Recent technological advances have allowed information processing and data storage capability to be distributed more easily from a central computer complex to remote user locations. Systems Network Architecture provides a unified systems structure for the contemporary teleprocessing environment that resulted from these advances. Using some current implementations as examples, this overview introduces the concepts on which the architecture is based and broadly describes the basic components of the structure. Specific architectural and implementation details can be found in the references.

Systems Network Architecture: An overview

by J. H. McFadyen

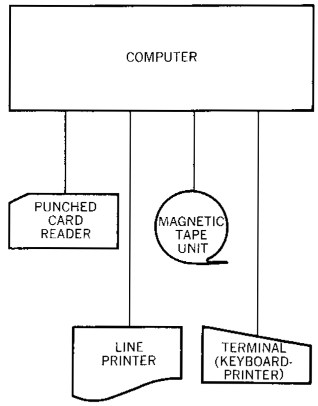
The first computers were developed for specific scientific data processing applications. Data punched into cards was read into the computer for processing, and the results of the calculations were printed at the conclusion of the "job". When these early computers evolved to provide more comprehensive information-processing services, the traditional techniques of interacting with the computer were inappropriate for certain classes of users. For example, business activities such as credit verification require quick response for customer acceptance; card punching and the bulk printing of reports introduce unacceptable delays. Many commercial environments and some scientific applications require computing facilities with which the user can interact directly.

As a result of such requirements, a typewriter-like device was designed for attachment to the computer in the same way as other devices such as the punched card readers, magnetic tape units, and line printers shown in Figure 1. Combining a keyboard and a character printer, this "terminal" allowed the user to request information-processing services without the intermediate step of punching cards. Errors in input data were quickly made known to the user who could immediately correct the faulty data. As soon as the processing was complete, the response to the user's request could be read directly from the "interactive" terminal.

remote users

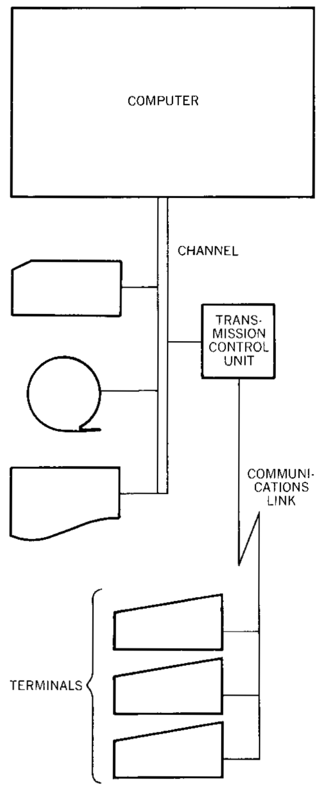
Users whose work locations were distant from the computer continued to transport punched cards and computer printout between their work site and the computer room. Interactive services did not become available to remote users until telephone lines were used to connect the computer and the terminal. A transmission control unit controlled the communications line using a protocol that intermixed line control and device control. Programming in the computer managed the control unit, formatted the data, and controlled the operation of the terminal. For example, with Basic Telecommunications Access Method support for a Start/Stop device, the application program had to provide Start/Stop line control characters and polling sequences. Application development was inhibited by the device-dependent support that had to be provided by each application program for each terminal type. Although teleprocessing (remote information processing) provided many user benefits, system modifications were both costly and time-consuming because of their effect on application programs.

Figure 1 Direct attachment of I/O devices



Such terminals were the forerunners of the extensive, complex networks of terminals in use with computers today. In this paper, a method to unify network operations, known as Systems Network Architecture (SNA), is discussed. In particular, the concepts of this structure are introduced. First, the events leading up to the development of SNA are briefly described. Then network characteristics and their relation to the SNA structure are discussed. The network nodes of SNA, their structure, and their operation are described later, and finally a brief description of the subsystems in an SNA network is given. Detailed descriptions of SNA and the Advanced Function for Communications are available elsewhere.[1-3]

Figure 2 Remote attachment of terminals



Historical perspective

As computer development progressed, the increasing numbers of remote terminals attached to expanding networks required additional links, while, unfortunately, the typical interactive terminal could not fully utilize the data rates of the available links. Multipoint configurations reduced the link costs since, as shown in Figure 2, several compatible terminals could share the same link just as locally attached devices shared a channel. Each application program had to manage the multiple terminals on the multipoint line. Terminals using incompatible protocols continued to require both unique programming support and separate communications lines, consuming valuable programming and system resources. The responsiveness of the system suffered because the information-processing capacity of the computer was diverted from application functions to network management. Large networks required new types of transmission control units designed to remove some of the repetitive, low-level network functions from the computer. These new control units allowed more computing capability to be devoted to application processing.

In order to support noninteractive applications, new terminals for job-oriented applications were developed to duplicate the local attachment environment for remotely located users. Each unique terminal placed an additional burden on computer performance, configuration flexibility, and programming support. Because application processing and network processing were combined in each application program, few teleprocessing networks could be cost-justified. Networks were limited to critical applications or to those providing a significant financial return on investment.

programmable control units

Recent technological advances, such as the development of microprocessors, have encouraged further development of control units, especially programmable communications control units and programmable device control units. Functions already provided by a nonprogrammable transmission control unit can be performed by the communications control unit with its network control program. The control program can reduce the communications management load on the computer by controlling network data traffic, formerly the responsibility of programs in the computer.

