EDP ANALYZER

JULY, 1976
VOL. 14, NO. 7

NETWORK STRUCTURES FOR DISTRIBUTED SYSTEMS

Distributed systems, as we use the term, will make extensive use of data communications. Last month we considered how application systems might be partitioned and distributed, based on the natural clustering of activities. In this report, we consider the structure of the data communications network for tying the partitions together. The trend today is toward network structures and protocols (control procedures) that are largely independent of specific applications and specific equipment. But even within this trend, there are several major schools of thought—private versus public networks and centralized versus distributed control, to name a few. Here is an overview of what is emerging in data communications networks.

Montgomery Ward, with headquarters in Chicago, Illinois, is the huge retail department store and catalog sales office arm of Marcor, Inc. Total corporate sales are over $4.6 billion annually, the bulk of which are Montgomery Ward sales. The corporation employs some 125,000 people.

Montgomery Ward's data processing has been regionalized, with processing centers in Chicago, Illinois; Baltimore, Maryland; Kansas City, Kansas; and Oakland, California. IBM 370 equipment is used in these four centers. In addition, the company has been using a variety of terminal types in its stores and catalog sales offices. These terminal types have included IBM 3270 CRT-type terminals, IBM System 3 remote job entry terminals, NCR 280 point of sale terminals, TRW credit authorization pads, and Teletypes.

But the variations did not end with the types of terminals; different communication line speeds and disciplines have been necessary. On the leased line network, these speeds included 1200 bits per second asynchronous operation, and 2400, 4800, and 7200 bps synchronous operation. In addition, the Teletypes have been operated over the dial network at 110 bps asynchronous.

The resulting network included a large number of lines, operating at different speeds and under different disciplines, for serving the variety of applications. The amount of line sharing that could be done among applications was quite limited. Moreover, transaction activity is growing. By 1978, Montgomery Ward expects to need a total of about 20,000 terminals in its stores and catalog sales offices. Operating costs have been growing right along with this transaction growth.

So, in 1974, Montgomery Ward decided to look at alternative solutions to the data communications problem. In 1975, they became interested in what NCR had to offer. NCR had developed their 721 communications processor and supporting software, and were in the process of formulating their Data Communications Util-

ity (DCU) approach to data communications. Montgomery Ward made significant contributions to key concepts in DCU, such as structuring it as a stand-alone system. So Montgomery Ward placed an order for a DCU network and is now in the process of installing the network, with the first phase already in operation.

Montgomery Ward's DCU network

The four regional centers—at Chicago, Baltimore, Kansas City, and Oakland—will continue to perform their data processing functions as before, and on the IBM 370 equipment. But the 370s will now interface to a data communications network that has been supplied by NCR. Further, some of the terminals that will be used will be IBM-compatible (in addition to other types that we will mention). The same network will serve all applications. Hence the network will be largely application independent and computer and terminal independent.

Here is a summary of the network configuration that Montgomery Ward has chosen. The network components, of course, can be configured in a wide variety of ways so that this configuration is just one example of what is possible.

Hardware configuration. A dual front end processor is being installed at each of the four regional centers; dual processors are being used for backup. This is the NCR 721 communications processor, operating under the transaction oriented executive operating system. Each of these regional front ends is tied to three remote communications processors (RCPs), which are also NCR 721s. In a later phase of the system, the four front ends will be tied together, allowing in theory any-terminal-to-any-terminal communication.

The NCR 721 is a mini-computer with a special hardware multiplexor for data communications use. The multiplexor handles all input, output, and communications checking. It stores characters directly into processor memory, interrupting the processor only after the complete message is assembled in memory. The processor is thus available for performing a large number of network management functions, to be discussed.

Connected to each RCP by a common trunk, and adjacent to it, is an NCR 725 distributed processor. This uses the same mini as the 721 but does not include the multiplexor. The 725 is used for capturing and storing transactions and routine credit authorization messages during the day, for transmission at night to the parent regional center. The 725 has a printer, card reader, and magnetic tape unit attached and functions as a remote entry station to a host computer.

The RCP also interfaces with a variety of terminals at the store or catalog office level.

At each store location is one or more concentrators for serving terminals in the store. Each concentrator is connected via the network to its parent RCP. One common type of concentrator handles a cluster of NCR 280 point of sale terminals and KSR Teletype compatible terminals in the catalog sales department. Another handles TRW credit authorization pads, used with older style cash registers. If the parent RCP is down or the communications line fails, the concentrator stores point of sale transactions on cassettes until the failure is corrected.

The RCP also interfaces two other types of units. One is a control unit for NCR 7300 CRT terminals, which are IBM 3270 compatible. The other is the ability of the RCP to dial up an ASR Teletype terminal at a catalog sales office and request the transmission of transactions that have been recorded on paper tape.

In the first installation phase, the DCU is being used for handling the types of message flow to be discussed shortly. The whole network has been brought under DCU management. Message switching, which was one of Montgomery Ward's original goals for the network, has been scheduled for later implementation, as have the restart capabilities.

Network control. There is no overall central point of control in a DCU network. Each node has the information it needs about the status of adjacent nodes and communications links. NCR currently offers over 30 software modules for use in the 721 by which networks can be configured as the user desires. These modules reside in the front end processors and remote communications processors.

The functions performed by network control include routing, alternate routing, acknowledgement, logging, packet formation, load smoothing, and restart. An end-to-end acknowledgement feature, optional by message, is provided. Each packet header contains a disposition code, to in-

dicate what should be done if the packet cannot be delivered.

