Transmission Control Protocol/Internet Protocol (TCP/IP)

In this report:

Datapro Summary

The Transmission Control Protocol/Internet Protocol, or TCP/IP, comprises two specific protocols existing within a protocol stack often referred to as the TCP/IP protocol suite, the DOD protocols, or the Internet Protocol suite. TCP and IP follow layered networking concepts, occupying middle layers four (transport) and three (network) of the OSI Reference Model. Each element of TCP/IP is a unique, functional layer of protocol: TCP provides traditional virtual circuit, byte stream-oriented communications services for programs using Application layer protocols, including end-to-end flow control, error control, connection setup, and status exchange; and IP provides a datagram-oriented gateway service between subnetworks, so that hosts can access other hosts. The network environment consisting of interconnected subnetworks is called an "internet." TCP/IP has steadily grown in popularity and has been implemented on nearly every type of computer, from PCs to mainframes. It represents a communications service's "lowest common denominator," providing peer communications and projection of higher level services. In LAN environments, it is most popular and successful in conjunction with Ethernet. Other protocols, including GOSIP, however, now vie for prominence in TCP/IP's realm.

Background

A surprising number of network products and nomenclature are rooted in a common event—the development of an interfacility communication network linking laboratories and research centers of the Defense Advanced Research Projects Agency (DARPA). Called ARPANET, the network was one of the very first layered communication networks, even though with its 1960s development date, it preceded the formal ISO Reference Model by nearly a decade. ARPANET, backed by government funding, was a prototype for many computer network concepts widely available today, including packet switching and peer communications.

As ARPANET advanced and matured, it often faced (and met) issues that were to be commercially significant only much later. One such issue was the general problem of reliable data interchange between unlike computer systems. ARPANET played host to computers of a dozen varieties, connected in various subnetworks, and the problem had to be solved across manufacturer lines. It

—By L. Michael Sabo
Communications Network Consultant, SSDS, Inc.

was, and the solutions were a guide for later commercial protocol development. But often the ARPANET environment was deemed too complex for the more financially sensitive mass market, and ARPANET protocols were rarely transported directly.

An important exception to this was the transport and internet portion of ARPANET, called the Transmission Control Protocol/Internet Protocol, or TCP/IP. TCP/IP is two specific protocols existing within a protocol stack often referred to as the TCP/IP protocol suite, the DOD protocols, or the Internet Protocol suite. The Internet is worldwide interconnections of numerous research networks including NSFnet, Defense Data Network (DDN), and networks from such agencies as the Department of Energy and NASA. It was technically feasible to internet these networks using TCP/IP as the "glue."

TCP, as a formal transport layer, evolved from the original NCP transport layer of ARPANET specifically to deal with the problems of reliable communication through essentially unreliable subnetwork interfaces, such as the linkages to datagram networks or packet radio environments. The Network Layer of TCP/IP is the Internet Protocol, designed to route information through networks of networks termed internets. TCP/IP, the "middle" of a full-layered protocol, seems an unlikely commercial star, yet its popularity has grown steadily, and now is experiencing what can only be called an explosion of implementations.

Commercial Success Story

Many TCP/IP product vendors agree that the protocol's technical characteristics have little to do with its market success. When the problems of LAN communications among dissimilar computers were becoming critical, there were fewer solutions than are available today. Xerox Corp.'s XNS, a proprietary competitor to TCP/IP as a LAN transport/internet product, was the obvious heir to the market, but Xerox failed to "open" XNS to developers quickly enough. TCP/IP was always open, thereby becoming the market standard.

Additionally, TCP/IP's acceptance at the commercial level is attributed to four significant events. First, commercial interest in formal transport/internet protocols is a relatively recent phenomenon, and by the time it matured, microprocessor and memory technology had advanced to the point where complex formal protocols such as TCP/IP could be implemented on nearly any type of product without crippling financial effects. Second, DARPA funded the development and integration of TCP/IP into the University of California's Berkeley Distribution (BSD) UNIX operating system, thus ensuring that TCP/IP would be widely deployed. Third, one environment that TCP/IP was designed to support was the contention-oriented ALOHA packet radio environment—an environment similar in many ways to the carrier-sense multiple access/collision detection (CSMA/CD) technology used for local area networks (LANs).

