidate data traffic and do not yet need to combine voice/video with data.

[34] Tim Wilson, "Six Vendors License APPN Source Code," *Communications Week*, p. 65, October 25 1993. Six additional non-IBM companies license source code to develop their own APPN products.