TCP/IP versus OSI

Douglas Meyer
George Zobrist

The battle of the network standards

During the past several years, the computing community has witnessed an unprecedented growth in the area of networking and its associated technologies. While such rapid bursts in technology are usually something to be savored in our profession, it is unfortunate that this one has occurred in such an unconstrained fashion. Unlike most breakthroughs of this type, where the ultimate winner is the consumer, the multitude of networking products and standards have created almost as many problems as they have solutions.

The problem statement is simple. What the user community demands from the computer industry is true inter-vendor connectivity, free of all the usual hacks and special boxes that installations have become so accustomed to when incorporating new hardware into their existing systems.

The days of installations being dedicated to one specific vendor's products are gone. The computing market place is no longer dominated by a select few who can dictate standards based solely on the width of their customer base. The time has come for the players involved to bury their hatchets and adopt a common standard for network communications. A standard which is robust enough to meet the network needs of the user community, and one which is flexible enough to adapt to the constantly changing technology.

In the course of time, two vendor-independent network protocols have risen to the top of the list of contenders for the title of "THE" standard. These are the Transmission Control Protocol / Internet Protocol (TCP/IP), which was developed in the late 1960s as a research project conducted by the United States Department of Defense, and the Open Systems Interconnection Reference Model (OSIRM, or simply OSI), developed by the International Standards Organization (ISO) in the mid-1970s. While each attempts to provide the end user with a standardized method of accessing the underlying physical network, they differ in their basic structure and in the way they go about achieving their respective ends.

Reprinted from *MIPS Magazine* (permission of Computer Metrics, Inc.), copyright 1989.

Medium access/Logical link control

If any given networking system is studied from the bottom up, the first topic discussed must be what's known as the "physical" (as opposed to the "logical") network. It is at this level of the network that the connecting hardware, its operational characteristics, and its basic modes of data transfer are considered. While an exhaustive coverage of the various types of physical networks is beyond the intention of this article, a brief discussion is in order since it is at this point that both TCP/IP and OSI must meet on common ground.

Developed in the mid-1980s by the Institute of Electrical and Electronics Engineers (IEEE), the 802.x standard set attempted to stringently define this level of the network structure for several of the most popular methodologies. The most notable of these are the CSMA/CD networks (IEEE 802.3), token bus (IEEE 802.4), and token ring (IEEE 802.5). Defining the methods by which the most fundamental units of data are transferred from host to host, these standards, along with their less common counterparts, collectively make up what is known as the Medium Access Control, or "MAC," layer. This layer, as will be seen, is essentially a common member of both TCP/IP and OSI implementations.

Since each member of the MAC layer is different operationally, it was thought advantageous to shield these differences from the higher level systems in order to make the transition from one type of physical network to another as painless as possible. To accomplish this, a layer was created that would sit logically on top of the MAC layer and present a standardized set of network functions (i.e., a "logical" network), regardless of the exact details of the lower level implementation. This layer is known as the Logical Link Control (LLC) layer and is defined by the IEEE 802.2 standard.

As defined by the standard, two classes of LLC exist, known as LLC-1 and LLC-2. The simplest of these, LLC-1, provides little more than the functions already described. It allows the transmission of datagrams with neither guarantees nor feedback to the outcome of the transmission. The remaining class, LLC-2, provide various enhancements to the services provided by LLC-1. These include functions such as error correction and more reliable forms of data transmission.

Since the level of abstraction provided by the LLC is both desirable and complete, further discussions will ignore the details of the MAC layer altogether.

The packet switched foundations of TCP/IP

During the design of TCP/IP, it was the intent of those involved to create a network protocol that would be capable of tying together an extremely large number of both military and academic organizations scattered across the continent. In a network of this magnitude, although the total connectivity of any pair of nodes was a fundamental requirement, it was obviously infeasible to provide direct connections from every host to every other host. Thus it was decided that the only way to efficiently accomplish this scale of connectivity while maintaining the

0278-6648/90/0002-0016$01.00©1990 IEEE

required high degree of reliability was to model the underlying structure as a "packet switched" network.

In the "packet switched" model, physical connections between the network's hosts are structured on an "N nearest neighbors" basis. Each node on the net only has direct connections to some number, N, of its geographical neighbors and as such can only communicate *directly* with these N hosts. Communications to any hosts other than these neighbors must take place indirectly, with the data being routed through some number of intermediate hosts. With this as the case, it should be noted each connected installation actually plays two roles in the operation of the network. First, as is true with any computer network, it is a provider of some type of resource which is sharable across the net. In addition to this basic role, however, it also is required to provide a certain amount of routing or "gateway" service for its neighboring hosts.