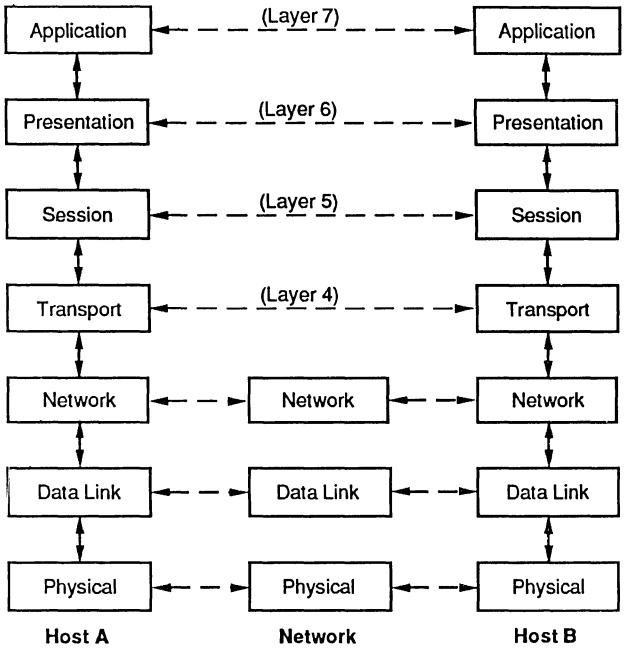
LANs have other common elements with the Internet environment, including the need to support multiple-vendor environments. LANs are often linked in internetwork environments, the very thing the Internet Protocol was designed to support. Additionally, TCP/IP is an attractive standard for cost-conscious vendors. TCP/IP's coding was developed with public funds; its specifications are in the public domain.

Layered protocols following the ISO Model have a characteristic called "peer protocol" interaction, which stipulates that each protocol layer maintains a conversation with a peer at the other side of the connection. Thus, for either station to use a protocol, both must. Because the TCP/IP suite was widely available on many minicomputer systems, it became a logical platform on which to base early full-service LAN environments. This has provided an impetus not only to maintain its use but to expand it to other systems that might want to communicate with those already supporting TCP/IP.

Transport and Internet Issues

Figure 1 shows an ISO Model connection, with each layer and peer protocol indicated. As the figure shows, Layers 1 through 3 of the protocol define "local" procedures, which have peer layer members in each network element or on both sides of any connection. Layers 4 through 7 represent "end-to-end" procedures, protocols which serve

Figure 1.
Formal Model of an ISO Standard Connection



Layers 1 through 3 define "local" procedures; Layers 4 through 7 represent "end-to-end" protocols.

the end user directly and reside in equipment serving that user only. Layer 2 is responsible for accurate transmission of information between any two communicating points. Layer 3 is responsible for the routing of information among the nodes of the network so that it reaches the intended destination. Layer 4 represents the lowest end-to-end protocol, and is responsible for the reliable end-to-end transmission of information.

That Layer 4 and Layer 2 have similar goals is clear, but the reason for the division of functionality is that the action of the routing and guiding functions of Layer 3 may create an environment where data is lost not while on a line but while in a queue at a node, or where data is discarded according to "congestion rules" when a node is heavily loaded. Layer 4 must "equalize" the risks attendant in Layer 3 activity, doing whatever is needed to ensure the user that one—and only one—copy of each information element sent is actually received.

Just how much Layer 4 has to do depends on what strategy is used at Layer 3. There are two primary alternatives:

- **Virtual Circuit** network and internet structures, where the network nodes establish a virtual path between communicating parties when a connection is established, and subsequent data messages follow that path.
- **Datagram** network structures, where each information element is addressed to its destination and dispatched to the mercy of the network nodes, to be routed by the best path available to the destination.

Virtual circuit networks have the following characteristics:

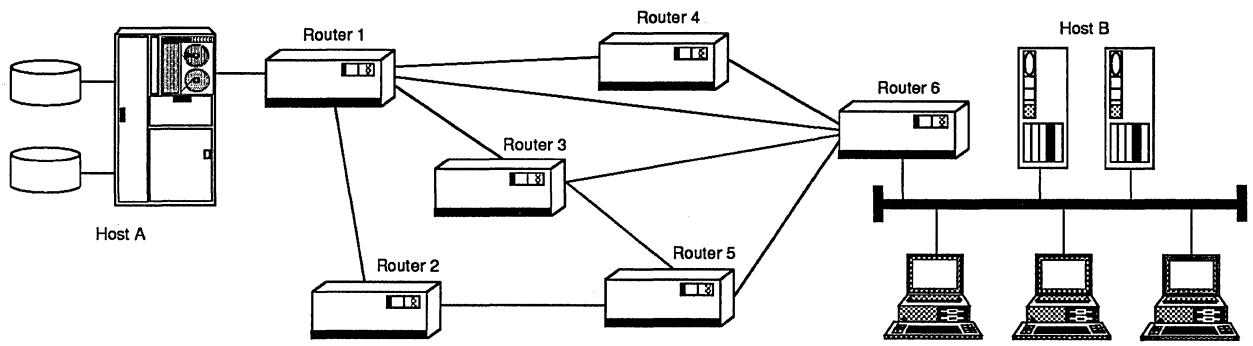
1. The network nodes must maintain a table of addresses and destination routes so that calls can be connected. The maintenance of these routing tables is a vital task, because an error or omission in the table will result in call failures.
2. A break in the routing chain disrupts the connection, because the destination address has been discarded and an alternate route cannot be looked up.
3. Information always follows the same path, so it cannot get out of order or be duplicated as long as the individual line procedures (Layer 2) ensure accurate transmission and as long as intermediate nodes do not discard anything. If they do, the nodes can "reset" the circuit and notify the users to recover the lost data.

Datagram environments are almost everything that virtual circuit environments are not, as Figure 2 shows. In the datagram network, a user simply pushes an information element into the network bearing the address of the destination. Routing algorithms within each node do the rest. There is no "memory" of a formal call setup.

Datagram networks have the following characteristics:

1. While routing tables may be used in some environments, datagrams may be routed in a "tableless" manner. One frequent concept is "broadcast routing," where every node receiving a datagram sends it on all its available lines. Such a technique may sound wasteful, but where network traffic is relatively light, it can simplify network routing tremendously.

Figure 2.
Virtual Circuit Routing



In comparison with complex virtual circuit call setups, datagram routing is simple. In the illustration above, messages destined from Host A to Host B are routed from router 1 to router 6, then over the LAN to Host B. As the illustration shows, alternate routes exist. Should the circuit from router 1 to router 6 go down, an alternate route will be selected "on the fly," without disruption of service.

2. There is no fixed route between nodes, so no special action is needed to reestablish a path if a line or node fails. If each node "knows" which output lines are working, it sends only on those lines. If any path at all can be found from source to destination, communication is possible.
3. Because consecutive information elements may take different routes to the destination, they may arrive in other than the sequence in which they were sent. Some routing algorithms, such as the broadcast scheme already mentioned, will result in duplicate message elements if two or more possible paths exist to the destination.

TCP/IP was designed for the datagram environment, an environment where the Network Layer may lose information, duplicate information, or shuffle it unpredictably. According to The Wollongong Group, leaders in commercial development of TCP/IP, robustness and reliability are the technical points that attracted early commercial vendors to TCP/IP. At a time when protocol issues were mysteries overall, transport and internet issues were simply magic; few developers cared to face the problems of failures at this layer. Selecting a protocol that had a proven record of performance in the most exotic and demanding environments simply made good sense.

The commercial growth of TCP/IP can thus be traced not to the network and transport issues but to the demand for higher layer services and improved connectivity at a time when these issues were really not well understood. On the face of things, TCP/IP would seem totally inapplicable to environments and of marginal commercial interest. In fact, it is neither.

TCP/IP and the LAN

Most companies do well over 75% of all their communicating at the local layer. Statistically, a company is thus likely to face most of its internetworking problems with LANs. Early attempts to link minicomputers and systems of other sizes to integrate information for centralized management and distribution ran afoul of the problems with protocols and connections. Even where systems had common link-layer access techniques to apply, these techniques failed to provide for reliable end-to-end communication. The classical example of the X.25 call which is cleared with packets queued in the network demonstrates that the ISO Model meant what it said about the responsibility for end-to-end delivery assurance—it does not reside in the network.

Host-to-host communication service is an application of the type that TCP serves with its concept of "sockets." A host can be viewed as having a collection of "sockets" into which connections can be "plugged." Some of these sockets represent standardized facilities (the well-known socket), and others are "free" to respond to the caller's requirements. This concept is much like the view of distributed network service provision

evolving in such commercial architectures as IBM's SNA—where it is called SNA Distribution Services, or SNADS. TCP/IP fits the requirements of the applications admirably.

