

*As an access method, VTAM is influenced by many non-SNA considerations. However, this paper focuses on the SNA functions implemented by VTAM, discussing the components involved in some detail. Also included is a brief discussion of the historical factors that led to the conception of VTAM.*

## **The Virtual Telecommunications Access Method: A Systems Network Architecture perspective**

**by H. R. Albrecht and K. D. Ryder**

The Virtual Telecommunications Access Method (VTAM) constitutes the main access method support in IBM's Advanced Function for Communications. A proper understanding of the role VTAM has in the IBM teleprocessing systems requires the comprehension of several factors.

The first factor that will be discussed is the historical situation that led to VTAM's conception. The second is its initial objectives. The third and fourth are another iteration on the history and expanded objectives that resulted in the introduction of the Systems Network Architecture (SNA). SNA is the architecture that underlies IBM's Advanced Function for Communications,<sup>1</sup> and the remainder of the paper describes the portions of SNA that fall within the scope of VTAM and have been implemented by the access method. The VTAM product design (component and subcomponent structure, etc.) will not be described, as this information is readily available through published sources.<sup>2-4</sup> Neither will a definitive description of SNA be presented here, since that information is also available in published form.<sup>1,5-10</sup>

### **Development of VTAM**

In 1971, the Basic Telecommunications Access Method (BTAM) and the Queued Telecommunications Access Method (QTAM) were IBM's principal access methods for teleprocessing. The Telecommunications Access Method (TCAM) had just become

**historical  
background**

available on OS/360 as the replacement for QTAM, providing a queued access method with superior performance, function, recovery, and device support, including support for binary synchronous communications (BSC) terminals. In addition, it provided specialized direct control support for terminals under the Time Sharing Option (TSO). It did not provide direct control support as a general-purpose facility and thus did not provide an alternative to BTAM for data base/data communications facilities (e.g., the IBM Information Management System (IMS) and the IBM Customer Information Control System (CICS)) or other application environments requiring that level of support. Also, it did not support the DOS/360 environment. Evaluation of these factors, and others, led to the conclusion that a new access method was required for continued growth in the general-purpose data base/data communications environment.

Also at this time, more and more data processing users were able to utilize multiple teleprocessing applications. For such users, it was becoming increasingly important to support sharing of network resources (e.g., lines and terminals) across multiple programming systems and application programs.

As the number of teleprocessing applications grew, terminal networks were also growing. As a result, systems were devoting a larger percentage of CPU resources to managing lines and terminals. A new family of communications controllers (the IBM 3704 and the IBM 3705) made it possible to reduce this load by performing line control and error-recovery functions in the controller. In addition, it was foreseen that emerging technologies would make it possible to develop programmable device controllers. This development would permit application functions to be distributed to the device controller itself, allowing much more significant reductions in the CPU load. Thus, new communications controllers were on the scene, and the prospect of programmable device controllers was on the horizon.

Finally, one other factor influenced the decision to develop VTAM. At the time that the decision was made, a new line of processing systems (System/370) with dynamic address translation features was in the offing. New operating systems, such as DOS/VS and OS/VS, would be introduced to support virtual addressing. In addition, OS/VS2 would be extended to provide more effective utilization of the increased real memory sizes and multiprocessing capabilities available on the larger models of System/370.

**VTAM  
initial  
objectives**

The above summarizes the major considerations that resulted in the decision to develop VTAM. It was to be a new teleprocessing access method that would meet the following objectives:

- Support IBM's general-purpose data base/data communications facilities (CICS and IMS), as well as other data communications facilities (including TCAM) and application program environments.
- Support the sharing of network resources (communications controllers, lines, and terminals) among application programs and data communications facilities.
- Improve the utilization of the new communications controllers and provide a base for further function distribution in the future.
- Improve the teleprocessing utilization of virtual addressing capabilities across DOS/VS, OS/VS1, and OS/VS2.
- Improve the teleprocessing utilization of multiprocessing capabilities (OS/VS2 only).

Based on the objectives, a VTAM system design was produced for each of the three virtual storage operating systems. The resulting definition was the VTAM Basic-Mode macroinstruction language and associated functions for communicating with BSC and start/stop terminals.

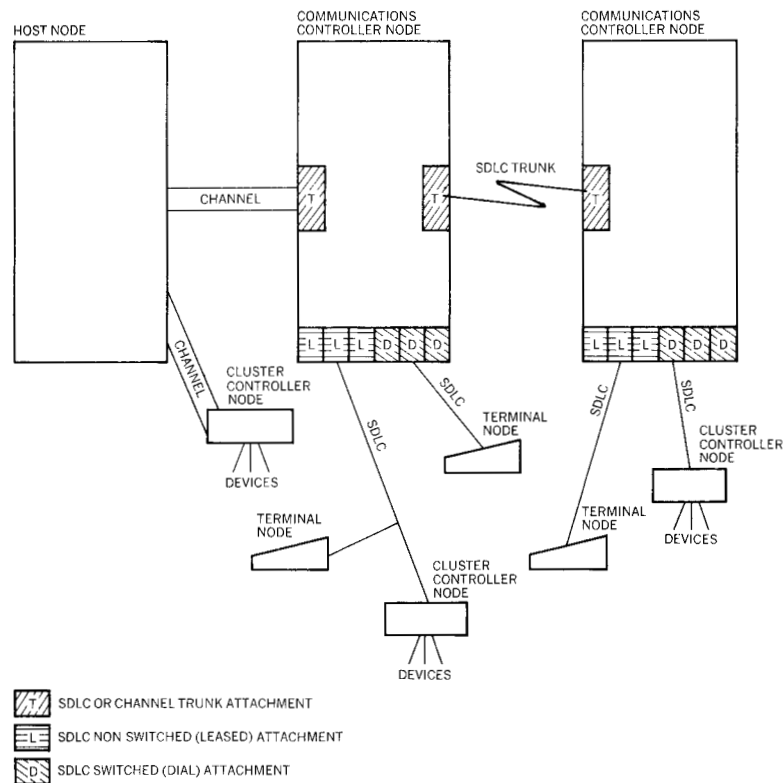
About a year after VTAM's inception, another major evaluation of IBM's technical direction for communications took place. This further evaluation was stimulated primarily by two related factors. First, the new-technology device controllers (referred to earlier) were well along in the development cycle. Second, a new line control discipline called synchronous data link control (SDLC) had been developed. SDLC offered performance, function, and reliability benefits to the new controllers.