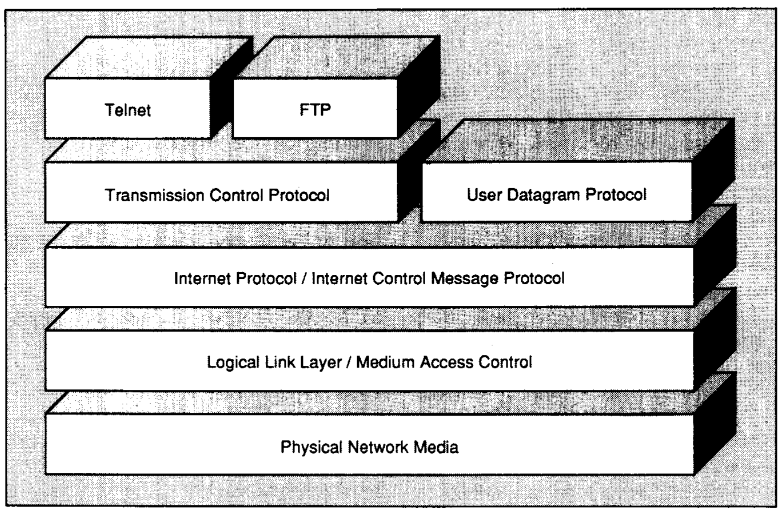
In a network of this type, the basic unit of information which is passed from host to host is known as a "packet" or "datagram." Each datagram has integrated into it the address of the destination host; an address which is, from a functional perspective, not too terribly different than the address one might place on a letter. It uniquely identifies the single host to which the packet is to be delivered, but says nothing about the route which must be taken to get there. The determination of this routing is instead delegated to the hosts through which the packet passes.

Each host has, within a limited perspective, knowledge of its local environment. With the proper handshaking between it and its neighbors, a host has the ability to compile statistics detailing the volume and status of network traffic in the immediate area. Using this information, when a host receives a packet which must be forwarded, it is able to intelligently choose one of its immediate neighbors as the "best" next step. This "handing off" continues, with local routing decisions being made at each step, until eventually the packet is passed to the host for which it was originally intended.

At first glance, it may appear that all this handling might decrease the effectiveness of the network. Such is not the case. In practice the packet switching philosophy actually improves the overall reliability of the network dramatically.

Functional layers of TCP/IP

To support a network as that described, TCP/IP is organized as a set of functional layers. Within each of these layers, a well defined set of network services are provided, services which are in turn implemented utilizing the services provided by the layers below it. This functional stratification and the levels of abstraction it provides is an important concept and worthy of note here. At the lower layers, services tend to be very simplistic and streamlined. As successively higher layers are studied, functions become more and more complex. By providing additional network services in stages such as this, users and applications can jump in at whatever level provides the degree of service necessary without encumbering the overhead of the higher, unnecessary functions. Admittedly, this structuring is not unique to the design of TCP/IP. As will be seen, it is also a fundamental concept in the design of the OSI model.



Structural layers of TCP/IP.

Internet protocol/Internet control message protocol

The lowest layer of TCP/IP, the layer which resides immediately above the MAC/LLC level discussed earlier, is known as the IP or Internet Protocol. The responsibilities of the IP, as indicated by its rank in the overall structure, are very simple. First of all, it provides mechanisms by which individual datagrams can be transmitted and received over the network. These services, however, are provided by the IP without any guarantees to the reliability or correctness of the data transferred. While a portion of this layer, an extension to the formal IP standard known as the Internet Control Message Protocol, does handle the *reporting* of certain low level errors, no action is taken at this level to correct errors. The creation of a truly reliable communications environment is instead delegated to the next higher layer in the TCP/IP structure.

The second function performed by the Internet Protocol is that of datagram routing. As was mentioned earlier in the discussion of packet switched networks, when one host receives a datagram which is addressed to another host, it must attempt to route it one step closer to that destination. It is the responsibility of the protocols at this level to perform this routing in an intelligent fashion, basing the decision on the current state of the network around it.

User datagram protocol

The next higher layer within TCP/IP is actually comprised of two, distinct protocols known as the User Datagram Protocol and the Transmission Control Protocol. The first of these, the UDP, is designed to provide the user with direct access to the services (and the shortcomings) of the lower, IP layer. Data transmitted via UDP functions are considered to be datagrams and, as such, are guaranteed neither delivery nor correctness. While these shortcomings are generally too serious for the casual user or application, this protocol does play a useful role in the TCP/IP hierarchy. For example, in the development of new network systems it is obviously helpful to have a standardized, streamlined interface to the physical network. Such an interface would ideally provide only minimal capabilities so as not to duplicate any functions or services of the new system under development. It is the UDP which provides for just this type of interface.

