IBM Communications Server
June 2002

Energize your business network.

Contents

Abstract

Communications Server is IBM's premier open solution for your networking needs, providing industrial strength SNA services on a workstation platform, over both SNA and TCP/IP networks. It is a high-performance multifunction multiprotocol gateway and networking application platform that enables workstations to communicate with other workstations and with host computers. It offers a robust set of communications and systems management features, to enable users to communicate through terminal emulation and client/server and distributed applications across local and wide area networks. It also offers a rich set of network management and client/server application programming interfaces (APIs). And Communications Server provides the widest range of connectivity across the industry.

Communications Server is the solution for companies who:

- run multiprotocol or multiple networks
- want to consolidate or change their backbone networks
- have SNA applications that they want to extend over TCP/IP networks
- have sockets applications that they want to extend over SNA networks
- want to provide SNA 3270 host access to TCP/IP users using TN3270E emulators
- want to provide IPX and TCP/IP attached clients access to SNA applications
- have users who want to get connected to a corporate intranet or to the Internet
- want to improve network availability
- want to access data from anywhere using familiar interfaces and protocols

Communications Server provides an essential foundation for networked computing by providing industrial strength support for the most widely used networking technologies, enabling customers and business partners to build client/server applications that are independent of networking protocol or hardware, while preserving their investments in existing systems. Throughout this paper, the term "Communications Server" will be used to refer to IBM Communications Server for AIX, IBM Communications Server for Windows NT and Windows 2000, and the IBM Communications Server for Linux PRPQ. The full product names will be used where appropriate to highlight any differences between the products.

IBM Host Integration Overview

IBM Host Integration software provides enterprise-class universal connectivity and information access for cost-effective network computing. By combining IBM's expertise in delivering industrial-strength solutions for the enterprise environment with the latest networking technologies, Host Integration software provides the foundation you need to capitalize on the latest technologies and ways of doing business.

IBM Host Integration software offers a full range of networking software products:

> The Communications Server product line connects people and applications, even when platforms and networks are diverse. This gives you the freedom to address business issues without being hindered by application dependencies or network design.

With a broad range of communication clients, Host Integration software puts information within the reach of all users. From Personal Communications, the industry's premier solution for emulation and desktop communication, to Host On-Demand, a Web-based solution for accessing host applications, you can select the access method that's best suited to your needs.

It's more than just breadth of products that makes Host Integration software unique in the industry. Host Integration software products are designed and built on the essential elements required to address your networking needs:

- Enterprise-class dependability
- End-to-end universal access
- Easy implementation and use
- Effective network utilization

With Host Integration software, you'll see your network as a competitive business advantage, not an IT constraint. For more information about IBM Host Integration software, visit our Web site at:
http://ibm.com/software/webservers/hostintegration

Communications Server: Energize your business network

What is Communications Server?
IBM has long been a leader in communication software. With the introduction of the Communications Servers IBM has implemented the broadest set of functions and connectivity, all based on industry-standard solutions and optimized for the

platform of your choice. Fundamental to today's client/server and networked computing environments, Communications Server builds on the leading peer-to-peer networking protocols of Advanced Peer-to-Peer Networking (APPN) and TCP/IP. Communications Server has been built from tried, tested and award-winning components, to provide advanced function and performance as a standalone communications gateway or in support of other communications and application servers. Communications Servers have evolved into multiprotocol, multifunction gateways designed for interoperability.

Communications Server meets the requirements of customers who need reliable and powerful networking support, to enable workstations to communicate with other workstations and with host computers over today's leading networking technologies, and who need to maintain a competitive edge.

Communications Server meets those requirements by providing:

- powerful multiprotocol gateway function to connect SNA and TCP/IP networks, enabling SNA and sockets applications on any platform and from any vendor to be transported across interconnected SNA and TCP/IP networks
- TN3270E server function to provide SNA 3270 access to host systems for TCP/IP users using TN3270 emulators
- powerful SNA gateway function for any product using the industry standard 3270 display and printer protocols
- powerful multiprotocol access node function, which gives customers protocol independence, enabling them to run SNA and TCP/IP applications

regardless of the transport network to which workstations and hosts are connected

- full support for APPN, providing state-of-the-art dynamic routing for SNA applications, such as client/server applications written using APPC or CPI-C, including 5250 and 3270 protocols, high performance routing (HPR), and dependent LU requester (DLUR) support
- Enterprise Extender which enables HPR applications to run over TCP/IP networks
- Branch Extender for very large APPN networks which hides branch topology from the WAN, reducing WAN traffic
- a rich set of programming interfaces for client/server and network management applications
- the most extensive range of connectivity options in the industry for local area (LAN) and wide area networking (WAN)
- Communications Server for Windows NT allows IPX and TCP/IP attached clients to access SNA APIs without SNA Protocol (split stack)

Today's networking challenge

In today's constantly changing environment, effective businesses depend for their prosperity, and even survival, on their ability to exchange information quickly and efficiently throughout the organization. The network that makes this possible should be and usually is invisible to its users until it slows down or stops altogether. In any organization, the challenge for the network manager is to keep that network invisible, delivering data where it's needed, when it's needed, while at the same time ensuring that future requirements can be catered for with minimum disruption.