**new  
communications  
technology**

It became apparent that a number of additional problems had to be solved in order to facilitate an orderly development of data communications systems utilizing the new device controllers. The solution to these problems are embodied in the concepts, structure, and operational protocols that constitute SNA. The problems are discussed briefly below.

The currently available BSC and start/stop terminals used a variety of specific line control and device control disciplines. This variety resulted in increased development cost for both IBM and the teleprocessing user in supporting the unique characteristics of each new terminal. It also resulted in increased line costs to the user because different line control disciplines could not be mixed on the same line. The new device controller technology, combined with the availability of SDLC procedures, facilitated the application of SNA concepts. The essence of these concepts is to separate data link control protocols, end-to-end data communication protocols, and device control protocols, all of which had been integrated with data link controls in the then-available terminals. The clean separation of data link control protocols

Figure 1 Physical network environment



also made it possible to introduce both node-to-node and end-to-end routing procedures. Thus, it was possible to introduce simple procedures for routing data through a series of nodes (interconnected by data links) to its final destination. By using these SNA concepts, it was possible to define two classes of nodes that would meet a wide variety of device controller requirements. By defining and supporting the new device controllers within one of these two classes, the requirement for unique terminal support code was largely eliminated. The two classes, SNA terminal nodes and SNA cluster controller nodes, may also share the same multipoint communications line or the same dial port, thus reducing line and modem costs.

One other key problem remained to be addressed. A network that includes controllers with significant functional capabilities requires a well-defined discipline for managing the operational state of the physical and logical connections between nodes. This requirement resulted in the introduction of another major SNA concept—separation of the network management functions from the main-line data transmission and data presentation functions. This concept allows flexible placement of the network

management functions as determined by the requirements of the particular type of network topology being supported. The initial requirement was to provide centralized network management in support of a single-host, tree-structured network. These separate network management functions are embodied in the system services control point of SNA.

With the advent of SNA, the VTAM objectives had to be extended. The previously stated objectives were still valid, and, in fact, the objective to support sharing of network resources was further strengthened. In addition, three major new objectives were introduced as follows:

**additional  
VTAM  
objectives**

- Support the new device controllers as SNA cluster controller nodes and SNA terminal nodes.
- Support the centralized network resource services required by SNA.
- Support the separation of data link protocols, end-to-end protocols, and device control protocols.

With the introduction of the above SNA-related objectives, it was required that VTAM support both the then-current network environment and the new SNA environment. Since the VTAM design for supporting both environments is well-documented, the remainder of this paper provides an overview of the functions provided by VTAM in an SNA environment.

### **Physical network environment**

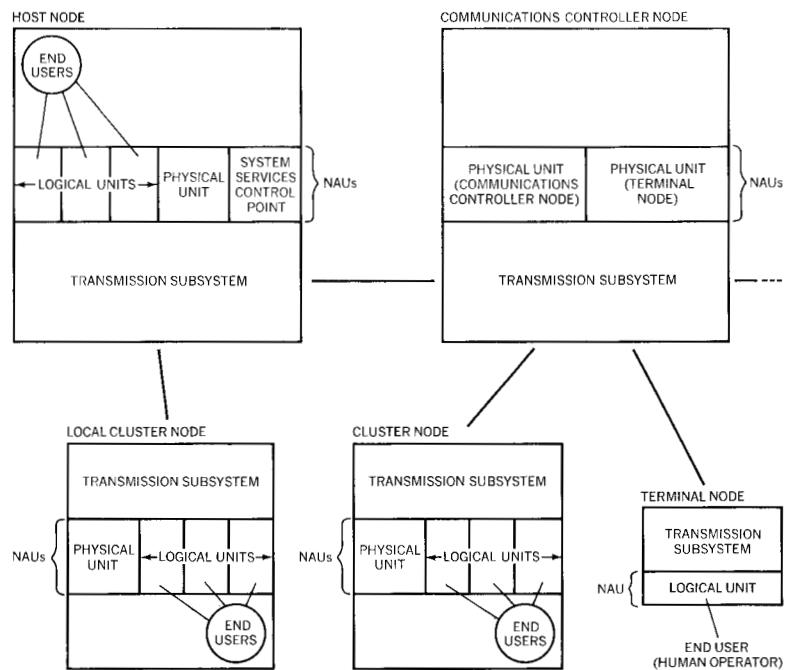
SNA supports a physical environment that encompasses a broad range of hardware units and attachments. Figure 1 is an example of such a network environment. A detailed discussion of its components, other than the host node, is beyond the scope of this paper.

The host node is based upon a System/370 CPU, and the physical attachment to this node is via channels. VTAM, in conjunction with the I/O supervisor of the host operating system, supports such channels as data links in the SNA network. In Figure 1, two distinct types of SNA nodes are shown attached via the System/370 channel. These nodes are a cluster controller node and a communications controller node.

Although SNA does not restrict the development of more complex network topologies, the VTAM implementation is limited to hierarchical, tree-structure networks of the type shown in Figure 1. Multiple local and remote communications controllers are supported, but only a single host node is allowed.

**complex  
networks**

Figure 2 Logical network environment



## Logical network environment

The logical configuration of the network appears quite different from the physical configuration. Distributed function within the SNA environment makes these distinctions possible (Figure 2).

network  
addressable  
units

Within each node in the network exists a set of logical ports to the network, called network addressable units (NAUs). These NAUs represent the addressable entities among which network traffic can flow. The actual transmissions to and from NAUs are achieved through the transmission subsystem<sup>8</sup> support within each node. Obviously, the sophistication of these transmission functions varies greatly, depending upon the type of node in which the functions exist. Only the transmission subsystem functions of the host node will be described in this article.

The NAUs support network access for three main types of functions:

1. System services control point—This function is a unique, host-resident entity that is responsible for overall management of the SNA network. Specific control point functions, as implemented by VTAM, will be described in a following section.

2. Physical unit—Each node in the network, i.e., host node, communications controller node, cluster controller node, and terminal node, possesses functions responsible for the physical management of that node and any adjacent, subordinate links or devices. Only the physical unit functions implemented in the host node by VTAM are within the scope of this paper.
3. Logical unit—This function is the network port provided for an end user, where an end user may be a program in the host, a program in a cluster controller, or, in the case of a terminal node, a human operator.