Data transmission. Messages are transmitted in packet form. Packets are variable in length up to a maximum of 255 characters. Transmission speeds from terminals to RCPs generally will be between 300 and 1800 bps asynchronous and 4800 and 7200 bps synchronous. From RCP to the front end, speeds generally will be 9600 bps, although 56 kbps is being considered. Montgomery Ward has chosen to use multiple 9600 bps lines over dispersed paths, for load sharing and backup. Geographically dispersed paths are being used to reduce the chance of both lines being down at the same time.

Message flow. During store business hours, the network is being used for credit authorization queries and responses, for credit referral queries and responses, and for some on-line file maintenance functions. For instance, when a customer wishes to purchase something on credit, the sales clerk enters the account number and amount on the NCR 280 POS terminal (or on the TRW pad, if a manual register is being used). The message goes through the store's concentrator, onto the network, to the RCP. The RCP can be hundreds of miles away. The RCP recognizes the message as a credit query and routes it to the parent front end processor—which in turn directs it to the TRW 4000/IBM 370 processor. If credit is satisfactory, a message to that effect is sent back along the same route and the sales transaction is completed. If there is a question about the customer's credit, a refer signal is given. The clerk then calls the credit office on the telephone, describes the transaction, and gives the account number. A credit authorizer, operating an NCR 7300 CRT-type terminal, enters a credit referral query. This goes up through the network to the regional center. The response is a multi-line message of the customer's name, address, buying and payment history. After looking at it, the credit authorizer may authorize the sale then and there. Or the credit authorizer may want to talk to the customer on the phone, to get additional information, before making a decision.

If the credit card has been reported stolen or for comparable reasons, the system notifies the sales clerk. The sales clerk keeps the card and calls the appropriate member of management.

When a credit sale has been authorized, or any other type of sale consumated, the message flows to the RCP where it is recognized as a sale transaction. The RCP directs the message to the 725 distributed processor. The 725 records it on magnetic tape, for transmission that night to the regional center.

What the network does for Montgomery Ward. This discussion has touched only on the highlights of the data communications network, but several points should be apparent. The same network is being used to serve all applications. A variety of terminal types are being accommodated. If Montgomery Ward wants to add a new terminal type or a new application, chances are good that these can be added with little concern for network characteristics. The data communications functions have been separated from the data processing functions. The main CPUs are not involved with data communications functions. And the network is quite robust. Alternate routings are available. Moreover, there is no central network control processor which, if it goes down, shuts down the network.

So Montgomery Ward is well along in the installation of a new generation of data communications networks.

Aspects of data communications networks

The subject of data communications networks is huge. In a report such as this, it is possible to touch on only a few aspects of networks. To give an idea of the scope of the subject, the following are some of the high level aspects that could be used to categorize networks.

ASPECTS OF NETWORKS

1. Ownership: public, private, shared private.
2. Structure: point-to-point, multi-point (party line), centralized (star), multi-star, hierarchy (tree), ring, or distributed (peer, with all processors of equal rank) network.
3. Protocols (control procedures)
   a) Inter-element: between user and host, user and terminal, host and host, host and terminal, host and node, node and node, node and terminal.
   b) Type of traffic: one-way messages (packets for fast transmission, store-and-forward for throughput), two-way messages (interactive), data transfer, process (job) transfer.
   c) Switching: circuit, virtual circuit, message, packet.
   d) Routing: dynamic, preferred, fixed.
4. Standards
   a) Interfaces: between host and node, terminal and node, and between one network and another.
   b) Canonical elements: virtual terminal, virtual host.

5. Controls: network command language, error control, routing control, load control.
6. Transmission media: cable, microwave, satellite.
7. Transmission: analog, digital.
8. Performance characteristics: integrity, reliability, recovery, security, throughput, response time.
9. Network evaluation: modelling, vulnerability analysis, growth analysis.
10. User aspects: user identification, user assistance, usage accounting and billing.
11. Legal and regulatory considerations.

It is clearly impractical to try to cover even the major alternatives from among the many combinations of the above aspects. So, in this report, we will concentrate mainly on *private* data communications networks, for reasons that we will give below. But to set the stage for discussing what is happening with private networks, it will help to briefly review how computer networks—both public and private—have emerged.

Evolution of data communications networks

This material has been drawn from McFadyen and Thomas (Reference 1) Davies (Reference 3), Blanc and Cotton (References 4a and 4b), Frisch and Frank (Reference 5a), and Sanders (References 3 and 13 plus an interview).

McFadyen and Thomas trace the development of private data communications networks. In the earliest days of computers, each application program had its own input-output routines for communicating with peripheral devices. The next step was to develop generalized input-output control software for all locally attached devices, via batch-type communications. The next extension was for the generalized I/O software to serve remote terminals, as well as local peripherals. Then came teleprocessing control systems, which were specific as to device type and line speed. The last stage before the present generation was tree-structured networks with node teleprocessing control; however, it still required specialized control routines for specific devices. Moreover, throughout all of this development, these specialized control routines for devices were embedded in the application programs. So there was tight coupling between the application programs and the data communications network.

We are just witnessing the arrival of the distributed, device-independent private network. These include IBM's Systems Network Architecture (SNA), NCR's DCU, Digital Equipment Corporation's DECNET, Computer Transmission Corporation's PACUIT, and others. While these offerings have quite similar objectives, they have quite different structures, as we will discuss shortly.

