Public data networks are now being designed and implemented to handle the expansion of data communications. The interaction of Systems Network Architecture (SNA) and the international standards now being developed is discussed. A provisional architecture model is used as the basis for discussion, and SNA is compared to each level of the model.

SNA and emerging international standards

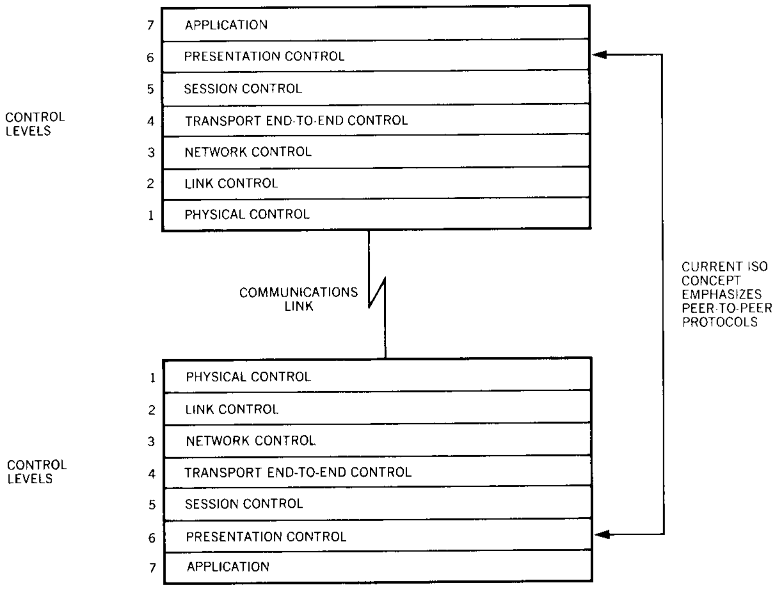
by F. P. Corr and D. H. Neal

The successful adaptation of telephone networks to data transmission has fostered two decades of rapid growth in data processing systems using telecommunications facilities. To augment the telephony base, new public networks are now being designed specifically for data communications. The international scope and technological aspects of these new services have been described elsewhere in this issue by Halsey, Hardy, and Powning.[1]

As a sequel, this paper examines the new services from the perspectives of emerging international standards and IBM's Systems Network Architecture (SNA). A provisional architecture model[2] is used as a framework for discussion, illustrating the respective roles played by public data networks, the emerging interface standards, and SNA. The roles of SNA and the new interfaces differ, and in some cases, overlapping functional capability occurs. However, there is a complementary interworking of SNA with the public data networks employing the new interfaces.

Beginning with a broad outline of the provisional seven-layered architecture model, the paper proceeds to a level-by-level comparison of SNA and the model. For each of the first three levels of the model, data network interface standards are used as implementation examples. Having depicted the architectural consistency, by level, of the model, the new interfaces, and SNA, the

Figure 1 ISO provisional architectural model



paper then deals in more detail with three specific interfaces: X.21, HDLC, and X.25. (Some of this interface discussion can be followed without prior knowledge of the three standards; however, background reading would be helpful in some instances; for example, see Chapters 11 and 17 of Cypser.[3]) In the case of X.25, several possible SNA interface techniques are discussed, and an existing SNA implementation of X.25 is described. The final section attempts to discern how the long-term objective of full open-system interconnection might be achieved.

ISO's provisional architecture model: A framework

In the International Organization for Standardization (ISO),[4] a provisional architecture model (Figure 1) has been recently introduced. Intended ultimately to provide a basis for open-system interconnection, the model consists of seven functional or control levels.[2] An example of the first level is the CCITT[5] interface recommendation X.21 for circuit-switched and leased-circuit services.[6] The second level is exemplified by ISO's high-level data link control (HDLC) procedures.[7] The CCITT interface recommendation X.25 for packet-switched services is an example of an implementation at all of the first three levels.

No international standards for the four levels above X.25 have yet been agreed upon. Of course, to implement any working system it is necessary to provide the functional equivalents of all levels even in the absence of the higher-level standards. Individual manufacturers and users have developed equivalent levels. This paper describes how one such manufacturer-developed logical structure, IBM's SNA, matches the ISO provisional model. X.21, X.25, and SNA map into certain layers of the ISO model; general architectural compatibility of all three concepts is manifest.

Since X.21, HDLC, and X.25 encompass only the lower levels of the ISO model, they do not, of themselves, assure full communications capability between any two end devices. An oversimplified analogy to X.21 and X.25 is representing them as standard envelopes in a postal system; other, more complex agreements, like language, are needed to completely specify user-to-user postal communications.

While standards for the higher levels of the ISO model are still under development, various interim measures may be taken to provide their functional equivalents. For example, the higher levels of SNA have been used in conjunction with adaptations (described in a subsequent section of this paper) to some IBM products to ensure full SNA-to-SNA communications through X.25 interfaces on two specific public packet networks. Another example is the set of CCITT recommendations, X.3, X.28, and X.29, that adapt packet-switched networks to some start-stop equipment.

The following section considers each of the seven ISO model levels and how X.21, HDLC, X.25, and SNA map into them.