Communication between NAUS is on the basis of logical connections called sessions. Once a session is established between a pair of NAUS, a normal flow of information may begin. VTAM supports the following sessions:

#### sessions

1. System services control point/physical unit (SSCP/PU)—As each physical node in the network is activated, a session is established between the physical unit function in that node and the control point to carry out network management activities that directly affect that particular node.
2. System services control point/logical unit (SSCP/LU)—As each logical unit is activated, a session is established between the newly activated logical unit and the control point. This logical connection allows an end user to communicate control information to the control point, e.g., requests to begin a session with a host application.
3. Logical unit/logical unit (LU/LU)—The normal data flow within the network is dependent upon the establishment of sessions that allow end-user-to-end-user communication. This type of session represents a logical binding of the appropriate pair of logical units.  
The logical unit representing the host application may be concurrently in session with multiple logical units in the network.

The VTAM interface to host application programs also represents end-user communication in logical, not physical terms. There are several important consequences for the VTAM user. As seen in Figure 2, the logical configuration, and hence, the end user, is not affected by physical attachment considerations. The type of link attachment, channel or SDLC, is transparent to the end-user programs. In a similar manner, dial versus leased attachment considerations are transparent to end-user programs.

Thus, the VTAM user is not confronted with management of physical connections, paths, or links. Rather, the user deals with symbolically named resources. The mapping of these names into network addressable entities and the management of both the

physical and logical network is achieved by the system services control point implementation within VTAM.

### Host node

As an access method, VTAM is influenced by many non-SNA considerations. As noted earlier, VTAM must optimize its operations within the environment provided by: (1) a System/370 CPU with dynamic address translation, (2) a DOS/VS, VS/1, or VS/2 operating system, and (3) a multiprocessor environment (VS/2 only).

However, for the purpose of this discussion, it is the SNA influence that is of most interest. Through its definition of the host node, SNA has stated the requirements that most directly affect VTAM.

SNA explicitly sets forth certain external functions of the host node and thereby has some implications on its internal structure. An implementation of the network architecture, such as VTAM, may modify the SNA internal structure so long as it continues to support the required external functions in the SNA-specified manner. VTAM does not follow an internal structure precisely as shown in Figure 3, primarily because of the late introduction of SNA into the VTAM objectives. Yet, VTAM fully supports the SNA functional requirements. Hence, Figure 3 provides a convenient basis for discussing the SNA host node functions that are provided by VTAM.

#### NAUs supported in the host

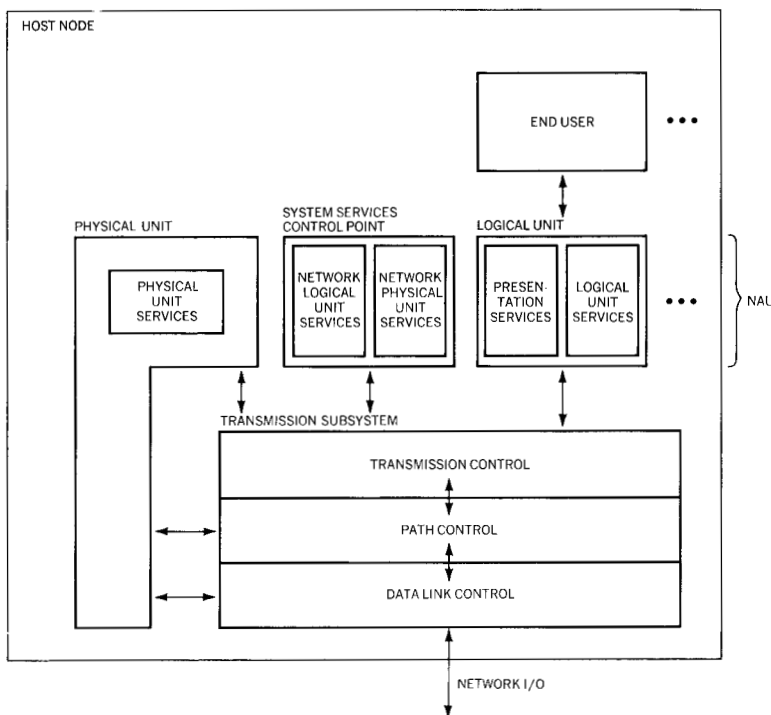
The host node is unique in the network in that it alone supports all three types of NAUs: system services control point, physical units, and logical units.

The system services control point is found only in the host node and is responsible for the logical and physical domain containing all NAUs. In the present network implementation, this domain implies a hierarchical, tree-like structure with control centralized in a single control point.

System services control point functions are conveniently separated into the two categories of network logical unit services and network physical unit services. The logical unit services support the initiation and termination of sessions between logical units in the network. The physical services are concerned with establishing, controlling, and monitoring the physical configuration of the network.

The second NAU type found in the host node is the physical unit. The system services control point achieves its management of the physical network through interactions with the physical

Figure 3 Host node structure



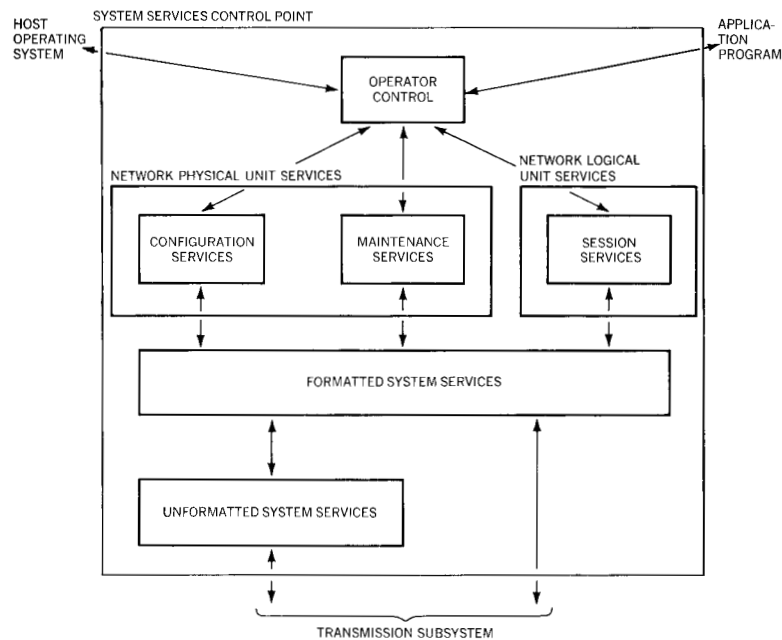
units in the appropriate network nodes. The functions involved include the activation and initialization of links and nodes, the handling of certain error conditions, and overall maintenance of the physical configuration.