In the early days of computing networks, while this was not always easy, it was usually relatively straightforward. The number of key business applications to be supported was small. The work force was often centralized. The choice of networking standard was usually dictated by the choice of mainframe - for very many companies, of all sizes, this meant SNA. Today, the challenges of interconnecting people and applications in an invisible network are considerably greater. Corporate networks are required to support a constantly increasing number of diverse and complex networked applications. The work force is increasingly dispersed. The choice of networking technologies and protocols is no longer clear cut. Many organizations now have multiple networks, whether as the result of mergers or acquisitions or reorganization, or arising from a business strategy to incorporate the best technology for individual business requirements.

Given this diversity, the network manager is now expected to

- accommodate complex and constantly changing application requirements, while protecting the investment in existing applications
- maintain existing service level agreements for response time, availability and reliability, while maximizing the efficiency of the network and minimizing its costs
- position the organization to exploit technologies, such as client/server applications, the Internet and high-speed networking such as ATM

Communications Server is designed to address these challenges by

- delivering protocol independence
- delivering advanced networking solutions
- delivering breadth of function and connectivity and capacity to support future needs

How it does this and the key technologies it uses are explained in the following sections.

Delivering protocol independence

Providing SNA 3270 application access to TCP/IP users
Communications Server provides TCP/IP and Internet users with easy access to IBM central computers through support of TN3270. This support provides Telnet 3270E server functions to TN3270E/TN3270 clients. It provides the functionality to convert Telnet client traffic to SNA format for communications to the host system and converts SNA traffic to Telnet format for communications to the clients. Supporting the industry-standard TN3270E, Communications Server provides 3270 terminal and printer emulation to TCP/IP users enabling support of locally attached printers or printers in their TCP/IP network. TN3270E support also includes confirmation of print requests, positive and negative responses, handling the ATTN and SYSREQ keys, and support for LU classes (which simplifies user access, groups users by applications needs, and maximizes host resources). Telnet 3270E clients have the ability to request specific host LUs. In addition, an administrator can dedicate host LUs to specific Telnet 3270 and Telnet 3270E clients providing a higher level of security. Communications Server supports any

industry-standard TN3270 or TN3270E client and is compliant with industry
standard Request For Comment (RFC) 1576, 1646 and 1647. The TN3270E Server function enables you to configure your network as shown in Figure 1.

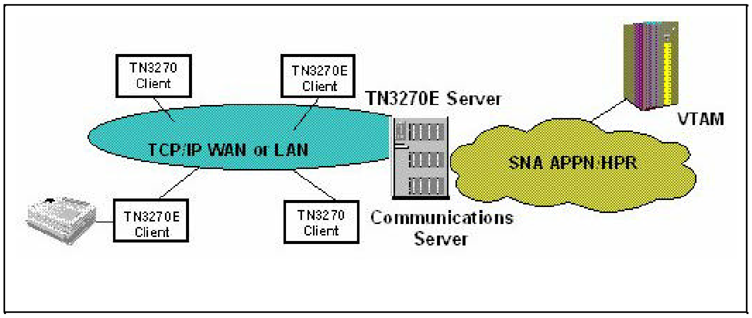


Figure 1. TN3270E Server with Wide Area TCP/IP Network

The TN3270E Server function supports:

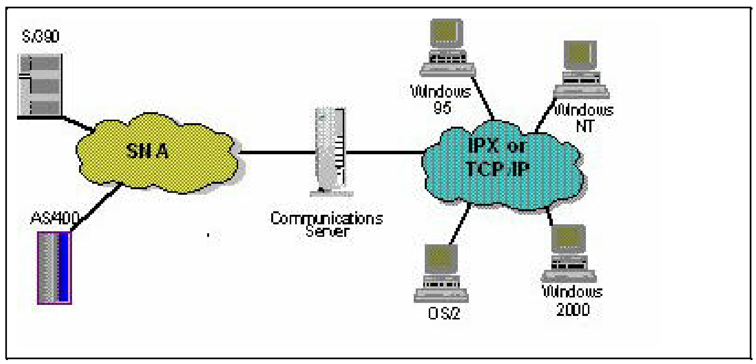
- Terminal emulation
- Host print
- Response handling
- ATTN and SYSREQ key handling
- LU classes

TN3270E Server supports both standard and extended Telnet 3270. Typical client programs emulate a 3270 display. Clients that support the TN3270E protocol can emulate LU1 and LU3 Printers.

SNA API client solution with Communications Server for Windows NT

The Communications Server SNA API client solution allows TCP/IP and IPX attached clients to access SNA APIs. This solution also gives you the ability to run SNA applications without installing an SNA stack on the client. Because almost all SNA configuration and processing is done on the server, you can reduce DASD, memory, and processor demands on your clients. And your System Administrator saves time by not having to configure SNA on every client.