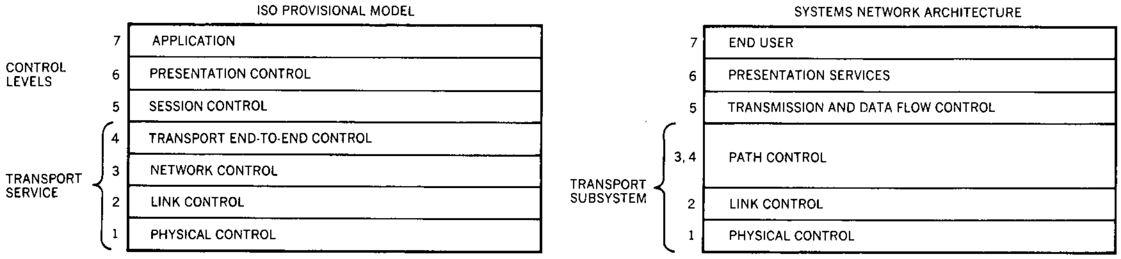
SNA and the ISO provisional model: A synopsis

In the ISO provisional model, seven control levels are posited for each user system connected to a data communications network. They should permit full communication of intelligible information through the network once the required standards within each layer are defined by ISO.

Current ISO work emphasizes clear definition of protocols and message formats for peer-to-peer communications; that is, for control coordination between equivalent layers in different user systems. According to the present concept, adjacent layers within an individual user system would have only loosely identifiable boundaries. Formats and protocols for adjacent-layer communications within a system would not and need not be standardized.

The seven ISO levels have SNA counterparts (Figure 2). This functional correspondence ensures structural compatibility of SNA

Figure 2 Control levels of the ISO provisional model and their SNA counterparts



and the model. It does not, of itself, guarantee bit-by-bit agreement of formats and protocols. That is, structural matching alone does not ensure the ability to communicate. As already noted, some of the higher levels of the provisional model have no detailed standards at this time.

control level 1

Level 1, or the physical control layer,[8] is the physical interface between system components generally called data terminal equipment (DTE) and data circuit terminating equipment (DCE).[9] Level 1 defines the electrical characteristics and signaling needed to establish, maintain, and disconnect the physical connection between DTE and DCE or between two DTEs. Current examples of Level 1 standards include CCITT recommendations V.24 for telephone networks and X.21 for data networks. In addition, X.21 is Level 1 of CCITT recommendation X.25.

ISO's Level 1 is functionally equivalent to the corresponding level of SNA. The SNA products currently implement V.24 (and V.25 in some cases where automatic calling on switched telephone service is needed); the implementation of X.21 in SNA should prove architecturally straightforward as X.21 services become available.

control level 2

Level 2, or the link control layer, provides transmission over a single data link between two user systems. Current examples of standards include the Advanced Data Communication Control Procedures (ADCCP) in the United States, and the ISO HDLC.

In SNA, Synchronous Data Link Control (SDLC) is a subset of HDLC. Details on how SDLC implements HDLC appear in a later section.

control level 3

Level 3, or the network control layer, provides control between two adjacent nodes. For example, the packet level of X.25 provides network control between an end-user node and an access node of a public packet network. An important aspect of Level 3 in X.25 is the virtual circuit. A single X.25 interface can control up to 4095 virtual circuits, each of which could, in principle, route data packets to a different destination. Path control in SNA encompasses functions similar to those in Level 3 of the ISO model. Level 3 protocols in SNA function on a DTE-to-DTE basis, whereas those in X.25 apply to communications between DTE and network.

control level 4

Level 4, or the transport end-to-end layer, provides control from user node to user node across a network. Included on a user-to-user basis are addressing, data assurance, and flow control. At present, ISO standards for Level 4 do not exist. The top of Level 4 provides a functional service interface to Level 5 that is independent of the network types employed at the lower four levels.

The functions of Level 4 are included in the path control layer of SNA. In SNA, the composite of path, link, and physical control layers is termed the transport subsystem, analogous to ISO's transport service (see Figure 2).

control level 5

Level 5, or the session control layer, establishes, maintains, and terminates logical connections for transfer of data between processes (end users in SNA terminology). ISO standards for Level 5 do not yet exist.

In SNA, the functions of ISO Level 5 are provided by transmission control and data flow control. Transmission control activates and deactivates SNA sessions between end users. Data flow control enforces dialog control between end users.

control level 6

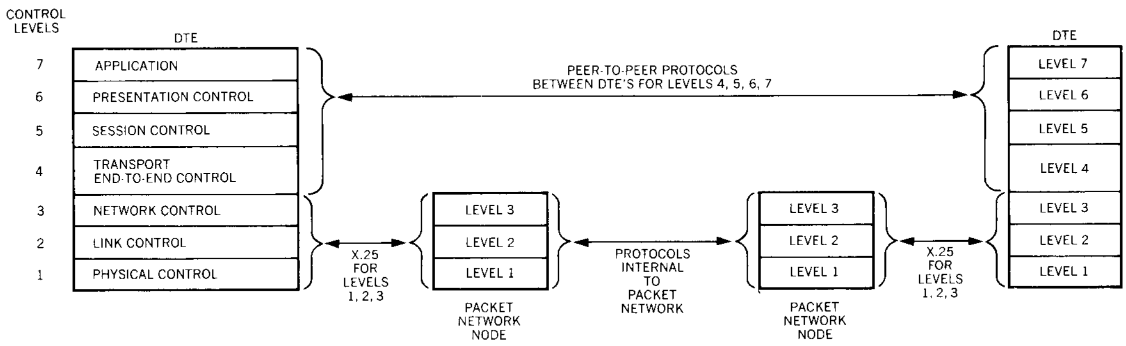
Level 6, or the presentation control layer, provides data formats and any needed data transformations. Data formats might include those required for video display screens or printers. Code translations would also be included in Level 6. Some Level 6 functions are found in the set of CCITT recommendations, X.3, X.28, and X.29.[10] ISO standards for Level 6 are still being developed.

