SNA directions—a 1985 perspective

by R. J. SUNDBROM, J. B. STATON, G. D. SCHULTZ, M. L. HESS,
G. A. DEATON, L. J. COLE, and R. M. AMY

IBM Corporation
Research Triangle Park, North Carolina

ABSTRACT

Since its announcement in 1974, SNA has evolved in terms of its functional content, configurational flexibility, and network management services. This paper briefly traces this progress to the present, and examines the more recent advances in greater detail. We then discuss known requirements for enhanced application and transaction services, for additional provisions for very large networks, for continuing exploitation of small-system and transmission media advances, for inclusion of additional management capabilities, and for further accommodation of network standards, all of which will shape future SNA developments.
INTRODUCTION

IBM's commitment to maintain and extend Systems Network Architecture (SNA) as its blueprint for the design of its communication products and for interconnecting them within networks has been reaffirmed and widely publicized over the years. From time to time, a checkpoint is useful to see where we have been and where we are going with SNA. One such checkpoint, given in 1980, traced the progress of SNA, both as an architecture and as a set of products, over its first six years. Today, a year after IBM has marked the tenth anniversary of SNA's introduction, it seems appropriate again to examine SNA's past and current status, and to consider its future. Because the family of SNA products has mushroomed to such an extent (both inside and outside IBM), a description of implementations would well merit a paper of its own; but we confine ourselves here primarily to the architectural aspects.

In looking at the innovations introduced by SNA, two conceptual notions independent of the functions provided stand out; both were borrowed from other disciplines within computer science, but were freshly applied together to the networking context.

The first of these was the basic notion of the network architecture itself. This notion, originating in computer design, resulted in the superimposition of a logical network on the underlying physical network. Thus, the interface presented to the user by the logical network could hide the actual underlying physical realities and provide additional functions not provided at the rudimentary physical level.

The second notion, originating in operating systems practice, was to structure functional offerings in a layered fashion. Together, these notions resulted in SNA being structured as a set of layered logical networks where the outermost network offers an interface for the end user, and each successive inner network offers distinct functions for the next higher layered network in a well-defined fashion. Changes to one layer need affect a higher layer only to the extent that new function is offered, not because old functions are achieved internally in a different way. The notion of layering continues to be fundamental to the orderly evolution of SNA.

In the next section, we review briefly the highlights of SNA's evolution since its introduction. In the ensuing sections, we discuss the recent advances in more detail, according to function. In each case, we examine trends and speculate on the future based on new requirements on SNA.

EVOLUTION OF SNA—HIGHLIGHTS AND TRENDS

In 1974, SNA began as a simple tree-oriented, single-host network. Today, it has evolved to support multiple, independent, mesh-configured networks separately administered, but interconnected by gateways into composite networks. End users can freely communicate with application programs anywhere in these composite networks without being aware of the network configurations. Some of the major highlights in the progress of SNA are shown in Figure 1.

When SNA was introduced in 1974, it was supported by a single operating system (DOS/VS) running on a single host connected to terminals through a front-end communication controller. At that time, the only SNA terminals available were on the IBM 3600 banking system. Host programs communicated with the 3600 controller.

Within a year, SNA coverage was expanded to include remote communication controllers, the MVS environment, and additional supermarket and retail point-of-sale controllers and their terminals. Still, SNA remained a leased-line system only—a situation that changed in 1976 with the addition of dial capability.
In 1977, with the release of IBM's Advanced Communication Function (ACF), more general networking function became available. SNA allowed multiple host "trees" to be interconnected using single links to connect front-end communication controllers. This capability opened the way for a terminal to access application programs in multiple host processors. Also in 1977, IBM introduced its initial support for the X.25 standard for public packet-switched networks. We discuss this in greater detail in a later section.

With the basic networking in place, it was time for more emphasis to be placed on the incorporation of network management services. This focus, which began in 1979, continues to the present. The need for such services is proportional to the complexity of network configurations and how critical network reliability, availability, and serviceability are to the users of the network. With the increasing reliance by businesses on their network operations and with the widening scope of their applications, network management services have taken on greater importance as the architecture and implementing product set have been extended.

In 1979, IBM also enhanced session capabilities to allow parallel sessions between two application subsystems such as CICS and IMS. Another new session feature was the support of the NBS Data Encryption Standard (DES) for session cryptography. In addition, SNA capability to handle non-SNA terminals was significantly advanced by the inclusion of the NTO software product on NCP in the 3705 communication controller.

In 1980, significant extensions were made to SNA configuration flexibility and to its transport services. Improvements included multi-routing, parallel links between nodes, priority transport, and global congestion control within the network; fully meshed connectivity within the backbone transport network was introduced.

The 1980s have brought several significant advances.

1. Advanced Program-to-Program Communication (APPC), or LU 6.2, and node-type 2.1 introduced new and more general session and peer-oriented capabilities.
2. Document Interchange Architecture (DIA) and Document Content Architecture extended SNA support for office-systems applications.
3. SNA Distribution Services (SNADS) provided a new store-and-forward, or asynchronous, distribution service to SNA that complements the synchronous delivery support of sessions between two end users.
4. SNA Network Interconnection (SNI) and extended network addressing (ENA) enhanced SNA routing and configuration flexibility, particularly for large networks.

These more recent advances will be discussed in more detail in the following sections.

In summary, some of the major evolutionary trends in SNA have been

1. Increasing configuration flexibility, particularly to exploit advances in transmission technology;
2. A burgeoning set of products;
3. Greater attention to network management services;
4. Inclusion of network standards, as these have become available;
5. Widening support for non-SNA devices;
6. Expansion of routing and transport services to keep pace with installation of ever larger networks; and
7. Increasing function available to end users.

The steady development of SNA has been tempered all along by a concern for compatibility of past products with new SNA releases. The care taken is manifested by the continued operation of the original SNA terminals under the most recent releases of SNA. This migration sensitivity is one of the hallmarks of SNA evolution.

In the following sections, we discuss how new requirements will likely bear on the above trends. Some emerging trends will also be discussed; these concern an increased focus on continuous (24-hour) operation with a reduction in system-definition down-time, a greater emphasis on peer communication independent of the traditional backbone network, and exploitation of local-area networking and other state-of-the-art technology.

