Network Models- OSI Model and TCP/IP Model

Reference:
Chapter 2 -Stallings
Chapter 2 - Forouzan

Review of Lecture 1

- Communications model
- Key tasks of communication
- Data communications model
- Data communications network i.e. WAN and LAN
- Standards

Lecture 2 Objectives

- To understand the need for a protocol architecture
- Describe the OSI reference model
- Describe the TCP/IP reference model
- Describe the advantages and disadvantages of a layered network architecture
- Describe the basic functions of each layer in the OSI Reference Model

Need for Protocol Architecture

- When computers or other data processing devices exchange data, the procedures involved can be quite complex
- Consider, for example, transfer of a file between two computers:
  - In addition to having a data path between the computers, directly or via a communication network, the following are also needed
    - The source system must either activate a direct data communication path or inform the communication network about the destination

Need for Protocol Architecture Contd.

- The source system must ascertain that the destination system is prepared to receive and store data for a particular user
- If the file formats used on the two systems are incompatible, one of the systems must perform a translation function

- In a protocol architecture, instead of implementing above logic in a single module, it is broken up into a vertical stack of sub tasks
- Each layer in the stack performs a related subset of the function required to communicate with the other system

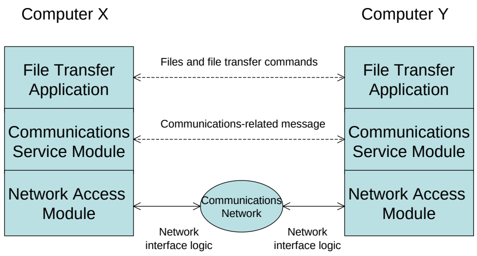
Need for Protocol Architecture Contd.

- Ideally the layers should be defined so that changes in one layer do not require changes in another layer
  - A lower layer performs more primitive functions and provides a service to the higher layer
- It is obvious that for two systems to communicate, the layered functions must exist in both systems
  - That is, communication is achieved by having the peer, or corresponding layers in the two systems to communicate

Key Elements of a Protocol

- The peer layer communication takes place by means of formatted blocks of data that obey a set of rules or conventions known as a protocol
- The key features of a protocol are:
  - Syntax Concerns of the format of the data blocks, signal levels
  - Semantics includes control information, error handling
  - Timing Includes speed matching and sequencing

Protocol Architecture Broken into Modules



Simplified Protocol Architecture For File Transfer

A Three Layer Model

- Generally, communications can be said to involve three agents:
  - Applications, Computers and Networks
    - Applications execute on computers that typically support multiple simultaneous applications
    - Computers are connected to networks and
    - Data to be exchanged are transferred by the network
- With these concepts in mind, it appears natural to organise communication task into three relatively independent layers
  - Network access layer, transport layer and application layer

Network Access Layer

- Network access layer is concerned with the exchange of data between a computer and the network to which it is attached
- The software in this layer depends on the type of network used such as circuit switched, packet switched, LANs etc
- By separating this function into a separate layer, the remainder of communication software, above this layer, need not be concerned about specifics of the network used

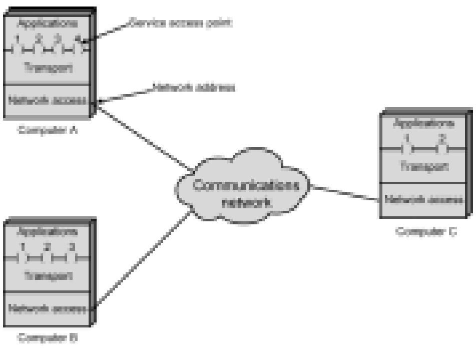
Transport Layer

- Regardless of the application, there is a need that data is exchanged reliably
- That is, all of the data should arrive at the destination and in the same order in which they were sent
- As the need for reliability is **independent** of the application, it makes sense to collect those mechanisms in a common layer shared by all applications
- This common layer is referred to as the transport layer