The SNA API clients provide support for CPI-C, APPC, EHNAPPC, and LUA request unit interface (RUI) API interfaces and are packaged with the Communications Server for Windows NT. Supported clients include OS/2, Windows 3.1, Windows 95, and Windows NT as illustrated in Figure 2.



F

igure 2. TCP/IP and IPX Attached SNA API Clients

Multiprotocol Support
In addition to providing access to SNA 3270 applications to TCP/IP users, Communications Server delivers network protocol independence, allowing application design decisions to be made based on the characteristics of the application and not on those of the existing network. In this section we will explore why this is important and how this is achieved.

With the growth of networking in general and local area networks in particular, many large customer networks have become collections of individual networks running different networking protocols. This situation occurs and will continue to occur for many reasons: as one company merges with another that already has a different kind of network; as a company decentralizes and increased autonomy encourages each department to make its own decisions relating to IT infrastructure; as customers become more concerned with choosing or developing the right business application than with the networking interface or protocol for which it was designed. In many cases, whether by accident or design, companies run duplicate physical networks. This considerably increases their IT infrastructure costs and requires them to invest in a multiplicity of increasingly scarce skills. It also makes network management more complex.

A number of solutions have arisen over the last few years to try to solve this problem. One of the most common techniques is encapsulation, or tunneling, in which one protocol is enveloped in another for transportation across the backbone network. This technique has been widely used,
partly because it is relatively easy to implement, but it has its drawbacks. Getting the application data onto the network involves processing the data through layers of code all the way from the application, through the original network protocol to the point of producing a transport-level packet, complete with network headers; this packet is then presented to another protocol stack as application data, and processed a second time to add another set of headers. Traversing two protocol stacks adds to the computing time required to process the data. The additional headers -10 bytes of SNA, for example, can get encapsulated into TCP/IP with 40 bytes of TCP and IP headers - add to the load on the network. And almost inevitably those applications using the encapsulated protocol receive inferior performance compared with those using the native protocol.

Another possible solution is to define a new programming interface, such as the X/Open Transport Interface (XTI), which can interface to any network transport protocol. This, however, does not take account of the wealth of existing applications, which would have to be rewritten for the new interface. Nor is there any guarantee that it would be widely adopted, given the popularity of the existing interfaces. Furthermore, in order to give true interoperability on any network this new interface could only support the lowest common subset of functions available on those networks, limiting the ability of the system designer to use the most effective techniques for the application.

IBM's solution, adopted as an open industry standard architecture by X/Open, is Multiprotocol Transport Networking (MPTN), implemented hitherto in the award-winning AnyNet products. MPTN adopts a more fundamental approach to protocol independence than previous solutions. It uses protocol conversion, whenever possible, rather than encapsulation. MPTN works at the API layer, converting, for example, the sockets interface to use SNA protocols instead of TCP/IP, or the APPC/CPI-C interface to use TCP/IP protocols instead of SNA. The application calls its preferred API, unaware of the true nature of the underlying network; MPTN then converts those calls to use the protocol of the installed transport network.
adopted, given the popularity of the existing interfaces. Furthermore, in order to give true interoperability on any network this new interface could only support the lowest common subset of functions available on those networks, limiting the ability of the system designer to use the most effective techniques for the application.

IBM's solution, adopted as an open industry standard architecture by X/Open, is Multiprotocol Transport Networking (MPTN), implemented hitherto in the award-winning AnyNet products. MPTN adopts a more

fundamental approach to protocol independence than previous solutions. It uses protocol conversion, whenever possible, rather than encapsulation. MPTN works at the API layer, converting, for example, the sockets interface to use SNA protocols instead of TCP/IP, or the APPC/CPI-C interface to use TCP/IP protocols instead of SNA. The application calls its preferred API, unaware of the true nature of the underlying network; MPTN then converts those calls to use the protocol of the installed transport network.

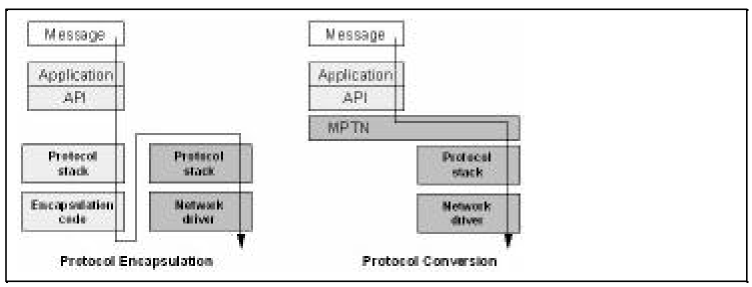


Figure 3. MPTN and Protocol Conversion

Of course, the different network protocols have essential differences which are reflected in their programming interfaces and in the way they are used by applications. SNA, for example, is record-oriented, where TCP/IP is stream-oriented; SNA supports expedited data, TCP/IP does not. MPTN incorporates compensation mechanisms to handle this and to ensure, therefore, that what began as TCP/IP traffic can still end up as TCP/IP traffic, even if it traverses an SNA backbone, but with the minimum of extra processing on the way.