Programmable device control units can relieve the application program in the computer from having to control device operation. The application program communicates with a program in the device control unit tailored to the specific requirements of the terminal users. These control units may also provide a computing facility for processing simple user transactions without interrupting the computer. This relocation of application processing can improve system responsiveness; also, the accessibility of information-processing services is less dependent on the reliability of the communications link. The control unit uses the link to gain access to complex services that are better performed by the more versatile computer. As an example, a control unit of the IBM Retail Store System relies on a System/370 computer for inventory management.

function distribution

The availability of programmable control units increases the possibilities for "function distribution," the technique of partitioning network functions among several nodes of the network. Communications control units support the distribution of network-processing function; programmable device control units are designed primarily to support the distribution of application-processing function.

Application program support for a multiplicity of devices and application involvement in communication details have made teleprocessing too complex and expensive for many organizations. The feasibility of distributed function further complicates network design when pre-SNA teleprocessing methods are employed. SNA was developed to provide a unified structure for the contemporary teleprocessing environment. SNA describes the division of network functions into discrete layers and defines protocols and formats for communication between equivalent layers. The products developed for use in SNA networks are combinations of hardware and programming whose design is based on the architecture.

An SNA network can accommodate a diverse variety of current and future configurations and devices. By clearly defining the functional responsibility of each network component and the protocols for communication between them, SNA provides a coherent network structure that can satisfy advanced user requirements.

Network characteristics

A network is a collection of computer, control unit, and device nodes that are connected by data links such as channels, satellite and microwave links, and switched or nonswitched communications lines.

System/360 network

Figure 3 illustrates the manner in which a network of terminals is supported by a System/360 computer. Operation of this network is primarily dependent on programming in the computer. The programs supplying services to the terminal user contain device-specific code. The Information Management System, for example, provides two different types of device control for the IBM 2741 and 2260 terminals. The application programs communicate with the terminals through a telecommunications access method. The choice of an access method depends on a number of factors such as the types of terminals supported, the sophistication of the application programmer, the types of application programs, and the performance requirements. Frequently more than one access method is in use because a single System/360 access method cannot satisfy all of the requirements. Each access method intermixes link control, device control, and user data characters appropriate to the particular link and terminal. In Figure 3, for example, if the 2741 terminals use different character codes (EBCDIC or ASCII), the access method must use different program sequences to manage the two terminals. Resource sharing is limited because of the specialized support required by the various terminals. Although the network satisfies the user's current requirements, it cannot be extended or recon-

Figure 3 System/360 teleprocessing network example

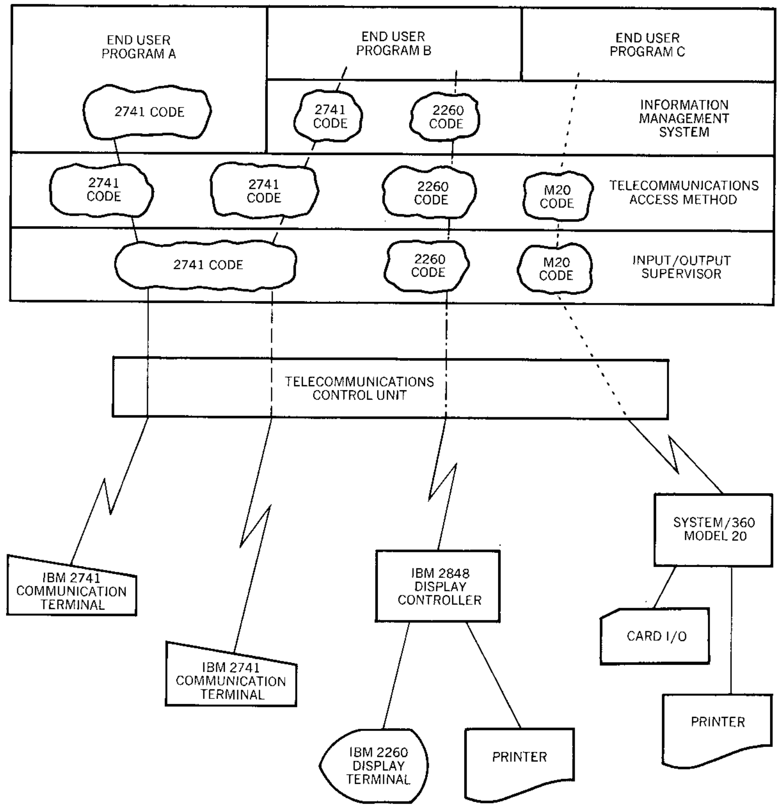
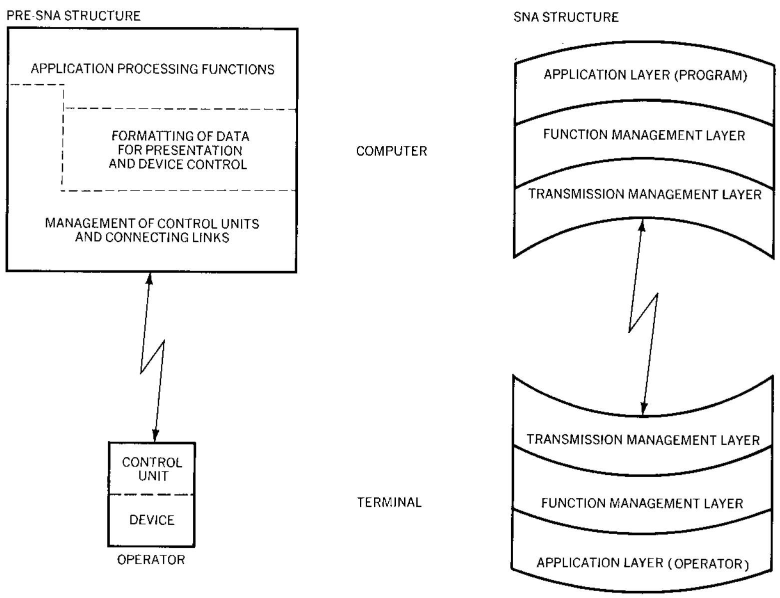


