Chapter I

The OSI Model and Switching

Vasilios A. Siris
Institute of Computer Science–FORTH, Greece

In this chapter we give the motivation and basic concepts of the OSI reference model, discuss its seven-layer architecture, the communication between systems using the OSI model, and finally the relationship between the OSI model and multilayer switching.

MOTIVATION AND BASIC CONCEPTS

The Open System Interconnection (OSI) reference model is a framework for defining the conventions and tasks required for network systems to communicate with one another. The work on the OSI model began in the late 1970s, mostly independently, by the International Organization for Standardization (ISO) and the International Telegraph and Telephone Consultative Committee or CCITT (which comes from the translation of the title in French). CCITT has been succeeded by the Telecommunications Standardization Sector of the International Telecommunications Union (ITU-TS). In 1983 the work of the two organizations was combined, and a single document describing the reference model for Open Systems Interconnection was produced. The term "open systems" refers to the fact that the specifications are publicly available to everyone.

The purpose of the OSI model was to assist vendors and communications software developers to produce interoperable network systems. Although the OSI model was designed to replace all previous computer communications standards, it is no longer viewed as such a replacement. Rather, the OSI model has succeeded as a tool for describing and defining how heterogeneous network systems communicate.

The OSI model is based on a widely accepted structuring technique called layering. According to this approach, the communications functions are partitioned into a vertical set of layers. Each layer performs a related set of functions, utilizing and enriching the services provided by the immediately lower layer. The layering approach was developed to address the following goals:

- Provide a logical decomposition of a complex communications network into smaller, more understandable and manageable parts.
- Provide standard interfaces between network functions and modules.
- Provide a standard language for describing network functions, to be used by network designers, managers, vendors, and users.

An important task in the development of the OSI model was to group similar functions into layers, while keeping each layer small enough to be manageable, and at the same time, keeping the number of layers small, since a large number of layers would increase the processing overhead. The principles used in defining the OSI layers are summarized in following list (Stallings, 1987):

1. The number of layers should not be so many as to make the task of describing and integrating the layers more difficult than necessary.
2. Layer boundaries should be created at points where the description of services is small and the number of interactions between boundaries is minimized.
3. Separate layers should be created in cases where manifestly different functions are performed or different technologies are involved.
4. Similar functions should be collected into the same layer.
5. A layer should be created where functions are easily localized. This enables the redesign of the layer to take advantage of new technologies.
6. A layer should be created where there is a need for a different level of abstraction in the handling of data.
7. Changes of functions or protocols of a layer should be made without affecting other layers.
8. For each layer, boundaries with its upper and lower layers only are created.

The application of the above principles resulted in the seven-layer OSI reference model, which we describe next.

THE SEVEN OSI LAYERS

The seven layers of the OSI reference model, are concerned with tasks ranging from how electrical signals are generated and bits are encoded, to the interface with user applications (Stallings, 1987; Tanenbaum, 1988) (Table 1).

The Lower Layers: Physical, Data Link, Network

The three lower layers of the OSI reference model are responsible for transferring the data between the end systems, hence constitute the communications portion of the model. These layers run on both end systems and intermediate nodes.

Table 1: The seven layers of the OSI model

1. Physical	Transmission of an unstructured bit stream over the physical medium
2. Data Link	Reliable transmission of frames over a single network connection
3. Network	End-to-end communication across one or more subnetworks
4. Transport	Reliable and transparent transfer of data between end points
5. Session	Control structure and management of sessions between applications
6. Presentation	Data representation (encoding) during transfer
7. Application	Information processing and provision of services to end users

Physical Layer

The physical layer is concerned with the transmission of bits between adjacent systems (nodes). Its functions include interfacing with the transmission hardware, physical connector characteristics, and voltage levels for encoding of binary values. Repeaters, which are responsible for reading and regenerating pulses, operate at this layer. Some well-known physical layer standards include RS-232 and its successor RS-449.