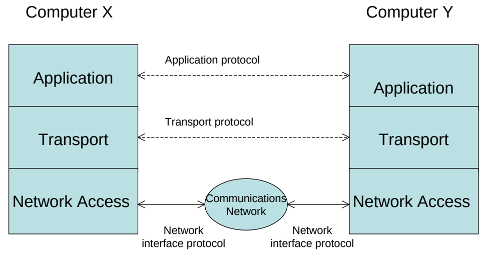
Application Layer

- Application layer contains the logic needed to support various applications
  - For different applications, a separate module is needed
- Each application on a computer must have an address that is unique within that computer
  - This allows the transport layer to support multiple application on a single computer
  - Each application that accesses the services of the transport layer is identified as a service access point (SAP) or port

Protocol Architectures and Networks



Protocol Communication



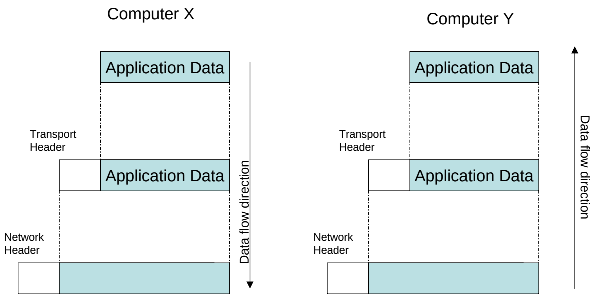
Protocol Communication Contd.

- Suppose that an application on computer X (associated with SAP1) need to send a message to an application (associated with SAP2) on Computer Y
- The application on X hands the message over to the transport layer with instructions to send it to SAP2 on computer Y
- The transport layer hands the message over to the network access layer, which instructs the network to send the message to computer Y

Protocol Communication Contd.

  - The network need not be told the identity of the destination SAP
  - All that it needs to know is that data is intended for computer Y
- To control the above operation, control information, as well as user data must be transmitted as shown in the next slide

Protocol Communication Contd.



Protocol Communication Contd.

- The sending application generates a block of data and passes it to the transport layer
- The transport layer appends a transport header containing protocol control information
- The combination of data from the next layer and control information is known as protocol data unit (PDU)
  - In this example it is known as transport PDU

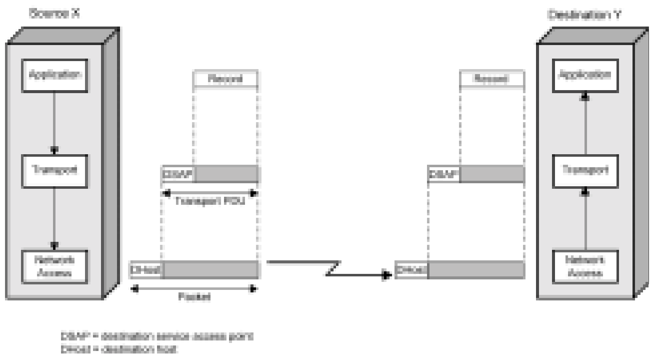
Protocol Communication Contd.

- The header in a transport PDU contains control information to be used by the peer transport protocol at computer Y
- Examples of items that may be stored in transport header include:
  - Destination SAP
  - Sequence number
  - Error-detection code

Protocol Communication Contd.

- The next step is for the transport layer to hand each PDU over to the network layer with instruction to transmit it to the destination computer
- The network access protocol appends a network access header to the data it receives from transport layer, creating network access PDU
- Examples of the items that may be stored in the header include:
  - Destination computer address
  - Facilities request

Operation of a Protocol Architecture



Standardised Protocol Architectures

- When communication is desired among computers from different vendors, the software development effort can be a nightmare
  - Different vendors use different data formats and data exchange protocols
  - Even within one vendor, different model computers may communicate in unique ways
- As the use of computer communications and computer networking proliferates, vendors are forced to adopt a common set of conventions

Standardised Protocol Architectures Contd.