Figure 4 Functional relationships: pre-SNA versus SNA



figured easily. The replacement or enhancement of a network function may affect other system components because System/360 teleprocessing networks did not formally separate functions that are logically independent. When programmable control units are added to these networks, the restructuring of the network to accommodate new devices or applications can be very costly and time-consuming.

functional layers

The basic concepts of SNA were developed to provide a unified communication system architecture that could satisfy both current and future network requirements. SNA increases network flexibility by clearly identifying and separating the functional responsibilities of three major functional layers:

- Transmission management
- Function management
- Application

A key concept of SNA is the structuring of each network node as a set of well-defined layers. SNA is used as a basis for the design of hardware and programming providing Advanced Function for Communications.[3] When a network is assembled from these products, the layer concept produces an essential symmetry throughout the network: Every SNA node has the same structure. Figure 4 contrasts the previous relationship of a computer controlling a terminal to the SNA concept of two network nodes performing complementary functions for the network user. Prior to the development of SNA, network and terminal management responsibilities were meshed with data manipulation and other application-processing functions. In the computer controlling pre-SNA networks, application programs included data-formatting functions, link-control functions, and device-control functions.

The SNA structure, however, specifies the responsibilities of functionally discrete layers in every node. SNA describes the interaction of each layer with a peer layer in another node. This matched pair of layers performs a specific function in the network, and the functional responsibility can be shared equally by the two layers. In other instances, the primary functional responsibility is allocated to one node, thereby simplifying the other node. This pairing of peer layers is the architectural feature that permits an SNA network to easily support the distribution of function without major network disruption. Also, functional changes can be made in one layer with minimum or no alteration of the remaining layers of the node.

transmission management layer

The layered structure allows terminal operators and application programs to obtain information-processing services without being involved in transmission details. Similarly the transmission management layer[4] controls movement of user data through the network, independently of the contents of the user data unit. Because the transmission management layer does not examine, use, or change the content of the data units, a change in the method of transmission between nodes requires no change in the data unit itself. A transmission management layer exists in every intermediate node through which the data units flow, and transmission management may utilize a variety of physical connections and protocols between the nodes of an SNA network. Paths through the network can consist of several nodes and data links that can be shared by many applications. The transmission management layer provides the control necessary to manage these shared resources without the intervention of the function management layer. A basic design concept of the transmission management layer is the integrity of the data unit exchanged between function management layers. That is, the data unit passed from the function management layer to the transmission management layer in the originating SNA node is passed intact from the transmission management layer to the function management layer in the destination node.

function management layer

The function management layer controls the presentation format of information sent from and received by the application layer. Function management components convert the data into a form convenient for the user. For example, application data originally arranged in a line-printer format can be converted to a display-screen format. The conversion can take place in either the System/370 virtual storage computer or in the control unit supporting the display terminal. SNA allows function management to be distributed between the computer and control units. A network designer can select products that optimize application performance with little or no change in terminal-user procedures.

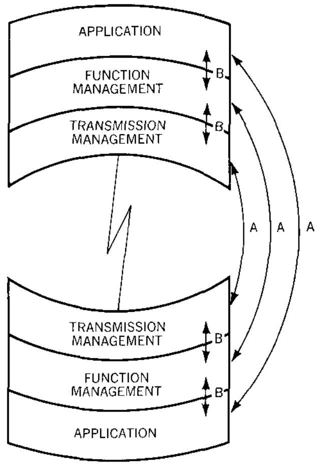
Function management also manages the protocols supporting the exchange of user information. Even though transmission management provides a full-duplex data flow, a terminal keyboard can be locked and unlocked to simulate the half-duplex data flow protocol that is more convenient for terminal operators. The function management layer can be designed to prompt the inexperienced terminal operator by using indicator lights or informational messages. Additional functions can be added to the function management layer in a programmable control unit to support some user requests for services, decreasing the application-processing load on the computer.

application layer

The services of the function management layer are invoked by requests from the application layer. In the computer, the application layer consists of the application programs from which the terminal user requests information-processing services. At the terminal, the application layer is represented by the terminal operator or an application program in a programmable control unit. SNA refers to these sources and destinations of information as "end users" and also includes in this group various unattended physical device media such as punched cards and magnetic tapes and disks. Application programming in the System/370 and the programmable control units can be provided by the user, by IBM, or by both.

In each SNA node, the application, function management, and transmission management layers operate independently. The layered structure of SNA nodes allows end users to exchange information without being involved in such procedures as controlling a communications line or routing data units through the network. Each layer communicates with a counterpart layer in another node, and SNA describes the means by which these peer layers communicate. This type of communication is shown in Figure 5 as peer layer communication.

Figure 5 Communication between layers



A: PEER LAYER COMMUNICATION
B: ADJACENT LAYER COMMUNICATION

The communication between adjacent layers in the same node is defined by the individual products. SNA describes functional relationships between these layers but does not define the format for adjacent layer communication. For example, SNA does not define the Virtual Telecommunications Access Method (VTAM)[5] macroinstructions that a System/370 application program uses to access the network. Nor, for example, does SNA define the program statements used by the programmer developing applications for the IBM 3790 Communication System. The programmer can use these statements to develop programs that facilitate use of the communication system by the terminal operator. The operator interaction with the function management layer of his terminal is an example of adjacent layer communication. Both adjacent and peer layer communication support the exchange of information between end users in different nodes of the network.