Data Link Layer

The data link layer provides reliable transmission of data (frames) between adjacent nodes, built on top of a raw and unreliable bit transmission service provided by the physical layer. To achieve this, the data link layer performs error detection and control, usually implemented with a Cyclic Redundancy Check (CRC). Note that the data link layer provides reliable transmission service over a single link connecting two systems. If the two end systems that communicate are not directly connected, then their communication will go through multiple data links, each operating independently. In this case, it is the responsibility of higher layers to provide reliable end-to-end transmission.

Bridges, which connect two similar or dissimilar local area network segments, operate at this layer. Some well-known protocols for the data link layer include High-level Data Link Control (HDLC), LAN drivers and access methods such as Ethernet and Token Ring, and the LAP-D protocol in ISDN networks.

Network Layer

The network layer provides the transparent transfer of data packets from the source to the destination system, thus relieving the higher layers from having to know about the underlying network configuration and topology. The end systems can belong to different subnetworks, with different transmission and switching technologies and procedures. It is the responsibility of the network layer to hide all the heterogeneous transmission and switching used to connect end systems and intermediate nodes from its upper layer (transport layer). Two basic functions performed by the network layer are routing, which involves determining the path a packet must follow to reach its destination, and packet forwarding, which involves

moving the packet from one subnetwork to another. Routing is performed based on the network layer address, which uniquely identifies each connection of an end-system with the network. Note that in the simple case where the two end systems are located on the same subnetwork (e.g., they are directly connected), there may be little or no need for a network layer.

Network protocols can be connection-oriented or connectionless. Connection-oriented protocols require some initial interaction between the communicating entities before data transfer begins. This interaction leads to the creation of a logical connection or virtual circuit between the communicating entities. On the other hand, connectionless protocols do not require any initial interaction between the communicating entities. Furthermore, one message is handled independently of any other messages between the same entities.

The network layer is also responsible for segmenting messages into data units that can be accepted by the data link layer. Such functionality is required due to the different technologies used in local and wide area networks. Furthermore, since it would be insufficient to enforce a single data unit size, segmentation can occur more than once. Reassembly, which refers to creating the original message prior to segmentation, can be performed in the intermediate nodes or the end systems. Finally, it is also possible for the network layer to perform error and flow control.

Routers, which provide the necessary functionality for connecting local area networks and/or wide area networks, operate at the network layer. Some well-known protocols for the network layer include the Internet Protocol (IP), the Internetwork Packet Exchange (IPX) protocol, and the X.25 Layer 3 protocol.

The Higher Layers: Transport, Session, Presentation, Application

The four higher layers of the OSI model provide services to users of end systems, hence constitute the end system or end-to-end portion of the model. These layers typically, but not always (e.g., in the case of gateways or Layer 4 switches which we discuss later), run on end systems.

Transport Layer

The transport layer provides a reliable and transparent transfer of data between end systems, on top of a possibly unreliable network layer. In order to provide a reliable transfer service, the transport layer uses mechanisms such as error detection and recovery, and flow control. Note that such mechanisms can also exist in lower layers, such as the data link layer. The difference is that the data link layer is responsible for the reliable transmission of data over a single link, whereas the transport layer is responsible for the reliable transmission of data from the source to the destination, which can involve a number of independent links.

The transport layer is also responsible for segmenting long messages into smaller units, or packets, that can be accepted by the network layer, and then

reassembling the packets into the original message. Furthermore, similar to network layer protocols, transport layer protocols can be connection-oriented or connectionless. Finally, transport layer protocols are capable of multiplexing data from different higher layer protocols.

The complexity of the transport layer depends both on the service it is expected to provide to the session layer and on the service it receives from the network layer. Hence, if the network layer provides an unreliable connectionless (datagram) service and the transport layer is to provide an error-free, in sequence and zero loss or duplications transmission of data, then the transport layer will need to implement extensive error and duplicate detection, retransmission and recovery, and congestion control mechanisms.

Examples of transport layer protocols include TCP (Transmission Control Protocol), which is a connection-oriented protocol, and UDP (User Datagram Protocol), which is a connectionless protocol (Feit, 1998).