VTAM-NCP TRANSPORT NETWORK

In our survey of the directions that will shape tomorrow's SNA, we start with the VTAM and NCP transport network. VTAM and NCP are two of the first three SNA products (along with the 3600 banking system); they provide the SNA transport network and many of the control and service functions needed to operate SNA networks.

Recent additions to the VTAM-NCP transport network include SNA Network Interconnection (SNI) and extended network addressing (ENA). SNI was announced in November 1983; it provides for the interconnection of autonomous SNA networks through gateways, which consist of specialized interconnection logic in VTAM and NCP. SNI is appropriate for inter-company communication, for companies experiencing mergers and acquisitions, and for situations where independence of company divisions is needed; it enables two or more networks to merge for the purpose of user communication, but to be independently managed and controlled.

To maintain the integrity of the component networks, constraints are built into the gateway to prevent one network from disrupting an adjacent network or gathering information it shouldn't have access to. Flow control prevents one network from flooding its neighbors with more traffic than they are prepared to handle. Names (e.g., of LUs, discussed later) and routing addresses are also independently assigned. To resolve potential conflicts in name assignment, Network Communications Control Facility (NCCF) provides an optional naming-aliasing facility. The SNI extensions to SNA are described in detail in other works. 4,5

Because each of the independent networks can use its full SNA address space, SNI can also be used to configure networks larger than would otherwise be possible. Each network can allocate a pool of addresses available as local aliases for destinations in other networks; dynamic assignment can result...
in using them as needed, thereby sharing the pool over a large number of destinations in other networks. Thus, SNI provides the network interconnection function, and at the same time, possible relief from the addressing constraints some customers were experiencing. This is why SNI was provided prior to a more direct solution to the addressing problem (which is our next topic).

As SNA networks grew in size, a requirement arose to extend the original 16-bit address space. These 16-bit addresses were partitioned into two pieces, a subarea address that identified the destination subarea node (containing VTAM or NCP), and an element address that was used by the destination subarea node for routing, for example, to the correct VTAM application program or to the intended terminal. In a particular SNA network, the subarea address can be chosen to be any size from one to eight bits; the remainder of the 16 bits is then used for the element address. In theory, over 64,000 destinations could be addressed; in practice, this could not be achieved because the subarea/element split has to be uniform throughout the network, and the optimum split varies from node to node.

In September, 1984, IBM announced extended network addressing (ENA), which provides for 23-bit addresses, thereby allowing over eight million destinations in a single SNA network. The subarea portion is fixed at eight bits (the previous maximum for subarea addresses), and the element portion is fixed at 15 bits (the previous maximum for element addresses). These sizes were chosen because they were already accommodated by the existing routing tables.

We view this extension as an interim step, and recognize the need to provide much larger subarea addresses. This next step will be relatively easy from an addressing perspective because space exists for 48-bit addresses in SNA formats, but it will raise more serious problems with the routing schemes. Under the current SNA routing implementation, generating and storing routing tables becomes increasingly difficult as the network grows in size. This brings us to the next topic, the requirement for dynamic routing in SNA.

Today, if you have a large SNA network and want to add to it, you must first decide where to locate the new addition and what links should connect it to the existing network. Then, usually with the help of an IBM program such as Routing Table Generator (RTG), you design the routes of the enlarged network. Once this work is completed, you load the new routing tables into the subarea nodes of the network through a system definition process. This procedure lends itself to very efficient routing, since all the routes are predefined to the network; but, because of definition time and complexity, it limits the size of the network that can be practically supported.

In addition to supporting very large networks, SNA has a requirement to make all networks easier to install and change. One of our long-term design directions is to reduce, and whenever possible to completely eliminate, system definition.

One potential solution is to have the network update itself through an exchange of node and link characteristics whenever a change occurs in any of these parameters. The network could then use the most current topology information to compute the best route between points at the time the route was requested. This route computation could also consider a class of service requested by the user; this is how the user specifies to the network whether he needs, for example, the least-cost route, the route with the least delay, or the route with the greatest bandwidth.

This dynamic routing capability will have a number of beneficial effects on SNA networks. First, larger networks will be possible, because the difficulties with the current route generation process will be eliminated and intermediate nodes will have to store routing information only for currently active routes, not for all potential routes. Second, networks will be easier to install and change by reducing the workload now experienced to do coordinated system definition. Third, this will be an important step toward continuous operation, because it will no longer be necessary to bring the network down for the purpose of updating routing tables. Further discussions of dynamic routing in SNA networks can be found in other works.57

The need for continuous operation runs deep in SNA, and is worthy of more discussion here. A number of features currently available in SNA can be used to increase the availability of networks and to insulate its users from outages; these include the following:

1. Pause and retry logic in Synchronous Data Link Control (SDLC), which allows SDLC links to remain operational across periods of transient errors on the links.
2. Multi-link transmission groups, which allow bundling a number of SDLC links into a single logical link. The sender schedules data traffic for the first available link in a group and the receiver reorders received messages if necessary to maintain the FIFO property of the logical link. Individual links can be dynamically added or deleted from a transmission group without disrupting the ongoing flow of information; a transmission group fails only when the last operational link in the transmission group fails.
3. Multiple routes. Networks can be configured with multiple pre-defined routes so that if the route serving a session fails, that session can be re-established over an alternate route.
4. Parallel sessions. SNA currently allows multiple simultaneous (parallel) sessions to be established between host application subsystems such as IMS and CICS. These sessions can traverse different routes through the network, and where supported comprise a resource pool; when a transaction program needs to communicate with a partner program at another host, the first available session with the desired class of service can be assigned from that pool. Should one session fail, other sessions in the pool will continue to provide session connectivity with the partner subsystem.
5. Host control-point (SSCP) takeover, which provides protocols in the NCPs for detecting failure of controlling hosts and informing their backups.
6. Distributed processing. By moving application programs and data closer to the user, the user can often continue working uninterrupted by link or node failures in the communication network.
While SNA today provides numerous functions that can be used to configure highly available networks, further requirements exist in this area. For example, while SNA provides multiple routes, should a route fail, the sessions it carries are deactivated prior to possible reactivation over a backup route. A possible extension is to have the network perform this route switch without disrupting the sessions.