In SNA, the functions of ISO Level 6 are provided in presentation services. These SNA services may include transformations (such as data compression), additions (such as column headings for displays), and translations (such as program commands into local terminal language). Examples of presentation services in IBM software can be found in the Customer Information Control System (CICS) and the Information Management System (IMS).

control level 7

Level 7 is the application layer. ISO standards for this level are not yet defined. In SNA, the end user provides this functional level. The end user is a person or process that wants to use an SNA network; an end user may be internal or external to an SNA node. External end users may be human operators; internal end users may be application programs resident in a node. As a user of SNA, an end user is not strictly a part of SNA.

Figure 3 X.25 and its usage in terms of the ISO architectural levels



In the following section these seven ISO control levels are used as the framework for discussing interfaces to public data networks.

Interfaces to public data networks

Both the nodes inside a public data network and the users' data terminal equipment may be considered as structured according to the ISO model. Figure 3 illustrates the X.25 case where the DTEs have the full ISO-model complement of seven levels, and the network nodes have only the three levels needed to provide X.25.

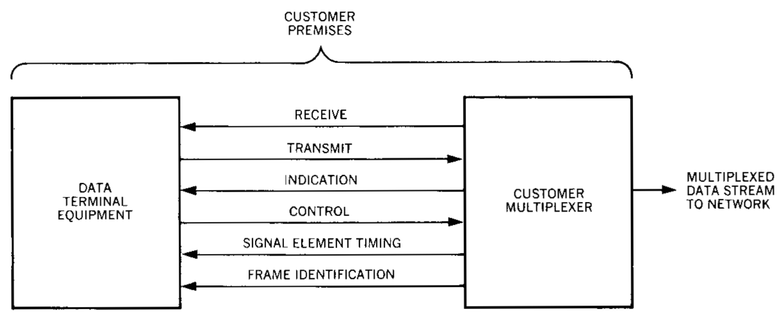
A similar figure could be drawn where the packet network offered a data-link control interface, HDLC for example. In this case, the network nodes would encompass only the first two architectural levels. For leased-circuit services using X.21, the equivalent to Figure 3 would show only one level at each network node.

In the rest of the paper, the focus becomes more narrow in two respects. First, since the protocols above Level 3 are not directly involved in the network-to-DTE interface, the four upper levels are no longer considered. Second, of the many existing interfaces to public data networks, only three will be emphasized: X.21 at Level 1, HDLC at Level 2, and X.25 at Levels 1, 2, and 3. All three are existing, detailed, international standards. Although the three are not the only standards used for public data network interfaces, they are convenient ones for SNA with its reliance on synchronous transmission and bit-oriented protocols.

SNA and X.21: a Level 1 interface

As a Level 1 standard,[11] the X.21 interface can be used for point-to-point, leased private-line services. New driver and terminator circuits, connectors, and cables will be required in DTE to take advantage of improved technical characteristics such as greater noise immunity and higher bit rates offered by X.21. However,

Figure 4 Proposed new interface for multiplexed X.21 link



the addition of this new silicon and copper does not represent changes to SNA itself; SNA products with new interface adapters will operate with X.21 leased services without architectural modification.

Since SNA already accommodates circuit-switched services, the necessary SNA extensions for X.21 switched service are minor. Needed extensions include some small additions to the SNA inoperative command.[13] These additions can accommodate the new call-progress information provided by X.21. For example, with X.21, the business equipment (DTE) will be informed of the reasons why a call could not be completed. Upon such notification, appropriate recovery actions could be invoked by the DTE.

Enhancements to X.21, under study by CCITT, include:

- Multipoint leased-circuit services
- A multiplexed link for leased-circuit and circuit-switched services
- A mini-interface like X.21 for use on telephone networks

Based on 1978 proposals,[14] it appears no architectural changes to SNA for leased-circuit, multipoint X.21 service are required.

The multiplexed X.21 link has been proposed to provide multiple logical channels on a single communications link. Its objective is to lower cost through the ensuing reduction in the number of access lines at a large information-processing installation. In one proposal, a special DCE called a customer multiplexer would be provided.[15]

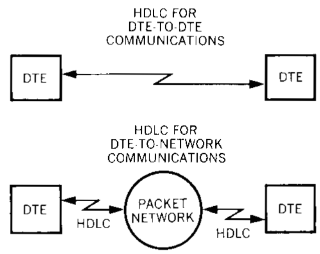
In the proposed interchange circuits of the customer multiplexer (Figure 4), the byte timing circuit of X.21 is replaced by a frame identification circuit. This circuit would have to be implemented by an SNA DTE, but this would not be a fundamental architectural concern.[16]

Table 1 Comparison of two basic characteristics of SDLC and HDLC

	Primary-secondary asymmetry	*Polling*
SDLC normal response mode	yes	yes
HDLC normal response mode	yes	yes
HDLC asynchronous response mode	yes	not needed*
HDLC asynchronous balanced mode	avoided	not needed

*A secondary station may send data without having been polled. However, as in all HDLC modes, it is possible to use polling to serve purposes other than permission to send data; an example might be a recovery procedure.

Figure 5 An example of Level 2, HDLC in two roles



Another proposed enhancement based on X.21 is a mini-interface for operation on telephone networks.[17,18] If certain technical problems, such as adapting to half-duplex terminals, can be solved without significant deviation from X.21, applications operating on both data and telephone networks could be enhanced.

The first X.21 public networks will become operational in 1979. According to a recent CCITT survey,[19] eight countries will offer X.21 services by the end of 1980 and eight others at some time after 1980. Thus, the first practical usage of X.21 on a large scale is about to occur. The nature of X.21 enhancements will undoubtedly be strongly influenced by this forthcoming experience.