Session Layer

The session layer is responsible for controlling the dialogue between the end systems. This involves establishing and terminating the dialogue, called session, between applications. The session layer can also include determination of the dialogue type used and synchronization between the end systems through a checkpointing mechanism.

Presentation Layer

The presentation layer is responsible for the encoding or bit pattern representation of the transferred data. Its objective is to resolve any differences in the format or encoding of application data. Two examples of the presentation layer functions are data compression and data encryption.

Application Layer

Finally, the application layer provides end user services, such as file transfer, electronic message transfer, virtual terminal emulation, etc. Some well-known examples of application layer protocols include TELNET (Remote Login), FTP (File Transfer Protocol), SMTP (Simple Mail Transfer Protocol), SNMP (Simple Network Management Protocol), X.400 (Message Handling System), and X.500 (Directory Services).

COMMUNICATION BETWEEN SYSTEMS USING THE OSI MODEL

Next we describe how layers interact with each other, and how end systems can communicate using the OSI reference mode.

Interaction Between the OSI Layers

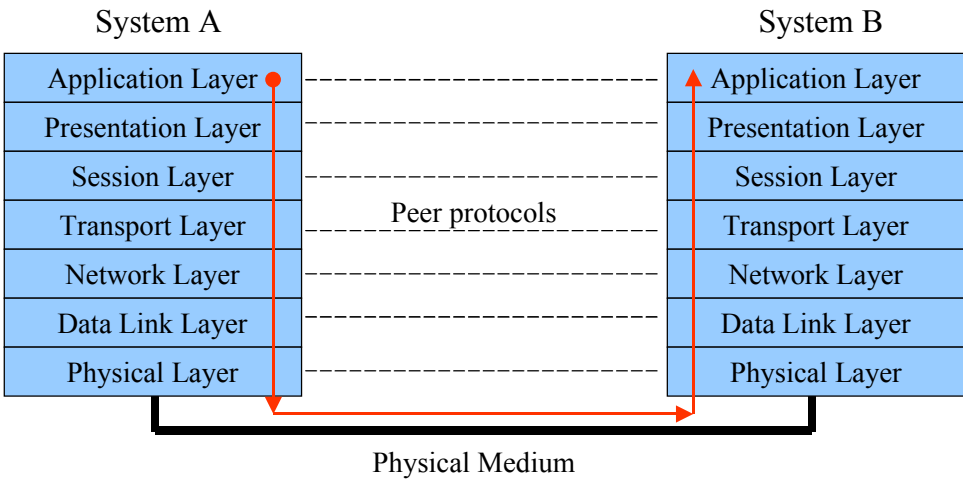
Communication of two end systems using the OSI model is depicted in Figure 1. The figure shows the case where two systems are directly connected through some physical media. Both systems implement all seven layers of the OSI model.

Let us first focus on a single system. In the same system, each layer (n) communicates with the layer (n-1) directly below it, through a well-defined interface. Each layer (n-1) is said to provide a service to layer (n). The service definition specifies the notation to be used by layer (n) to reference procedures and messages belonging to layer (n-1). Layer (n-1) offers services to layer (n) through service primitives.

Now let us consider both end-systems. Observe that the two systems are physically connected only at the physical layer, through the physical medium. However, for the two end systems to communicate, the corresponding or "peer" layers in the two systems need to interact using a well defined set of conventions and rules that form a protocol.

It is important to note the difference between a service definition (or interface) and protocol. A service definition or interface refers to the vertical relationship and interaction between neighboring layers in the same system. On the other hand, a protocol refers to the horizontal relationship between peer layers of adjacent systems. The actual communication of the two systems originates at the application layer of the sender (System A). The message to be sent proceeds down the seven-layer protocol stack of System A (sender) until it reaches the physical layer, where it is encoded for transmission over the physical medium. From the physical layer of System A (sender), the message is sent over the physical medium to the physical layer of System B (receiver). At System B, the message proceeds up the seven-layer stack to the application layer.