an SNA network

The advantages of a layered architecture[6] are evident in the types of SNA networks that can be designed. SNA permits configurations, such as that shown in Figure 6, which support many diverse applications. However, not all individual products are capable of supporting a full spectrum of network configurations or a full complement of specialized and general-purpose subsystems. The Supermarket System in the figure is an example of a system designed for a specific application that does not support all general-purpose devices.[7] In the same network, programmable subsystems such as the 3790 Communication System can provide the specialized control required for the input/output devices while supporting multiple different applications concurrently. Figure 6 has been simplified by not showing the wide range of devices that can be attached to these subsystems. The Finance System, for example, can include not only a keyboard display and document printer but also a consumer transaction facility that can issue variable amounts of money. In many cases,

Figure 6 Versatility of an SNA network

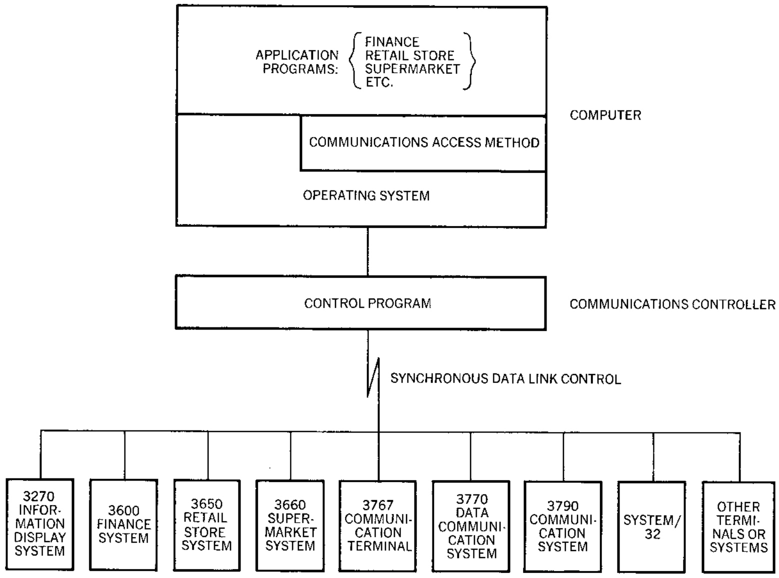
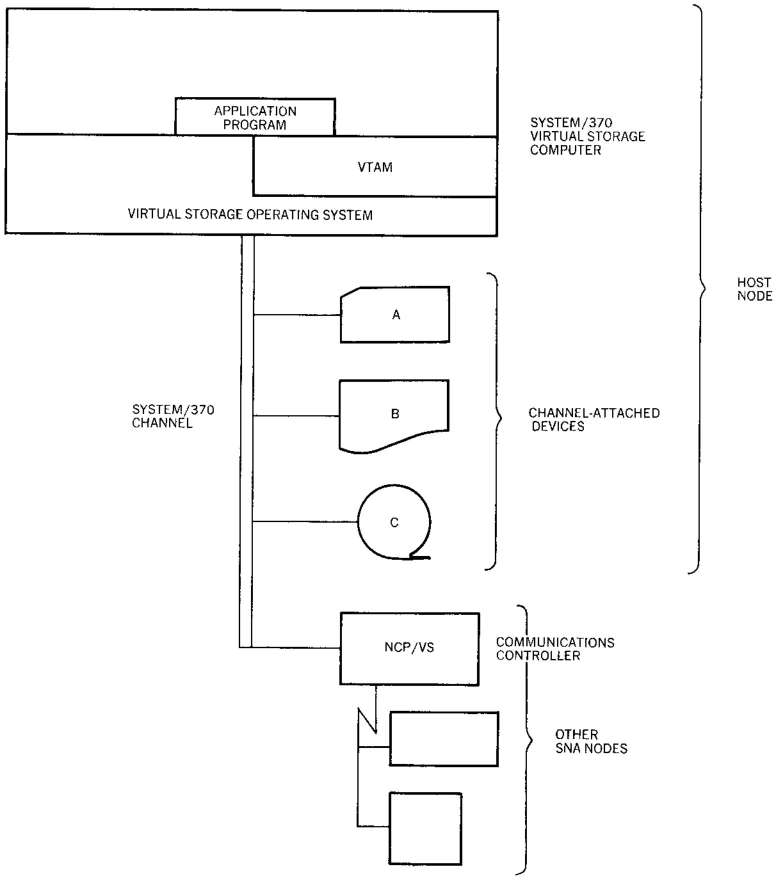


Figure 7 Coexistence of SNA nodes and host devices



support for new devices attached to these subsystems and reconfiguration of the network can be accomplished without alteration of the application programs in the computer.

SNA network nodes

System/360 networks emphasized communication between a terminal and a program in the computer. SNA networks continue this support but are also designed to support communication between programs in different nodes. The functions and protocols defined by SNA to support this communication operate in the context of a physical network composed of nodes interconnected by data links. SNA distinguishes different types of nodes by their network capabilities and logical relationships rather than by geographical location or physical configuration. An SNA node is classified as a host node, a communications controller node, or a cluster controller node. A fourth type of node, the terminal node, is a version of the cluster controller node with reduced capability for network interaction. A communications controller node performs some additional network functions for the terminal node.