Somewhat in parallel with these developments, large semi-private and public networks were being developed by telecommunications agencies and quasi-public bodies in a number of countries. Mr. R. W. Sanders, President of Computer Transmission Corporation, reviewed these developments for us. Among the first of modern packet-type networks was the SITA network, developed in the mid-1960s for joint use by international airlines; we discussed the SITA network in our February 1975 issue. This was followed shortly by the ARPA Net (discussed in our November 1971 and January 1973 reports), which has tied together universities and research centers across the U.S. and in the United Kingdom. The ARPA Net has had a very significant influence on network development. TYMNET, developed by Tymshare, Inc., originally was set up to serve the company's time sharing customers, but soon expanded into a value-added type of operation through joint use agreements (as we discussed in January 1973). The CTNE network in Spain became operational in 1972. The Telenet Communications Corporation began offering public packet switched communications services in the U.S. in 1975, as a common carrier. In the U.K., the EPSS network is well along toward operational status, as is the Datapac network in Canada and the CYCLADES and TRANSPAC network in France. (We discussed the Telenet, CTNE, EPSS, Datapac, and CYCLADES networks in our February 1975 issue and expect to describe the TRANSPAC network in the not-distant future.)

Computer Transmission Corporation (TRAN), of El Segundo, California, was active throughout most of this period. They questioned the effectiveness of some of the goals in the "pure" packet switched networks—goals such as dynamic routing and virtual terminals. What TRAN has developed is a hybrid system that they call PACUIT, for a combination of *pac*ket and circ*uit* switching. We will have more to say about PACUIT later in this report. It is interesting to note that TRAN has supplied switching equipment for Canada's Dataroute network (the first phase of Canada's total digital network), and is supplying the switching

equipment to Pacific Telephone for the State of California's new data communications network.

Sanders reviewed for us the goals of much of the early development work in networks. These were:

GOALS IN NETWORK DEVELOPMENT

1. A universal network access arrangement—by using the concepts of "virtual terminal" and "virtual host," designers hoped to be able to connect just about any terminal and any host computer to the network; within the network, everything would be handled in a standard (virtual) manner.
2. Independence from any particular supplier, of computers or of terminals.
3. Highly effective error control in an environment of noise and disruption of communications services, by means of automatic error detection, retransmission, and the use of alternate routes.
4. More effective use of communication bandwidths by the concentration of many "virtual circuits" on high-speed inter-node trunk circuits.
5. Accessing multiple terminals without the use of polling, since polling is a wasteful operation.
6. Controlling the flow of traffic into the network and through the network, so that storage facilities do not become overloaded and message queues too long—or worse, messages lost.

Two other aspects were mentioned to us by William Helgeson of Honeywell Information Systems. One has been the emphasis on fast response time, rather than thoughput (as was characteristic of store-and-forward systems). The other aspect was the emphasis on end-to-end control of transmission and communications, in addition to point-to-point control.

It is evident from this list that the designers were seeking complete *data communications services*, not just data transmission services. Packet switching was seen as a way of accomplishing these goals, as contrasted with circuit switching or conventional message switching. For instance, the concepts of virtual terminals and virtual hosts probably could not be implemented in a regular circuit switching environment.

The point here is important, so perhaps we had better reiterate it. The telecommunication agencies witnessed the development of private data communications networks in which they provided only the data transmission services. But telecommunication agencies generally feel that private networks are wasteful; customers can be served more effectively by public networks. Packet switched public networks were seen as a way by which the telecommunication agencies could offer complete data communications services.

But this goal leads to further complexities. Design standards are needed, so that the public (packet switched) networks of one country will interface properly with those of other countries—just the way that the dial telephone networks interface. So some type of common structure was and is needed. Is this an impossible dream?

Surprisingly, it is not out of the realm of possibility. In October 1975, the French PTT, the Canadian Trans Canada Telephone System, and Telenet in the U.S. offered a proposal to CCITT for a standard structure. This was an informal proposal because, for one thing, Telenet cannot speak for all telecommunications agencies in the U.S. (CCITT is the standards organization of the national telecommunication agencies and telecommunication equipment manufacturers.) The proposal was very well received, we were told, and a formal proposal to the same ends was submitted later in 1975 by the French PTT and the U.K. Post Office. The proposal, code named X-25, was approved in March of this year by the responsible CCITT study group. We understand that approval by the CCITT plenary session in September seems assured—in which case, X-25 will become an international standard. So we might just see a common structure for public data communications services adopted in the not-distant future. But it may take a number of years for such services to be implemented in any particular country.

While public data communications networks are important and might become *the* mechanism for data communications by, say, 1990, we see private networks as playing a dominant role for at least the next five to ten years. Telecommunication agencies will offer more and better data transmission services, such as the Bell System's DDS and Canada's Dataroute. Some will be offering packet switched services, such as Telenet, Datapac, TRANSPAC, and EPSS, but probably users will use these as data transmission services at the outset. The reason users will do so is that they often will have to couple such services with more conventional data transmission services; the packet switched services will not be available, end-to-end, to all remote user locations.

This, then, is why we will be concentrating on developments in *private* data communications networks in this report. We see them as providing the bulk of data communication services for at least the next five to ten years. But public networks are coming, and users would be wise, we think, to take advantage of them at least as data transmission services as they become available.

Let us now consider the structure of some of the leading private network offerings.

Variations in network structure

The Montgomery Ward network, described earlier in this report, is an example of a private network—for the private use of the company. The hardware/software complexes for switching messages, performing error control, and so on were obtained in this instance from NCR. The transmission services are obtained from the common carriers.

We will give a summary of several offerings in overall network structure, following which we will describe some line control protocols.

IBM's SNA

IBM's Systems Network Architecture is a centrally controlled, distributed intelligence network structure. It provides a common structure for a data communications network in much the same way that the IBM 360/370 has provided a common structure for data processing.

There are four levels of intelligence in SNA, but the user need not use all four levels. The top level is the Virtual Telecommunications Access Method (VTAM), located in the main CPU, which provides overall control for the whole network. (Optionally, this function can be performed by TCAM.) At the next level is the Network Control Program (NCP), which provides local network management under the direction of VTAM. The NCP resides in the front end processors and intermediate node processors (currently IBM 3704 or 3705). The next level is the cluster control level—the terminal control level. The fourth level is the terminal level, should intelligent terminals be used.