Figure 1: The seven-layer OSI model



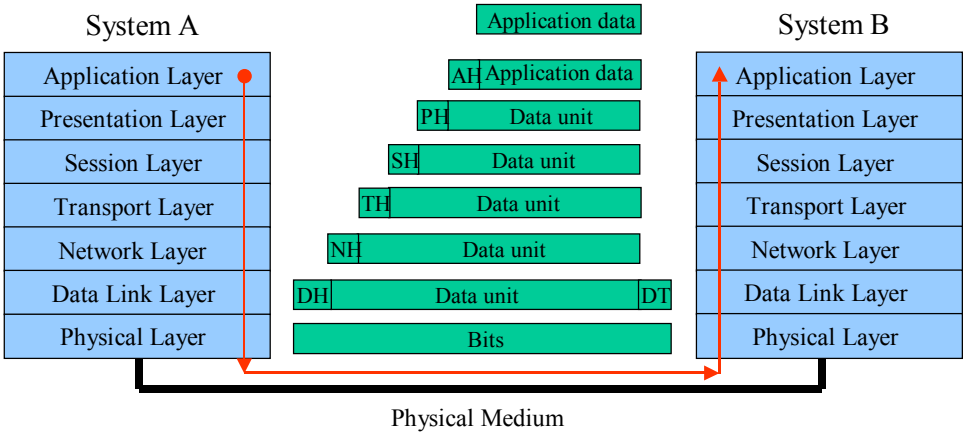
Information Flow Between the OSI Layers

Information pertaining to peer protocols is communicated through headers. This is illustrated in Figure 2. As data travels down the layer stack at the sending system, each layer adds a header with information specific to the protocol at that layer. This addition of a header to the higher layer data unit is also called encapsulation. Hence, the application layer adds an application header (AH), the session layer a session header (SH), and so on. Note that the data link layer adds both a header (DH) and a trailer (DT); the latter contains a Cyclic Redundancy Check (CRC) and a flag that is used for identifying frame boundaries. At the bottom of the stack, the physical layer transmits the Layer 2 data unit, called frame, over the physical medium. At the receiving system, the above steps are followed in reverse order.

There are two important mechanisms that are related to the information flow we described above: connection-oriented (or virtual circuit) and connectionless (or datagram) transmission, and segmentation and reassembly. Data transfer between peer layers can proceed in a connection-oriented fashion, in which case there is an initial interaction between the peer layers that leads to the establishment of a logical connection or virtual circuit. After this initial interaction, which is referred to as the call setup (or establishment) phase, data transfer can take place. The data transfer phase is followed by the call tear-down (or termination) phase, which is responsible for removing the logical connection. Unlike connection-oriented transmission, connectionless transmission does not require any initial interaction for establishing a logical connection, or a final interaction for removing the logical connection. During connectionless transmission, a message between peer layers is independent of previous and later messages.

Segmentation refers to the breaking of data units into smaller data units. Segmentation can occur at the network and data link layers when the message to be sent is larger than the maximum size of the packet allowed at the data link layer (e.g., for Ethernet local area networks the maximum packet size is 1500 bytes).

Figure 2: Information flow and encapsulation



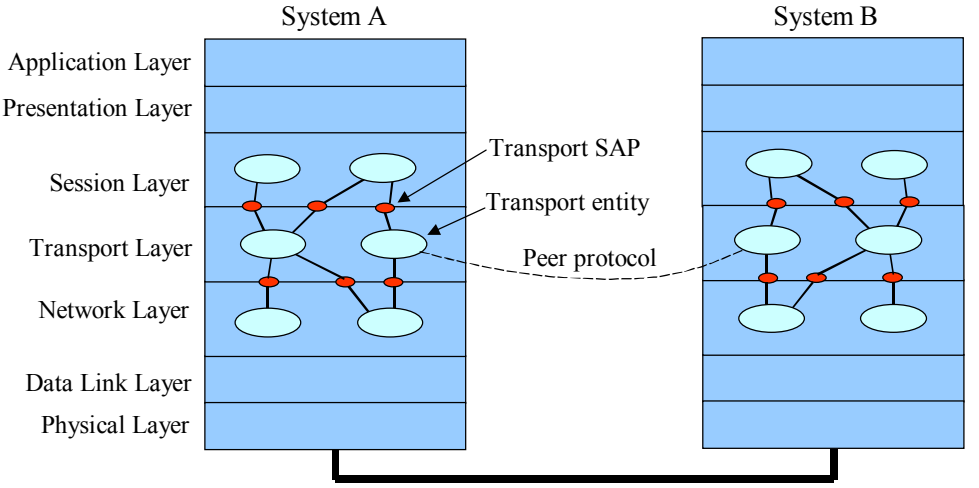
At each layer of the OSI model there might be more than one *entity* that may implement different protocols. One entity can communicate with one or more entities in the layer below through *service access points* (SAP). Furthermore, it communicates with its peer entity through some protocol. This is shown in Figure 3.