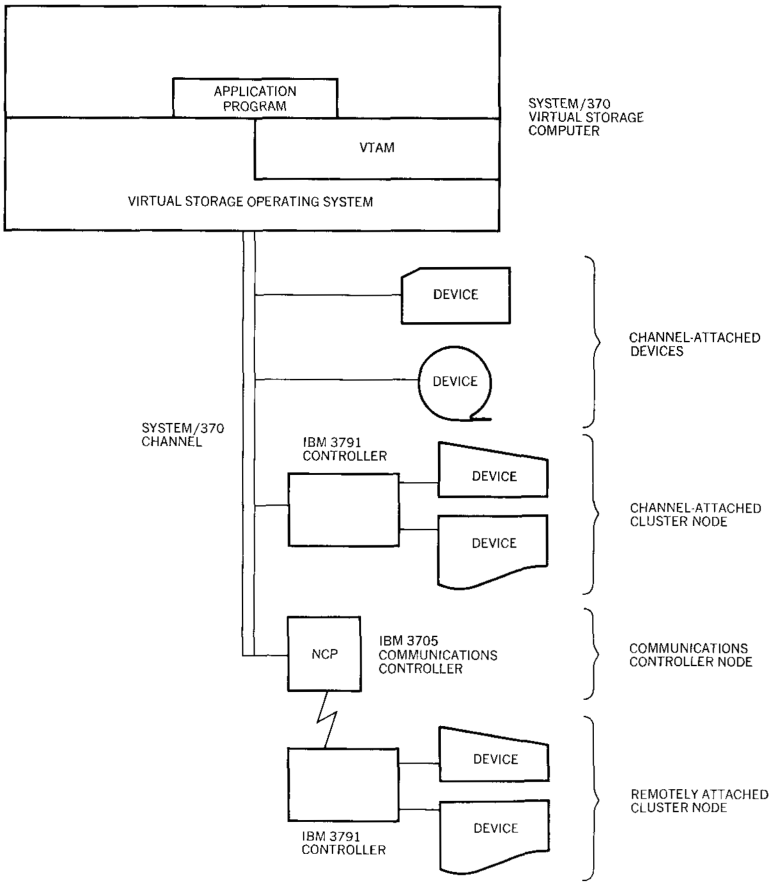
host node

A host node is a job-oriented general-purpose computer supporting the SNA formats and protocols. A current implementation is the System/370 virtual storage computer with VTAM. To communicate with end users in other nodes, the application program uses facilities of the host function management and transmission management layers. The application program may use host devices in responding to an end-user request for services. Most channel-attached input/output products such as card readers and printers are part of the host node. These devices are controlled by programming in the host. In Figure 7, devices A, B, and C are host node resources rather than individual nodes that can be managed as separate network resources. Unlike cluster nodes, these devices are not directly addressable from other nodes of the network.

cluster node

Cluster control units are operator-oriented, and some implementations of the cluster concept are programmable. A programmable cluster is usually a specialized subsystem of input/output devices whose processing unit is a control unit, but a cluster can also be a computer such as the System/32. Devices attached to clusters are not directly addressable from other nodes of the network. The devices connected to the IBM 3791 Controller in Figure 8 are controlled by it and are part of the IBM 3790 Communication System. Some clusters can be attached either directly to a System/370 channel or to a communications line. The application program communicates with end users of the channel-attached Communication System cluster in the same way as with end users of the remotely attached cluster.

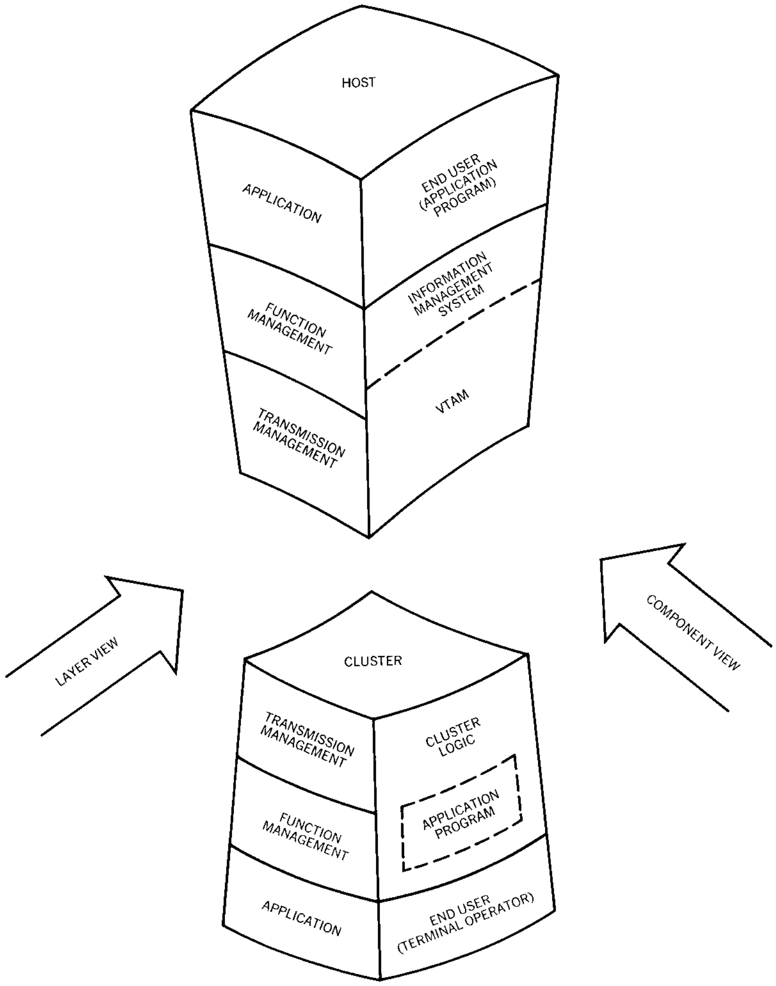
Figure 8 Channel and remote attachment of clusters



The layered structure of host and cluster nodes is compared in Figure 9. The architectural organization of the two nodes is identical, but the capability of each layer and the details of the implementation depend on the specific functions of each node. Cluster logic can consist of a mixture of "hardware" logic and application programming; the amount of hardware function supplied by IBM depends on the type of cluster node. The IBM 3770 Data Communication System, for example, has both programmable and nonprogrammable models. In a nonprogrammable model the operator uses the cluster hardware to communicate with application logic in the computer.