SNA uses Synchronous Data Link Control (SDLC) as the line control protocol. We will discuss SDLC shortly. As will be pointed out, SDLC is not compatible with IBM's previous offerings such as the binary synchronous line protocol (BSC).

With programmable control units in the network, device control functions are moved out of the host CPU into the network controllers. The structure of SNA determines which network functions can be located in which processors, in order to achieve compatibility throughout the network and among the various applications. Thus, the functions to be performed at each level of the network are precisely defined.

User programs need not be concerned with the physical makeup of the network nor with specific transmission services. The user program simply provides the data to be transmitted and calls for the desired service by name. The network has the responsibility of getting the data to the desired destination.

As is true of this generation of network management systems, SNA requires an appreciable overhead as compared with, say, a point-to-point communication system for one application. The rationale for something like SNA is when an organization is using a variety of terminal types, operating at different speeds and under different disciplines, for a variety of applications.

It should be noted that not only are some network functions moved out of the CPU into the network processors but also some application logic can be similarly moved. For instance, selected data can be stored at the local level for answering most queries and validating most input. Only in exceptional cases need reference be made to the central master files or data base. However, top level network control still resides in the host CPU.

NCR's DCU

We discussed a number of the characteristics of the DCU structure earlier in this issue, in connection with the Montgomery Ward network. We will try not to repeat too much.

DCU is a distributed control, distributed intelligence network structure. The structure has three potential levels of intelligence, not all of which need be used. The top level is the node processor, the NCR 721, operating under the TOX operating system (Transaction Oriented Executive). This node processor operates either as a front end for a host computer or as the network management processor at a remote node of the network. Control is distributed; there is no processor that is in overall control of the network; DCU is stand-alone and data transparent. Should one

node processor go down, the remainder of the network can continue to operate. Some applications logic can be distributed to the node processors, but typically a separate mini-computer processor will be connected to the node processor for handling applications-oriented functions.

The next level is the terminal control unit level. The intelligence at this level typically is in the form of firmware. Should the parent node processor or communications line go down, the terminal control unit stores the transactions for later transmission to the node processor when the failure has been corrected.

The next level is the terminal level. Terminals may be intelligent, semi-intelligent, or non-intelligent. By semi-intelligent, we mean terminals that operate under a fixed program, such as point of sale devices.

DCU operates under a variety of line disciplines. These include the NCR terminal disciplines, IBM's BSC, Teletype, IBM 3270 I/O and poll inward I/O, multi-leaving, and (later this year) SDLC.

DEC's DECNET

DECNET is a distributed control, distributed intelligence network framework. It can use versions of the DECsystem-10, DECsystem-20, PDP-11, and PDP-8 computers, as well as other DEC products, and a variety of operating systems. In developing DECNET, DEC strove to create as general an inter-connection mechanism as possible, limited only by cost and state of the technology considerations.

DECNET uses a set of network protocols. One is the Digital Data Communications Message Protocol (DDCMP) which handles link traffic control and error recovery. We will have more to say about DDCMP shortly.

Another protocol is the Network Services Protocol (NSP). It handles network management functions, routing of messages, and such. NSP uses a distributed philosophy. Each node is kept informed of the current state of the other nodes and links.

The Data Access Protocol (DAP) allows a program at one node to use input-output services at other nodes, for using remote files and devices.

TRAN's PACUIT

Computer Transmission Corporation's PACUIT is a distributed control, distributed intelligence network framework. As mentioned earlier, it is a hybrid system, making use of a combination of packet switching and circuit switching disciplines.

Each source node in the network (a node at which traffic is originating) is responsible for selecting the routing for all packets originating at it. Thus, each node maintains the current status of the other nodes and links in the network. The network uses preferred routing. The preferred route between one node and another will always be used unless a line outage, high error rate, or severe traffic load requires the use of an alternate route. Moreover, once a route has been selected, it is used for all packets transmitted during that "virtual call" communications session.

PACUIT provides a data communications service similar to what telecommunications agencies will probably provide. It is a largely transparent packet switched service that operates under the virtual call discipline (which we will discuss later in this report). No source-to-destination protocol and no code conversions are performed by the node processors. As an example of the transparency provided, the people at TRAN say that if a terminal and a computer currently operate on a point-to-point basis, they should have no trouble communicating over a PACUIT network. PACUIT can be expanded to support the X-25 packet network access protocol of the CCITT.

Line control protocols

A good discussion of major line control protocols will be found in References 4c and 7. In addition, Donnan and Kersey (Reference 8) give a good discussion for the rationale behind the SDLC protocol.

Asynchronous protocols

A large fraction of today's keyboard operated devices operate on a character-by-character asynchronous basis. There is a variety of asynchronous protocols, and the characteristics of some are widely known. We will touch on only a few of the highlights of a Teletype protocol, as an illustration.

The Model 35 Teletype, for instance, uses an 11-bit frame for each character. This frame is made up of one start bit, 7 data bits, one parity bit, and two stop bits. In many instances, no par-

ity check is performed by a receiving terminal. If a check is performed and an error is detected, a standard error character is inserted for the data character. Asynchronous data received by a processor, of course, can have an error check made on a character-by-character basis. If an error is detected, generally all characters starting with the word or block in error must be retransmitted. Asynchronous devices can be operated on a full duplex or half duplex basis.

The asynchronous protocols are important because so many of today's terminal devices use them. Networks incapable of handling asynchronous protocols will be incapable of handling these existing devices, and only new, compatible terminals will be usable.