Surprisingly, it also fits the network requirements as well. The datagram environment, which TCP/IP is designed to support, bears a surprising resemblance to the broadcast LAN technologies, particularly CSMA environments:

- **Routing.** LANs of most types do not really "route" at all; they merely assume that each station will see every message and acquire those that are addressed to it. This is philosophically much like the broadcast datagram routing concept.
- **Connections versus connectionless.** LAN activity is rarely "call" oriented. Instead, it often is generated by programs that "think" they are interacting with a local resource, such as a disk or printer—one to which no call would be necessary.
- **Administrative load sensitivity.** LAN administrators are not unknown, but few firms would desire assigning someone to maintain routing tables and lists of names.
- **Missing and extra data.** Although LANs do not traditionally lose or duplicate data, gateways and internet portals may sometimes provide multiple paths between environments, causing both conditions.

In short, TCP/IP was designed for an environment very much like the one presented by a multivendor LAN. And that is exactly the environment for which it is proving most commercially interesting.

TCP/IP Concepts

ARPANET, as a network-linking research center, had from the start a strong host-to-host and "roving terminal" orientation. Telnet, a famous ARPANET protocol, defined the first commercial version of a "network virtual terminal" and let network users access hosts without regard for the local host protocol. The socket concept provided a means of projecting network services by identifying communications points at which they could be accessed and protocols to connect with those points. The ARPANET environment tended to encourage its members to develop their own site networks, linked to ARPANET through a common host. These networks were not controlled by DOD specifications and, therefore, took many forms.

TCP/IP was first designed to allow the members of the ARPANET environment to interact between ARPANET and private subnetworks developed by each of the members. The network environment consisting of interconnected subnetworks is called an "internet." It was assumed that these subnetworks would be the result of independent developments and would, therefore, be compatible only by accident, and that each might be made up of different types of computers and use different low-level communications technologies (low layer in the ISO sense meaning at layers below Layer 3). For this reason, a basic design premise of TCP/IP was to provide for the ability to communicate using nearly any environment that made a reasonable effort to transmit and receive bits. This "robustness" is a very rare quality in communications products because its development cost is often prohibitive.

But a review of an ISO protocol model reveals that there are two interfaces at any layer—to the one below and to the one above. At the one above layer, TCP/IP was designed to present a program-layer interface that was relatively independent of any architectural features of the host, including memory, performance, etc. Thus, TCP/IP provides for the interfacing of host (program)-layer processes of any power and performance layer.

It is the combination of these features—independence of host performance at the higher layer interaction points and of communications protocols and facilities at the lower layer interaction points—that forms both the design basis and the commercial justification of TCP/IP.

TCP/IP Architecture

TCP/IP is almost always referred to in this unified form, when, in fact, each element (TCP and IP) is a unique, functional layer of protocol:

- **Transmission Control Protocol (TCP)** provides traditional virtual circuit, byte stream-oriented communications services for programs using Application Layer protocols, including end-to-end flow control, error control, connection

setup, and status exchange. Without it, programs could not be ensured of send/receive order, correctness, etc.

- **Internet Protocol (IP)** provides a datagram-oriented gateway service between subnetworks, so that hosts can access other hosts. IP does not enhance the reliability or exactitude of the datagrams—it only lets them be bridged from one subnetwork to another. IP also provides fragmentation and reassembly so that large IP datagrams can be transferred over networks with small maximum packet sizes. Figure 3 depicts the TCP/IP model.

Transmission Control Protocol

TCP is an OSI Layer 4 Transport Layer function and provides the following functions:

1. Data transfer support, providing a virtual circuit connecting the called and calling user, regardless of the lower layer delivery system. (IP, in fact, is a datagram service.)
2. Error checking, including detecting lost information elements, duplicate information elements, and out-of-sequence information elements.
3. Flow control, to prevent the "swamping" of one user by another faster or more powerful user.
4. Multiplexing and demultiplexing Application Layer connections.
5. Status and synchronization control, including the ability to set up and break connections, to mark significant points in the dialog, etc. This includes the ability to signal an unusual event (interrupt).