Transmission control protocol

Parallel to UDP in this level of the TCP/IP structure is the Transmission Control Protocol. The purpose of this protocol is twofold. Its first and foremost job is to perform whatever steps are necessary to take the functionality of the lower, IP layer, as unreliable as they might be, and create a reliable transport mechanism that guarantees the correct delivery of all data. To accomplish this, TCP performs a complex series of "handshaking" operations for each transmitted datagram to ensure that it was correctly received. When an error does occur, the datagram is continually retransmitted until it arrives successfully. While an exacting discussion of just how these "handshaking" operations take place is beyond the scope of this article, let it suffice to say that they are sufficient to provide the required guaranteed level of service.

TCP also provides what is known as "connection-based" or "virtual circuit" communications to the other hosts on the network. Due to the dynamic routing which takes place in a packet switched network, fixed communication paths between hosts seldom exist. Several packets originating from one host and addressed for another host could very well take different routes to their common destination. This mode of operation, while advantageous for reasons stated earlier, is often difficult for users to comprehend and work with. They often demand the creation of a fixed circuit between themselves and another host. Since a physical implementation of this circuit is usually impossible (and certainly undesirable), it is instead simulated by TCP. Users request and receive what appears, from their point of view, to be a direct data path to the desired machine. Data exchanged over that path would seem to flow directly between the hosts on either end. In actuality, TCP intervenes in the transmission process. It fragments the data into some number of sequenced datagrams, and sends them (in normal datagram fashion) to its counterpart layer on the other end. Once there, the remote TCP layer assembles the packets so as to recreate the original data and passes it on to the recipient.

TCP/IP applications

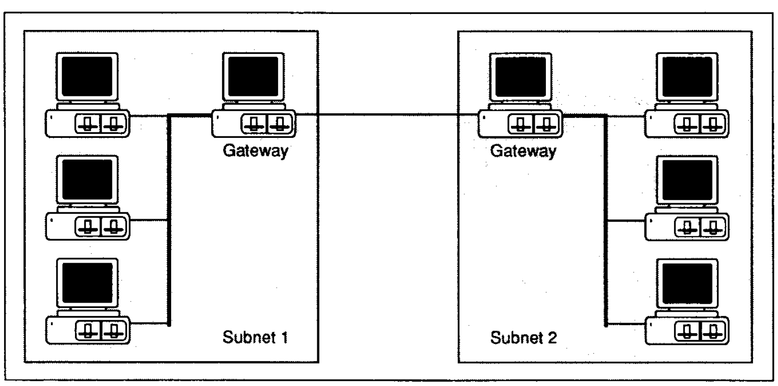
Built on top of the TCP services, residing at the highest level of the TCP/IP structure, are usually a set of network applications or utility programs. While the number and function of these applications vary from installation to installation, some have fallen into such common use so as to be thought of as integral parts of TCP/IP. Among these are TELNET, a terminal emulation program used to establish remote login sessions and FTP, a file transfer program which supports the high speed transfer of both text and binary data between hosts.

OSI reference model

The International Standards Organization was established in 1978 to bring about greater conformity in the design of communications networks. The ultimate goal of the ISO was to develop a standard that would reduce connectivity costs by increasing inter-vendor compatibility. The Open Systems Interconnection Reference Model was designed to meet this goal. The model is comprised of a series of standard protocols that enable application compatibility and full communication between participating systems of different manufacture and design. Regarding the role of this standard, ISO formally states: "The purpose of this International Systems Interconnection is to provide a common basis for the coordination of standards development for the purpose of systems interconnection, while allowing existing standards to be placed into perspective within the overall Reference Model."

Physical network model of the OSI standard

The OSI model was originally designed to operate over standard packet switched networks, and in that regard it shares many of the low level characteristics of TCP/IP. The theoretical way in which OSI views the underlying network, however, differs slightly from TCP/IP. OSI structures the overall network as a hierarchy of several interconnected "subnetworks." Within each of these subnets, the common characteristic shared by all member machines is the ability to communicate *directly* with each other. In addition, one or more machines on each subnet are designated as gateway nodes and are provided with a link to a counterpart machine on some other subnet. When in operation, network traffic between hosts on distinct subnets is identified and explicitly routed by the OSI protocols through the gateways connecting them.



Subnet relationship of the OSI standard.

Functional layers of OSI

The OSI protocol suite is a set of seven layered, modularized protocols organized by logically related functionality. Each layer is composed of an ordered set of subsystems and is responsible for providing a selected set of network services to the layers above. The advantages to such a layered design were detailed in the discussion of TCP/IP, which was organized in a similar fashion. Obviously, those same advantages can be applied here to the OSI model equally well.

The lowest two protocol layers,

the physical and data link layers, are practically identical in function to the lowest two layers of TCP/IP. The physical layer, at the bottom of the OSI hierarchy, provides basic mechanisms for the encoding and transmission of data between network nodes. Various methods for the exchange of information are allowed at this level, such as the CSMA/CD, token ring, and token bus methods mentioned earlier. The task of shielding the upper layers from the operational differences of these architectures falls on the layer above; the data link layer.