IBM's binary synchronous protocol

IBM's BSC protocol is probably the workhorse of today's synchronous data transmission. By sending multiple characters per block, the relative overhead is reduced as compared with asynchronous transmission.

Bsc operates only in the half duplex mode. For each block of data transmitted, at least two line turnarounds are needed, of 100 to 300 milliseconds each. After a block has been transmitted, the line direction must be turned around for sending the acknowledgement, and then turned around again for sending the next block.

Blocks can be of variable length, and control characters are used to delimit the fields. Synchronism is achieved by sending at least two SYN characters preceding each block. The protocol is character oriented and uses ASCII (7 data bits plus parity bit), EBCDIC (8 data bits) Transcode (6 data bits), or transparency mode.

Bsc uses block-by-block acknowledgement for acknowledging either correct or incorrect receipt. A simple form of sequence acknowledgement is obtained by using two forms of positive acknowledgement—ACK0 and ACK1—alternately in a series of blocks. NAK is used when a transmission data error is detected.

Transparent operation for the transmission of text in which control character bit patterns may occur is obtained by the use of DLE characters before and after the code to be sent transparently. If a DLE bit pattern occurs within this text, the system "stuffs" in an extra DLE character. The receiving end recognizes two DLEs in a row, discards the first, and accepts the second as data.

Bsc allows for polling and addressing on multipoint lines. It also includes a rigorous set of rules for establishing, maintaining, and terminating a communications session.

Bsc *could* be used for asynchronous or parallel data transmission but IBM does not provide these options.

IBM's Synchronous Data Link Control

IBM's SDLC is one of the new generation of line control protocols. It is quite similar to, but not compatible with, two proposed standard protocols—HDLC and ADCCP—which we will mention shortly. It is an important element in IBM's Systems Network Architecture. Further, it is not compatible with BSC.

SDLC can operate with either half duplex or full duplex transmission. Blocks can be of variable length. Each block has a fixed length header and a fixed length trailer section, of 24 bits each. Only the information field varies in length. SDLC is bit oriented, not character oriented, so the information field can be any number of bits. Were IBM to change to a 9-bit character, for instance, this could be accommodated in SDLC.

The first field in the header is an 8-bit flag, consisting of a 0 bit, six 1 bits, and a 0 bit. *This is the only control character used by this protocol.* It is repeated at the end of a block and is thus used for providing synchronism, for identifying the beginning of a block, and for indicating the end of a block. The trailer portion of a block consists of a 16-bit cyclic redundancy check and the 8-bit flag. The redundancy check method used is the one recommended by CCITT, which will probably become an international standard.

One of the controversial parts of SDLC is the feature that allows transmission of up to 7 blocks before an acknowledgement is required. The purpose is to reduce the number of line turnarounds in half-duplex operation. In theory, too, it would aid transmission by satellite; efficiency would be very low if the system had to wait for the acknowledgement of one block before the next could be transmitted. However, the seven block limit is too low for satellite transmission, as we will discuss, so it is possible that IBM will change this aspect of SDLC.

An acknowledgement can be for multiple blocks and can be included in the header of a message going in the opposite direction.

When transmission data errors are detected, the receiving station does nothing. When the sending station does not receive an acknowledgement within a specified time limit, it retransmits the block(s) in question.

Each block is assigned a sequence number—currently 0 through 7. When the receiving end detects an out-of-sequence receipt, it sends a NAK to the sending station. Currently, SDLC requires that all blocks subsequent to the erroneous (or out of sequence) block be retransmitted but it is possible that this feature will be changed if and when the number of blocks allowed before acknowledgement is increased.

SDLC has a relatively low overhead. Currently this is 48 bits per block, equivalent to six 8-bit characters. As a comparison, BSC has an overhead of 64 bits per block (or eight 8-bit characters) plus more line turnarounds than SDLC.

For transparency, SDLC uses a "bit stuffing" technique. The only control character used by SDLC is the 8-bit flag, as mentioned. Should that same bit pattern appear in the information field, it must be automatically sensed by the transmitting equipment and an extra 0 bit inserted after the fifth 1 bit. This means that one more bit is added to the information field. At the receiving end, the equipment senses for a string of 1 bits. If five in a row are detected and if the next bit is a 0, the receiver automatically deletes it, restoring the information field to its original length. If the receiver detects six 1 bits in a row, that is the flag.

Because bit stuffing increases the number of bits, SDLC cannot be used for asynchronous transmission or for parallel transmission.

SDLC is based on the idea of central control. Each link has an assigned primary station and one or more secondary stations. A secondary station can transmit to the primary only when authorized to do so by the primary. In addition, SDLC has been designed to handle multi-point lines that are in a hub or loop arrangement.

SDLC does *not* perform the following functions: break records or messages into blocks, handle device addressing or device status or device control, provide end-to-end control, establish or maintain or terminate a communications session, or exchange supervisor signals between modems.

DEC's DDCMP

Digital Equipment Corporation's Digital Data Communications Message Protocol can be used in either half or full duplex modes of operation. It has variable length blocks. The header is fixed format and length, with its own cyclic redundancy check. The header is considered particularly sensitive since it controls the routing of the block; hence the use of its own CRC. The information field can be of variable length.

DDCMP can handle synchronous, asynchronous, or parallel transmission. Synchronism is provided by the use of two ASCII SYN characters. The next character is start of header and indicates which of three block types the block is.

DDCMP is character oriented. The control characters are ASCII. The information field can be any number of 8-bit characters. DDCMP uses the CRC-16 type of cyclic redundancy check, which is not the same as the CCITT recommended CRC.

This protocol can transmit up to 255 blocks of information before an acknowledgement is required. So blocks are assigned sequence numbers ranging from 2 to 255.