The final type of NAU, the logical unit, provides a port for network access by end users. In the case of the host node, these end users are in fact programs. Each logical unit provides specific sets of services to its end user. Logical unit services refers to that group of functions that interact with the system services control point in creating and terminating sessions between logical units. Presentation services is a set of functions that support data and control communication between the end users of a session. It is presentation services that supports the transformations required due to specific characteristics of end users.

The actual flow of traffic between the network ports is achieved via the transmission subsystem. The highest level of function within the transmission subsystem is transmission control, which provides a duplex data transmission capability on a session basis. At the path control level, traffic routing is performed. Finally, at the level of data link control, the direct management of the physical transmission link occurs. In the host node, the

transmission  
subsystem

Figure 4 System services control point structure



data link control functions imply management of the attached channels. This responsibility is shared between VTAM and the host operating system.

### System services control point

As was suggested in Figure 3, the major categories of functions within the system services control point are network logical unit services and network physical unit services. However, before entering a more detailed analysis of these areas, it is convenient to discuss the manner in which these functions are accessed.

A more detailed view of the functional structure of the system services control point is to be found in Figure 4. This diagram is not intended to portray the subcomponent structure of VTAM. Rather, this view presents logical groupings of the SNA functions that appear in VTAM's implementation of the control point.

The system services control point communicates with logical units and physical units in the network via the transmission subsystem. This communication can take two forms. It may be done through the use of formatted, bit-encoded SNA commands, or it may be done through the use of character strings. The latter form is provided primarily for multipurpose operator terminals. This unformatted, character-string form provides a more conve-

nient means for a human operator at a terminal to exchange control information with the system services control point. The unformatted system services subcomponent of the control point receives this character string input and converts it to a formal, formatted SNA command. Replies from the control point back to the terminal operator will also be in the form of character strings. The unformatted system services support will accept character strings, e.g., LOGON/LOGOFF, that cause the initiation and termination of sessions between a remote logical unit and a host logical unit.

SNA cluster controllers have the ability to accept input from their attached devices and convert commands directly into the formatted form, which requires bit-level encoding of information. Hence, communication between the system services control point and a logical unit in a cluster controller is normally via formatted system services. These commands are processed by formatted system services without the need for any handling by unformatted system services. Once a command has reached formatted systems services, it is then routed to the control point subcomponent responsible for the requested function.

Communication between the control point and the host physical unit or between the control point and host logical units is not via the transmission subsystem. Instead, these intra-host linkages are achieved in a direct, node-dependent manner.

The second major path for communication with the system services control point is via the network operator interface. The VTAM support defines the network operator station as the system console. Through the multiple console support facility in OS/VS1 and OS/VS2, the network operator stations may be designated as any subset of system consoles.

**network  
operator  
interface**

Routing of network commands to the control point is generally achieved by adding a special NET parameter to the operating system VARY, MODIFY, and DISPLAY commands.

A recent functional enhancement to VTAM also allows an application program in the host to submit operator commands to the control point. Through this host application, remote logical units may be provided with a capability for requesting operator functions.

The first major grouping of system services control point functions is network logical unit services. The host node implementation of these functions is primarily in the area of session services. When a host logical unit or a remote logical unit wishes to establish a session or to terminate a session, it is done via interaction with session services in the control point.

**network  
logical  
unit  
services**

The initiation of a session by a remote logical unit provides a specific example of the role of session services. If it is assumed that the remote logical unit is contained within an SNA cluster controller node, then a formatted SNA command, called INITIATE, would be sent from the logical unit to the control point. This command would be routed to the session services portion of the control point; as a result the following would occur:

1. Session services causes the symbolic resource name specified by INITIATE to be mapped to the address of a particular logical unit in the host.
2. Session services notifies the logical unit services portion of the selected host logical unit that a session is desired.
3. Logical unit services drives an application routine, via the LOGON exit, to notify the application of the session request.
4. The application, assuming that it desires to consummate the session, will use the OPNDST (Open Destination) macro to signal logical unit services.
5. Logical unit services sends an SNA command, BIND, to the remote logical unit to indicate acceptance of the session request.
6. Logical unit services notifies session services in the system services control point that the session has been successfully bound.

Similar functions are performed when a session is initiated by a host logical unit. These functions will be discussed in a later section on logical unit services.

A session may be terminated by either the host logical unit or the remote logical unit. The following sequence is observed when the SNA command TERMINATE (conditional) is sent to session services by a remote logical unit:

1. Session services causes the symbolic resource name specified by TERMINATE to be mapped to the address of a particular logical unit in the host.
2. Session services notifies the logical unit services portion of the selected host logical unit that the designated session is to be ended.
3. Logical unit services drives an application routine, via the LOSTERM exit, to notify the application of the request to end the session.
4. The application uses the CLSDST (Close Destination) macro to signal to logical unit services that session termination should proceed.
5. Logical unit services uses the CLEAR command to purge all data flowing on the session and then sends another SNA command, UNBIND, to the remote logical unit to indicate that the session has been concluded.

6. Logical unit services notifies session services in the system services control point that the session has been successfully unbound.

The symmetry between INITIATE and TERMINATE processing is obvious. It is important to note that the existence of an LU/LU session supporting user data flow has in no way supplanted or impaired the use of the SSCP/LU session by the remote logical unit. That is, the flow of user data on the LU/LU session has not affected use of the SSCP/LU session for INITIATE or TERMINATE command flow.

The second major grouping of system services control point functions is network physical unit services, which can be broken down into two subtopics, configuration services and maintenance services.

**network  
physical  
unit  
services**

Configuration services is responsible for the overall management of network resources and maintains tabular information relative to the symbolic name, network address, and status of all NAUS within the control point domain.

The NAUS may be activated explicitly by operator action or implicitly as a result of the activation of related NAUS. Configuration services can be invoked to activate a specific resource, or it can be requested to activate a symbolically named resource group, as defined by the installation.

Activation of a physical unit, as in the activation of a cluster controller node, involves the following interchange:

1. The system services control point sends an SNA command, CONTACT, to the communications controller node to which the cluster controller is attached. The CONTACT command causes the communications controller to perform link-level procedures to establish contact with the cluster controller node.
2. The communications controller sends a CONTACTED command to the control point, indicating that the CONTACT procedure has been successfully completed.
3. An SNA ACTPU (Activate Physical Unit) command is now sent from the control point to physical unit services in the cluster controller. This command causes the SSCP/PU session to be established, thereby allowing the control point to communicate with the physical unit.