A service access point (SAP) at the interface of two layers operates like an address, allowing an entity residing at the lower of the two layers to identify the entity residing at the higher layer to which it must forward a message it receives. As shown in Figure 3, a single entity at one layer can communicate with two or more different entities residing at the higher or lower layer through a corresponding number of independent SAPs.

As an example, consider the IP, TCP, and UDP protocols. A system can have two entities implementing the TCP and UDP protocols, which through independent SAPs use the services of a lower entity that implements the IP protocol. In turn, the TCP and UDP entities offer services to higher layers through SAPs. The SAPs of the TCP and UDP protocols are referred to as ports. There exists a set of specific ports called well-known ports. These enable one system to access services, such as FTP and TELNET, offered by another system.

To understand how and why ports are needed, consider a System A that wishes to transfer a file from a System B. File transfer along with many other applications use the TCP protocol. Due to this, when the TCP entity at System B receives data from System A it will need to know to which higher layer entity to pass it to. This is achieved through the notion of well-known ports, as follows. Common applications, such as file transfer, terminal access, and www access, have an associated well-known port number. These numbers are known to all systems that implement the TCP/IP protocol suite. Hence, when System A wishes to transfer a file from System B, it includes the port number that is associated with file transfer in the TCP

Figure 3: Interaction of layers and service access points



header of the message it sends to System B. The TCP entity at System B, through the port number, knows to which higher layer protocol (file transfer in our example) to pass the message.

Communication Across a Network

In the previous subsection we described the communication between adjacent systems, i.e., systems that had a direct physical connection. The OSI reference model also pertains to the case where the two end systems communicate through intermediate nodes. These intermediate nodes run a subset or all seven layers of the OSI model. Depending on which layers are implemented in the intermediate nodes, we have the following types of intermediate devices:

- **repeaters, hubs**: These devices implement only the functionality of the physical layer. Repeaters amplify or regenerate the physical signal and are used to extend the physical range of networks. Segments connected using repeaters logically behave as a single network segment. Hubs are essentially multiport repeaters, with port management capabilities for assigning ports to different network segments.
- **bridges, LAN/Layer 2 switches**: These devices implement the functionality of the physical and data link layers. Layer 2 devices are used to interconnect two or more network segments. They implement two algorithms: a self-learning algorithm that enables them to associate ports to data link addresses, and a spanning tree algorithm that is responsible for detecting and breaking circular paths, thus preventing frames from travelling in circles.

When a Layer 2 device receives a frame, it looks at its data link address and, based on the table it created using the self-learning algorithm, decides whether to forward the frame to a particular port or to filter it. For shared access physical media (such as Ethernet or Token Ring), the data link addresses reside on the Media Access Control (MAC) layer. This layer is an IEEE standard and lies at the upper part of the physical and the lower part of the data link layers of the OSI model. The purpose of the MAC layer is to define different methods to control the access to shared physical media.

Finally, Layer 2 devices flood (i.e., forward an incoming frame to all output ports) all multicast and broadcast traffic. The latter is used for status, availability, and address resolution related information.

- **routers, Layer 3 switches**: These devices implement the functionality of the physical, data link, and network layers. Layer 3 devices are used to connect different subnetworks and create separate administration domains. They implement three basic functions: route table updating, route table lookup, and packet forwarding.

