Fundamentals of Computer Networks

Understand computer network technology

Lesson 2

OSI Model

Open System Interconnection Reference model

OSI Model (recall)

- Open System Interconnection (OSI) model
- A reference model developed by **International Organization for Standardization (ISO)** that provides a common basis for coordination of standards development for the purpose of systems interconnection.
- 7 layers reference model that defines and separates networking hardware and software into distinct layers and functions
- International Organization for Standardization (ISO)
  - An independent, non-governmental, international standard development organization composed of representatives from national standards organizations of member countries.
  - ISO attempts to standardize and define items that increase communication and compatibility in many different countries

OSI Reference Model (recall)

Layer			Protocol data unit (PDU)	Function
Host layers (end devices, hosts, servers, computers)	7	Application	Data	Interfaces directly with end-user applications, providing network services like email, file transfer, and web access
	6	Presentation		Translates data formats between systems; handles encryption, compression, and encoding.
	5	Session		Establishes, manages, and terminates sessions between applications
	4	Transport	Segment / Datagram	Provides end-to-end communication, error recovery, and flow control
Media layers (network devices, routers, Switches, Network Interface card (NIC))	3	Network	Packet	Manages routing of data packets between devices across different networks
	2	Data link	Frame	Ensures reliable data transfer across the physical link; handles framing, error detection, and MAC addressing
	1	Physical	Bits	Transmission and reception of bitstream over a transmission medium (physical medium)

Responsibilities of Each Layer (recall)

- The purpose of each layer in the OSI model is to provide services to the layer **above** it while shielding the upper level from what happens below.
- The higher layers do not need to know how the data got there or what happened at the lower layers.

Network Layer

Layer			Protocol data unit (PDU)
Host layers	7	Application	Data
	6	Presentation	
	5	Session	
	4	Transport	Segment / Datagram
Media layers	3	Network	Packet
	2	Data link	Frame
	1	Physical	Bits

- The network layer is responsible for the source-to-destination delivery of a packet, possibly across multiple networks. Whereas the data link layer oversees the delivery of the packet between two systems on the same network, the network layer ensures that each packet gets from its point of origin to its final destination.
- The network layer does not recognize any relationship between the packets. It treats each one independently, as though each piece belonged to a separate message.

Network Layer – Other responsibilities

- **Logical addressing.** If a packet passes the network boundary, the network layer adds a header to the packet coming from the upper layer that includes the logical addresses of the sender and receiver. The logical address helps to distinguish the source and destination systems.
- **Routing.** When independent networks or links are connected together to create internetworks (network of networks) or a large network, the connecting devices (called routers or switches) route the packets to their final destination. Network layer provides this mechanism.

Network Layer

- If two hosts are connected to the **same** network, network layer is needed to accomplished source-to-destination delivery, although no routing is needed.
  - Sender – create IP packet with source and destination IP address
  - Based on the destination IP, the system will decide whether the destination is on the same or different network.
  - If same network, it uses Address Resolution Protocol to find the destination MAC address, Data Link Layer sends the frame directly to the destination host
  - No routing is performed (no router), packet remains in the network

Network Layer

- If the two systems are attached to **different** networks with connecting devices (routers) between the networks, there is a need for the network layer to accomplish source-to-destination delivery with routing.
  - Sender – create IP packet with source and destination IP address
  - Based on the destination IP, the system will decide whether the destination is on the same or different network.
  - If different network, the packet is sent to a router (default gateway). Each router will examine the destination IP and decides on the next hop.
  - The packet travels across multiple networks until it reaches the destination host.

Transport Layer

Layer			Protocol data unit (PDU)
Host layers	7	Application	Data
	6	Presentation	
	5	Session	
	4	Transport	Segment / Datagram
Media layers	3	Network	Packet
	2	Data link	Frame
	1	Physical	Bits

- The transport layer is responsible for process-to-process delivery of the entire message. A process is an application program running on the host.
- The transport layer ensures that the whole message arrives intact and in order, overseeing both error control and flow control at the source-to-destination level.

Transport Layer – Other responsibilities

- **Service-point addressing**. Computers often run several programs at the same time. For this reason, source-to-destination delivery means delivery not only from one computer to the next but also from a specific process (running program) on one computer to a specific process (running program) on the other.
- The transport layer header must therefore include a type of address called a service-point address (or port address). The network layer gets each packet to the correct computer; the transport layer gets the entire message to the correct process on that computer.