The preceding scenario assumes that the link between the communications controller and the cluster controller had already been activated by the control point.

Activation of the logical units associated with the cluster controller is achieved via an ACTLU (Activate Logical Unit) command sent from the control point to the logical unit. This command causes the SSCP/LU session to be established, thereby allowing the logical unit to initiate and terminate sessions for end-user communication. Deactivation of physical units and logical units is also done via explicit commands from the control point.

Configuration services also plays a major role in network-level error recovery. When a physical unit detects the failure of an adjacent link or node, it will notify the system services control point of this inoperative condition. The control point will cause logical unit services to drive affected host applications via LOSTERM exists. In this manner, the applications are made aware that a network outage has occurred and that recovery, if possible, will be attempted.

The current level of configuration, as per the control point network definition tables, is used by configuration services to restore the logical and physical environment to its prefailure state. Failing nodes, such as communications controllers, are loaded as required. These recovery operations will cause physical units and logical units to be selectively activated and SSCP/PU and SSCP/LU sessions to be restored. However, physical units and logical units on dial links will not be activated as part of the configuration recovery process.

When configuration recovery procedures are complete, configuration services will again cause logical unit services to drive the application via a LOSTERM exit. The exit will be scheduled with an indication as to whether the error was permanent or was successfully recovered. In either case, the application is expected to reset its session representation via a CLSDST macro. If configuration recovery was successful, the application now has the option of reestablishing the LU/LU session by issuing an OPNDST macro. The OPNDST macro results in the system services control point and logical unit services interaction required to recreate the session. Once the session is reestablished, it is up to the application to resynchronize operations with the other end user.

The information necessary for configuration recovery is normally maintained in virtual storage. Optionally, it may also be maintained on auxiliary storage. In the latter case, the previous configuration may be restored after a failure and restart of the host node.

Other functions provided by configuration services allow access to information related to network resources and activity. Application programs in the host can inquire as to the status of logical units and related data or control flows.

The final set of functions within network physical unit services is known as maintenance services. These activities are generally carried out through the SSCP/PU sessions with the various nodes outside the host. For testing of SDLC links and various adapters, the Terminal On-Line Test Executive Program (TOLTEP) uses SSCP/PU sessions to pass control information to communications controller nodes. In this sense, TOLTEP may be viewed as a logical extension of the system services control point. Some SNA cluster controller and terminal nodes are also tested by TOLTEP, but this testing is achieved by a normal LU/LU session and does not involve any special use of the control point. Most SNA cluster controllers, because of their high function capability, can perform testing functions without host participation.

Another aspect of maintenance involves statistics collection. The control point will, for example, initiate line trace activities in the communications controller via commands sent to the physical unit in that node. This trace data is then sent from the communications controller's physical unit to the control point for recording. Test data and error records are also sent on the SSCP/PU sessions to be recorded in the host. The control point directs certain maintenance activities in the host node itself by communication with the host physical unit.

### **Physical unit services**

The services provided by any physical unit in the network represent specific node counterparts of the network physical unit services in the system services control point. For that reason, the services of a physical unit may be conveniently subdivided, as with the control point, into configuration services and maintenance services components.

The host physical unit is responsible not only for its specific node but also for all links and nodes that are directly attached to the host node. As part of its configuration services function, the host physical unit is responsible for the activation and deactivation of attached links. These procedures are carried out under the direction of the control point.

Also at the behest of the control point, configuration services will perform the link-level procedures preparatory to activation of a node attached to the host. These activities include the establishment of link-level contact with the node, determining whether or not the node requires a program load, and actually performing the program load, using program text supplied by the control point. This initialization brings locally attached nodes up to the level that permits them to receive SNA communications directly from the control point, e.g., ACTPU (Activate

Physical Unit). The configuration services of the physical unit play a similar role in the deactivation of local links and nodes and in the link-level dumping of communications controller nodes.

Whenever link-level errors cannot be successfully recovered, it is the responsibility of configuration services to notify the system services control point of the inoperative condition of affected local links or nodes. Subsequent recovery actions, if any, will be under control of the control point.

Another major function of configuration services is the maintenance of routing information. This information is obtained from the configuration definition tables of the control point and is used to determine the correct link for any transmission outbound from the host node. The physical unit also establishes the necessary information to allow routing of inbound data to logical units within the host node.

The host physical unit also plays a role in controlling the traffic flow to locally attached communications controller nodes. Whenever a communications controller node detects that its communications resources are being exhausted, it will notify the control point that it is entering a slowdown state. The control point will, in turn, notify the host physical unit that a slowdown has occurred. It is now the responsibility of the host physical unit to ensure that the transmission subsystem allows no more traffic to the affected communications controller until the slowdown state has been exited.

The maintenance services functions of the physical unit are primarily controlled through the control point and involve traces and data recording within the host node. The tracing capability allows the network operator to specify, on a node or session basis, that transmission units (I/O Trace) or data areas (Buffer Trace) are to be recorded. In either case, information on origin, destination, and direction of flow is supplied.

Another responsibility of maintenance services is the recording of permanent errors encountered at the local link level. Under some circumstances, e.g., counter overflow or end-of-day, temporary error conditions will also be recorded.

### **Logical unit services**

The logical unit services within the host have already been introduced through the discussion of the system services control point. Logical unit services perform functions complementary to the network logical unit services of the control point.

One means of categorizing logical units is as *primary* or *secondary*. In simplified terms, a primary logical unit is one that can send out SNA BIND and UNBIND commands, establishing and terminating a session. At its current level of definition, SNA supports only those sessions that involve one primary and one secondary logical unit. Further, the VTAM implementation requires that only primary logical units may exist in the host node.

Requests for session establishment and termination by a secondary (nonhost) logical unit have been discussed in the system services control point description. Session initiation by a primary (host) logical unit involves the following operations:

1. The application issues a VTAM macro, OPNDST (Open Destination), that specifies the symbolic name of a secondary logical unit.
2. Logical unit services receives control as a consequence of the OPNDST macro and interacts with the control point to cause the symbolic name to be mapped to the address of a particular secondary logical unit.
3. Logical unit services sends an SNA command, BIND, to the secondary logical unit. This action notifies the secondary logical unit of the desire to establish a session and communicates certain parameters that are to be in effect for the session. The secondary logical unit has the option of accepting or rejecting the BIND command.
4. Assuming that the BIND command was accepted by the secondary logical unit, the host logical unit services then notifies the control point that the session has been successfully bound.