MPTN thus delivers protocol independence without the need to change the applications and with the greatest efficiency of the techniques available today. Whether the transport network

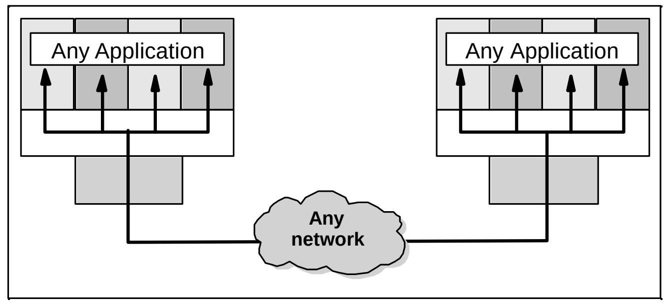


Figure 4. MPTN Access Node

An MPTN gateway concatenates networks running different protocols so that they function like a single network. Thus an SNA client/server application, such as DB2/2, for example, can be installed such that the clients are installed on workstations attached to a TCP/IP network and communicate with the server on an SNA network. Or a TCP/IP client application, such as an Internet web browser, can be installed on an SNA workstation and communicate with its server on an IP network; the IP network could be the Internet.

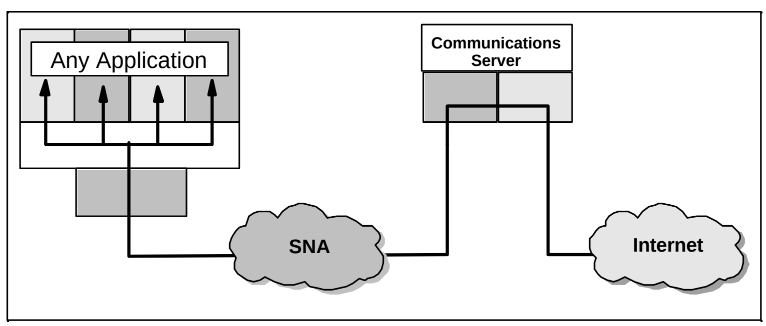


Figure 5. MPTN gateway connecting two unlike networks

A pair of MPTN gateways connects like networks through a backbone that uses a different protocol. Thus a TCP/IP client and server, each on an IP network, can communicate across an

SNA backbone; equally, an SNA client/server application can communicate across a TCP/IP network.

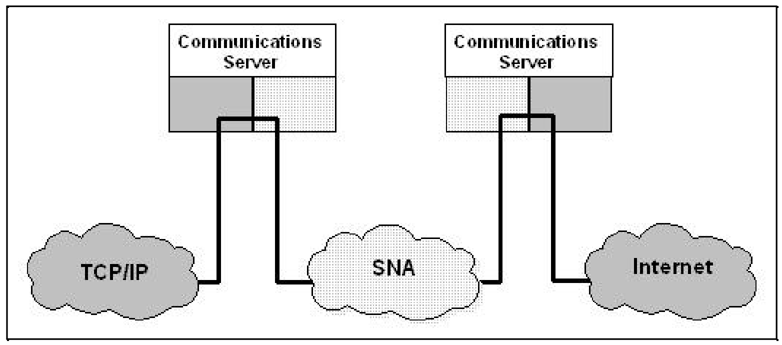


Figure 6. MPTN gateways connecting two like networks with unlike backbone

Delivering advanced networking solutions

For some years, as TCP/IP has spread from its roots in the academic world into the business environment, there has been an occasionally strident debate comparing TCP/IP and SNA. Much has been said and published about both, not always entirely factual, and often colored by the authors' backgrounds and priorities. Each protocol has strengths and both architectures are evolving to support the strengths of the other.

IBM understands both SNA and TCP/IP, and indeed offers the most comprehensive set of world-class products for both protocols. However, IBM's support of TCP/IP and multiprotocol networking, embodied in IBM's Open Blueprint, does not mean that IBM places any less emphasis on SNA as the superior networking protocol for mission-critical business applications. IBM continues to evolve SNA to respond to the rigorous requirements of the professional corporate network. While the growth of TCP/IP networks will, of course, be phenomenal with the pervasive interest in the Internet, SNA will continue to grow at the rhythm of and to suit the needs of the business community. Users will continue to use SNA, even though TCP/IP is in many cases free. Why? Because SNA networks are known to be efficient, stable and predictable.

Communications Server supports both TCP/IP and SNA, and therefore makes it possible for you to ignore the debate and to deploy whichever protocol meets your business needs. However, it delivers its most advanced networking capabilities over SNA, and especially over the latest evolution of SNA - APPN and High Performance Routing (HPR).

In the rest of this section we will briefly compare SNA and TCP/IP, introduce APPN/HPR for those who may be familiar with traditional subarea SNA but not with this relatively new technology, and describe how Communications Server fully exploits the features of SNA to deliver advanced networking for business today and into the foreseeable future.