Transport Layer – Other responsibilities

- **Segmentation and reassembly**. A message is divided into transmittable segments, with each segment containing a sequence number. These numbers enable the transport layer to reassemble the message correctly upon arriving at the destination and to identify and replace packets that were lost in transmission.
- **Connection control.** The transport layer can be either connectionless or connection oriented. A connectionless transport layer treats each segment as an independent packet and delivers it to the transport layer at the destination machine. A connection oriented transport layer makes a connection with the transport layer at the destination machine first before delivering the packets. After all the data are transferred, the connection is terminated.

Transport Layer – Other responsibilities

- **Flow control.** Like the data link layer, the transport layer is responsible for flow control. However, flow control at this layer is performed end to end rather than across a single link.
- **Error control.** Like the data link layer, the transport layer is responsible for error control. However, error control at this layer is performed process-to-process rather than across a single link. The sending transport layer makes sure that the entire message arrives at the receiving transport layer without error (damage, loss, or duplication). Error correction is usually achieved through retransmission.

Session Layer

Layer			Protocol data unit (PDU)
Host layers	7	Application	Data
	6	Presentation	
	5	Session	
	4	Transport	Segment / Datagram
Media layers	3	Network	Packet
	2	Data link	Frame
	1	Physical	Bits

- Provide the ability to the presentation entities to organize the communication for multiple communication sessions taking place at the same time.
- **Session initiation and teardown.** The session layer is responsible for starting the sessions between the communicating entities. And when the session is over, the session layer is responsible for the release of the communication. Data transfer takes place in-between.

Session Layer

- **Token management.** This function is related to the communication mode used in the specific session (simplex, half-, or full-duplex). The session layer controls which entity own the token and can transmit data at this time. The token is the license to transmit in an environment or service where only one entity can transmit at a time. This is not the case for all applications. Some applications operate in full-duplex mode, others operate in half-duplex mode.
- **Session-connection to transport-connection mapping (in connection-oriented transfer only)** The function provides the session layer the ability to map between the transport layer connections and the sessions currently taking place. This way the session layer can tell which data goes to what session.

Presentation Layer

Layer			Protocol data unit (PDU)
Host layers	7	Application	Data
	6	Presentation	
	5	Session	
	4	Transport	Segment / Datagram
Media layers	3	Network	Packet
	2	Data link	Frame
	1	Physical	Bits

- Responsible for the way data is presented to the application. At the start of the communication, the presentation layer negotiates the form of data to be transferred with the other entity, the transfer syntax. After this negotiation, the presentation layer can provide additional services such as compression, encryption, and translation. The choice of which service(s) to be used is up to the application itself.

Application Layer

Layer			Protocol data unit (PDU)
Host layers	7	Application	Data
	6	Presentation	
	5	Session	
	4	Transport	Segment / Datagram
Media layers	3	Network	Packet
	2	Data link	Frame
	1	Physical	Bits

- Defining the services presented at the user-end.
- Main functions:
  - Identification of services provided to the user
  - Defining QoS parameters required for the application
  - Defining security mechanisms to be employed - access control, authentication
  - Synchronization of communication applications (connection-oriented services)