Highly available systems typically avoid being sensitive to the failure of a single component. To eliminate this sensitivity in SNA, it would be desirable to attach peripheral nodes to the transport network through multiple links to different NCPs. This capability exists today by making the peripheral node appear to the network as multiple nodes with different sets of destinations; comparable support is also necessary in order to provide optimal routing when peripheral nodes are connected to the network using X.25, local-area networks, or other facilities that provide connectivity between large numbers of nodes. In these situations, data traffic could be routed directly from the peripheral node by the intermediate, high-connectivity communication facility to the NCP closest to the destination.

To eliminate having a single point of failure, provision will also have to be made for backup application subsystems. However, just having a backup application subsystem ready to start taking over the moment the primary application subsystem fails will not always be sufficient. Some critical application programs can support thousands of simultaneous users. It could take a number of minutes for the backup application subsystem to re-establish and resynchronize all the user sessions. For these critical applications, a need exists to pre-establish the backup sessions and have them available for immediate use should the primary application subsystem fail. For IMS applications, this capability is planned for availability in 1986 using the recently announced Extended Recovery Facility (XRF).

**SMALL SYSTEMS**

Thus far, we have been focusing on the VTAM-NCP transport network. We now look at SNA requirements and directions from the perspective of the peripheral nodes of the SNA network. In the past, these peripheral nodes have predominantly been display terminals, printers, and remote job entry stations. With the steadily decreasing cost of mini- and microprocessors and storage, more and more of the peripheral nodes are small systems, such as personal computers, distributed processors, intelligent workstations, and office systems, which because of their more general nature, have greater connectivity requirements than traditional terminal devices.

One such requirement is more flexible session connectivity. Figure 2 shows the sessions connectivity in SNA today. Host application subsystems can have sessions with other host application subsystems and with outboard terminals and small systems. The host-to-host connections can employ parallel sessions, which can be used to increase transaction bandwidth (each session can serve one active transaction), to provide for a distinct class of service selection, and to improve performance and availability by fanning out traffic across different routes between the hosts. Peripheral nodes, on the other hand, are currently limited to a single session per LU (a user port discussed later), and that session must be with a host application subsystem. A requirement exists for small systems to enjoy the session connectivity that hosts enjoy today, namely the ability to use parallel sessions and to have direct session connectivity with any other destination in the network.

These small systems also have requirements for communication outside the VTAM-NCP environment. One simple but important form of communication is direct peer-to-peer: just two nodes and a link between them. While this is the simplest possible configuration, it is increasingly important because of the current trend in the communications environment towards high-connectivity multi-access facilities such as local-area networks, X.25, and ISDN, which are discussed later.

In 1983, a new peripheral node type, 2.1, was introduced into SNA to provide this peer-to-peer form of communication. Implementations of this new protocol are now available on the System/36, System/38, IBM 5520 Administrative System, Displaywriter, and Scanmaster. Direct peer-to-peer communication had been available earlier on IBM SNA products such as 8100/DPPX and previous releases of the 5520. The new node type 2.1 protocols provide the capability to carry LU 6.2 sessions (including parallel sessions), and provide the SNA direction for compatible, small-product, peer-to-peer communication.

In designing networking solutions for small systems, it is important to recognize the differences in operating environments between large and small systems. Procurement and operational decisions for small systems are generally decentralized and dynamic, resulting in frequent change. Yet, technical support from systems programmers and network operators is far more limited. Another difference affecting design decisions is that small systems typically need not support the high-traffic volumes of large systems; on the other hand, small systems have more stringent entry-cost requirements.
We are currently investigating architectural solutions to meet the special needs of small-systems networking; further discussion on this topic can be found in the work of Baratz, et al. Among the approaches being explored is a non-hierarchical, peer-to-peer scheme that uses dynamic routing protocols similar to those discussed earlier for the large-systems environment. This would support frequent network changes without the need for coordinated network definition of routes. Dynamic, distributed directories are also envisioned to avoid the need for coordinated network definition of the locations of network resources. This topic is discussed in more detail later.

The evolution of small-systems networking can hardly end with stand-alone networks; we anticipate a requirement to connect these networks to the high-capacity transport networks as shown in Figure 3. These connections should allow sessions between small-system A, for example, and an application in one of the S/370 hosts; they should also allow small systems to communicate peer-to-peer across the SNA backbone networks (for example, from A to F) and share the high-capacity links that are often in place.

Provisions are also needed for managing networks of small systems. These network management techniques should be consistent with the network management functions that are in place for SNA backbone networks so that when networks of small systems are connected to networks of large systems, the entire consolidated network can be centrally managed.

**LOGICAL UNITS**

Logical unit (LU) is one of the more abstract terms used in SNA, but it has a straightforward role, namely as the intermediary between the transport network and the people, devices, and application programs using or attached to the network. Figure 4 shows the position of the LU in the layered structure of SNA.

To this point, we have discussed functions mainly in the transport network, which provides global protocols and services such as network-wide flow control and routing. By contrast, the LU is concerned primarily with session protocols for paired end users. LUs serve as the attachment points for the ultimate destinations of data (application programs, databases, and devices) and provide the end-to-end session protocols in support of communication between these network resources. While the transport network provides global flow control so that links and intermediate nodes in the network are not overloaded, the LU provides end-to-end flow control so that, for example, an application program does not send data to a printer faster than the printer can handle it. Other functions include session cryptography, name-to-address translation (using a distributed directory services capability of the network), and blocking and subdividing message units for network efficiency.

A number of LU types are defined in SNA, with the earlier ones (LU types 1, 2, and 3) optimized for asymmetric host-application-to-device communication. With today's trend toward personal computers, office workstations, and other small systems, new communication requirements exist at the LU level as well as at the transport network level. The key requirement at the LU level is for general program-to-program communication. A single set of protocols is needed for communication between all types of network nodes, including host-to-host, host-to-small-system, and small-system-to-small-system. It should provide a range of functions suited to such products as the IBM Personal Computer and the IBM 3820 Page Printer at the low end, and CICS and the System/38 at the high end. The protocols should also provide a new base for device support, allowing for ease in function distribution to the devices.