SNA and HDLC: a Level 2 interface

Level 2 procedures provide transfer of data between two DTEs communicating directly with each other across a circuit-switched network or on a private line. Alternatively, in the case of packet or store-and-forward networks, a Level 2 procedure is implemented between the network and each individual DTE (Figure 5). In either case, three basic data-link control functions are needed: distinguishing start- and end-of-information fields, addressing of sender and receiver, and providing recovery from errors. A flexible and reliable data-link control procedure for meeting these needs, HDLC can be used between two DTEs or between each DTE and a public packet network.

Table 1 provides a summary comparison of SDLC and three modes of HDLC. SDLC in SNA products conforms to the HDLC unbalanced normal class of procedure. In this procedure, normal response mode and normal disconnected mode in both SDLC and HDLC have two basic characteristics: control of the link by a station-designated primary and the existence of a polling protocol. Normal response mode is used on multipoint lines, where polling by

Figure 6 Required frame structure for SDLC and all HDLC classes with nonextended control field

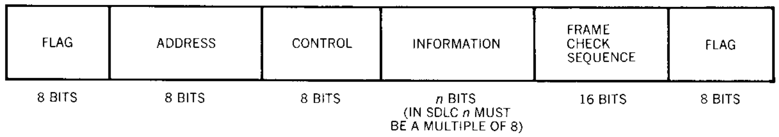


Table 2 Basic repertoire of commands and responses for HDLC unbalanced, normal response mode

	Commands from primary		*Responses from secondary*	
	HDLC	*SDLC*	*HDLC*	*SDLC*
Information Frame	X	X	X	X
Receive Ready	X	X	X	X
Receive Not Ready	X	X	X	X
Set Normal Response Mode	X	X		
Disconnect	X	X		
Command Reject			X	X
Unnumbered Acknowledgment			X	X
Disconnect Mode			X	X

the primary station creates disciplined line sharing. It is also useful where DTEs have no need for two-way simultaneous operation.

Another HDLC mode in the unbalanced class of procedure, asynchronous response mode, avoids polling but retains the primary-secondary relationship among the data stations.

In HDLC, a second class of procedure, the asynchronous balanced class, avoids both the primary-secondary asymmetry and the requirement to poll for data transmission. As its prime advantage, the asynchronous balanced mode of HDLC gives each station on a point-to-point link equal capability to initiate error recovery. The HDLC asynchronous balanced mode was produced by ISO subsequent to its development of the asynchronous response mode of the unbalanced class. In many applications, asynchronous unbalanced mode appears to have no major advantage over the newer asynchronous balanced mode.

SNA products using SDLC conform to the HDLC unbalanced normal class of procedure by adherence to the ISO standard on frame structure and by implementing the basic repertoire of commands and responses. The HDLC frame structure is specified by ISO document IS-3309[7] and is common to all HDLC classes. Thus, SDLC conforms to the frame structure of all HDLC classes (see Figure 6) but conforms to the required commands and responses for only the unbalanced normal procedures (see Table 2).

In summary, HDLC can be used for data-link control both in DTE-to-DTE and DTE-to-network communication. SDLC conforms to HDLC in frame structure and in the basic repertoire of commands and responses for unbalanced normal response mode. HDLC is further discussed below in the context of X.25.

SNA and X.25: a three-level interface

Recommendation X.25 was first published by the CCITT in 1976;[6] a revised version appeared in 1978.[10] Further changes and enhancements are under study for another revision in 1980. All three versions, while differing in details, may be considered in the context of the first three control levels of the ISO provisional model.

X.25 defines a three-level protocol between a DTE and a public packet network. X.25 thus constitutes an implementation example of the corresponding Levels 1, 2, and 3 in the ISO model of Figure 1. For Level 1 of X.25, either X.21 or a V.24-compatible interface is recommended. Level 2 of X.25 may be LAP,[20] a special protocol developed by the CCITT, or LAP "B", a subset of the HDLC asynchronous balanced mode. Level 3 of X.25 comprises the packet format and control procedures for the exchange of control and data packets on a virtual channel between a DTE and a packet network.[21]

At each of its three levels, X.25 is a protocol between a DTE and a packet network.[22] In contrast, SNA protocols ensure information and control exchange at six levels between two DTEs. In spite of this fundamental difference in role, both X.25 and SNA can be treated in terms of the ISO model.

The layered structure of an X.25 interface in SNA must follow the conceptual configuration of Figure 3. By definition, the packet network as perceived through the X.25 interface has three levels. The SNA DTEs have seven levels, the lower six properly SNA, and Level 7, the application layer, a user of SNA. It follows that the set of required protocols are those depicted in Figure 3, where it is assumed that both DTEs in communication use the X.25 interface.[23] Since the protocols internal to the public packet network are not directly visible to the user at the X.25 interface, they may, in theory, take any form that provides adequate performance.

The choice of Level 1 protocol in X.25, either X.21 or the V.24-compatible version, is stipulated by the public network design. The current implementation of an SNA-X.25 interface (discussed below) to each of two networks uses the V.24 version of X.25 Level 1. However, as explained earlier, SNA is compatible with the X.21 approach as well.

At Level 2, a single data-link control is implemented for each X.25 interface. This may be either LAP or LAP "B". According to the CCITT survey published in November, 1978,[19] there are 11 countries planning LAP "B" and three that will have both versions. The existing SNA implementation of X.25 uses LAP to match the specification for the two networks to which it provides an interface. However, provision for LAP "B" would not be difficult in the current SNA implementation; moreover, the trend toward LAP "B" observed in the CCITT survey is sound because it should lead to HDLC compatibility.