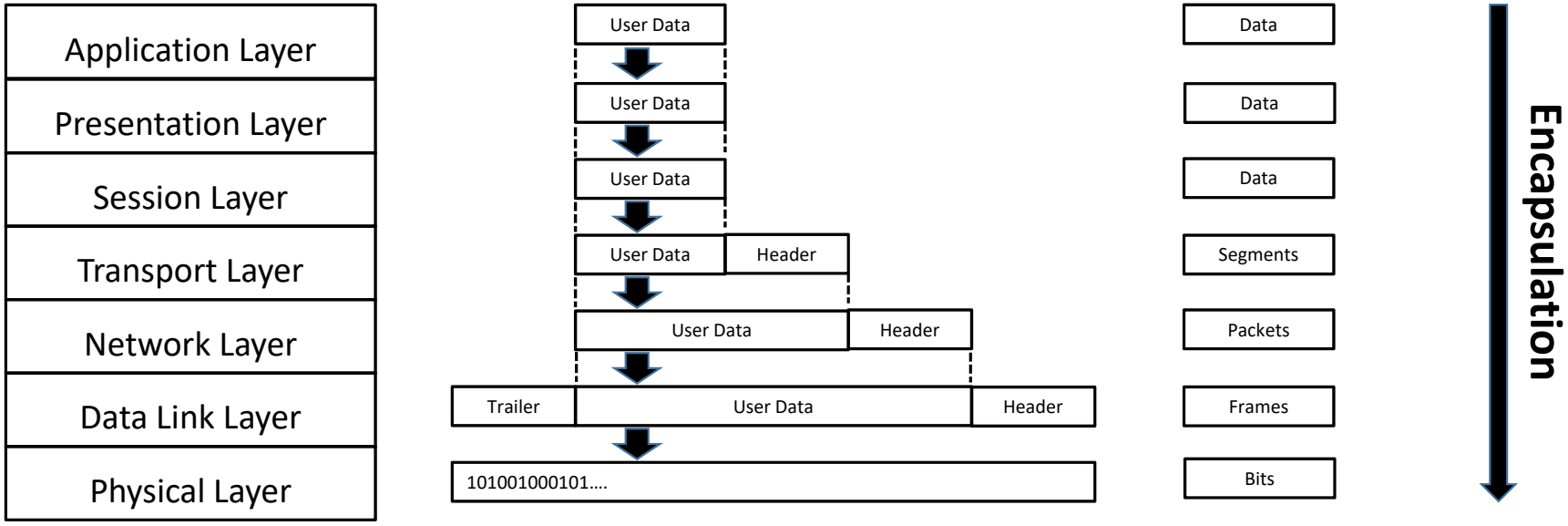
OSI Model

Data Encapsulation/ Decapsulation

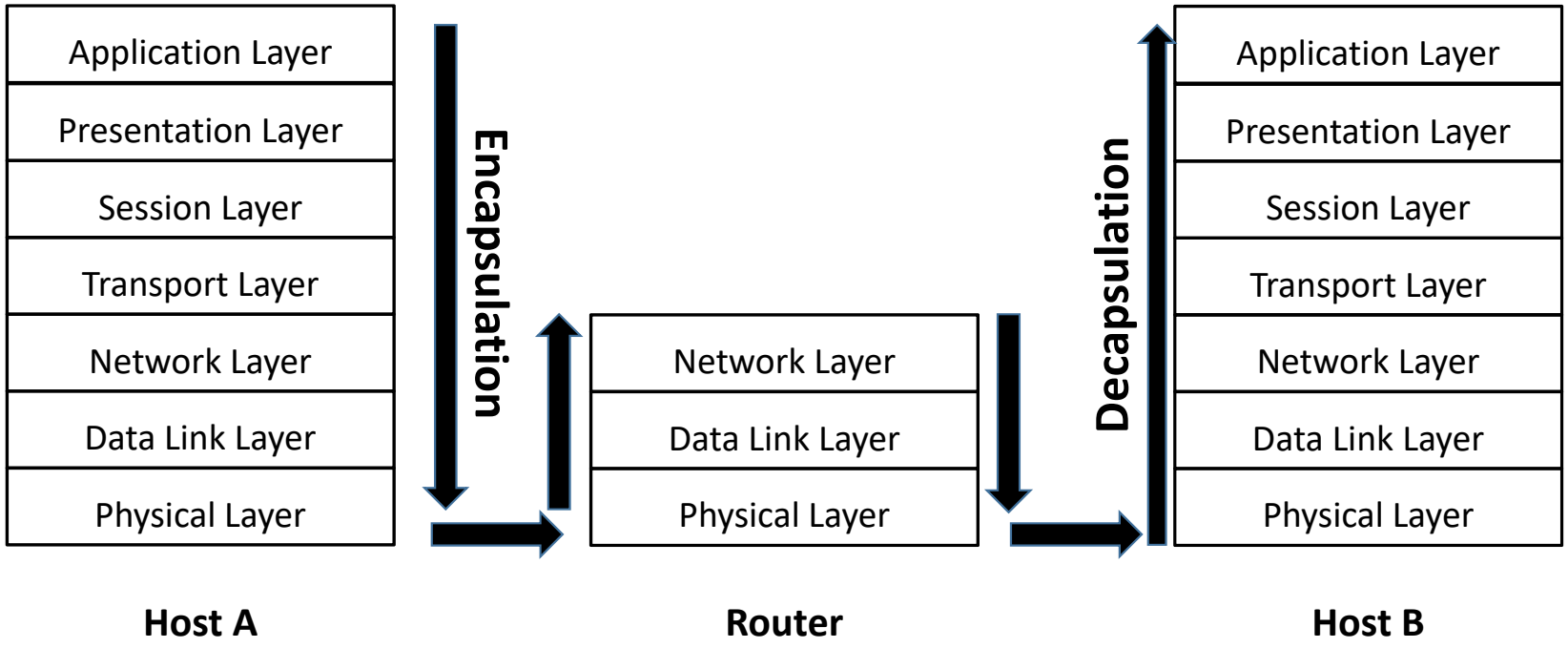
Describes how data is wrapped with headers/trailers as it moves **down** the OSI layers (encapsulation) and **unwrapped** as it moves **up** the layers (decapsulation) during transmission and reception.