To meet these requirements, a new LU type, 6.2, was introduced in 1982 as Advanced Program-to-Program Communication. Initial IBM implementations included CICS, System/36, System/38, Displaywriter, Scanmaster, the IBM 5520 Administrative System, the IBM 3820 Page Printer, Print
596 National Computer Conference, 1985

Services Facility, and the IBM Local Network PrintManager on the IBM PC.

To support a broad range of distributed applications across any set of LU 6.2 products, LU 6.2 defines a set of generic commands (verbs) implemented by each LU 6.2 product in its own syntax, but with common semantics that application transaction programs can issue to communicate, independent of the details of the underlying configurations and protocols. The services provided by these verbs are defined in an IBM manual.9

LU 6.2 was designed to include SNA-defined transaction programs such as those for SNADS and DIA (discussed later) to serve device- or product-specific transaction programs such as those used to communicate with the IBM 3820 Page Printer, and to be used by user-written transaction programs. Users can provide their own transaction programs on LU 6.2 implementations such as CICS, System/36, and System/38, that provide for customer programmability. Such implementations are said to have an open application program interface (API); implementations in which the LU 6.2 implementation is limited to serving only prepackaged transaction programs are called closed API products.

To meet the low entry cost and the high function requirements, LU 6.2 has a base set of functions and a limited set of options. Every open-API implementation supports the base and may also support any of the option sets. One option implemented by CICS can synchronize updates to multiple databases so that all updates either succeed or fail together as an atomic unit of work. Closed API products need provide only the functions used by their coupled transaction programs.

LU 6.2 has recently been enhanced with the addition of option sets for transaction program security. The foundation of these protocols is a two-way verification exchange used to check the identity of the session partner. This is accomplished by having each LU generate and transmit a random number, and having the partner return that value encrypted under a shared session password. Later, when a transaction program initiates a conversation with a remote partner using the session, it can include a user ID and another password. Based on the trust established in the earlier verification, these fields need not be encrypted; they are used by the receiver to verify that the conversation initiator has the authority to gain access to the requested transaction program. Transaction programs with very high security requirements can additionally transmit fully-encrypted data throughout their sessions. Further information on LU 6.2 can be found in other works.10-12

In the following sections, we explore the progress in the transaction services layer, the top layer of the LU. The basic application interface of LU 6.2 is facilitating progress in several areas of transaction services.

SNA DISTRIBUTION SERVICES

SNA Distribution Services (SNADS), made available in 1984, consist of a set of IBM-supplied transaction programs that use a network of LU 6.2 sessions to provide a store-and-forward distribution capability in an SNA network. The LU 6.2 base provides a synchronous, connection-oriented, logically point-to-point service for application programs, analogous to that of a telephone service. SNADS builds on this basic service to provide a connectionless, noninteractive distribution service, analogous to that of a mail service.

This service allows users needing to distribute material to submit their request to a distribution service unit (DSU), a SNADS component, to initiate full delivery. It frees the users from further responsibility. A network of DSUs connected by LU 6.2 sessions handles the complete distribution process. SNADS manages the subsequent distribution, including queuing for resources, fanning out the distribution at the appropriate point in the network topology, and notifying users or their agent transaction programs at the destinations of the distribution's arrival.

SNADS is particularly appropriate for batch-oriented, one-way flows found in such applications as document distribution, file transfer, and job networking. By queuing for resources until they are available, SNADS can transport material through a network where facilities are not all simultaneously active. For example, portions of a network may be connected only by switched links that operate at certain times of day. SNADS frees the user of concerns about network resource availability.

A SNADS user is referred to by a two-part name, which is generally independent of the network topology; user location is determined by reference to a directory. SNADS allows this directory to be distributed in such a way that the complete directory database need not be replicated at every DSU. Incomplete directories have entries that point to DSUs where more complete information can be found.

SNADS allows up to 256 distinctions to be specified for any given distribution request. The user's material is transported in such a way that only one copy is sent to any distribution service unit in the path to a final destination. Copies are made only when necessary as paths to the multiple recipients diverge.

SNADS is currently implemented for a number of products that will use it primarily for document distribution; these are DISOS/370, System/36, and 5520 Administrative System. IBM statements of direction exist for System/38, Series 1, and 8100. Because document distribution is the most common application in these first implementations of SNADS, the SNADS formats were designed to be compatible with those in Document Interchange Architecture (DIA), one of IBM's architectures for the office. Further information about SNADS may be found in other works.13,14

ARCHITECTURES FOR THE OFFICE

To handle situations generic to the automated office, IBM has developed two architectures specific to this important application, Document Interchange Architecture,15-17 and Document Content Architecture.17-19

Document Interchange Architecture

Like SNADS, Document Interchange Architecture (DIA), introduced in 1982, is also part of the transaction services
layer of SNA. It provides a set of protocols that define how several common office functions are performed cooperatively by IBM products. These include the filing, searching, and retrieving of documents and memos as part of DIA's document library services. DIA's document distribution services provide for the sending and receiving of documents or memos via SNADS or the basic LU 6.2 sessions, and include listing items pending receipt, canceling or resequencing their delivery at the recipient's request, and allowing access to software mail boxes and files by other authorized users. The formatting and processing of documents are defined in application processing services. The implementation status of DIA is similar to that of SNADS, but also includes IBM PC, Displaywriter, and Scanmaster implementations.

Document Content Architecture

A vital component of the office architectures is the Document Content Architecture, which provides the formats for describing the form and meaning of objects that are managed by DIA. Currently, two forms are implemented.

1. Final-form Text provides primitive format controls within a data stream in a generic fashion to allow presentation fidelity in a device-independent manner. This allows the sender of a document to control the formatting and print integrity of a document at its final destination without having to know the destination's print device characteristics.

2. Revisable-form Text allows interchange of draft documents in a form that is suitable for revision. Text processing indicators are included with the text, and may themselves be revised.