- For the above to happen, standards are required
- Two protocol architectures have served as the basis for the development of interoperable protocol standards:
  - Transmission Control Protocol/ Internetworking Protocol (TCP/IP) protocol suite
  - Open System Interconnection (OSI) reference model
- TCP/IP is by far the most widely used interoperable architecture, while OSI was well known but never lived up to its early promise

OSI Reference Model

- A widely accepted structuring technique, layering, is used by ISO for OSI model
  - The communications functions are partitioned into a hierarchical set of layers
  - A lower layer performs more primitive functions and conceals the details from the above layer to which service are provided
  - Ideally, the layers should be defined so that changes in one layer do not require changes in other layers
    - The problem is divided in to manageable sub problems

OSI Reference Model Contd.

- The task of ISO in defining OSI was to define a set of layers and the services performed by each of them
- The partitioning should group functions logically and should have enough layers to make each layer manageably small
- Should not have too many layers so that processing overhead is burdensome
- The resulting reference model has 7 layers as shown in the next slide

OSI Layers

Application Provides access to the OSI environment for users and also provides distributed information services
Presentation Provides independence to the application processes from differences in data representation
Session Provides control structure for communication between applications; establishes, manages and terminates Connections (sessions)
Transport Provides reliable transfer of data between end points; provides error recovery and flow control as well
Network Provides upper layers with independence from data transmission and switching technologies used to connect systems
Data Link Provides reliable transfer of information across the physical link
Physical Concerned with transmission of unstructured bit stream over physical medium

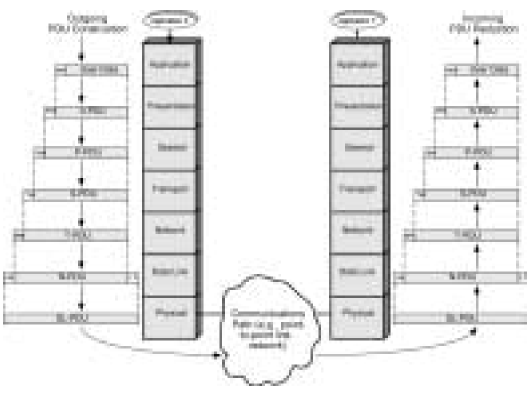
OSI Architecture

- Each of the two systems that communicate contains the seven layers
- An application on one machine invokes the application layer in the other
  - The two layers on the two machines establish a peer relationship
- These peer relationships are created in all layers down to the physical layer
- However, there is no direct communication between peer layers except at the physical layer

OSI Architecture Contd.

- Even at the physical layer, the OSI model does not stipulate that the two systems be directly connected
  - A packet switched or circuit switched network may be used to provide the communication link
- Similar to the 3 layer model discussed earlier, Protocol Data Units (PDUs) are created in all bottom 6 layers encapsulating user data
- At each layer, data units may be segmented into several parts to accommodate its requirements

The OSI Environment



Standardisation within OSI Framework

- The principal motivation for development of the OSI model was to provide a framework for standardisation
- The model facilitates the standards-making process in two ways:
  - As the functions of each layer are well defined, standards can be defined independently and simultaneously
  - As the boundaries between layers are well defined, changes in standards in one layer need not affect already existing software in another layer

Standardisation within OSI Framework Contd.

- The principles used essentially amount to using modular design
- The lower layers are concerned with greater detail while the upper layers are independent of these details
- Each layer provides services to the next higher layer and implements a protocol to the peer layer in other systems
- The nature of the standardisation required at each layer can be described using 3 key elements:

Standardisation within OSI Framework Contd.