SNA - connection-oriented business communications

SNA was designed to meet the needs of business computing, delivering reliable and high-performance data transport, and enabling confidentiality of data throughout the network.
SNA is based on a connection-oriented transport. Sender and receiver are connected in the same way as a telephone call: there is a defined network path for transmitting data, a virtual circuit which is secured for the duration of the call. This brings a number of significant benefits:

- Availability of network resources is known at the setup of the virtual circuit. This makes it possible to reserve bandwidth on a per session basis and thus to ensure a predictable transit time through the network - the basis for predictable response times for which SNA is noted.

- The path through the network can be determined based on the required class of service, making it possible to select the optimal routes for individual sessions and to route traffic according to desired priority or specific requirements such as the security of the links. Path information also facilitates tuning, problem determination, network control and network management.
- Knowledge of the route enables congestion-avoidance mechanisms. It makes it possible to report the availability of network resources to the traffic-origination points at the boundary of the network, thus making it possible to decrease or increase the flow of data as it enters the network, and making it unnecessary for nodes simply to discard data as the network becomes more busy.

The downside of connection-oriented transports, however, is that they cannot easily adapt to changing traffic patterns or outages without resetting routes, with consequent loss of sessions.

TCP/IP - connectionless flexible connectivity

TCP/IP was designed to be an inexpensive and simple way of linking dissimilar computers for use by government and academic institutions. TCP/IP is based on connectionless transport, similar to the way mail is delivered, in which individual packets can travel over different paths. One significant benefit of this, of course, is that it makes it very simple to automatically and transparently route data traffic around network failures. However, since packets can travel over different distances or congested paths or even get discarded by congested nodes, arrival time is unpredictable. A connectionless transport cannot guarantee response times, and the absence of predetermined route information makes it much more difficult to deliver effective tuning, congestion avoidance and network management.

APPN/HPR - the best of both worlds

SNA and TCP/IP each have their strengths, and the latest evolution of SNA, HPR, an extension of APPN and known at one time as APPN+, has borrowed from TCP/IP and Frame Relay, combining their best features with the traditional strengths of SNA and APPN to deliver the best of both worlds. HPR has been described as "one of the most significant and dramatic metamorphoses undergone by SNA ... HPR has been designed from the ground up not only to be appreciably faster in terms of data routing than SNA, APPN, or TCP/IP, but also to have the 'weaving and darting' dynamic networking characteristics hitherto associated with TCP/IP". [1]

APPN was introduced by IBM in 1987 as a powerful, flexible and easy-to-use networking solution for SNA client/server applications. It provides peer networking between independent nodes, without the need for the traditional SNA reliance on a central host and without the need for static resource definitions.

APPN defines two types of node, end node and network node, classed as SNA type 2.1 nodes within the SNA architecture.

A network node acts as a server to end nodes, providing networking services such as directory and routing services. Directory services locates partners for the end node and frees the end node from requiring any definitions of where its partners are located. Route selection services finds the optimal path for the session through the network, based on user-specified class of service and transmission priority. A

[1] Anura Guruge, “APPN HPR: the future SNA?”, Xephon, June 1994

network node is also a router, acting as an intermediate node and forwarding traffic to other nodes.

End nodes register their network resources with their server, making it unnecessary to predefine the network. An APPN network is thus self-defining, and can expand or contract or change its shape without the need for constant changes to network resource definitions, nor for the network to be stopped and restarted with consequent disruption to its users.

APPN is an open standard. The end node architecture was published in 1991, the network node architecture in 1993. Also in 1993, IBM established the APPN Implementers' Workshop (AIW), to foster high-quality APPN implementations from a variety of vendors. Some of the more recent APPN developments, including Dependent LU Requester/Server (DLUR/S) - which defines a technique for routing dependent LU traffic such as 3270 data stream natively across an APPN network - and HPR, were developed in consultation with and approved by the AIW. IBM continues to evolve SNA and APPN according to requirements identified not just by IBM but also by our customers and by most major networking vendors.

HPR adds the flexibility of TCP/IP and Frame Relay to APPN, at the same time streamlining the protocol to make significant performance improvements and to position APPN/HPR to address the key requirements of high-speed networking.

The two main components of HPR are Automatic Network Routing (ANR) and Rapid Transport Protocol (RTP).

ANR provides a low-overhead, connectionless routing mechanism for forwarding packets through the network along

a predetermined path. Initial session setup follows traditional APPN partner-location and route-calculation mechanisms. However, instead of the intermediate nodes
having to establish and maintain sessions with each other, incurring processing and memory overhead, the data packets are prefixed with a short header identifying to each node the link on which it should forward the packet. All the ANR node has to do is to strip this identifier off the packet and transfer the packet to the appropriate link: very little processing and no need for pre-allocated memory.