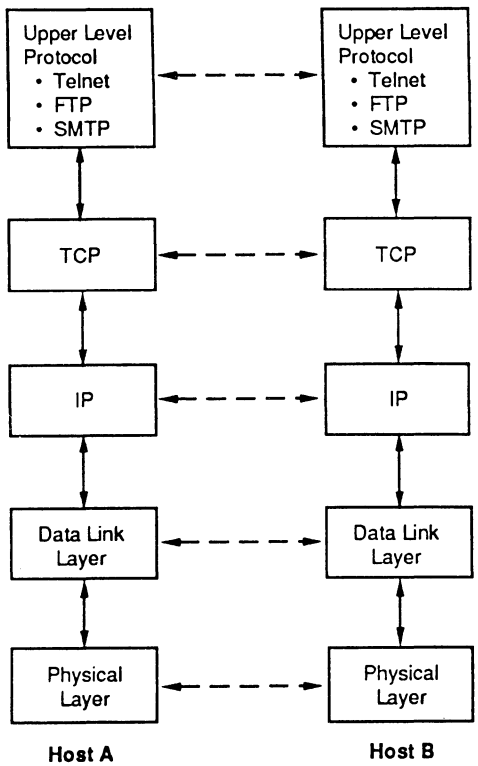
The data flow for TCP is unusual with respect to other protocols in that it provides only one type of Transport Data Unit (TPDU), termed TCP segment. Figure 4 shows the TCP/IP header, which is 20 to 24 bytes in length—long by any protocol standard.

Internet Protocol

IP is a Layer 3, or Network Layer, function in the ISO parlance. Such a layer is normally responsible for routing and delivery. In a multisubnet environment, an Internet Protocol is responsible for the following general communications areas:

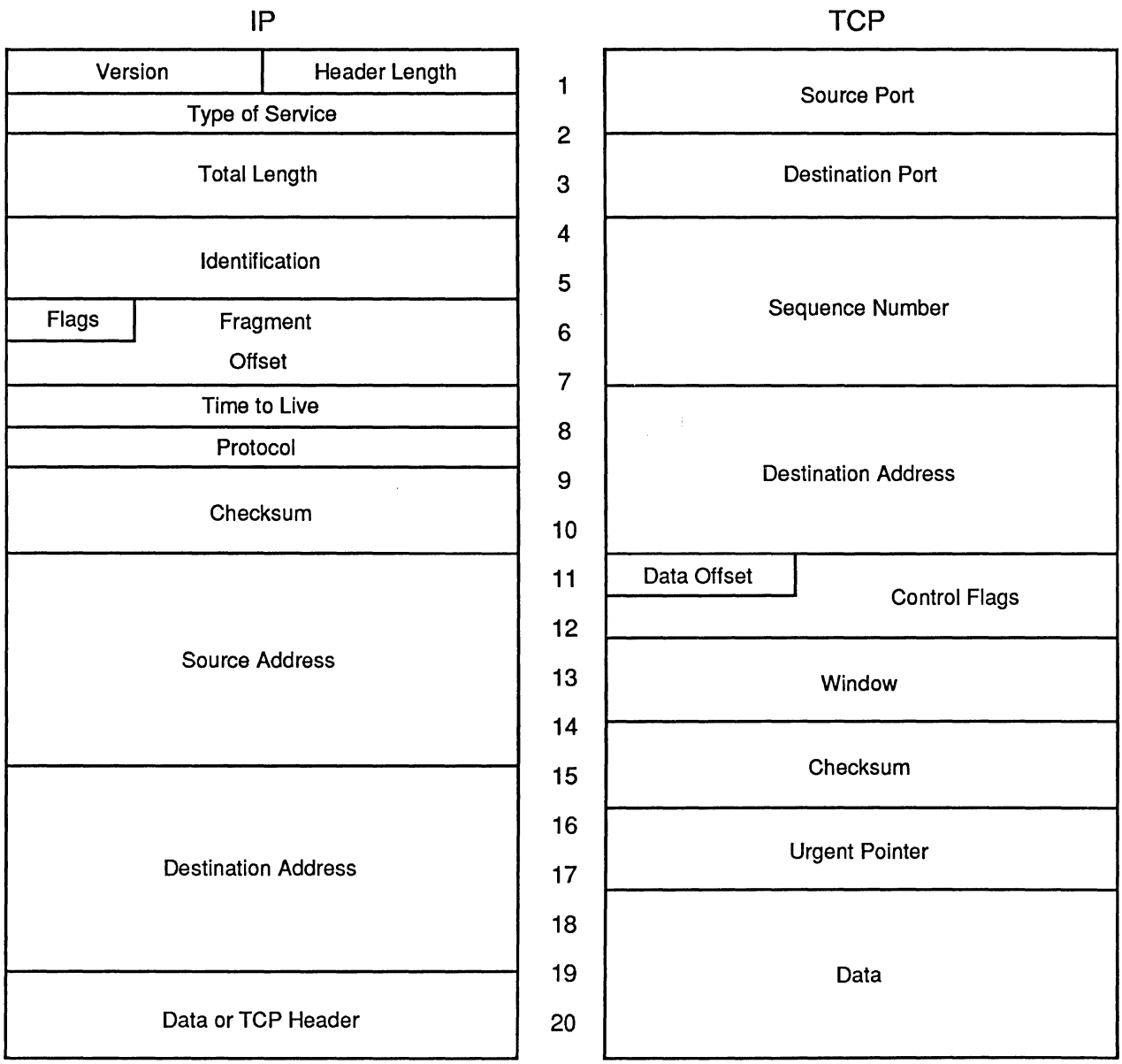
1. *Name control and translation.* Members of subnetworks may not have the same naming conventions, and there may not be common naming control. IP assumes an internet-wide naming convention generated by combining a network ID assigned by the network administrator and a local host address.
2. *Status translation and communications.* The result of internet operations must be communicated to the local user in a form that the user can understand. In addition, the operations that a user performs must return status conditions to alert the user to problems. IP provides four types of status messages: destination unreachable/invalid, time-out, parameter error, and redirect requested. The first three are largely self-explanatory; the last means that another gateway has a shorter route to the destination.
3. *Routing.* IP provides for the exchange of Gateway-Gateway Protocol (GGP) messages