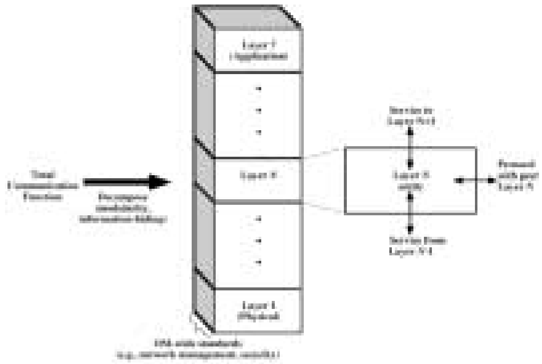
- Protocol specification
  - Two entities at the same layer in different systems cooperate and interact by means of a protocol
  - As two different open systems are involved, the protocol must be specified precisely
- Service definition
  - In addition to the protocols that operate at a given layer, standards are needed for services that each layer provides to the next higher layer
  - Typically, the definition of services defines what services are provided, but not how they are provided

Standardisation within OSI Framework Contd.

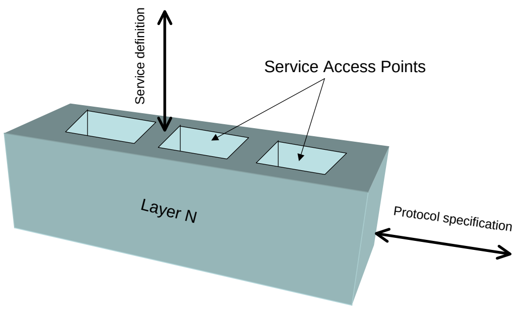
- Addressing
  - Each layer provides services to entities at the next higher layer by means of service access points (SAPs)

- The interaction between two adjacent layers takes place within the confines of a single system and is not the concern of any other open system
- Thus, as long as peer layers in different systems provide the same service to their next higher layers, the details of how the services are provided may differ without loss of interoperability

OSI as Framework for Standardization



Standardisation within OSI Framework Contd.



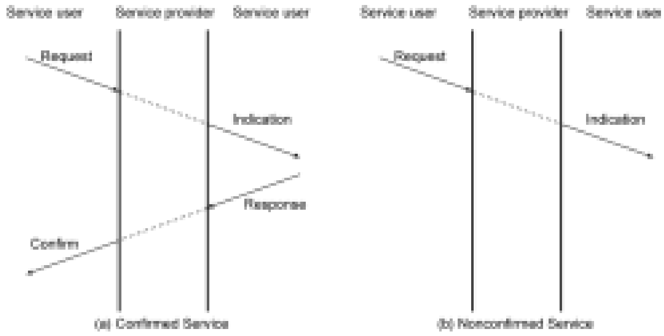
Service Primitives and Parameters

- Services between adjacent layers expressed in terms of primitives and parameters
- Primitives specify function to be performed
- Parameters pass data and control info

Primitive Types

REQUEST	A primitive issued by a service user to invoke some service and to pass the parameters needed to specify fully the requested service
INDICATION	A primitive issued by a service provider either to: indicate that a procedure has been invoked by the peer service user on the connection and to provide the associated parameters, or notify the service user of a provider-initiated action
RESPONSE	A primitive issued by a service user to acknowledge or complete some procedure previously invoked by an indication to that user
CONFIRM	A primitive issued by a service provider to acknowledge or complete some procedure previously invoked by a request by the service user

Timing Sequence for Service Primitives



OSI Layers - Physical Layer

- Moves bits between physically connected end-systems
- Physical interface between devices
  - Mechanical
  - Electrical
  - Functional
  - Procedural
- Standard prescribes
  - coding scheme to represent a bit
  - shapes and sizes of connectors
  - bit-level synchronization

OSI Layers - Datalink layer

- Means of activating, maintaining and deactivating a reliable link
- Error detection and control
- Flow control
- Higher layers may assume error free transmission
- Introduces the notion of a *frame*
  - set of bits that belong together
  - *Begin* and *end* markers delimit a frame
- Datalink layer protocols are the first layer of software

OSI Layers - Network Layer

- Allows an end-system to communicate with any other end-system by computing a route between them
- Provides unique network-wide addresses
- At end-systems primarily hides details of datalink layer
  - segmentation and reassembly
  - error detection
- At intermediate systems
  - participates in routing protocol to create routing tables
  - responsible for forwarding packets
  - scheduling the transmission order of packets
  - choosing which packets to drop