In programmable clusters, programming can be provided by IBM or by the user, or it may be a combination of IBM and user programming. Programs to be excuted in the cluster are prepared using an IBM-supplied assembler in a host node. Some programmable clusters such as the 3790 Communication System can

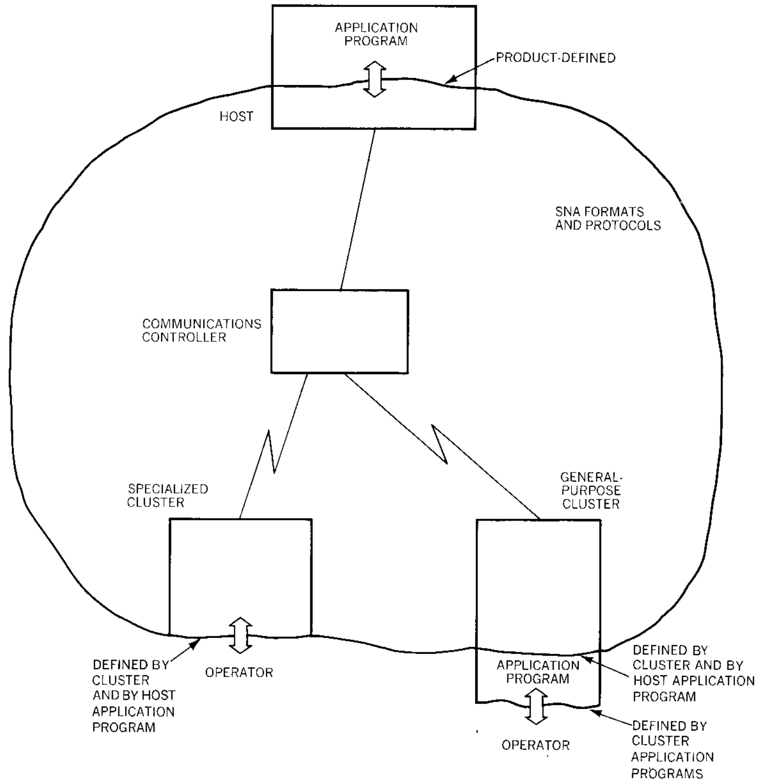
Figure 9 Structure of host and cluster nodes



support multiple applications simultaneously. Application programs allocate processing functions between host and programmable cluster nodes to best utilize the network to satisfy application requirements. Application programs that are not dependent on the host computer can continue to operate even though contact with the host has been lost. A programmable cluster node can accommodate various groups of users since the operator interface shown in Figure 10 is partially defined by cluster application programming.

The cluster concept encompasses a wide variety of control units and devices, both specialized and general purpose. A simple cluster like the IBM 3771 Communication Terminal can be based on a single input/output mechanism and include only basic function management and transmission management layers. As shown in Figure 11, a larger and more versatile cluster can sup-

Figure 10 End-user interactions with an SNA network

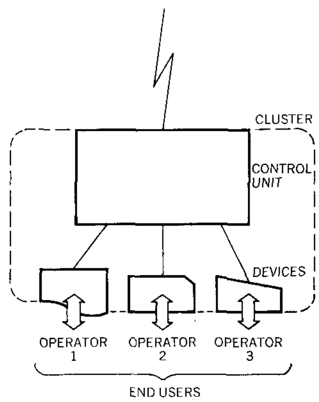


port multiple end users who obtain information-processing services through various types of devices supported by the cluster. The IBM 3774 Communication Terminal, for example, can include a line printer and a card reader as well as a keyboard-printer.

communications controller node

A communications controller node supports end-user communication by interconnecting other network nodes. The transmission management layer of this node reduces the network management responsibilities of the host node. Acting as a store-and-forward node, the communications controller contributes important integrity and recovery characteristics to an SNA network. It can provide two basic types of facilities. As a relay facility, the communications controller acts as an intermediate node that routes messages to the next node in the path through the network. As a boundary facility, the communications controller node shields the cluster from the complexities of network operation. The simplification of the network functions that must be performed in the cluster leaves more processing capacity available in the cluster for application-related functions. The communications controller, for example, converts full network addresses into formats more convenient for cluster operation and paces data flow based on the dynamic requirements of the cluster. Communications controllers connect host, cluster, terminal, and other communications controller nodes together into a unified network. The IBM 3705 Communications Controller operating under the direction of the Network Control Program/Virtual Storage is an example of such a controller.[8]

Figure 11 Cluster support of multiple end users



Network node structure and operation

SNA defines an orderly flow of information between an application program in a host and either a terminal user or another application program. To facilitate this communication and ease application program development, an SNA network insulates end users from the details of network operation. An end user need only be aware of the protocols and data formats required for interaction with function management in his node. There are three different types of function management components in an SNA network. These are the logical unit, the physical unit, and the system services control point.

logical unit

A logical unit is the set of function management services directly supporting an application program end user or an operator end user. The logical unit provides the services and protocols necessary for the end user to communicate with other end users. Before two end users can exchange information, a logical relationship called a *session* must be established between their respective logical units. The establishment of a session commits the appropriate resources such as a physical path, buffer storage, and processing capacity for the duration of the session. User protocol, the characteristics of the user interaction with the function management layer, involves end-user agreement on data syntax and meaning, full or half duplex data flow, and methods for controlling the rate of user data transfer. The logical unit performs appropriate data conversion for the end user. When an error condition has been detected, the logical unit assists in error recovery and reports uncorrectable errors to the end user. Function management services can be distributed between the two communicating logical units.