When the receiving station detects a transmission data error, it sends a NAK acknowledgement to the sending station, giving the block number. For out-of-sequence errors, no ACK or NAK is sent; when nothing is received within the allotted time, the sending station recognizes that a sequence error has occurred. In case of either data or sequence errors, all blocks subsequent to the error must be retransmitted.

For transparency, a count field (of the number of characters in the information field) is maintained in the header. Even though the information field may contain bit patterns that are the same as control characters, they are treated as data as long as they fall within the limits of the information field.

DDCMP can be used for point-to-point and multi-point operation, as well as in a hierarchical, ring, or network-type network.

Other protocols. References 4c and 11 tell something of the status of other important protocols. The Advanced Data Communications Control Procedures (ADCCP) has been developed under the auspices of the American National Standards Institute and is being considered as a U.S. national standard. The High-level Data Link

Control (HDLC) has been developed under the auspices of the International Standards Organization. Both of these are quite similar to SDLC but differ from it in significant ways. The Europeans are taking a very active leadership role in this area, so it is quite possible that something like HDLC will win out.

It can be seen that much activity is going on within the field on network architectures and protocols.

The debate on network structures

Because of all of this activity, it is not surprising that there are some significant differences of opinion on the structures of networks and protocols. We will briefly cover some of the major issues.

Overall structure

Farber (Reference 10) discusses the concept of a *hierarchy of clusters*, which he sees as the structure of the future. A cluster is a group of computers, or a group of computers and one or more other clusters. (We discussed the concept of natural clustering last month.) Clusters would communicate with each other by some form of data communications. A cluster must have stand-alone capability, says Farber, in case the data communications system is inoperative. The structure of the data communications within a cluster may vary from cluster to cluster; the same is true for communications among clusters. Messages can be sent between clusters by addressing the desired clusters, says Farber, and the communications system must insure proper delivery.

The structure of a cluster is not fixed, as Farber sees it, but may be a tree, star, ring, network, or open bus.

The idea of clusters ties in to the concept of electronic mail. This idea of electronic mail has proved to be very popular on the ARPA Net, we have been told. "Mail boxes" are assigned to users of the network. One user may send a message to another user by sending it to that other user's mail box, where it is held until the other user asks the system for all of his mail. The process is relatively inexpensive and quite effective. Users find they use it more than the long distance telephone or the mails.

Farber pointed out that messages for other clusters could be accumulated within a cluster. Then perhaps once or twice a day, a cluster would forward all "mail" to the other clusters, perhaps using the telephone dial network.

Hence Farber has discussed a structure that transcends what is normally discussed in network structures—namely, tree versus star versus ring versus network, etc.

Blocks or packets

In our February 1975 issue, we discussed the characteristics of packet switched networks. A packet is a block of data organized for transmission which has both a minimum and a maximum length. Typically the maximum lengths have been set at between 1000 and 8000 bits, depending upon the network. A record or message that is longer than the network maximum must be divided into two or more packets, each with its own header and trailer fields. In some networks, such as the ARPA Net, each of these packets might follow a different path from source to destination through the network.

A block (or "frame," in SDLC terminology), on the other hand, theoretically does not have a maximum length. Like a packet, it has header and trailer sections. Unlike the packet, the information section can be of any length, limited only by practical considerations. In practice, very long records or messages might be divided into multiple blocks—or could be treated as packets, for that matter.

There is a controversy on whether network structures should be based on the concepts of packets or blocks. But Helgeson of HIS has pointed out that future networks should allow *both* types of service. Packets (with some maximum length) would be handled by packet switching, for fast transmission through the network. Messages (perhaps with no theoretical maximum length) would be handled by store-and-forward switching, for greater throughput. SDLC would seem to be in harmony with this viewpoint.

Routing protocols

Fixed routing, as the name implies, involves a one-only route between two stations in a network. For backup, the dial telephone network may be used. Fixed routing has been built into many of today's private networks which use leased lines and (perhaps) multi-point stations.

Circuit switching. The dial telephone network is probably the best example of circuit switching.

A user dials a desired telephone number. The network switches assign a circuit from source to destination. If the desired number is not already busy, it is signalled; if answered, the circuit is established. The circuit remains assigned to that call whether any information is being transmitted or not, until the call is terminated.

The equivalent of circuit switching can be provided in private networks. For instance, in multipoint operation, a primary station authorizes a secondary station to transmit, which is the equivalent of assigning a circuit.

Packet switching. There are two main philosophies being debated in packet switching networks. One is *datagram* service, and the other is *virtual call* service. We discussed the concepts (but not under these names) in our February 1975 issue. Davies (Reference 3) and Pouzin (Reference 1) give good discussions of the concepts.

In datagram service, long messages or long records are divided into two or more packets, each no greater than the maximum permissible length. Each packet has its own header and trailer section, and each is individually routed through the network. Two undesirable types of events can occur. The several packets of a message may arrive at the destination out of sequence. Or a packet may get into an "endless loop" in a series of nodes and never get delivered. Adherents of datagram service advocate that the network be kept as simple as possible, and leave the cures for such troubles largely up to the users.

In virtual call service, a specified path is set up from source to destination at the time the call is set up. In general, this might be the preferred path between the source and destination nodes. However, in the case of line outage, excessive noise, or excessive load on a channel, an alternate path might be picked. But once that path is selected, it is retained for all packets involved in that call until the call is terminated. So packets cannot arrive out of sequence due to routing, nor will they get in an endless loop. (Of course, if a packet in the middle of a message has a transmission error in it, it will have to be retransmitted—and will be received out of sequence.)

As indicated earlier, the X-25 packet network access protocol (that CCITT will vote on in September) is based on virtual circuit service. Expectations are that it will become an international standard at that time.