While the two current types of Document Content Architecture are a good beginning, others are needed to describe the mixing of information types within a document. Text, graphics, image, and voice annotation data are all amenable to the same design approach. A Document Content Architecture serving mixed data would allow documents that integrate a variety of data types to be exchanged by future office workstations.

DIRECTORY SERVICES

One of SNA's design goals has been to insulate the user and the user's application program from the characteristics of the communications network, and to allow users and applications to be moved among processors without impacting other users and applications that communicate with them.

To do this, SNA has distinguished between resource names and their addresses. A name is a relatively stable identifier that users and applications can apply to other users or programs. By contrast, an address can vary according to operations decisions made in the network. Users and their application programs access resources by name. The system uses the name as a key to a directory that provides the current address of the requested resource.

In early SNA networks, the directory tables were defined in the system services control point in a System/370 processor. The control point acts as a mediator in LU-LU session initiation, translating names to addresses and checking resource availability. The use of a central directory in the control point simplified the management of the directory for additions, deletions, and changes.

With the introduction of multiple-host networks in 1977, the control points cooperated to provide directory services, with each control point being responsible for the detailed address information on a subset of the pool of LUs in the network; each control point knew all LU names and the associated control point that could resolve a specific name to an address. Successive designs of the control point have reduced the amount of coordinated predefinition of resources by eliminating redundancy among the directories in different control points. In an SNI environment, a control point can perform a trial-and-error search of other control points for LUs not found in its own directory; this further reduces the number of control points that need to be updated when new LUs are brought on-line.

In SNADS, a user directory is referenced to determine the distribution service unit (address) of an intended recipient. SNADS products have implemented their directories, which are manually maintained, as part of the DSU, rather than in the control point.

Two trends stress the current design point of manually maintained directories; as networks become larger, the frequency of directory updating increases and the number of directories that must be consistently maintained grows; furthermore, the trend toward peer connectivity among small systems requires that small systems also maintain directories, whereas they may have formerly depended upon a large host directory. Both trends result in many more directories and increased update activity in a network, and point to a growing need for the directories to be automatically and dynamically maintained.

Resources such as application programs, files, and LUs need to be registered only at their local (home) directories, at a minimum. Automatic network-wide searches of the various directories eventually results in the finding of the resource if an active path to it is available. Replication of directory entries throughout the network can have several benefits:

1. Performance in finding resources can be improved;
2. Switched links can be activated in order to complete required synchronous connections; and
3. Asynchronous distributions can be forwarded as far as possible into the network when an active path to a destination is not available.

To meet the requirement for more dynamically maintained, distributed directories in an SNA network, new protocols will be needed for resource registration and network searching. These protocols will apply between directory users and their directory providers and among the directory providers themselves.
MANAGEMENT SERVICES

The previous sections of this paper discussed the advances in the functional richness and configuration flexibility allowed by SNA. Now SNA users can connect multiple SNA networks, small systems to large systems, and various other devices, all within the same composite network. The actual attachments may be over various transmission media (e.g., SDLC telecommunications links, S/370 channels, and X.25). These enhancements exact a price in that they make the network more complex to manage.

Management services, otherwise known as network management or communications network management, include the monitoring and controlling of the SNA network and its associated resources. IBM is integrating management services into SNA to address the management problems inherent in the ever-increasing complexity of the communication environment.

History

The early releases of SNA provided limited management services in VTAM, supplemented in 1977 by the Network Operator Support Program (NOSP). In 1979, in the next major release after multiple-host support was introduced, the Network Communications Control Facility (NCCF), replacing NOSP, and the Network Problem Determination Application (NPDA) were included. NCCF was created to aid in operator control of the multi-host environment; NPDA, to help manage problem determination for the many different products that could be attached. In 1982, Network Logical Data Manager (NLDM) was introduced to help manage the logical resources (such as sessions and virtual routes) in SNA.

The original functions performed by NCCF, NPDA, and NLDM were largely on a product-by-product basis rather than being general functions defined by SNA. With the continued enrichment of SNA, management services have become a more integral part of that enrichment and go beyond the early product-specific design.

Requirements

The network owner or the provider of network services must be given the tools and resources to provide the reliable, high-performing secure services that users require. The network owner must also be able to manage the network configuration, effect changes, and monitor the use of network resources. The network operator (programmed, or human) must be able to invoke these functions from a central site, or to distribute control of the functions. Of course, all the functions must be performed in a cost-effective manner.

Major Categories of Management Services

The requirements for managing an SNA network fall into four major management services categories.

1. Problem management—the function of managing a problem from its detection through its resolution. The steps of problem management are: (1) problem determination, (2) problem diagnosis, (3) problem bypass and recovery, (4) problem resolution, and (5) problem tracking and control.
2. Performance and accounting management—the process of quantifying, measuring, reporting, and controlling the usage, responsiveness, availability, and cost of a network.
3. Change management—the planning, control, and application of changes (additions, deletions, and modifications) to the resources of a network.
4. Configuration management—the control of information necessary to both logically and physically identify network resources, and to indicate their relationships to one another.

Management Services Components in SNA

The functions provided by NCCF, NPDA, and NLDM are represented in the architecture by a management services component in the control point, physical unit, and individual layers of SNA. This structure provides a framework by which the aforementioned requirements can be satisfied.

Each layer of SNA (see Figure 4) is responsible for controlling the resources associated with that layer. This is accomplished through a component called local management services. For example, routing information that is used for problem management is gathered by the local management services component in path control. Once gathered, this information is sent to the management services component in the PU.

Physical unit management services is responsible for gathering management services information local to its node or attached links, performing some services such as reformatting and time-stamping the data, and sending the information to the control point for processing. Control point management services is responsible for collecting management services data from the network, analyzing the data, and taking the appropriate action based on that analysis.

Current Management Services Functions

Alert

The Alert is a problem management function that is used to report a loss or impending loss of availability of a resource. Once an error is detected that requires corrective action by a network operator (human, or programmed), the Alert is sent to the control point. The Alert contains a general classification of the problem, a description of the cause of the condition, identification of the failing resource, and addi-
tional details pertaining to the problem. The management services component of the control point analyzes the data and reports the condition along with a recommended action to the network operator.