The first function of Layer 3 devices is to run routing protocols, which exchange information with other routers in order to maintain routing tables. The second function of Layer 3 devices, which is called route lookup, is to select the path (or next hop) a packet must follow in order to reach its final destination. This routing decision is based on the addresses of the source and destination systems, which are

part of the network layer, and uses the information in the routing table. Finally, the third function of Layer 3 devices is to actually forward packets from input ports to output ports. Unlike Layer 2 devices, Layer 3 devices do not flood multicast and broadcast traffic.

- **gateways**: These devices implement all seven layers of the OSI model. Gateways are responsible for connecting incompatible application systems, such as electronic mail systems, and converting and transferring data from one system to another. Hence, gateways are application-specific devices.

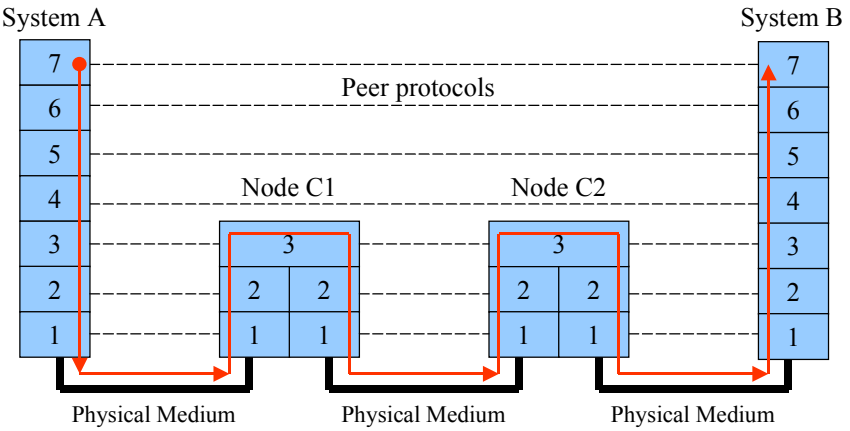
Figure 4 illustrates the communication of two systems when the intermediate nodes implement Layers 1-3 of the OSI model. Note that more than two intermediate nodes can exist. These intermediate nodes include two stacks of the bottom three layers of the OSI model, with the linking of the two stacks occurring at the network layer (Layer 3). The three layers of one stack of Node C1 are peers of Layers 1-3 at System A, while the three layers of the other stack are peers of Layers 1-3 at Node C2. On the other hand, the peers of the higher Layers 4-7 at System A reside on System B.

The flow of information at the two end systems is similar to the case of Figure 2 where the two end systems had a direct physical connection. The difference in this case is that the flow travels through the intermediate nodes, up the left stack and down the right stack.

Note that Layers 1 and 2 at the intermediate nodes can have different protocols. This is the case when the intermediate node connects two systems that reside on subnetworks of different technology. For example, the left and right physical medium in Figure 4 can be based on Local Area Network (LAN) technologies, whereas the middle can be based on Wide Area Network (WAN) technologies.

Figure 5 shows the communication of end systems through a network with intermediate nodes implementing Layers 1 and 2 (data link layer) of the OSI model. Observe that with Layer 3 intermediate nodes (Figure 4), the peer of the end systems resides on the first node they are connected to. On the other hand, with Layer 2 intermediate nodes, the Layer 3 (network layer) of the two end systems are peers.

Figure 4: Communication across a network through layer 3 intermediate nodes



THE OSI MODEL AND SWITCHING

In this section we discuss the relation of switching with the OSI reference model. Recall that the OSI model has succeeded as a descriptive and explanatory tool, rather than as an implementation guideline. As such, it can also be used to explain the idea of switching at the various layers.

We start with a general definition of "layer X switching." The expression includes two terms: "layer X" and "switching." The second term, "switching," for the discussion of this section, can be defined as fast or wire-speed forwarding (usually hardware-based, using ASICs or Application Specific Integrated Circuits) of packets or frames. The first term, "layer X," identifies the information used to process (or switch) the packets or frames. This processing typically involves classifying data units into different categories (or classes). Data units belonging to the same category (or class) are treated similarly, e.g., they can be assigned to the same queue or given the same priority. It is interesting to note that the term "switching" can be used independent of the type of data being "switched"; for example, the data can consist of small, fixed-size cells or variable-size structures such as data link layer frames or network layer packets.