Network control

Should network control be centralized or distributed? It is interesting to note that IBM's current version of SNA has reasonably centralized control. The central control resides in VTAM (or TCAM), and second level control is delegated to the network control program in the intermediate processors. One advantage of centralized control is that there is no need for each node to get process status reports from all adjacent nodes. A disadvantage is that if the central processor goes down, the network is down.

It is also interesting to note, as was pointed out to us, that IBM has retained its options on this question of control. For instance, the phrase "in this implementation" has been used repeatedly in SNA documentation. So, if the market starts to favor decentralized control, it looks like IBM may be able to change SNA in that direction rather quickly, we were told.

Distributed control requires that each node know the current status of the adjacent normal and alternate links and nodes in the network. Each node makes its own routing decisions. These routings could be dynamic, preferred, or fixed, according to the discipline of the network.

Error control

Some people point out that end-to-end error control is essential for reliable communications. The receiving station must assure itself that it has received every packet or block, correctly and in sequence. Because of this, say these people, the node-to-node error control of some networks is redundant and probably is not needed.

So the question is raised: is node-to-node error control necessary, or can the same function be performed effectively by an end-to-end method?

We suspect that the eventual decision will be in favor of both node-to-node and end-to-end protocols. The different links in a data communications network are the major sources of error, from what we have heard, so it would be best to detect and correct errors as close to their source as possible. Further, node-to-node error control allows for the rapid selection of an alternate path, in case of link outage, excessive errors, or link overload conditions. On the other hand, end-to-end control oversees the delivery of all packets, in their proper sequence, at the destination. It can cause the re-

transmission of a packet if the original transmission has been lost for any reason.

Standardization

Standards can play a very important role in both public and private data networks. For private networks, interface and protocol standards would allow users to connect, say, terminals from a number of suppliers. For public networks, interface and protocol standards would allow the telecommunication agency networks in the several countries to be interconnected.

Sanders (Reference 3) discusses some of the areas in which standards are being considered. But he argues that the telecommunication agencies should not jump too quickly. For instance, the X-25 protocol being voted on in September by CCITT is a good starting point, he says, but it must be expanded in the future. Further, nothing like X-25 probably would have been proposed as recently as three or four years ago, which indicates how quickly the field is changing.

Some of the issues are just coming into focus. While standards are needed, a too hasty adoption of standards will probably be regretted later. Which is to say that users should not expect to see other standards adopted at the speed with which X-25 apparently is being adopted.

Network protocols

We have mentioned above that HDLC is being proposed as an international standard, ADCCP as a U.S. standard, and SDLC already shows signs of becoming an ad hoc standard.

One issue in these alternative protocols deals with the delay involved in satellite transmission. The delay, from earth to satellite and back, is about one-quarter of a second. From the time that a first packet or block were transmitted, it would be one-quarter second before the addressed receiving ground station would get the first packet. Assuming that the acknowledgement went back via satellite, that would mean another quarter second. (Conceivably, acknowledgements might be sent by terrestrial circuits to cut the delay.) If the transmission rate were 50 kbps and if packets were 1000 bits in length maximum, this means that about 25 packets could have been transmitted before the first one was acknowledged. Assuming that groups of packets would have to be retained at the sending end until each packet in the group has been acknowledged, one can see that upwards of 100 packets or even more might be "in process" at any one time.

This is the reason that SDLC's limit of seven packets before an acknowledgement is required is considered too low. SDLC has a nice, clean block structure. It will almost surely be made messier by allowing for a larger number of blocks to be transmitted before an acknowledgement is required. But with IBM getting into the satellite communications field, we suspect that such a change will be made.

Conclusion

Private data communications networks will be in use for years. Telecommunications agencies may desire to replace all private networks with public data networks, and they may be developing services toward that end—but the private networks will take a long time to disappear.

We do think that users should carefully consider the public offerings developed by the telecommunications agencies. For instance, the Telenet, DDS, and Datran offerings in the U.S., to name just a few (and not comparable ones, at that), should be considered.

In the area of private networks, it seems to us to be a matter of the network structure and protocols advocated by IBM versus the structures and protocols offered by others. Of the network structures that we have examined, none seem to use centralized control to the degree that IBM currently does. But migration to IBM's SNA might well prove to be more costly than migration to other network structures and protocols—one reason being that SDLC is not compatible with previous protocols. As far as we can tell, migrating to SNA means a complete changeover to SDLC terminals and network processors. Other network structures, on the other hand, seem to accept asynchronous protocols, BSC, as well as SDLC. Time will show whether IBM retains this posture.

Our view is that we generally favor the flexibility and the expected lower costs of the distributed control, distributed intelligence network structure. We think it will offer more freedom of choice to users. But with IBM pushing the centralized control approach, it seems very likely that a large number of user organizations will follow that approach.