Problem determination statistics

Statistics are gathered by the data link control layer and are sent to the control point when a threshold is reached, or when solicited by the network operator. These statistics can be used as additional detail in problem management. The statistics include counters such as those for total transmit data, transmit errors, total receive data, receive errors, and polling information. For X.25, counters include statistics pertinent to the access links and virtual circuits.

Modem tests can be performed using the link problem determination aid (LPDA). This function allows the status of the modems and remote DTE interface and the results of the modem tests to be determined.

The link resource control function allows the network operator to dynamically change the threshold values of link statistics. It also allows the operator to disable the use of LPDA.

Response-time monitoring

Response-time monitoring (RTM) allows the network operator to validate certain predetermined end-user service levels for specified LU-LU sessions. Response time is measured by the time elapsed between LU recognition of an end-user request and reception by the LU of the resulting reply from the session-connected partner. The predefined values can be changed dynamically by the network operator. The response-time values can be sent unsolicited upon threshold overflow, or can be solicited by the network operator.

The network operator can request a summary of the response-time data for a specific LU over a user-defined period of time, detailed data for a specific session for a single collection period, and the long-term trend for a specific LU.

Managing logical resources

Because the protocols involve distributed participants, protocol violations and logical errors in SNA implementations can be very difficult to diagnose. Some information about the logical resources in an SNA network is captured by the control point and used for problem management.

Data captured by the control point on logical resources include session start information, normal session termination information, abnormal termination information (e.g., sense data), and virtual route information. This information is available for sessions that were started through the assistance of one or multiple control points.

With the advent of SNA network interconnection, it is possible to have a session established by a network wherein the endpoints of the session reside in different networks. In this case, additional session information is required for problem management. Retrieval of session information includes the gathering of session control-block information from gateway nodes (NCPs) for cross-network sessions.

Future Considerations

The requirements for SNA management services increase as SNA protocols and telecommunications complexity continue to advance; management services will need to be an integral part of each SNA enhancement. Common solutions are sought whether managing small or large systems interconnected via SDLC, X.25, or local-area networks.

A trend in both IBM SNA products and SNA management services has been to provide more granular monitoring and control of network components. Hence, such products as the 386x series of modems and the 3710 Network Controller offer extensive problem management features integral to their design. In turn, the architecture allows telecommunication links, for example, to be monitored and controlled at the level of their most basic subsystems: component adapters, modems, line concentrators, and transmission media. This has vastly improved problem management in SNA networks.

In the future, similar attention to management services features will be vital, particularly in the local-area networking arena. Management services enhancements for managing the logical resources would extend the ability to diagnose system definition and implementation incompatibilities. The capability to trace routes as they are established dynamically, and to track users as they switch between sessions would also add to the serviceability of SNA networks.

LOCAL-AREA NETWORKS

Even as architectural solutions for improved connectivity continue at the higher layers of SNA, the announcement of the IBM Cabling System in 1984 promises major improvements in dynamic connections at the physical and data link control layers.

For the immediate future, the IBM Cabling System allows an SNA network to be physically reconfigured without running new coaxial cable to specific locations in a building. By using a structured-wire approach and wiring closets to pre-wire a building with either twisted-pair copper conductors or optical fibers, today's workstations can be moved from office to office by simply plugging them into a wall and reconfiguring at a conveniently located wiring closet.

Although ease of reconfiguration is a desirable goal, the ultimate objective is to eliminate entirely the need for manual intervention by a systems professional when moving a workstation from one office to another. This objective could be accomplished by the incorporation of a token-ring local-area network (LAN) on the IBM Cabling System.

A token-ring LAN would consist of the wiring system, a set of communication adapters (stations), and an access protocol that controls the sharing of the physical medium by the stations attached to the LAN. The token-ring LAN is one of several LAN standards currently being developed by the IEEE 802 committee for submission to the International Standards Organization (ISO). (Other standards include one for...
CSMA/CD on baseband cable and a token-bus standard on broadband cable.) A token-ring LAN is unique among these in that the nodes are physically connected serially by a transmission medium, such as twisted pairs or optical fiber. Access to the transmission medium is controlled through the use of a unique bit sequence (token) that is passed from one station to the next. When a station has a frame to transmit, it modifies the token to a frame by changing the bit pattern of the token to a start-of-frame sequence; the frame is then transmitted. When the station has completed frame transmission, and after appropriate checking for proper operation, it initiates a new token so that other stations have an opportunity to gain access to the ring.22

An important part of the token protocol is the ability of a station to reserve the token for use at a specified priority. This ensures that the next token issued will be at the highest priority requested, and allows a station to gain faster access to the ring for frame transmission than would otherwise have been possible.

To ensure that a token is always available on the ring, one station is elected as the token monitor. The function of the token monitor is to detect error conditions in token operation, such as a continuously circulating frame or the absence of a token on the ring. The capability to be a token monitor resides in each station, and is determined by an election process when normal token operation is disrupted.23

Several advantages exist in choosing a token-ring configuration for a LAN; these include the ease of fault isolation, performance stability under load, the use of predominantly digital rather than analog engineering, and the promise of optical fiber technology.24

To take full advantage of the peer-to-peer connection capabilities inherent in a shared physical medium, a station on the token ring could use the data link control, called a logical link control (LLC), as defined by the IEEE 802.2 committee for LANs. This LLC employs the Asynchronous Balanced Mode of operation (like that in HDLC) when a link connection is established, thereby allowing either station to send data link commands at any time, and to initiate responses independently of the other link station. This provides for a balanced type of data transfer between two link stations that operate as equals on a logical point-to-point link;25 the number of logical links sharing the same ring equals the number of distinct pairs of communicating stations.