OSI Layers - Transport Layer

- Provides reliable exchange of data between end systems
- Creates the abstraction of an *error-controlled*, *flow-controlled* and *multiplexed* end-to-end link
- Error control
  - message will reach destination despite packet loss, corruption and duplication
  - retransmit lost packets; detect, discard, and retransmit corrupted packets; detect and discard duplicated packets
- Flow control
  - match transmission rate to rate currently sustainable on the path to destination, and at the destination itself
- Multiplexes multiple applications to the same end-to-end connection
  - adds an application-specific identifier (*port number) to* packets for delivery to the correct application

OSI Layers - Session Layer

- ❖ Control the dialogue between applications
  - order of communication
  - packing of information to avoid overflow
- ❖ Reestablish connection if a failure occurs
- ❖ Provides *session synchronization*
  - allows users to place marks in data stream and to roll back to a prespecified mark

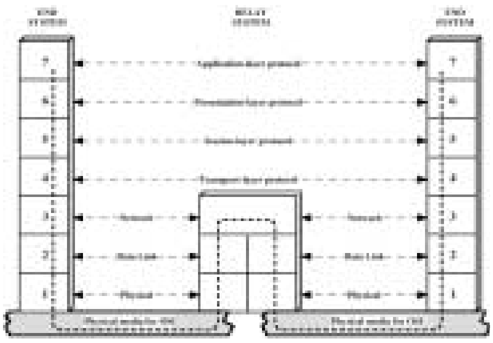
OSI Layers - Presentation Layer

- ❖ Unlike other layers which deal with *headers* presentation layer touches the application data
- ❖ Hides data representation differences between end-to-end applications
- ❖ Deals data compression and encryption

OSI Layers - Application Layer

- ❖ Means for applications to access OSI environment
- ❖ Defined by the user

Use of a Relay



TCP/IP Protocol Architecture

- The TCP/IP protocol architecture is a result of protocol research and development on the experimental packet-switched network, ARPANET
  - It was funded by Defence Advanced Projects Agency (DARPA) and generally referred to as the TCP/IP protocol suite
- This protocol suite consists of a large collection of protocols that have been issued as Internet standards by the Internet Architecture Board (IAB)

TCP/IP Layers

- The TCP/IP model organises the communication task into five relatively independent layers
  - Physical layer
  - Network access layer
  - Internet layer
  - Host-to-host, or transport layer
  - Application layer

TCP/IP Layers Contd.

- The physical layer covers the physical interface between a data transmission device and a transmission medium or network
  - Specifies the characteristics of the transmission medium, the nature of the signals, the data rate etc
- The network access layer is concerned with the exchange of data between an end system and the network to which it is attached
  - The software used at this layer depends on the type of network to be used (circuit switching, packet switching, LANs etc)

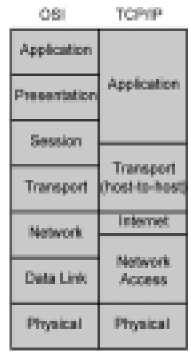
TCP/IP Layers Contd.

- The network access layer is concerned with routing data across a network for two end systems attached to the same network
- In the cases where two devices are attached to different networks, procedures are required to allow data to traverse multiple interconnected networks
  - This is the function of the Internet layer, which uses the Internet Protocol (IP) for it
  - This protocol is used not only in the end systems, but also in routers (a processor that connects 2 networks)

TCP/IP Layers Contd.