REFERENCES

1. *Communications networks*, published by Online (Cleveland Road, Uxbridge, Middlesex, England), 1975.
2. *Network systems and software*, State of the Art Report 24, Infotech Information Limited (Nicholson House, Maidenhead, Berkshire, England), 1975, price £50.
3. *Proceedings of European Symposium on Large Scale Computer Networks*, organized by Datel GmbH (Darmstadt, Germany) and Tecsi, S. A. (Paris, France), 1975.
4. Reports prepared by U.S. National Bureau of Standards; order from Superintendent of Documents, U.S. Printing Office, Washington, D.C. 20402;
   a) Blanc, R. P., "Review of computer networking technology," T. N. 804, SD Cat. No. C13.46:804, price $1.55.
   b) Cotton, I. W., "Network management survey," T. N. 805, SD Cat. No. C13.46:805, price $1.20
   c) Neumann, A. J. et al, "A technical guide to computer-communications interface standards," T. N. 843, SD Cat. No. C13.46:843, price $1.50.
   d) Neumann, A. J., "User procedures standardization for network access," T. N. 799, SD Cat. No. C13.46:799, price 70¢.
5. Proceedings of the National Computer Conferences; order from AFIPS Press (210 Summit Avenue, Montvale, New Jersey 07645):
   a) *Proceedings of 1975 NCC*, price $50 paper, $15 microfiche.
   b) *Proceedings of 1973 NCC*, price $40 paper, $15 microfiche.
6. *Proceedings of 1975 symposium on computer networks, trends and applications*, order from IEEE Computer Society (5855 Naples Plaza, Suite 301, Long Beach, Calif. 90803), 1975, price $8.
7. *Introduction to minicomputer networks*, published by Digital Equipment Corp. (146 Main Street, Maynard, Mass. 01754), 1974.
8. Donnan, R. A. and J. R. Kersey, "Synchronous data link control: a perspective," *IBM Systems Journal* (IBM Corporation, Armonk, New York 10504), Vol. 13, No. 2, 1974; price $1.50.
9. *Proceedings of Fourth Data Communications Symposium, October 1975*, order from IEEE Computer Society (address above), price $20.
10. Pouzin, L., "The communications network snarl," *Datamation* (1801 S. La Cienega Boulevard, Los Angeles, Calif. 90035), December 1975, pages 70-72.
11. *Everything you always wanted to know about SNA*, published by Sanders Associates, Inc. (Daniel Webster Highway South, Nashua, New Hampshire 03060), 1975, price $1.
12. "System Network Architecture" (four papers), *IBM Systems Journal* (IBM, Armonk, New York 10504), Vol. 15, No. 1, 1976, pages 4-80; price $1.75.
13. Sanders, R. W. and V. G. Cerf, "Compatibility or chaos in communications," *Datamation* (1801 S. La Cienega Blvd., Los Angeles, Calif. 90035), March 1976, p. 50-55.
14. Carlson, D. E., "ADCCP—A computer-oriented data link control," *Compcon Fall 75 Digest of Papers* (IEEE Computer Society, 5855 Naples Plaza, Suite 301, Long Beach, Calif. 90803), p. 110-113; price $20.
15. *Introduction to Data Communications Systems* (student materials), IBM Corporation, Form No. ZR20-4542; see IBM local branch office or write IBM, 1133 Westchester Avenue, White Plains, N.Y. 10604 (for U.S. only), or to IBM World Trade Corporation, 821 United Nations Plaza, New York, N.Y. 10017 (for international).

SUBJECTS COVERED BY EDP ANALYZER IN PRIOR YEARS

1973 (Volume 11)

Number

1. The Emerging Computer Networks
2. Distributed Intelligence in Data Communications
3. Developments in Data Transmission
4. Computer Progress in Japan
5. A Structure for EDP Projects
6. The Cautious Path to a Data Base
7. Long Term Data Retention
8. In Your Future: Distributed Systems?
9. Computer Fraud and Embezzlement
10. The Psychology of Mixed Installations
11. The Effects of Charge-Back Policies
12. Protecting Valuable Data—Part 1

1974 (Volume 12)

Number

1. Protecting Valuable Data—Part 2
2. The Current Status of Data Management
3. Problem Areas in Data Management
4. Issues in Programming Management
5. The Search for Software Reliability
6. The Advent of Structured Programming
7. Charging for Computer Services
8. Structures for Future Systems
9. The Upgrading of Computer Operators
10. What's Happening with CODASYL-type DBMS?
11. The Data Dictionary/Directory Function
12. Improve the System Building Process

1975 (Volume 13)

Number

1. Progress Toward International Data Networks
2. Soon: Public Packet Switched Networks
3. The Internal Auditor and the Computer
4. Improvements in Man/Machine Interfacing
5. "Are We Doing the Right Things?"
6. "Are We Doing Things Right?"
7. "Do We Have the Right Resources?"
8. The Benefits of Standard Practices
9. Progress Toward Easier Programming
10. The New Interactive Search Systems
11. The Debate on Information Privacy: Part 1
12. The Debate on Information Privacy: Part 2

1976 (Volume 14)

Number

1. Planning for Multi-national Data Processing
2. Staff Training on the Multi-national Scene
3. Professionalism: Coming or Not?
4. Integrity and Security of Personal Data
5. APL and Decision Support Systems
6. Distributed Data Systems
7. Network Structures for Distributed Systems

(List of subjects prior to 1973 sent upon request)

PRICE SCHEDULE

The annual subscription price for EDP ANALYZER is $48. The two year price is $88 and the three year price is $120; postpaid surface delivery to the U.S., Canada, and Mexico. (Optional air mail delivery to Canada and Mexico available at extra cost.)

Subscriptions to other countries are: One year $60, two years, $112, and three years $156. These prices include AIR MAIL postage. All prices in U.S. dollars.

Attractive binders for holding 12 issues of EDP ANALYZER are available at $4.75. Californians please add 29¢ sales tax.

Because of the continuing demand for back issues, all previous reports are available. Price: $6 each (for U.S., Canada, and Mexico), and $7 elsewhere; includes air mail postage.

Reduced rates are in effect for multiple subscriptions and for multiple copies of back issues. Please write for rates.

Subscription agency orders limited to single copy, one-, two-, and three-year subscriptions only.

Send your order and check to:
EDP ANALYZER
Subscription Office
925 Anza Avenue
Vista, California 92083
Phone: (714) 724-3233

Send editorial correspondence to:
EDP ANALYZER
Editorial Office
925 Anza Avenue
Vista, California 92083
Phone: (714) 724-5900

Name_____

Company_____

Address_____

City, State, ZIP Code_____