RTP operates at the endpoints of the HPR network. It is responsible for establishing a connection, often called a "transport pipe", with an RTP node at the other end of the network. This pipe traverses the ANR nodes described above. Having established the connection, RTP nodes are then responsible for all

- error detection and recovery - with emerging high-speed links with low bit-error rates it is counterproductive to perform error control at each intermediate node - including selective retransmission of only missing or corrupted packets, and not all packets after the last one successfully received
- end-to-end flow and congestion control, using an adaptive rate-based (ARB) technique that constantly monitors and adjusts the rate at which data flows through the RTP connection
- non-disruptive path and session switching, in which a new RTP connection is established if the original one fails for any reason, without the loss of the sessions using the connection; any data that may be lost in the network is automatically recovered using the RTP error recovery mechanisms.

HPR thus improves network availability through non-disruptive session (re-)routing, and improves network performance through reduced error checking, selective retransmission, its adaptive rate-based flow control, and reduced processing cycles and storage required in intermediate nodes. All of these position HPR as the ideal protocol for any high-speed network.

Exploiting SNA to deliver advanced networking
The IBM development laboratory for Communications Server works closely with that for host VTAM to take advantage of the latest host networking enhancements. Many of the benefits of SNA are realized from this synergy between workstation and host. Communications Server fully exploits these to deliver the most advanced networking available on a workstation platform.

Among the most important SNA features supported by Communications Server are:

APPN and **HPR**. The features and benefits of these have been described above. All three Communications Servers function as full APPN end nodes or network nodes and deliver full ANR and RTP HPR support.

Enterprise Extender. With Enterprise Extender,the Rapid Transport Protocol (RTP)endpoint views its interface with the UDP layer of the stack as just another data link control, and treats the connection across the IP network the same as it would any SNA connection. The IP layer handles packet forwarding for Enterprise Extender, providing the following advantages:

- The use of native IP routing maximizes router efficiency..
- By using EE, SNA applications are positioned to take full advantage of advances in IP routing technology.

- Enablement of a single network transport reduces costs, simplifies network management, and simplifies network architecture..

Branch Extender is a APPN border node subset that is designed to interconnect a remote branch office to an APPN WAN backbone network. Branch Extender optimizes the peer-to-peer communications environment for customers who want to connect LAN-based branches to one large WAN, primarily based on a switched network.

Dependent LU Requester (DLUR) allows dependent LU traffic, including 3270 and LU type 0, to flow over an APPN network. 3270 and dependent SNA applications no longer need to reside on a node adjacent to their host, and can take full advantage of the enhanced performance and non-disruptive session capabilities of High Performance Routing. In addition, DLUR makes it possible to support multiple PUs on a single physical link, thereby removing the limitation of 254 LUs (or dependent LU sessions) per logical host connection.

These multiple PUs can be mapped to PUs downstream of the Communications Server, and, coupled with the gateway's "PU passthrough" capability, this makes the downstream PUs visible to and accessible from the host for network management purposes.

SNA transmission priority has been referred to in the discussion on protocol independence. By using the SNA class of service mechanism, it makes it possible to assign different priorities through SNA nodes, and thus through the network, to different types of application.

SNA security. Communications Server provides architected SNA security for those applications that require it. It should be noted that with some proprietary non-IBM implementations of SNA gateway function this can only be effected between the gateway and the host, leaving the traffic between client and gateway potentially exposed; with the Communications Servers, this security is end-to-end, from client through to the final destination.

SNA data compression is a facility of SNA that allows data to be automatically compressed before it enters the network, using open industry-standard algorithms such as Lempel-Ziv (LZ9) and Run Length Encoding (RLE). Depending on the application this can significantly reduce the amount of data being transmitted, and is especially useful over relatively slow links where the computational cost of compression and decompression are outweighed by the reduced transit time of the data. SNA data compression is supported by Communications Server for Windows NT.

Self-defining dependent LU (SDDLU) - also known as dynamically-defined dependent LU (DDDLU) - is a VTAM enhancement that allows dependent logical units to be known by VTAM when they connect to the host, rather than having to be predefined. This means that new LUs can be added without interruption to other users in the network. It also helps to free up memory in the front-end communications processor, which now only has to allocate memory for Lus actually being used rather than for all LUs previously defined in the network.

Extending advanced networking to all applications
All of these make SNA the present and future advanced networking platform that almost all companies in the Fortune

500 rely on for their essential business communications needs.

With the multiprotocol support described earlier, Communications Server makes it possible to extend many of these benefits, especially high-performance routing and traffic prioritization, to TCP/IP applications as well as to SNA.

Delivering breadth of function and connectivity

In addition to multiprotocol support and advanced SNA networking, Communications Server also provides comprehensive programming support, an extensive range of connectivity options, and support for almost any client on any platform.

Programming support
Communications Server supports the major distributed and client/server Application Programming Interfaces (APIs) in current use. It supports applications written to the sockets interface. In addition, it provides support and the libraries and headers for a comprehensive range of SNA-based client/server and general network programming requirements.