When the ring reaches its capacity either physically, in terms of the number of stations it is capable of supporting for the required distance, or when the bandwidth is exhausted and the performance is not acceptable, a bridge can be added to combine two token rings into one logical ring. A bridge is a device that copies a frame from one ring and transmits it on the other. Bridges can be used to combine a number of small rings to preserve the integrated connectivity in an establishment while providing better fault independence and performance. Locating stations on a ring, or on multiple rings connected by bridges can be performed dynamically by broadcasting requests for specific station addresses. Once the station is located, routing data through bridges can be done efficiently by including the routing information to the destination station with each frame; this allows bridges to copy frames from one ring to another based on routing information in the frame format without building, referencing, and maintaining complex tables. Thus, expansion of the LAN to include additional rings need affect only the connecting bridge, and can be transparent to all other stations.

Other LANs could be connected to the token ring as well. Those that comply with the IEEE 802 standards could be connected using bridges similar to those used to connect two token rings. Thus, an SNA station implemented on a token bus could communicate to an SNA station on a token ring as though both were attached to the same LAN.

One desirable goal is the connection of LANs to the SNA backbone network. This could be done by several methods. For example, in an SNA network, any SNA node containing intermediate (forwarding) path control function could be attached directly to the LAN, thereby allowing SNA stations to gain access to the entire SNA network. Special gateways are not required because the LAN is providing the functions of the physical and data link control layers of SNA, and not higher layers such as path control. Connection of the intermediate-routing SNA node to the remainder of the backbone network could be accomplished either by channel attachment or via a remote communication link using SDLC.

The effect of the token-ring LAN on SNA would not be disruptive. To attach an SNA node to the token ring, very few changes would be required. Because SNA is a layered architecture, supporting a new physical and data link control layer has little effect on the rest of the architecture. Thus, path control, for example, could remain unchanged, thereby allowing SNA nodes attached to the token-ring LAN to participate in an SNA network immediately.

The long-term effects of LAN attachment could be far reaching. Improving dynamic connectivity and reducing system generation requirements in SNA products become even more important when physical connectivity in an establishment creates the possibility of a "hot pluggable," fully-meshed network. That is, once an SNA workstation is plugged into an office wall, it could have immediate physical access to all SNA workstations and other SNA nodes attached to the LAN. To translate this physical access into intelligent communication, a consistent application program interface and a set of protocols allowing peer attachment to work stations and mainframes, are critical. Thus, LU 6.2 and node type 2.1 protocols in SNA will become even more important.

LINK SUBSYSTEMS

SNA started with terrestrial links and channel attachments, and soon added dial capability. Local-area networking was discussed in the previous section. A number of other link-level options have been or could be accommodated by SNA. Some of these such as X.25, satellite technology, and ISDN, are described below.

Recommendation X.25 defines a packet-mode interface for attaching data terminal equipment (DTE) such as host computers, communications controllers, and terminals to packet-switched data networks (PSDNs). The International Tele-
graph and Telephone Consultative Committee (CCITT) introduced X.25 in 1976 and updated it in 1978, 1980, and 1984. IBM products that offer X.25 capability comply with the 1980 version of the interface. A PSDN provides connectivity to other DTEs using X.25 virtual circuits. Permanent virtual circuits provide fixed connectivity between DTEs, whereas switched virtual circuits provide dynamic connectivity using virtual call set-up and clearing capabilities. The X.25 interface also defines user facilities such as interface parameter negotiation, reverse charging, and closed user groups.

Having participated in the development of X.25, IBM announced the capability in 1977 for attachment of several DTE products to PSDNs in Canada and France. One of the early products was a network interface adapter, a stand-alone unit that converts between the link control protocol of SNA nodes (SDLC) and the X.25 protocols. This allowed most IBM products that communicate with System/370 hosts to use X.25. Initially, the communication controller (IBM 3705) for SNA System/370 hosts used a specific software adaptation for X.25. In 1980, an X.25 program product for the communication controller was introduced, allowing packet-switched communication with other SNA products and connections with non-SNA DTEs.

The IBM direction with respect to X.25 has been to integrate the interface into SNA products where required, so that the customer can choose the most economical communication medium. If network tariffs favor X.25, one can choose packet-switched services; otherwise, traditional leased or switched services can be used. By the end of 1984, 14 IBM products had been announced supporting the X.25 interface in more than 25 countries.

All SNA products that offer an X.25 interface conform to an IBM-defined specification for attachment of SNA products to PSDNs. The most recent enhancement to this specification is Enhanced Logical Link Control (ELLC). Sometimes, virtual circuits are interrupted by the PSDN, causing inconvenience to the users of certain products. This inconvenience is reduced with the implementation of ELLC in low-end computers and terminals that provide a dynamic packet error detection and recovery procedure across one or more PSDNs between SNA nodes.

The IBM X.25 interface specification will be updated to include aspects of the CCITT 1984 recommendation to be supported in SNA. Some of the new functions in the 1984 recommendation that are candidates for inclusion in the IBM SNA specification are:

1. Multiple links between the DTE and the PSDN;
2. Forwarding a virtual call by the PSDN from the called DTE to an alternate DTE;
3. Hunt groups that allow the network to assign a call to one of several target DTEs; and
4. A subaddressing capability that allows a DTE on a private packet-switched network to call a DTE on a PSDN.

The SNA nodes discussed above are DTEs that attach to packet-switched networks. Some customers need equipment from different vendors to communicate with each other over packet-switched networks. If the customer has a mix of SNA and non-SNA traffic, the SNA backbone network could be used to carry the non-SNA traffic. Studies are underway to determine the requirement to add a PSDN appearance of the X.25 interface to SNA. In such a configuration, the SNA network would provide X.25 permanent virtual circuit, virtual call, and user facility services to using DTEs.

A key aspect of X.25 is that it is an interface specification, not a network architecture. The internal operation below the X.25 interface, such as routing, flow control, and management services, is not specified by standards. Inclusion of an X.25 DCE capability within SNA is a relatively straightforward and natural step.