The termination of a session by a primary logical unit involves the following steps:

1. The application issues a VTAM macro, CLSDST (Close Destination) to signal logical unit services that the session is to be ended.
2. Logical unit services uses the CLEAR command to purge all data flowing on the session and then sends another SNA command, UNBIND, to the secondary logical unit to indicate that the session has been concluded.
3. Logical unit services notifies the control point that the session has been successfully unbound.

For sessions that are being requested by secondary logical units, host logical unit services also support session mode specification. A mode name can be supplied on the INITIATE command that is sent out by the secondary logical unit. This mode name will cause host logical unit services to utilize a predefined set of parameters that the secondary logical unit desires for the ses-

sion. The host application can interface with logical unit services through the VTAM INQUIRE macro to determine these mode-related parameters. The application can therefore examine this information before sending out the final session specification via the BIND command.

### **Presentation services**

In addition to logical unit services, each logical unit may have various presentation services associated with it. Presentation services is a generic term for those functions that support the specific requirements of end-user-to-end-user communication. As such, sets of presentation services functions are selected and assigned at each establishment of an SNA session between pairs of logical units. Depending upon the set of presentation services selected for a given LU/LU session, session characteristics such as the degree of device independence, the controls on the data flow, and other parameters are determined. In addition, presentation services provide data transformations between the end-user interface and the transmission subsystem.

A given set of presentation services is called a presentation class. The present VTAM implementation supports a single such SNA function group, the RECORD presentation class.

#### **requests/ responses**

Before discussing specific details about RECORD, it is desirable to introduce some additional SNA concepts. All units of information that flow between end users must be categorized as requests or responses. In SNA terms, a request can be either data or a command. The commands that can be instigated by the user, either directly or indirectly, serve to affect the flow of data within the LU/LU session. A response is a level of acknowledgment sent out to a previously received request. A response, which may be asked for and sent out selectively, can contain sense information about a particular request to which it is related.

#### **chaining**

Requests may have an important attribute associated with them, known as chaining. Through the use of chaining, a logical relationship among request units may be expressed. This relationship allows the logical units to treat an entire chain of requests as an entity, using the chain as the unit of recovery. For example, a positive response to each element of a chain is not required. It is sufficient from a recovery standpoint to send only error responses, if any, and reserve a positive response for only the final element in a chain. In this manner, the sender can be notified of the successful or unsuccessful transmission of an entire chain with minimal network overhead for acknowledgments.

The RECORD presentation class defines a set of presentation services that allow the user to control directly the generation of requests, responses, and chains of requests. In addition, RECORD provides capabilities for generating commands to control the LU/LU data flow.

RECORD may be thought of as a fundamental, low-level set of presentation services. The user of RECORD has a broad range of SNA functions explicitly available. For this reason, RECORD is particularly well-suited for use by subsystems such as IMS and CICS.

The user of the RECORD presentation class has three macros, SEND, RECEIVE, and SESSIONC, that are of primary importance in sending and receiving units of information. A brief overview of the capabilities provided by each will illustrate their functional significance.

Through the SEND macro, an end user may specify that a unit of information is to be conveyed to a particular secondary logical unit. Key SNA parameters associated with the SEND macro include the following: (1) designation of the unit as a request or a response, (2) for a response, identification of the response type and specification of sense information, (3) for a request, specification of chaining information that identifies this unit as a first, middle, last, or only element in a chain, and (4) for a request, designation of the unit as user data or control information. This control information is expressed via a set of specific parameters that allow the user to control the full duplex flow of information between the logical unit pair that is in session.

The RECEIVE macro provides the user an interface to obtain units of information that have arrived at the host logical unit. In terms of SNA-related parameters, a RECEIVE operation may specify that it is to be satisfied only by user data, only by control information, or by either.

The user of the RECORD interface has very powerful capabilities available through the SESSIONC macro. This macro allows the user to send out special commands that affect the operation of the entire session, including all data and control flows. The SNA functions that can be invoked through the SESSIONC macro include the following commands:

1. **START DATA TRAFFIC** — At the user's option, the flow of data and control information may be either allowed or inhibited at the time a session is established. If normal traffic flows are inhibited, it may be because the user wishes to perform some further initialization or resynchronization procedures before

entering normal operation. The START DATA TRAFFIC command is used to signal that all inhibitors to normal flow have been removed.

2. CLEAR—This operation is used to purge requests and responses that may be flowing within a session. CLEAR also inhibits further flow of normal traffic until a START DATA TRAFFIC command is issued. The CLEAR command is useful in resetting data flows to a known state prior to recovery/resynchronization operations.
3. SET AND TEST SEQUENCE NUMBERS—The general concept and function of sequence number generation and checking will be described in some detail in the transmission subsystem section. These sequence numbers are essential to achieving transmission assurance between pairs of logical units. The SET AND TEST SEQUENCE NUMBERS command allows the user to selectively test and reset these values during the course of recovery/resynchronization procedures.

The application program interface presented to the user of VTAM is, of course, much broader than the limited set of macros and parameters that have been discussed here. This overview, however, has highlighted the SNA aspects of that interface.

### **Transmission subsystem**

VTAM provides transmission subsystem functions for the host node. These functions will be described in terms of three layers: transmission control functions, path control functions, and data link control functions.

*Transmission control.* Transmission control provides end-to-end transmission of request/response units between NAUS. VTAM provides one end of the function for request/response unit transmissions between an NAU within the host and an NAU within another node. The portion of transmission control that provides end-to-end data transmission is called the connection point manager and will be discussed in more detail later.

The other major function of VTAM transmission control is called session control. It supports startup, shutdown, and recovery control for the request/response unit flow between NAUS. Within SNA, this control responsibility is always associated with the primary NAU. Only host nodes support primary NAUS.

Note that the SNA structure provides for another set of transmission control functions called network control.<sup>8</sup> The VTAM transmission control layer implementation provides no such functions.

The term session, as noted earlier, refers to the logical transmission path between two NAUs. Session control functions are provided under the direction of logical unit services. Session control includes the functions that establish a transmission path between a primary NAU (within the host node) and a secondary NAU. They include establishing network addressability between the primary transmission control (within the host) and the secondary transmission control (within another node). This operation requires communication with the secondary control to identify the network addresses of the two NAUs that are entering into session. This communication is accomplished via the BIND command sent by primary logical unit services. In addition, both ends must establish the initial state information for the transmission-level session. This operation includes initializing sequence number counters for the request unit flows. Sequence numbering (discussed later) is used to provide end-to-end transmission assurance and request/response correlation. Once this process is complete, the transmission of request/response units between the two NAUs may proceed under control of the connection point manager.