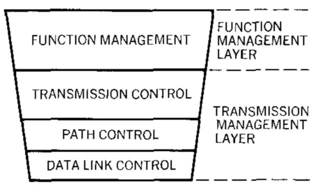
physical unit

A physical unit is a set of function management services that manage the SNA resources of a node in response to requests from the system services control point. Such requests include the initialization of the node during which the physical unit obtains the resources necessary for the node to operate as part of the SNA network. The physical unit in a communications controller node, for example, is responsible for activating other links in the network. In a cluster node, the physical unit establishes an environment in which logical units can be activated and provides the facilities for logical unit communication with the system services control point. The physical unit assists in recovery when network communication fails and can also contain measurement services.

system services control point

Management of an SNA network is the responsibility of the system services control point, and the network operator directs the overall operation of the network by communicating with the control point. This component manages the network elements by means of sessions established between it and all logical units and physical units. Through these sessions a logical unit can request the establishment and termination of a session with other logical units. The session services component of the control point supports these sessions by resolving symbolic names to network addresses and allocating the appropriate resources using system services control point-physical unit sessions. The configuration services component of the control point is used to alter the network status of physical units, logical units, and links. The system services control point also detects and recovers from various error conditions. Maintenance services provide for testing the network facilities as well as collecting and recording error information.

Figure 12 Components of an SNA node



SNA describes several levels of protocols that are rules for sequences of requests, actions, and responses. Some of these sequences occur only infrequently as, for example, when a cluster node is being activated. In this sequence, the node is first contacted at the communication link level, then the system services control point activates the physical unit and, finally, the control point activates one or more logical units. One of several sequences is then used to establish a session between logical units, depending on the particular function management component that originates the request for the session.

transmission control

The three types of function management components—logical unit, physical unit, and system services control point—communicate with each other by invoking the services of the transmission management layer. As shown in Figure 12, the three major components of transmission management are transmission control, path control, and data link control. In addition to the user and function management levels of protocol already described, session communication also requires transmission control protocols. These protocols are supported by the physical transmission of information and are used to manage the transmission control components. These components assist function management in session initiation, termination, and recovery. Transmission control also controls the flow of data to and from other transmission management components and manages the details of the physical transmission for the function management layer.

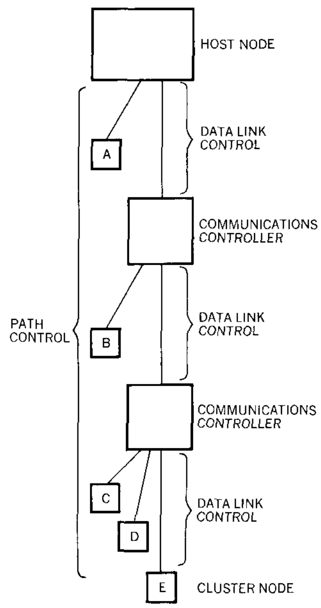
path control

The path control component provides a full duplex path, independent of the physical configuration. Path control performs the path-selection functions, ensuring that the correct data link is selected and that the transmission format is appropriate for that link. Path control routes data over the available links and through intermediate nodes enabling many end users to share common network resources. If path control cannot recover from a transmission error, the transmission management layer informs the logical unit, physical unit, or system services control point of the transmission difficulty. When such an event occurs, the components of the function management layer may attempt recovery, record system errors, and notify the network operator and end users as appropriate.

data link control

The data link control component manages a physical link connecting two nodes and delivers the data to the next node on the path selected by path control. A single link protocol, synchronous data link control (SDLC), has been developed to provide improved link operations for all telecommunications links.[9-11] SDLC maintains the integrity of the data units transmitted on point-to-point or multipoint lines or on a loop configuration. Data link control informs path control of errors only when the errors cannot be corrected by data link control alone. Path control components cooperate in routing data between end points, whereas data link control manages data flow only between adjacent nodes. Figure 13 shows that data link control moves data on the individual links of the path between a host application program and cluster E.

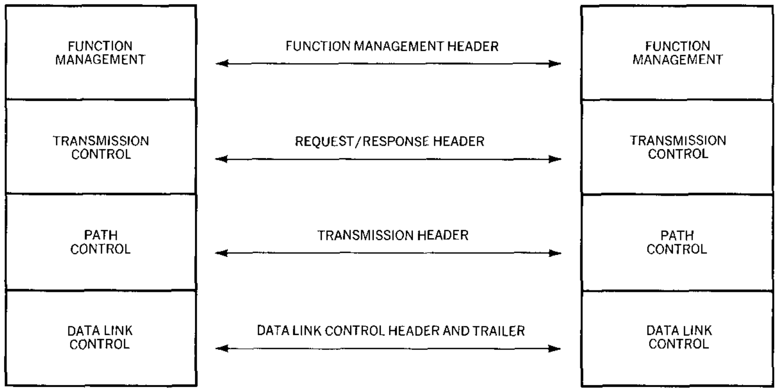
Figure 13 Path control routing



SNA components communicate with their counterpart components in other nodes by means of the headers and trailers shown in Figure 14. Headers contain control fields and describe the destination and type of information to which the header is prefixed. Trailers may simply signal the end of a "frame" or may contain a field used for verifying the correct transmission of the frame. On an SDLC link, data link control in one node communicates with data link control in another node using both headers and trailers. Data link control in the receiving node interprets only the header and trailer and does not examine or alter data passed from path control in the sending node. When a node is being initialized, the data may be the initial program load for the node. Figure 15 illustrates the format of a typical frame that flows on the link after the node has been initialized. In the receiving node, the data link header and trailer are stripped from the frame, and the remaining information is passed to path control.