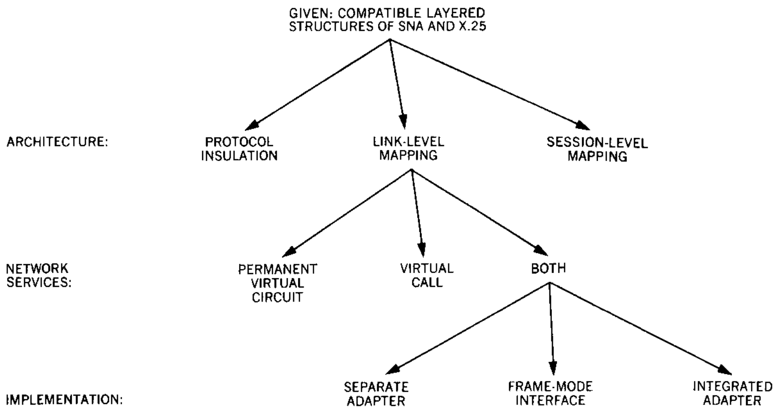
The conceptual base for Level 3 of X.25 is the virtual circuit. When an X.25 virtual call is set up, one of 4095 virtual circuits is assigned to the call. Interleaved packets are used to set up and clear virtual calls as well as to transfer data packets on individual virtual circuits. Since each virtual circuit has its own flow control, any one circuit user can be prevented from excessively degrading service to other circuit users. Another important user advantage in virtual circuits is that a central processing site may communicate with many remote terminals by means of a single X.25 interface. In effect, a single real circuit can be treated as multiple virtual circuits. Although the theoretical limit of 4095 simultaneous virtual calls will not be achieved in practice,[24] the multiple-call capability wrought by the virtual-circuit concept is useful.

To achieve the advantages of virtual circuits at Level 3 of X.25, several diverse implementations of an SNA-to-X.25 interface are possible. Figure 7 is a simplified overview of implied design decisions related to architecture, network services, and implementation technique.

From an architectural perspective, the many possible designs of an SNA-X.25 interface may be typified by three: protocol insulation, link-level mapping, and session-level mapping. Protocol insulation implies encompassing every SNA frame completely within the data fields of X.25 packets. Disadvantages of protocol insulation include duplication of HDLC procedures in X.25 and SNA and the excessive number of X.25 packets generated by polling frames.

Although protocol insulation can be quickly rejected, choosing between link- and session-level architectural mapping is more subtle. Session-level mapping implies a one-to-one or many-to-

Figure 7 Simplified decision tree for design of the X.25 interface in SNA

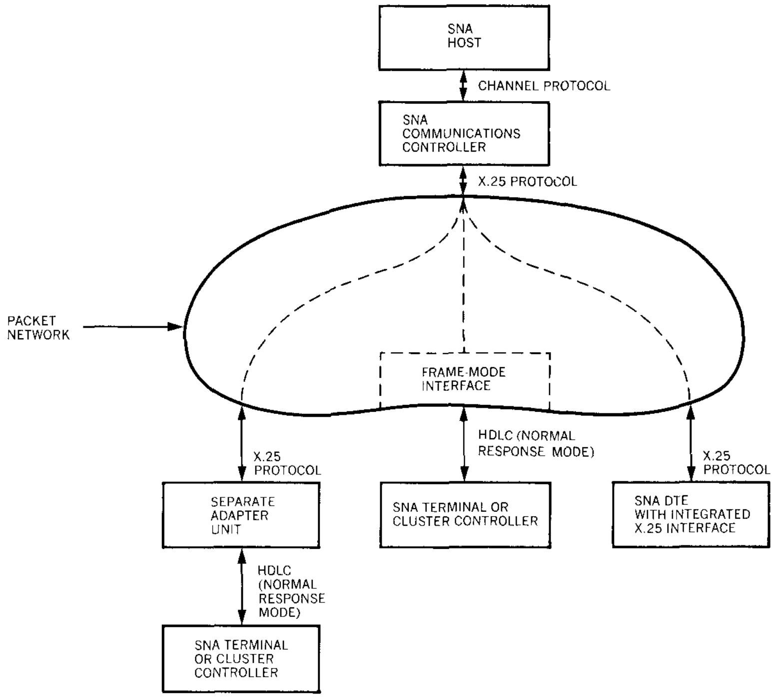


one correspondence of X.25 virtual circuits to SNA sessions. The apparent advantage of session-level mapping is lower overhead. One drawback is the lack of end-to-end control in X.25. (It directly controls DTE-to-network flows, not DTE-to-DTE.) Thus, much of SNA session control may have to be retained in any case. Another disadvantage results from different implementations of flow control parameters, maximum packet lengths, time-outs, and recovery procedures in public networks. To adapt a DTE to these differences among networks in a single stage at the link level is easier than managing this variability on a session-by-session basis. Other concerns with session-level mapping include the difficulties in setting up sessions to multiple end users at the same DTE (with a single X.25 address) and accounting for undelivered packets when virtual circuits are reset.

In sum, most of SNA session control is not redundant with virtual-circuit capability, and adapting to different networks is most easily handled by a single procedure at the link level rather than for each session. A decision to map at the link level means that virtual circuits can be treated by SNA in essentially the same way it handles real circuits with an X.21 interface.