The data link layer performs two functions within the OSI model. First, as already mentioned, it attempts to compensate for the differences in the various physical network implementations. It does so by providing the upper layers with a common set of interface functions with which to transmit data within a subnet. It accomplishes this in the same way that TCP/IP did, through an implementation of the MAC and LLC layers as described by the IEEE 802.x standard set. Also, when using a class 2 LLC (LLC-2) it is able to perform the error checking needed to ensure error free transmissions of datagrams between nodes within the subnet.

Immediately above the data link layer is the network layer. The responsibility of this level is to support the transmission and routing of datagrams between hosts on disjoint subnets. Information is maintained here concerning the location and status of other subnets and their associated gateway machines. When datagrams are encountered which span subnets, this information is used to establish a route, possibly through several intermediate gateways, by which the message can be delivered to the proper subnet. By way of comparison, this function is roughly equivalent to that performed by the Internet Protocol within TCP/IP.

The fourth layer of the OSI structure is the transport layer. This layer performs yet another level of error correction on the data transmissions, checking the reliability of end-to-end communications spanning distinct subnets. It ensures not only that all data sent by the source host arrives at the destination host, but it also makes sure that it arrives in sequenced order without duplication.

The next layer, known as the session layer, builds on the reliability provided by the layer immediately below it to support virtual circuit types of connections between hosts. All the layers below this still operate on the assumption that the underlying network structure is packet switched based. It is only here, at layer five, that assumption can be altered. With the services provided by this level, full-duplex virtual circuits (or "sessions" as they are known in OSI) can be established between hosts on disjoint subnets. Over them data can be transferred in either a synchronous or asynchronous fashion. In comparison, these functions, combined with those provided by the transport layer, are roughly equivalent to the capabilities of the Transmission Control Protocol within TCP/IP.

Layer six of the OSI reference model is the presentation layer. In the transmission of data onto the network, it is the responsibility of this level to translate the information into a site-specific transportable form before handing to the lower layers. Conversely, upon the receipt of a message, this layer handles the conversion necessary to put the information back into a usable form. Typically, these "translations" vary from installation to installation, and in many cases are absent altogether. When they are used, they often involve either the compression or encryption of the transmitted data.

The seventh layer of the OSI model is the application layer. As was the case with the comparable layer within TCP/IP, this level is composed of a set of utilities, each of which performs some fixed, network-oriented function. Four such utilities have been defined as part of the OSI model. These are Job Transfer and Manipulation (JTM) which provides services for the execution and control of programs on remote systems, File Transfer, Access, and Manipulation (FTAM) which, as the name implies, allow for the remote access and transfer of data files. Virtual Terminal (VT) which provides services for establishment of login session on remote systems, and Message Oriented Text Interchange System (MOTIS) which provides for the exchange of electronic mail. In addition to these, users and installations are free to include at this level their own site specific applications and utilities as well.

And the winner is. . .

Structurally speaking, in the design of the OSI standard the OSI has attempted to provide a flexible interface for all types of foreseeable network communication needs. The resulting seven layers of the model, and the functionality they contain, are the product of numerous hours of deliberation by an organization known for its ability to construct workable, vendor independent standards.

TCP/IP, on the other hand, was not designed to be all things for all times. Its DOD developers designed it to be streamlined yet adaptable, allowing for new functionality to be "bolted on" as technology as the user community demanded it. Such was the case with the Internet Control Message Protocol (ICMP) which was added after the fact to the base IP layer.

Hence, gauging these two standards from a structural perspective hinges on whether or not OSI was complete in its view of the future. If it was, and it has indeed provided a standard that is both complete and flexible enough to keep pace with time, then it is probably safe to say that the OSI model is better structured than TCP/IP. This, however, is a purely theoretical metric.

If the two network standards are compared from a more practical perspective, the OSI model doesn't fare nearly as well. To date, the only organization that has accepted it wholeheartedly is the government. In spite of this powerful ally, commercial software developers have tended to avoid the OSI standard, citing a less than enthusiastic acceptance by the private sector. TCP/IP, on the other hand, is the older of the two protocols and has had time to develop a substantial user base, especially in the UNIX community. Here TCP/IP has already become an unspoken standard and clearly dominates the market.

As is the case with most disputes of this type, theoretical advantages of one side over the other matter very little. The ultimate winner will be decided in and by the marketplace.

About the author

Douglas Meyer is a Ph.D. candidate in Computer Science at the University of Missouri-Rolla and is Network Manager for the Intelligent Systems Center located there. Dr. George Zobrist is a Professor in the Department of Computer Science at UMR and an Associate Editor for *IEEE Potentials*. □