Figure 3.
TCP/IP Model



Although referred to by a single acronym, TCP/IP is actually two separate protocols, each occupying a different layer of the OSI Reference Model.

Figure 4.
TCP/IP Header



Protocol headers for TCP and IP, showing the complexity of internetworking protocols. The IP header precedes the TCP header (or appends on top) in the protocol stack.

to determine the status of gateways and their related hosts. GGPs can also be used to verify the operability of each interface; "probes" are sent to interfaces, and failure to return several successive probes causes the interface to be marked down. In addition, IP provides a time-to-live parameter, a number that is decremented as the IP datagram is processed by a router. Once it reaches zero, the datagram is discarded. Although seemingly crude, it is an effective method of stopping a datagram from endlessly looping around the network.

4. *Management.* IP provides a vehicle for operation center management of the internet environment, a vehicle that provides for control and information gathering.
5. *Fragmentation and Reassembly.* IP is capable of routing datagrams into networks that are incapable of handling the datagram's size via a technique called fragmentation. The original

datagram is subdivided into pieces small enough for transmission over the destination network.

6. *Type of Service.* This indicates that a datagram would prefer any or all of the following services: low delay path, high bandwidth path, and high reliability path.

The IP appends an additional header on the TCP information element, as shown in Figure 4. This header is a minimum of 20 bytes and can be 24 or more. Source and destination addresses are composed of a one-byte network ID and a three-byte host ID. The definition of subaddresses within a host is the responsibility of the TCP layer.

TCP/IP Applications

TCP/IP has been implemented on nearly every type of computer; therefore, it represents a communications service's "lowest common denominator" that provides for peer communication and projection of higher layer services.

On the wide area side, TCP/IP is often used with X.25—more often, in fact, than X.25 is used with its "formal" ISO Transport Protocol companion. Vendors have successfully layered ISO protocols above TCP/IP as well, protocols such as the ISO File Transfer and Management (FTAM) protocol, Common Application Services Element (CASE), and the X.400 message exchange standard.

Heterogeneous networks most likely exist as local networks, however, because corporations most often encounter dissimilar product interface requirements when coordinating application-specific computer systems. In a LAN environment, TCP/IP has been most popular and successful in conjunction with Ethernet. The reasons for this follow:

1. Ethernet is the most common LAN for personal computers and minicomputer products, succeeding those systems on which the original DOD ARPANET network was based. Since TCP/IP evolved with the systems and is available on Ethernet-compatible computers today, the marriage naturally developed.
2. The distance limitations of Ethernet (a product of the propagation delay effects on the CSMA environment) dictate the use of network bridges and internet structures when applied to a campus or industrial park environment. This multisubnet environment is just what TCP/IP was designed to support.
3. AT&T UNIX system vendors have tended to adopt Ethernet as a kind of LAN counterculture to the IBM LAN solutions, and UNIX software has long supported both TCP/IP and multivendor environments.