**session  
control**

Session control provides functions for resynchronizing the logical transmission path after a failure. This provision includes the ability to set sequence number counters to the appropriate restart values and to reinitialize request/response correlation information.

Session control also provides the function of disestablishing network addressability once the session is no longer required by the two NAUs. This function is accomplished via the UNBIND command sent by primary logical unit services.

As introduced earlier, the connection point manager provides end-to-end transmission of request/response units between NAUs by supporting two request unit flows: one from the primary NAU to the secondary NAU and one from the secondary to the primary. In addition, each request unit flow has an associated response unit flow. Thus, the VTAM connection point manager supports the transmission of request units to a secondary NAU and the reception of associated responses. It also supports the reception of request units from a secondary NAU and the transmission of associated responses. The following discussion will treat these two separately, under the headings of outbound request processing and inbound request processing.

**connection  
point  
manager  
functions**

Outbound request processing supports the transmission of request units from an NAU within the host to an NAU within another node. It also supports the reception of associated response units. The major functions are discussed briefly below.

**outbound  
request  
processing**

A request unit header is generated based on parameters that were passed to the connection point manager with the request unit. The parameters include such things as request unit chaining indicators and response indicators. The request unit header together with the request unit itself forms the basic information unit.

A request unit sequence number is generated. It will be a unique, sequentially assigned number used to provide transmission assurance and request/response correlation for this flow.

Optionally, depending upon the characteristics of the secondary NAU, request unit scheduling will be performed. The particular form of scheduling utilized by SNA for NAU-to-NAU transmissions is referred to as *pacing*. If the scheduling algorithm is satisfied, request unit processing will proceed. If not, the request will be queued awaiting a pacing response. Once the pacing response is received, processing will proceed as discussed below.

The address of the basic information unit, the associated sequence number, the origin and destination network addresses, and other parameters will then be passed to path control for routing to the destination connection point manager. When a response unit is received in the host, it will be passed to the connection point manager (from path control) as a basic information unit with appropriate parameters, including a response sequence number. This information will be passed to the NAU. The sequence number may be used by the NAU to correlate with the record of an associated request for which a definite response was requested. (This correlation is possible because response units have an associated sequence number that is identical to the one for the corresponding request unit.)

**inbound  
request  
processing**

Inbound request processing supports reception of request units and the transmission of response units. When a basic information unit is received from path control, the request/response unit header is interpreted. If it is a response header, it is processed as discussed under outbound request processing. Otherwise, the basic information unit, with associated parameters, is processed as discussed below.

The request unit sequence number is checked to ensure that its value is one greater than the last-received request unit for this flow. If the sequence number is incorrect, exception processing will be performed and will result in the discarding of the current request unit and any succeeding request units until higher-level (subsystem or application program) recovery procedures can be initiated. If the sequence number is correct, the request unit, its sequence number, and other parameters will be passed to the NAU (the user of the transmission subsystem). (The user of trans-

mission services is either a logical unit or the system services control point. NAU-to-NAU communication within the host is performed directly without using the transmission subsystem.)

When the NAU wishes to send a response to a particular request unit, it passes a corresponding function request to the transmission subsystem (connection point manager). Based on the function request and associated parameters, the connection point manager will generate a basic information unit of the appropriate type. For a positive response, it consists only of a response header. For a negative response, it consists of a response header plus sense information. The address of the basic information unit, the associated response sequence number, the origin and destination network addresses, and other parameters will then be passed to path control for routing to the destination connection point manager.

To summarize, transmission control provides two major categories of function. One is the primary session control functions, which support startup, shutdown, and recovery of the logical transmission path between two NAUs. The other is the connection point manager functions, which utilize the logical transmission path to support duplex request/response unit flows between two NAUs. Within this context, one of the NAUs is always a VTAM NAU (a logical unit or the system services control point). The other is an NAU within another node (communications controller node, cluster controller node, or terminal node).

**transmission  
control  
summary**

*Path control.* VTAM/SNA path control provides both the end-to-end routing and the node-to-node routing of basic information units. The term "end-to-end routing" refers to the routing of basic information units between two transmission control elements (or connection point managers). The term "node-to-node routing" refers to the routing between the physical nodes that make up, or support, the end-to-end path.

End-to-end routing will be discussed in terms of outbound processing and inbound processing. Outbound processing supports the end-to-end routing of basic information units from a VTAM connection point manager to a connection point manager within another node. This operation involves passing the basic information unit and associated parameters to the particular node-to-node router that supports the next node in the path to the destination.

**end-to-end  
routing  
functions**

Inbound processing involves accepting basic information units and routing them to the appropriate connection point manager. A basic information unit is received from node-to-node path control with parameters that include origin and destination network addresses and the associated request or response sequence

number. An internal routing mechanism is used to pass the basic information unit to the VTAM connection point manager for the particular session.

**node-to-node  
routing  
functions**

Node-to-node routing involves routing between the host node and each of the attached cluster controller nodes and communications controller nodes. The functions differ depending upon the capabilities of the particular node type.

For routing to and from locally attached cluster controller nodes, the host functions include address transformation and Format 2 transmission header generation and interpretation. For outbound processing, the origin and destination network addresses are converted to local addresses. The Format 2 transmission header is generated, and the basic transmission unit, consisting of a transmission header plus a basic information unit, is then passed to data link control for transmission to the cluster node. For inbound processing, the transmission header is interpreted, the local addresses are converted to network addresses, and the basic information unit is passed to end-to-end path control with appropriate parameters.

For routing to and from communications controller nodes, the host functions include Format 1 transmission header generation and interpretation, as well as node-to-node flow control. For outbound processing, the Format 1 transmission header is generated and prefixed to the basic information unit to form a basic transmission unit. The basic transmission unit is then passed to data link control or placed on a request queue, depending upon the flow control status of the communications controller. For inbound processing, the transmission header is interpreted, and the basic information unit is passed to end-to-end path control with appropriate parameters.

The discussion of path control is thus completed. To summarize, path control provides two levels of routing function. One level is session-related and involves routing complete basic information units to the connection point manager for the particular session. The second is next-node-related and involves routing between the host node and the next node in the path.