Each header contains several fields of control information. For example, one of the fields in the transmission header identifies the final destination of the data. Path control in a receiving node uses the address field to determine whether the data is intended for that node. If not, as is usually the case in a communications

Figure 14 Communication between peer layers

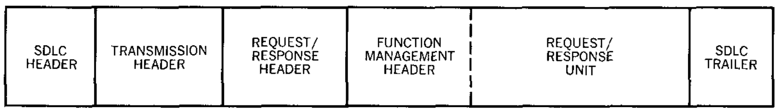


controller node, path control modifies the transmission header as necessary and selects an outbound link on the path to the destination node. The data link control component responsible for the outbound link constructs a new frame, and the process continues until the data reaches the destination specified in the transmission header. Path control in the destination node examines the transmission header to determine the transmission control component for which the information is intended.

The request/response header is used for communication between transmission control components and is not examined by data link control or by path control. Transmission control uses the request/response header to control the data flow and to direct the remaining control information and data to the appropriate node component.

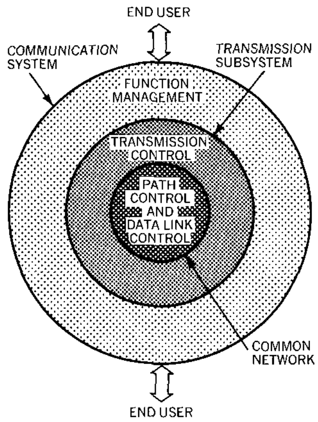
The function management header can be used to describe the contents of the request/response unit, which contains the end-user information being transferred through the network. To communicate with another end user, an end user need understand only the content of the request/response unit and the protocol for interacting with function management in his node. In each node component, the appropriate header is processed indepen-

Figure 15 Structure of an SDLC frame



dently of the data field contents. The sequential processing of successive headers and data fields complements the functional separation of the SNA layers.

Figure 16 Subsystems of an SNA network



SNA network subsystems

The layered structure of an SNA node allows an end user to be independent of and unaffected by many network services and facilities. An individual application program can be written without reference to specific implementations of devices or networks. This independence was difficult to achieve in pre-SNA networks because the services provided by different System/360 telecommunications access methods were not functionally equivalent.

In an SNA network, all nodes have a similar structure, and matched layers of the SNA node can be combined into the network subsystems. It is the design of these subsystems that simplify the concurrent use of an SNA network by multiple diverse applications. SNA describes three network subsystems—the communication system, the transmission subsystem, and the common network. End users share the facilities provided by these subsystems.

communication system

The SNA *communication system* is composed of the combined function management and transmission management layers of all SNA nodes in the network. As shown in Figure 16, end users exchange information using the facilities made available by the communication system. The end-user interaction with the communication system is described in reference manuals for the individual products; the structure and operation of the communication system is defined by SNA.

transmission subsystem

The outermost layer of the communication system, the function management layer, encompasses an inner subsystem, the *transmission subsystem*. Consisting of the transmission management layers of all SNA nodes, this subsystem includes transmission control, path control, and data link control components. The transmission control layer is the outermost layer of the transmission subsystem and connects the function management components of each node to the transmission subsystem. Transmission control elements are session-oriented and manage the flow of information into and out of the common network which is shared by all active sessions. The *common network*, composed of path control and data link control components of all SNA nodes, is the innermost subsystem described by SNA.

The implementation details of the network subsystems are irrelevant to the effective use of the network by many different application programs. Consequently, end users need not be sensitive to alterations in network configuration or operation, allowing the owner of the network to adapt the network to meet changing user requirements.

Summary

By relieving the user of network management concerns, Systems Network Architecture provides a flexible and extendible structure enabling multiple applications to share a single network. The SNA concepts encourage the distribution of information processing from computers to control units by establishing an environment that is independent of the physical configuration. In geographically dispersed networks, application programs and data bases can be moved from the host to another node closer to the terminal operator. Programmable control units can be optimized to allow persons with little or no data-processing training to utilize computers.

The communication system formats and protocols enhance the exchange of information among a diverse group of operators, application programs, and devices. Application programs and devices designed to SNA specifications solve specific problems rather than being involved in the entire communications process. Network users can concentrate their resources on specific applications without understanding the increasingly complex information-processing systems that assist them in their work.

CITED REFERENCES

1. *Systems Network Architecture—General Information*, No. GA27-3102, IBM Corporation, Data Processing Divisions, White Plains, New York.
2. C. R. Blair and J. P. Gray, "IBM's Systems Network Architecture," *Datamation* **21**, No. 4, 51–56 (April 1975).
3. *Advanced Function for Communications—System Summary*, No. GA27-3099, IBM Corporation, Data Processing Division, White Plains, New York.
4. P. G. Cullum, "The transmission subsystem in Systems Network Architecture," in this issue.
5. H. R. Albrecht and K. D. Ryder, "The Virtual Telecommunications Access Method: A Systems Network Architecture perspective," in this issue.
6. E. W. Dijkstra, "The structure of the 'THE' multiprocessing system," *Communications of the ACM* **11**, No. 5, 341–346 (May 1968).
7. P. V. McEnroe, H. T. Huth, E. A. Moore, and W. W. Morris III, "Overview of the Supermarket System and the Retail Store System," *IBM Systems Journal* **14**, No. 1, 3–15 (1975).
8. W. S. Hobgood, "The role of the Network Control Program in Systems Network Architecture," in this issue.
9. R. A. Donnan and J. R. Kersey, "Synchronous data link control: A perspective," *IBM Systems Journal* **13**, No. 2, 140–162 (1974).

10. *IBM Synchronous Data Link Control—General Information*, No. GA27-3093, IBM Corporation, Data Processing Division, White Plains, New York.
11. J. R. Kersey, "Synchronous data link control," *Data Communications* **3**, No. 1, 49–60 (May/June 1974).

Reprint Order No. G321-5024