Some customers find it advantageous to send their SNA traffic over satellite circuits. Because most communication satellites are in geostationary orbits above the equator at an altitude of about 23,000 miles, the delays in sending information from one earth station to another are long compared to those for a terrestrial circuit that connects the same two points on the earth’s surface. Consequently, communication protocols must be designed to accommodate the long propagation delay of satellite circuits. Detailed studies of SNA show that interactive and batch applications can use satellite links satisfactorily at speeds of up to 19,200 bits per second when the satellite link is attached to an SNA peripheral node. The IBM 3710 Network Controller provides satisfactory SNA performance over satellite links at speeds of up to 64,000 bits per second when the satellite link connects the 3710 to an IBM 3725 Communications Controller (using SDLC with a modulus of 128). Batch and interactive traffic over a satellite link that connects two SNA 3725 communications controllers is carried satisfactorily at link speeds of up to 256,000 bits per second. A special adaptation is available that allows two SNA 3725s to communicate over satellite links at speeds of up to 1,344 million bits per second.

At satellite speeds above 256,000 bits per second, special consideration must be given to the types of protocol and the value range of protocol parameters at several architectural layers of the system. Because high-speed satellite circuits are becoming more widely available, a requirement exists to enhance SNA to accommodate them. Capabilities such as support for larger link-level sequence numbers (an SDLC modulus of 128) have been added; selective retransmission of information and other protocols optimized to the high-speed, long-delay environment are being studied.

The rapidly approaching feasibility of high-speed digital communication (in units of 64,000 bits per second) will have significant impact on both data communication and telephony, and will open possibilities for interactive video applications in the foreseeable future. The ISDN (Integrated Services Digital Network) will provide voice and data services and will open possibilities for interactive video applications in the foreseeable future. The ISDN (Integrated Services Digital Network) standardization of the user-to-network interfaces for these applications has been going on for several years with CCITT.

IBM has consistently represented the needs of data communication applications in this effort. We continue to actively cooperate in the development of a single set of world-wide standards. The CCITT Recommendations of 1984 are a significant step in the advancement of these new digital transmission services.
IBM recognizes the widespread interest on the part of users in interconnecting networks using different communication architectures, IBM favors such interconnection and publishes extensive information about SNA, including formats and protocols, which facilitates interconnection by other systems to IBM SNA networks.

Widespread interest also gave rise to the Open System Interconnection (OSI) standards project aimed at providing communication protocols for interconnecting systems of different communication architectures, such as SNA with systems of other architectures. IBM has been involved from the start in this work. We have contributed what we have learned about layered communication architectures in the past several years, and increased our understanding of advances elsewhere in this area. Some capability for interconnecting heterogeneous systems is clearly desirable. From a vendor's perspective, a single international protocol for interconnecting heterogeneous systems is preferable to a number of national protocols.

IBM has stated that for industrial communications, IBM supports the National Bureau of Standards specifications for OSI Transport Layer Class 4 used over IEEE 802.4 LAN. The capability was demonstrated at the National Computer Conference in July, 1984.

IBM Europe has software under development that will provide IBM System/370 support for selected functions in the OSI 4 (Transport) and 5 (Session) layers. Testing in conjunction with third parties is planned to start during 1985. This represents a further step in IBM's commitment to provide IBM System/370 products capable of system interconnection in conformance with OSI standards.

OSI protocols could be implemented in SNA using a gateway concept; that is, we may transform the SNA protocols to and from the OSI protocols through gateway nodes in order to allow attachment to other networks. Indeed, IBM Japan, in cooperation with Nippon Telephone and Telegraph (NTT), has judged gateways using OSI as intermediate protocols to be a viable way to interconnect SNA networks with networks that use the DCNA protocols. This approach of using gateways, as opposed to adopting OSI protocols as internal operation protocols for SNA networks, reflects the fact that OSI protocols today are intersystem protocols, as opposed to network architecture.

CONCLUSIONS

SNA, now eleven years old, has evolved continually and will do so as long as new technology, applications, and requirements unfold. The layered structure of the architecture and of the implementing products allows this process to be natural and nondisruptive.

We have cited some of the historical trends. For example, configurational flexibility and accommodation of larger networks have been ongoing concerns. In addition to the generalized topology and larger address space now available, specific offerings such as the recent IBM 3710 Network Controller, which offers a new remote link concentration capability, have resulted in cost-performance advantages.

Additional offerings have resulted in the extension of SNA capabilities into areas that are important to many customers. A recent example is the inclusion of VTAM as an integral component of the native VM environment, thereby enhancing performance of SNA network operation from the VM viewpoint. Another area of traditional concern to customers is non-SNA device support. One technique, which uses format envelopment, was incorporated into the Non-SNA Interconnection (NSI) program product on the NCP; it allows BSC remote job entry terminals and BSC network job entry subsystems to communicate through an SNA network and share the SNA links. Another technique employs protocol conversion; aside from the long-time NTO support for pre-SNA terminals in the NCP, new capabilities such as those of the 3710 will allow concentration of start-stop links onto SDLC links, and continue to enhance SNA coverage in this important area.

The trend in general has been to perform protocol conversion to SNA as close to the non-SNA interface as possible in order to gain the SNA benefits of resource-sharing and network management quickly in the operation.

Of course, one of the most visible areas of non-SNA protocol support is national and international standards. IBM will continue to cooperate in the formulation of such standards, and include support for such standards in SNA products subject to appropriate business decisions.

Network management services have generally been and clearly must continue to be integrated into SNA and its implementing products. This commitment increases as the richness of the architecture grows; to control problems associated with increasing complexity, the pace of extensions in this vital area will likely be stepped up.

Other trends will become more prominent. The need for continuous network operation will foster even more features that promote high availability. The Extended Recovery Facility mentioned earlier in this paper is an archetypal feature in this category. Advances in route dynamics and distributed directories will also play a significant role in meeting requirements in this area. The matter of reducing the static nature of network definition is an ongoing concern; a long-term goal is to eliminate the need for static definition entirely.

Finally, SNA will continue to exploit advances in technology as they come along. A known requirement is to extend the peer-to-peer operation in SNA. This need follows from developments both in processor design, especially in small systems, and in different types of transmission technology, such as local-area networks, satellites, PSDNs, and ISDN. Other developments, not yet evident, will affect future requirements. The process will continue and will undoubtedly cause much interesting evolution of SNA for a long time to come.
REFERENCES

Note: All IBM publications, unless otherwise noted, are available through IBM branch offices.