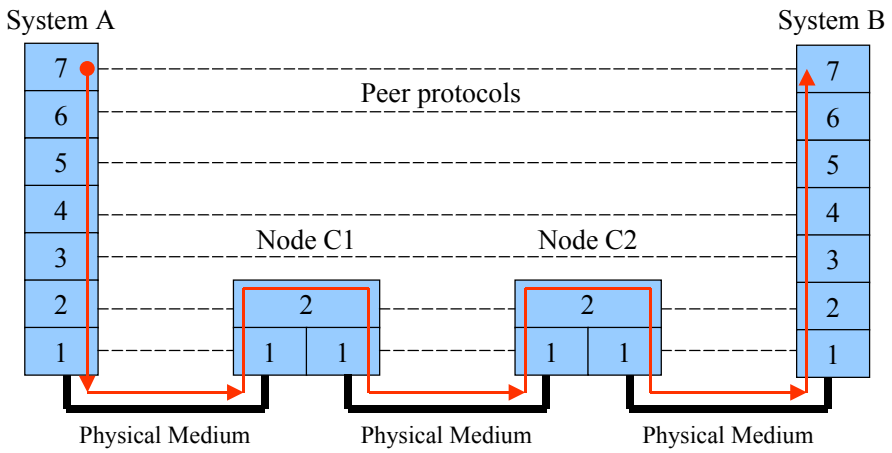
Having the above general definition of "layer X switching," we continue with a more detailed discussion of switching at various levels of the OSI model.

Layer 2 Switching

Layer 2 or LAN switching refers to the fast forwarding, from the input port to the output port, of frames based solely on Layer 2 information. Such information can include the Medium Access Control (MAC) address (e.g., Ethernet addresses) in Local Area Networks (LANs).

Layer 2 switches also implement all other functionality implemented by a bridge, which also operates at Layer 2. Learning is one such functionality through which the Layer 2 switch or bridge learns the port a particular MAC

Figure 5: Communication across a network through layer 2 intermediate nodes



address is reachable from. Indeed, if only one system (i.e., one MAC address) is connected to each port of a Layer 2 switch, collisions of packets from different systems can be avoided. This technique is known as segmentation of collision domains. Hence, the switch can operate by having the capacity of each port dedicated to the system connected to that port, thus increasing the aggregate throughput that can be achieved. Indeed, as a hub is essentially a multiport repeater, a Layer 2 switch is essentially a multiport bridge. Note that, similar to bridges, Layer 2 switches flood all multicast and broadcast traffic and create "flat," i.e., non-hierarchical, networks.

Layer 3 Switching

Based on our definition of layer X switching, Layer 3 switching refers to the fast forwarding of packets based on Layer 3 information, such as IP addresses. The high forwarding speeds of Layer 3 switches are achieved using hardware-based (ASIC) packet processing. On the other hand, traditional routers use software-based packet processing.

Layer 3 switching is not only about speed, but also about adding more functionality and capabilities to simple datagram networks that are limited to offering the same service (best-effort) to all users. Layer 3 switches can be used to build networks offering differentiated quality of service to different groups of end systems, identified by some prefix of the network layer address. The offered quality of service can include a minimum bandwidth guarantee, a maximum end-to-end packet delay or delay variation (jitter). Such capabilities are also referred to as policy-based routing. Other functions that can be supported with Layer 3 switching include security mechanisms based on access control lists.

In the above cases, both Layer 3 switches and routers perform packet-by-packet processing, i.e., each packet is handled independently of any previous packets. A technology that departs from such packet-by-packet processing, but is considered a type of Layer 3 switching, is cut-through switching, a representative of which is MultiProtocol Label Switching (MPLS) (Rosen et al., 2001). MPLS networks include a shim layer between the data link and network layers that adds the ability to define logical connections, called virtual paths, in connectionless network protocols. With MPLS, packets are not processed independently, but are processed based on a label included in the intermediate shim layer. Hence, we can have all packets originating from a particular network and/or destined to a particular network assigned the same label, hence processed similarly by all MPLS-capable nodes. Labels can be assigned, for example, based on routing and topology information. The support of such logical connections adds the ability to implement versatile traffic engineering and security mechanisms.