- The transport, or host-to-host layer is responsible for reliable exchange of data between end systems, regardless of the nature of the application
  - The Transmission Control Protocol (TCP) is the most commonly used protocol for this functionality
- The application layer contains the logic needed to support various user applications

OSI v TCP/IP



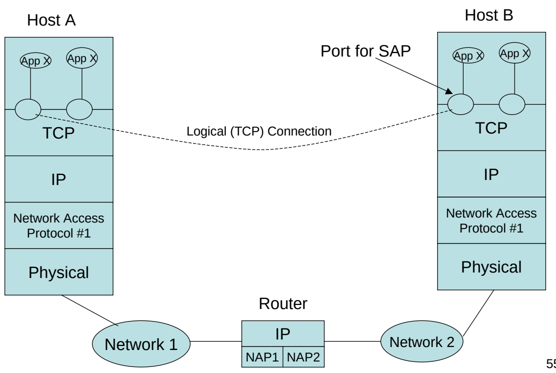
TCP and UDP

- For most applications running as part of the TCP/IP architecture, the transport layer protocol is TCP
- TCP provides a reliable connection in the form of a temporary logical association to transfer data between applications
- A TCP PDU is called a TCP segment, which includes source port and destination port in its header
  - Ports serve the same service as SAPs in the OSI model, and identifies the respective user applications

TCP and UDP Contd.

- In addition to TCP, the other commonly used transport layer protocol that comes as part of TCP/IP suite is the User Datagram Protocol (UDP)
- UDP does not guarantee delivery, preservation of sequence or protection against duplication
- UDP enables a procedure to send messages to other procedures with minimum of protocol mechanism / overhead
- Essentially UDP adds port addressing capability to IP

Operation of TCP and IP



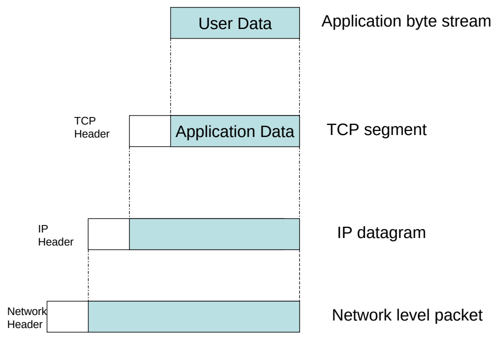
Operation of TCP and IP Contd.

- Communication facility that uses TCP/IP may consist of multiple networks, which are referred to as subnetworks
- Some sort of a network access protocol, such as Ethernet, is used to connect a computer to a subnetwork
  - This protocol enables a host to send data across the subnetwork to another host or, if the target host is on another subnetwork, to a router that will forward the data
- IP is implemented in all of the end systems and routers while TCP is implemented only in the end systems

Operation of TCP and IP Contd.

- For successful communication, every entity in the overall system must have a unique address
- Two levels of addressing are needed for this:
  - Each host on a subnetwork must have a unique global internet address
    - This allows the data to be delivered to the proper host
  - Each process with a host must have an address that is unique within the host
    - This allows host-to-host protocol (TCP) to deliver data to the proper process

Operation of TCP and IP Contd.



Operation of TCP and IP Contd.

- As shown in the previous slide, to accomplish data transmission, control information as well as user data must be transmitted
- The control information is added at each layer by means of a header appended
- Examples of items in a TCP header includes:
  - Destination port
    - When the TCP entity at the receiver receives data, it must know where the data to be delivered

Operation of TCP and IP Contd.

- Sequence number
  - TCP numbers the segments it sends it sends to a particular destination port sequentially
  - As a result if the segments arrive at the receiver out of order, they can be reordered
- Checksum
  - The sending TCP includes a code that is a function the contents of the remainder of the segment
  - The receiver TCP performs the same calculation, and compares the incoming code for any error detection

Operation of TCP and IP Contd.

- An example of an item stored in the IP header is the destination host address
- The header appended at the network access layer contains the information that is needed to transfer data across the subnetwork
- Examples of the items that may be contained in this header includes:
  - Destination address within the subnetwork
  - Facilities requests such as priority

Operation of TCP and IP Contd.

- At a router, the packet header (network header) is stripped off and the IP header is examined
- On the basis of the destination address in the IP header, the datagram is directed out across the subnetwork
  - To achieve this, the datagram is again augmented with a network access header
- Some applications that have been standardised to operate on top of TCP are Simple Mail Transfer Protocol (SMTP), File Transfer Protocol (FTP) and TELNET

Some Protocols in TCP/IP Suite