These include:
- Common Programming Interface for Communications (CPI-C) and Advanced Program-to-Program Communications (APPC), both of which enable client/server programming across SNA networks; CPI-C provides a set of high-level calls that are portable across multiple platforms, APPC provides lower level calls optimized for specific platforms
- Programming interfaces to secondary dependent SNA LUs, which make it possible to build a wide range of

applications that use traditional SNA protocols, such as banking and finance terminal emulation, and including 3270 terminal and printer emulation

  - Communications Server for Windows NT and Linux support the conventional LU Application Interface (LUA) RUI and SLI interfaces and Communications Server for AIX supports the LUA-RUI interface.

- Systems Management interfaces that provide support for applications to monitor and control the network and attached workstations and servers
- Programming interfaces and calls to start and stop communications services and to make use of other functions provided by the server, such as configuration and data conversion services

The widest range of connectivity in the industry

Communications Server uses an open link architecture: all types of link connectivity can be used regardless of the customer's network architecture. Thus LAN and WAN links can be used upstream or downstream of the gateway and Communications Server can be used equally as a network concentrator at the host or as a gateway at a remote site. You could, for example, install Communications Server for Windows NT on a LAN attached to the central computer, or Communications Server for AIX on a host channel, and connect downstream systems to them over an X.25 network. With multidrop SDLC links downstream, Communications Server can serve as a very effective single branch concentrator for many SNA SDLC devices, such as the 4702, 3174s, and SNA Automated Teller Machines (ATMs). Communications Server is the only complete product available today that provides such extensive connectivity for the workstation platform.

Communications Server for Windows NT and AIX connectivity options include

- LAN: Token-Ring, Ethernet and FDDI
- ATM (LAN emulation)
- SDLC: switched or leased, point-to-point or multidrop
- X.25, X.32
- Frame Relay
- Direct attachment to host, via ESCON channel

Communications Server for Windows NT also supports:

- Asynchronous
- AutoSync
- Twinax
- ISDN
- Intelligent adapters from OEMs (such as SDLC, X.25, Frame Relay and ISDN)

Communications Server for Linux is currently limited to:

- LAN: Token-Ring and Ethernet

Powerful SNA gateway with comprehensive client support
Communications Server delivers industrial-strength SNA gateway function, supporting both SNA and TCP/IP-based 3270 and 5250 emulators, as well as SNA and TCP/IP client/server applications. Communications Server uses standard open interfaces and protocols between client and server, unlike some communications servers or gateways on the market today. It does not rely on proprietary interfaces and protocols, so it does not lock the customer into a restricted set of products and platforms, and can be used by any client on any platform that supports SNA or TCP/IP.

Supported clients include:

- SNA-based 3270 or 5250 emulators, such as the IBM Personal Communications family of products which run on OS/2, Windows 3.1, Windows 95 and Windows NT, as well as virtually any emulator product from other vendors
- TCP/IP-based 3270 emulators, conforming to the TN3270 and TN3270E standards, also available in the IBM Personal Communications products and from many other vendors
- SNA gateways, including any workstation gateway that uses SNA for its upstream connection, and including the IBM 3174 range and other SNA terminal controllers
- Any SNA client/server application, together with application servers such as the IBM Transaction and Database Servers
- TCP/IP client/server applications, including Telnet, File Transfer Program (FTP), Network File System (NFS), and many others

And the number of clients and applications that can be supported concurrently is equally impressive. Communications Server has proven itself in critical, bet-your-business environments in terms of reliability, performance and capacity. Communications Server is highly scalable, supporting from small branch environments to very large enterprise environments of tens of thousands of users.

Integrating Server Administration using the Web

IBM Communications Servers can now be administered over an intranet or the Internet. Either from a remote or local workstation, the administrator can manage Communications Server through a Web Browser. The Web Administration feature provides the Web administration function for

Communications Server. You can use this feature to manage Communications Server over an intranet or the Internet. Using a Web browser, an administrator can query node status, obtain information about resources, modify resources, display configuration files, display message logs, and perform other administrative tasks. This facility is not available wiht Communications Server for Linux.

The Communications Server Family

The IBM Communications Servers currently include solutions for AIX, Windows NT, Windows 2000, and Linux, and are fully interoperable with OS/390, OS/400 and other SNA and TCP/IP platforms.

Communications Server for Windows NT and Windows 2000

IBM Communications Server for Windows NT and Windows 2000 takes advantage of IBM's experience with SNA and communications servers, and provides a high performance, high quality communications solution for the Windows NT environment. It contains similar capabilities as our other Communications Servers which have been described earlier. The key features include:

- integration of AnyNet SNA over TCP/IP access node and gateway
- integration of AnyNet Sockets over SNA access node and gateway
- TN3270E Server
- significant advanced SNA functions, including:
  - APPN network node and end node support
  - support for HPR as both an endpoint (with RTP) and an intermediate node (with ANR)
  - discovery of service providers

- Dependent LU Requester (DLUR), enabling dependent LUs (including 3270 and 3270-based applications) to operate within an APPN network
- Powerful SNA gateway
- SNA API client services for TCP/IP and IPX/SPX
- SNA data compression
- Designed for Microsoft BackOffice**
- local and remote configuration and administration support
- Web-based server administration via easy-to-use graphical interface
- 32-bit application programming interfaces (APIs) including CPI-C, APPC, and LUA (RUI and SLI)
- data security