Upper Layer Protocols

Most commercial implementations of the TCP/IP protocol suite include at least three standardized upper layer protocols: Telnet, File Transfer Protocol (FTP), and Simple Mail Transfer Protocol (SMTP). These protocols are easy to use, and users are shielded from the particulars of the underlying TCP/IP protocols.

Telnet

Telnet builds on TCP services to provide virtual network terminal services. Telnet implementations at both ends of the TCP connection translate from their local terminal types into the standard network representation. This permits each end of the communication to implement only what is known locally, without needing to translate into every other possible terminal type available.

Telnet is composed of two independent protocol entities: a client portion and a server portion. The client portion implements the connection to the actual terminal or terminal emulation requesting a connection from the user's location. Telnet's server portion is on the receiving end of the session. Server Telnet is present in heterogeneous systems that permit users to log in over a network.

File Transfer Protocol (FTP)

FTP uses TCP services to control the exchange of files between two hosts. Like Telnet, FTP consists of both a server portion and a client portion. Server FTP listens at a Well-Known Port for the incoming sessions from FTP clients. Client FTP is initiated by the user process and makes the connection with a remote FTP server. Files may be transferred in either direction; ASCII and binary file transfers are supported. FTP can also support third-party transfers where a user establishes an

FTP session on a local host to transfer a file between two remote hosts.

Simple Mail Transfer Protocol (SMTP)
SMTP uses TCP connections to transfer text-oriented electronic mail. SMTP supports the exchange of electronic mail between users on the same host or between hosts over a network.

The Future

The U.S. government endorses the OSI protocols through the Government Open System Interconnection Profile (GOSIP). GOSIP is a superset of the OSI standards, and had been specified for all government procurements after August 1990. That is not to say that the U.S. government is dropping TCP/IP in favor of OSI; quite to the contrary. Recently, the Department of Treasury and the Department of Veterans Affairs awarded major programs which were based on TCP/IP, with gateways to support OSI interoperability.

Other protocols vie for prominence in TCP/IP's realm.

- **ISO Transport Protocol**, the offshoot of an international effort to develop a Layer 4 protocol suitable for environments of widely varying complexity, has now been implemented successfully in a number of wide area network environments. The protocol has somewhat less overhead than TCP/IP, is well specified, and has the valuable mantle of ISO approval. Digital Equipment Corp. is now standardizing on the ISO Transport Protocol within its DECnet environment, and Digital represents the cornerstone of the Ethernet market.
- **IBM SNA**, particularly in its Advanced Program-to-Program Communications (APPC) or LU6.2 peer communications protocol, is a growing influence in the LAN market. IBM has a product for the PC that implements LU6.2, and similar support is available on most minicomputers. In many ways, APPC/LU6.2 is superior to TCP/IP as a universal high-layer protocol. Although it is widely supported, it shares with ISO Transport Protocol a general lack of LAN support. IBM's APPC/PC product is supported on its PC LANs, but most other computer vendors support LU6.2 only as a wide area SNA gateway protocol.
- **Versatile Message Transaction Protocol (VMTP)** has made significant advances over existing transport protocols, and in the next several years may be adopted as a new protocol standard. VMTP is designed to offer extremely high throughput for transaction-oriented applications and supports process migration, multicast transmissions, security, datagram service, and a priority mechanism.
- **Xpress Transfer Protocol (XTP)** is designed to address the need for reliable and efficient multicast communications over high-speed connections. XTP's draft specifications are also available from Protocol Engines, Inc. of Santa Barbara, CA. ■

This Datapro report was updated by L. Michael Sabo, a communications architect with SSDS, Inc., Littleton, CO. Mr. Sabo is currently consulting on various networking and internetworking projects. Previously, he participated in porting TCP/IP to the emerging ANSI High-Performance Parallel Interface (HIPPI) Gigabit/sec. LAN standard. Mr. Sabo has been active in integrated network management. He participated in developing an object-oriented and SNMP-based network management architecture for Lockheed Integration Services. This effort included defining numerous private enterprise management information base (MIB) objects to support system management functions.

Mr. Sabo is a member of the SNMP working group and has been active in the Internet for six years. He is a member of the board of advisors for *Datapro Network Management*. He holds a master's degree in Data Processing Management from the University of Denver and a bachelor's degree in Computer Science from Wright State University.