*Data link control.* SNA data link control provides the physical node-to-node transmission functions, including the control functions required to support startup, shutdown, and recovery of the physical transmission path between two nodes. VTAM, in conjunction with the I/O Supervisor component of the operating systems, provides data link control functions for the host node. VTAM provides the primary station support required to utilize the System/370 channel as an SNA data link. The I/O Supervisor provides the channel scheduling functions and the channel I/O

initiation (Start I/O processing) functions. The remainder of this discussion will focus on the primary station functions supported by VTAM.

VTAM data link control supports two types of functions: transmission functions and control functions. The transmission functions are those required by path control as a user of data link control. They support protocols for the transmission and reception of information records. The control functions are those required by physical unit services. They support the configuration services and maintenance services functions for the data links. Both types of functions are discussed in more detail below.

As indicated above, the control functions support configuration services and maintenance services for the data links. Each data link may be viewed as a point-to-point communications channel between a primary station and a secondary station, with each secondary station (cluster or communications controller) having a unique channel-unit address. The configuration services control functions support startup, shutdown, and recovery of communications with the secondary stations. The form of support is somewhat different for communications controllers and cluster controllers. The major difference is in the area of startup, which establishes the logical connection between the primary and secondary stations. Support for cluster controllers allows the logical connection to be initiated via an asynchronous signal from the secondary station, whereas for communications controllers it must be initiated by the primary station. This support for cluster controllers was provided as part of an overall requirement to facilitate dynamic transitions between off-line and on-line modes of operation. Below is a summary of the major functions supported in the area of configuration services control:

**control  
functions**

- Determine the operational status of the physical interface and the initialization status of the secondary station.
- Request the initiation of initial program load (IPL) procedures, if required, based on the initialization status (communications controllers only).
- Request the initiation of activation procedures, based on asynchronous status from the secondary station (cluster controllers only).
- Establish the link-level connection, including required link-level transmission parameters (cluster controllers only).
- Report unrecoverable link or station failures, thereby initiating network-level recovery procedures.
- Disestablish link-level connection, thereby allowing secondary node to enter off-line mode of operation (cluster controllers only).

In addition to the above, the control functions (of data link control) provide support for maintenance services. These services were mentioned previously under physical unit services. They include the following:

- A trace of all basic transmission units sent to or received from a secondary station.
- Recording of I/O error information.

**transmission  
functions**

The remainder of this section will describe the transmission functions of data link control. These functions support path control for the purpose of transmitting information records to, and receiving records from, a path control element in another node. VTAM provides the primary station functions required to support both channel-attached communications controllers and cluster controllers for this purpose.

VTAM utilizes normal System/370 channel I/O programming techniques to accomplish the transmission functions. Actual data transfer is accomplished using channel programs consisting of one or more WRITE or READ commands, depending upon the direction of the transfer. These commands affect communication with the controller only and do not relate to any devices attached to the controller.

Asynchronous Attention status is utilized by the secondary station to indicate that it has records available for transmission to the host. The details of these and other techniques are described in VTAM publications.<sup>4</sup> Below is a brief description of the functional capabilities provided:

- Transfer of one or more basic transmission units to the secondary station.
- Temporary suspension of outbound data transfer due to buffer depletion in the secondary station (inbound data transfer proceeds independently).
- Resumption of outbound data transfer upon signal from the secondary station.
- Transfer of one or more basic transmission units from the secondary station to the host (inbound data transfer).
- Support of alternating acknowledgment procedure and other transfer verification procedures for assuring transmission integrity.
- Error recovery (retry) procedures for both inbound and outbound data transfer.

In summary, data link control supports two internal users with two different types of functional capabilities. One user is physical unit services. The control functions are provided to assist physical unit services in managing startup, shutdown, and net-

work recovery of the data links. The other user is path control. The transmission functions provide path control the capability to transmit and receive basic transmission units, without being exposed to the channel link procedures and associated concerns for transmission integrity.

## Summary

Even before the advent of the Systems Network Architecture, there were strong motivations for a new teleprocessing access method. The introduction of VTAM provided both sharing of network resources and direct control of communications. VTAM also represented an optimization for the recent enhancements in CPU technology, e.g., dynamic address translation and multiprocessing. In conjunction with the queued control capabilities of TCAM, VTAM formed a base for consolidation and further enhancement of IBM's communications facilities.

The technological advances that made distributed function possible introduced new requirements. A new network architecture, SNA, was needed to combine software and hardware components into an efficient, coherent system. The function distribution and protocol definitions implied by SNA offered a further motivation for the creation of a new access method such as VTAM. Clearly, no existing teleprocessing access method possessed the functional capabilities that SNA demanded within the host node.

The SNA functions realized in the host through VTAM include: overall network management, done by the system services control point; management of the host node, achieved by the physical unit; support for end-user sessions, provided by logical unit services; data transformations necessary for end-user communication, done by presentation services; and data transmission, accomplished via the transmission subsystem.

Just as SNA has provided coherency and long-range direction to the overall communications environment, so has VTAM established the long-term base for network access and control.

## CITED REFERENCES

1. *Advanced Function for Communications System Summary*, No. GA27-3099, IBM Corporation, Data Processing Division, White Plains, New York.
2. *Introduction to VTAM*, No. GC27-6987, IBM Corporation, Data Processing Division, White Plains, New York.
3. *VTAM Concepts and Planning*, No. GC27-6998, IBM Corporation, Data Processing Division, White Plains, New York.
4. *Introduction to VTAM Logic*, No. SY27-7256; *DOS VTAM Logic*, No. SY27-7262; *OS/VS1 VTAM Logic*, No. SY27-7257; *OS/VS2 VTAM*

*Logic*, No. SY28-0621, IBM Corporation, Data Processing Division, White Plains, New York.

5. J. H. McFadyen, "Systems Network Architecture: An overview," in this issue.
6. *Systems Network Architecture General Information*, No. GA27-3102, IBM Corporation, Data Processing Division, White Plains, New York.
7. J. P. Gray and C. R. Blair, "IBM's System Network Architecture," *Datamation* **21**, No. 4, 51-56 (April 1975).
8. P.G. Cullum, "The transmission subsystem in Systems Network Architecture," in this issue.
9. W. S. Hobgood, "The role of the Network Control Program in Systems Network Architecture," in this issue.
10. R. A. Donnan and J. R. Kersey, "Synchronous data link control: A perspective," *IBM Systems Journal* **13**, No. 2, 140-162 (1974).

Reprint Order No. G321-5027