The next class of decisions concerns which network service to use. Of the multiplicity of services, Figure 7 indicates one example, the choice of using X.25 permanent virtual circuits,[25] virtual calls, or both. Once a decision in favor of link-level mapping is made, adaptations to both permanent virtual circuits and virtual calls may be provided. The same SNA mechanisms that handle conventional private and switched lines can be readily adapted to their X.25 analogs when link-level mapping is used. Although not all packet networks provide both services, use of both by SNA DTEs appears appropriate since one or the other may prove more effective depending upon the application.

Figure 8 Three implementation alternatives for SNA terminals and cluster controllers



The last choice in Figure 7, implementation alternatives for DTE, is elaborated in Figure 8. The X.25 interface at a host location is best handled by modified system software. However, the implementation of X.25 at a terminal location may be done in any of three ways. At the left of Figure 8, a separately housed adapter unit performs a protocol transformation. This transformation process accepts a frame from an SNA terminal or cluster controller fully adhering to SNA formats and protocols. Then the unit breaks the message into correct X.25 packet sizes and creates appropriate X.25 packet headers. Since the Level 2 protocol between the adapter and the SNA terminal is SDLC normal response mode, the adapter unit polls the terminal. The terminal reacts to the poll just as it would if received from a distant host; data may flow in either direction between terminal and adapter unit. The separate adapter unit then converts to the version of X.25 Level 2 line control that the packet network designer chooses. None of the network Level 2 procedures employs polling; therefore, no packets with polls are transported across the network.

An advantage of the separate adapter unit is that the same terminal can be used on conventional leased and switched facilities by simply bypassing the unit. A second advantage is that the same separate adapter unit can be used with any SNA terminal or cluster controller that acts as an SDLC secondary station, which includes nearly all of the SNA family. However, the unit adds another delay to the transit time of the packet network; analysis to date indicates the additional performance degradation is not major. Perhaps more important is the inconvenience of another piece of equipment on the user's premises.

At the right of Figure 8 is a second implementation example where the function of the adapter unit is integrated directly into the SNA DTE. This integration would probably mean adding X.25 microcode within the terminal in such a way that the code would be logically bypassed when the terminal uses leased- or circuit-switched facilities. The elimination of a piece of equipment is clearly advantageous; however, each SNA terminal would have its own characteristics in terms of available microcode memory and logic-card space. The choice between the separate adapter unit and a physically integrated one is a relatively narrow one, once the architectural decision for link-level mapping has been made. According to a recent analysis for the Transpac packet network in France,[26] there are eleven manufacturers providing an integrated X.25 terminal adaptation and nine (including IBM) providing a separate adapter unit.

The middle implementation alternative in Figure 8 illustrates a concept similar to the separate adapter unit except that the adapter functions are provided by a protocol converter in the network itself. IBM has submitted a paper to the CCITT describing in detail how this conversion might be accomplished.[27] It may be thought of as a packet assembly-disassembly unit with a bit-oriented, frame-mode interface. Its advantage lies in eliminating the need for HDLC terminals to provide an interface designed specifically for packet networks.

The frame-mode interface used on permanent virtual circuits could allow current HDLC terminals to use packet services without modification to the terminal. Such a terminal could communicate on a conventional leased line to a host processor and then later be attached to a packet network without change to the terminal. For HDLC terminals that operate as secondary (i.e., polled) stations in normal response mode, this implies that the packet network assumes responsibility to poll the terminal. Network polls would have to be organized to provide acceptable performance.

Virtual calls, with a need for call establishment and call clearing, would require some terminal modifications even with the frame-mode interface. International discussions are now underway to consider several ways of providing virtual-call capability for the frame-mode interface using ideas based on X.21, HDLC, and X.25. For both virtual calls and permanent virtual circuits, the frame-mode interface holds the promise of an economical, flexible interface for any HDLC terminal requiring only one virtual circuit.

An early implementation of an SNA-X.25 interface

In February 1977, IBM released SNA adaptations for the Transpac network in France and the Datapac network in Canada. The basic design choices at that time, in terms of Figure 7, were link-level mapping, support for permanent virtual circuit and virtual call consistent with existing product support of leased and switched service, and use of a separate adapter.

The 1977 release is represented in Figure 8 by the SNA communications controller, the IBM 3705, and the left-most terminal configuration. At the terminal end of a virtual circuit, adaptation to X.25 is handled by a separately housed unit (the IBM 5973 adapter). At the host end of the circuit, the adaptation is done by special X.25 software in the network control program of the communications controller. The controller can implement up to 486 virtual channels, but the adapter is constrained to operate with a single virtual channel.

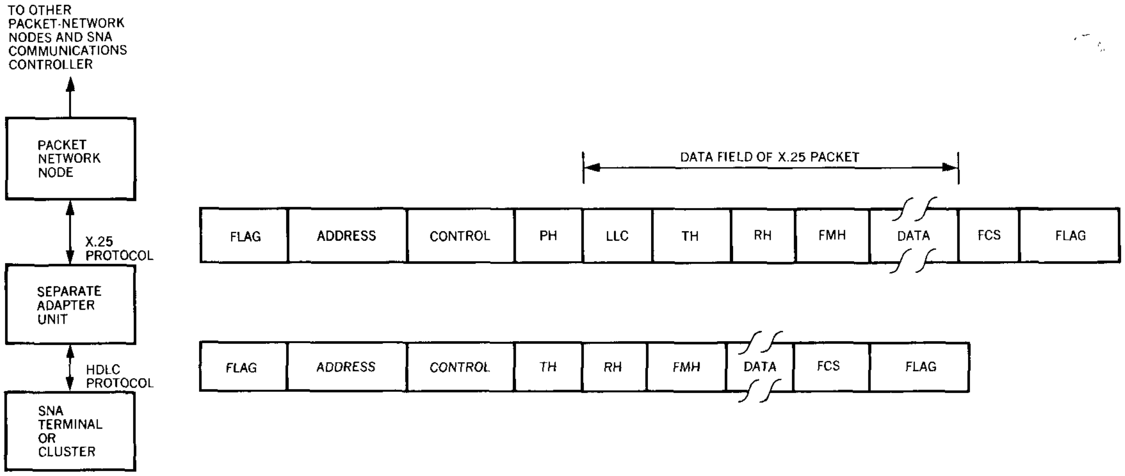
Since the adapter is intended for use with only the classes of SNA terminals called node Types 1 and 2,[28] simultaneous virtual circuits to multiple hosts are not used. Node Types 1 and 2 are configured in SNA to communicate directly with only one host at a time.