- local and wide area connectivity support

Communications Server for AIX

IBM Communications Server for AIX, Version 6.1, is the latest product in the evolution of AIX SNA products, including AIX SNA Services/6000, AIX SNA Server/6000, SNA Server for AIX, and Communications Server for AIX, Key features include:

- An integrated TN3270E Server function supports both TN3270E and TN3270 protocols, providing access to 3270 sessions for clients using the telnet protocol over TCP/IP.
- Multipath channel (MPC) support is provided for the ESCON adapter.
- Supports frame relay over a token ring or Ethernet interface with the TPS**/SoftFRAD product.
- The high-performance routing (HPR) feature of Advanced Peer-to-Peer Networking (APPN) now includes rapid transport protocol (RTP) in addition to automatic network routing (ANR). This feature provides end-to-end support for nondisruptive

rerouting of session traffic around route failures or congestion. This function is supported over all link types.

- DLUR support has been extended to sessions that use generic SNA, secondary LU 0, and LUA API.
- DDDLU (Dynamic Definition of Dependent LUs) enables CS/AIX to dynamically define LUs to VTAM systems, reducing the need for host configuration. This function is sometimes known as SDDLU(Self-Defining Dependent LUs),
- Motif Administration program provides complete configuration and management facilities for CS/AIX in an easy-to-use interface for graphical X-terminals. This program simplifies CS/AIX administration and provides online help for configuration and management tasks. Configuration changes made using the Motif administration program, the command-line administration, and the NOF (Node Operator Facility) API are applied immediately to the node configuration file.

CS/AIX provides enhanced APIs that are more compatible with the APIs provided by members of the Communications Server family running on other operating systems. (Older API support is continued for existing applications; but you should write new applications using the new APIs.) CS/AIX includes the following application programming interfaces (APIs):

- The LUA API enables application programmers to write applications that communicate with host applications at the request unit and response unit (RU) level, and to send and receive data on both the SSCP-LU session and the PLU-SLU session. This API can be used to support LU 0, 1, 2, or 3 communication with the host.

- NOF API can be used to write applications that administer CS/AIX configuration and management resources.
- The Common Programming Interface for Communications (CPI-C) provides CPI-C 2.0+ support and is backwards compatible with existing CPI-C applications written for CS/AIX.
- The advanced program-to-program communication (APPC) API supports LU 6.2 communication using either independent sessions for peer-to-peer communication or dependent sessions for host communication.
- Common Service Verb (CSV) API provides utility verbs that enable an application program to perform functions such as character set conversion and trace file control.
- The MS (Management Services) API supports network messaging functions.

In addition to LAN and SDLC connections, Communications Server for AIX also supports direct channel attachment from a mainframe to a pSeries or SP2 system, over an ESCON channel, allowing the pSeries or SP2 to be used as a high-capacity network controller.

Communications Server for Linux PRPQ
IBM Communications Server for Linux PRPQ is built on the same code base as Communications Server for AIX and is designed to run on the Redhat 7.2 for Intel 32-bit Linux distribution. It contains most of the key features provided by Communications Server for AIX with the exception of the AnyNet functions, web-based administration, SNMP support, Service Level Protocol (SLP) load balancing.

The key features of Communications Server for Linux PRPQ include:

- A full-function SNA gateway

- TN3270E Server
  - SSL data encryption
  - Telnet Redirector
- Advanced Peer-to-Peer Networking (APPN),including both end node and network node
- Dependent LU Requester (DLUR)
- High Performance Routing (HPR)
- Enterprise Extender (HPR over TCP/IP)
- Branch Extender
- Support for Ethernet and Token-Ring connections
- A rich set of application programming interfaces (APIs)
  - LUA, including both RUI and SLI interfaces(LU0,LU1,LU2,LU3)
  - APPC, Common Service Verbs(CSV), Network Operations Facility(NOF), and Management Services(MS)
  - CPI-C and Java for CPI-C (J-CPIC)
- English language only

Summary

The IBM Communications Server provides the powerful and comprehensive networking support required to support the traditional line-of-business applications of today through support of both SNA and TCP/IP networks. It uses tried and tested components to provide reliable yet advanced function. It supports the most popular communications protocols in use today, while at the same time incorporating the latest in networking architectures and technology to support emerging high-speed networks. It can be used on its own to provide multifunction multiprotocol gateway support and also to work seamlessly to provide the underlying communications support for the whole family of IBM middleware Servers. As the inventor, architect and developer of SNA, IBM is uniquely positioned to provide the most efficient communication with host systems on all platforms; for workstation platforms, the networking software of choice is Communications Server.

Further information

For further information on the Communications Servers and other networking products from IBM, please visit our web sites, at:

http://ibm.com/software/network/commservers

Special Notices

IBM Corporation
Software Group
Route 100
Somers, NY 10589
U.S.A.
Produced in the United States of America
08-01