Layer 4 Switching

As discussed previously, service access points (SAPs) and ports are features located at Layer 4 of the OSI model. SAPs are used to identify different

higher layers (applications). In the case of TCP/IP networks, recall that applications are assigned well-known port numbers. Examples include e-mail, www, and file transfer, which use the SMTP, HTTP, and FTP protocols, respectively. Hence, switching that takes into account Layer 4 information allows the implementation of advanced traffic management capabilities such as offering differentiated quality of service to different applications.

Other applications of Layer 4 switching include filtering, security, load balancing, and bandwidth allocation based on application type. For example, Layer 4 switching can be used to forward packets belonging to different types of applications (e.g., e-mail, file transfer, www) to different servers, each tuned to better handle the corresponding application type.

Another capability provided by Layer 4 switching is the collection of detailed accounting information that includes not only the aggregate load per source/destination network, but also information based on application type. Such a capability can be used for billing, when charges are based on volume. Interactive traffic can receive higher priority than bulk transfer traffic, hence it would be fair to charge more for the former. Accounting using Layer 4 information can provide the necessary information for such a charging scheme.

It is interesting to note that traditional routers with software-based packet processing were also capable of viewing Layer 4 related information. However, due to their low processing capability, such information could only be used to implement crude filters for providing some basic form of security.

Layer 5-7 Switching

Although service access points or port numbers allow some differentiation of applications, such identification can be rather crude. For example, both bulk and interactive transfers can use the http protocol. Hence, port numbers are not sufficient for differentiating such transfer types. To allow finer differentiation of user applications, hence providing finer differentiation of service quality, one would have to take into account information at layers above the transport layer, namely Layers 5-7.

Using the information in Layers 5-7 to classify data, one could build networks able to differentiate various sessions, such as web sessions, thereby providing different performance or Quality-of-Service (QoS) to different sessions.

Multilayer Switching

Multilayer switching typically refers to the ability of a network to switch data at more than one layers of the OSI model. In particular, multilayer switches operating at Layers 2 and 3 have the intelligence of knowing at which of the two layers to switch data. Hence, if a multilayer switch receives data destined for a system residing on the same subnet as the source, the data is switched at Layer 2. On the other hand, if it receives data destined for a system that resides on a different subnet, then the data is switched at Layer 3.

Table 2: Switching at the various layers of the OSI model

Switching at	Classification based on	Enables
Layer 2 (Data Link layer)	Data Link addresses (e.g., MAC addresses)	Dedicated (switched), not shared bandwidth per host
Layer 3 (Network layer)	Network addresses (e.g., IP addresses)	Differentiation of services based on source/destination address
Layer 4 (Transport layer)	Transport layer service access points (e.g., ports)	Differentiation of services based on application type
Layers 5-7 (Session, Presentation, Application)	Session and application layer information	Differentiation of services based on session and application related information

A summary of the information used for traffic classification and the capabilities that are enabled by switching at the various layers of the OSI model are shown in Table 2.

REFERENCES

Feit, S. (1998). *TCP/IP*. New York: McGraw Hill.

Rosen, E., Viswanathan, A., and Callon, R. (2001). MultiProtocol Label Switching Architecture. RFC 3031 *Internet Engineering Task Force*.

Stallings, R. (2000). *Data and Computer Communications*. Englewood Cliffs, NJ: Prentice Hall.

Stallings, R. (1987). *Handbook of Computer-Communications Standards, Volume I: The Open Systems Interconnection (OSI) Model and OSI-related Standards*. New York: Macmillan Publishing Co.

Tanenbaum, A. S. (1989). *Computer Networks*. Englewood Cliffs, NJ: Prentice-Hall.