In conventional operation, without a packet network, communication between nodes of Types 1 and 2 and multiple SNA hosts is accomplished by using SNA networking among hosts and communications controllers or by using one-host-at-a-time dial-in capability.

When practical usage of X.25 services takes place, a need for simultaneous virtual circuits to multiple hosts from a single controller will probably arise. There are several possible SNA solutions. When the terminal controller in question is a reasonably sophisticated processor, the most straightforward solution is to implement that DTE as an SNA Type 4 or 5 node.

The format translation performed by the SNA adapter unit is shown in Figure 9. The HDLC normal-response mode headers of SNA are simply converted into the HDLC headers of the Level 2 procedure of the network. The other SNA formats, the transmission header, the request-response header, and function-management header, if present, are sent essentially without change. A packet header in conformance with X.25 is added by the adapter

Figure 9 Protocol conversion effected by separate adapter unit in the SNA-X.25 interface implementation for Transpac and Datapac (Legend: PH = X.25 Packet Header, LLC = Logical Link Control, TH = SNA Transmission Header, RH = SNA Request/Response Header, FMH = SNA Function Management Header, FCS = Frame Check Sequence)



unit. Finally, a special logical-link control provides assurance of packet delivery on an end-to-end basis. It also provides certain functions, like identification exchange, that are provided at the data-link control level on circuit-switched services.

If SNA and X.25 headers in the format shown in Figure 9 are all regarded as overhead, then the X.25 formats account for about one third of the overhead and SNA for two thirds. Total overhead represents about 10 percent of all bits transmitted when data packets have a length of 256 bytes. For data packets of 128 bytes, overhead represents about 20 percent of all bits. Other forms of overhead, such as exchange of control packets, are application-dependent.

In its 1977 release, IBM has, in effect, provided the levels above X.25 by means of SNA. In the absence of standardization of these higher levels for synchronous DTEs, two solutions are feasible. One is to provide a set of existing formats and protocols matched at an appropriate level to X.25 as in the 1977 release. A second approach is to provide each user a programming interface at the packet level of X.25 and allow the user to provide whatever coding is needed to match the user's DTEs to each other. This user capability was provided by IBM in 1978 in an X.25 adaptation of the Series 1 processor.

The CCITT survey cited previously[19] reveals that ten countries should be providing X.25 services by the end of 1980, and four more have plans to provide them later. Clearly, the X.25 services are entering upon a phase of practical usage. Undoubtedly, this experience will lead to an evolution in the nature of these services. Concurrently, SNA-X.25 interface concepts can take advantage of this learning process.

Summary and prognosis

The long-term objective of full intersystem compatibility is sought by ISO. A first step, based on the ISO provisional model, is an encouraging early thrust at a global architectural solution. SNA and the international standards X.21, HDLC, and X.25 map into the ISO architecture. Until a complete family of standards is ready, these three standards can serve as the basis for operating on new public data networks.

Thanks to broad architectural compatibility, SNA can be adapted to the current standards. For X.21, the adaptation is largely physical with X.21 circuit-switched service readily compatible with SNA. For X.25 the adaptation is best made at the link level. IBM's 1977 release is based on link-level adaptation with a separately housed adapter unit for cluster controllers. This release uses SNA protocols for those higher levels not yet standardized. A later IBM release offers the user the capability to program his own higher-level protocols.

As difficult as prognostication may be in this fast-moving area, some forecasting can be done. Two objectives may ultimately be achieved: the so-called universal interface and any-to-any communication.

A universal interface would allow a terminal or host computer access to any public communication service using a single standard interface. Today, in a number of countries, we have X.21 for new leased private-line and circuit-switched services and X.25 for packet services; in almost all countries we have V.24 and V.25 where the conventional telephone network is used. Since X.21 is part of X.25, there has been, at least, some movement toward a common interface for public data networks. In parallel, the recent mini-interface proposals might make it possible to design a terminal with an X.21 interface that will operate either on a telephone network or a data network. Convergence to a single interface for all services may take several years; meanwhile, the ISO architectural model may keep network differences from appearing at higher levels. The mapping of X.25 into the SNA data-link level discussed previously is an example of how a layered structure can be used to prevent network differences from having a significant effect at higher levels.

The long-sought goal of any terminal being able to access any application can only be achieved in steps. Currently, existing standards can be used for lower-level compatibility. For synchronous terminals, private protocols provided by manufacturers or users have been shown to be partial solutions. That is, users and manufacturers can implement, on a case-by-case basis, selected higher-level protocols. IBM's implementation of X.25 interfaces on machines using SNA protocols is an example of how specific higher-level protocols may operate with X.25. Ultimate realization of the long-sought ideal of open-system interconnection awaits the completion of the ISO model and the implied layer-by-layer standards.

ACKNOWLEDGMENTS

The authors are not, in general, the originators of the work and ideas reported in this paper. In addition to the references cited, the following individuals have made important contributions: V. Ahuja, J. Aschenbrenner, W. Brodd, B. Forsyth, and J. Merkel. S. Chappel helped to structure figures and text. The existing SNA-X.25 interface described in the paper was developed by a special engineering team at IBM's Centre d'Etudes et Recherches in La Gaude, France.

CITED REFERENCES AND NOTES

1. J. R. Halsey, L. E. Hardy, and L. F. Powning, "Public data networks: Their evolution, interfaces, and status," *IBM Systems Journal* **18,** No. 2, 223-243 (1979), this issue.
2. *Reference Model of Open Systems Architecture*, International Standards Organization Document ISO/TC97/XC16/N117 (November 1978).
3. R. J. Cypser, *Communications Architecture for Distributed Systems*, Addison-Wesley Publishing Company, Inc., Reading, MA (1978).
4. The International Organization for Standardization is composed of standards groups from over 20 countries which together develop standards to facilitate international exchange of goods and services. Standards for data communication are developed by ISO Technical Committee 97.
5. The International Consultative Committee for Telegraph and Telephone, with headquarters in Geneva, Switzerland, is part of the International Telecommunications Union. The ITU operates under the United Nations Economic and Social Council.
6. *Orange Book*, Volume VIII.2, Public Data Networks, CCITT Sixth Plenary Assembly, Geneva (September 27-October 8, 1976).
7. *Data Communication—High-level data link control procedures—Frame Structure*, ISO Standard ISO/3309-1976.
8. In the ISO documentation, the word "level" is used with numerical descriptors, and the word "layer" with textual descriptors.
9. This terminology, DTE and DCE, has been accepted by both CCITT and ISO. A DTE corresponds to a business machine such as a data terminal or a communications controller. A DCE corresponds to a modem or its functional equivalent in a public data network.
10. *Provisional Recommendations X.3, X.25, X.28 and X.29 on Packet-Switched Data Transmission Services*, CCITT, Geneva (1978).
11. X.21 can also be considered an example of a protocol at Levels 1, 2, and 3 since the call establishment dialog with the network involves elements of functional Levels 2 and 3. In the interest of simplicity, this paper treats X.21 as though it were only a Level 1 interface. A lucid treatment of the X.21 circuit establishment spanning the three levels appears in a recent paper by Folts.[12]

12. H. C. Folts, "Interface standards for public data networks," *Compcon 78 Proceedings*, IEEE Catalog No. 78CH1388-86, 246–252 (September 1978).
13. V. Ahuja and A. S. Barclay, *Adaptability of Systems Network Architecture for the CCITT X.21 Interface* (to be published in 1979).
14. *Centralized Multipoint Facility for Leased Circuit Service*, U.S.A. Contribution No. D31 to CCITT X.21 Rapporteur, CCITT Document COM. VII-No. 282, X.21 Special Rapporteurs Report (November 1978).
15. *Draft Proposal for a Recommendation on Multiplexed Links to User Classes 3–6*, CCITT Rapporteur for X.21 Multiplexed Links, CCITT Document COM. VII-No. 234E (October 1978).
16. Because the byte-timing circuit option of the conventional single-channel X.21 interface serves no useful function in an SNA environment, it would probably not be implemented in SNA DTEs.
17. *Mini-Interface: Some Initial Proposals*, U.S.A. Contribution to CCITT SG XVII, CCITT Document COM. XVII-No. 127 (September 18, 1978).
18. *X.21 Compatible Mini-interface*, contribution of the Netherlands Administration to CCITT SGXVII, CCITT Document COM. XVII-No. 127 (September 18, 1978).
19. *Final Report on the Responses by Administrations to CCITT Circular Letter No. 68*, CCITT Rapporteur for Question 33/VII, CCITT Document COM. VII-No. 247-E, (November 21, 1978). This report does not refer to networks in the United States.
20. LAP stands for link access procedure. LAP is unique to X.25 and not part of HDLC.
21. There are several good references with a more detailed explanation of packet procedures in X.25. Halsey, Hardy, and Powning[1] is one example. Chapter 17 of Cypser[3] provides more depth.
22. In some packet networks, certain aspects of the X.25 protocol are used on an end-to-end or DTE-to-DTE basis. However, this is an optional use, and the only "universal significance" of the procedures is the local DTE-to-network protocol.
23. In most existing packet-network implementations, X.25, when used, is implemented by sophisticated data terminals or communications controllers. Many different interfaces accommodate simpler data terminals. In this discussion it is assumed that SNA will use either X.25 or the so-called frame-mode interface.
24. P. T. F. Kelly, "Public packet switched data networks, international plans and standards," *Proceedings of the IEEE* **66**, No. 11, 1539–1549 (November 1978).
25. A permanent virtual circuit may be considered a degenerate case of a virtual call wherein a virtual circuit is provided in permanence when a user subscribes to the X.25 service.
26. J. Trottin and M. Gayraud, *Equipment Connectable to Transpac* (in French), Volumes 1 and 2, published by GERPAC in Paris (July 1978).
27. *Frame Mode DTE Interface*, CCITT Document COM. VII-No. 193 (May 1978).
28. Chapter 12 of Cypser[3] describes SNA node types. Type 1 is a relatively low-function terminal; Type 2 a cluster controller; Type 4 a communications controller like the IBM 3705; Type 5 a host processor.

Reprint Order No. G321-5095.