Data Encapsulation

- The flow of data from the application layer to the physical layer is called encapsulation.
- Header and trailer information is added to the data in various layers at the end and start of data.



Data Decapsulation



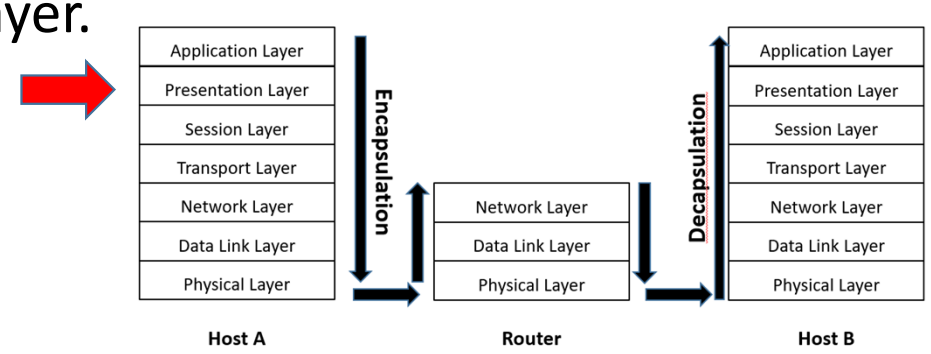
OSI Model

End-to-End Data flow

Walks through how data is prepared, sent, routed, and received using the OSI model—from **Host A** to **Router(s)** to **Host B**.

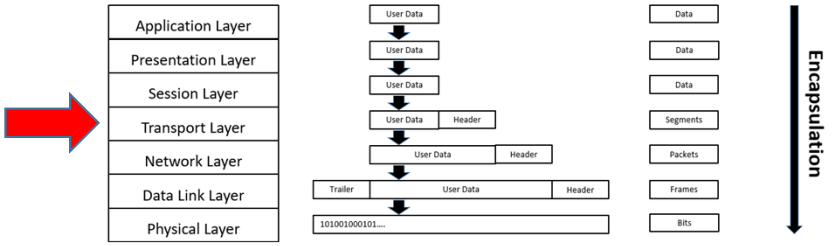
End-to-End Data Flow (Host A Source)

- Transmission starts from application layer at Host A. The application decides that it needs to communicate with Host B and passes the data down to the presentation layer.
- The presentation layer does the required transformations that need to be done on data, like compression, encryption, or translation. Then the data is passed down to the session layer.
- The session layer starts initiating the communication session and passes the data to the transport layer.



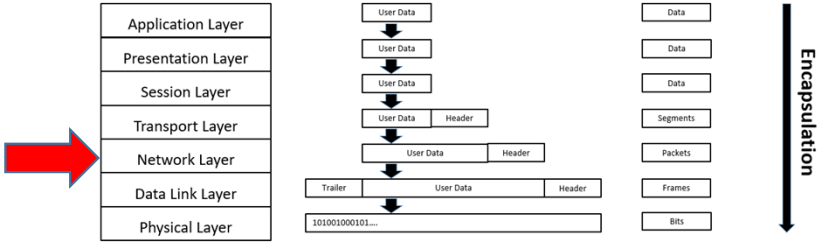
End-to-End Data Flow (Host A Source)

- At the transport layer, the data is segmented and a header is added to each segment of the data.
- This header contains transport control information such as the sequence number and acknowledgement number.
- The segment along with its header is passed down to the network layer.
- The header added by the transport layer is meant to be read by the transport layer at the receiving end.



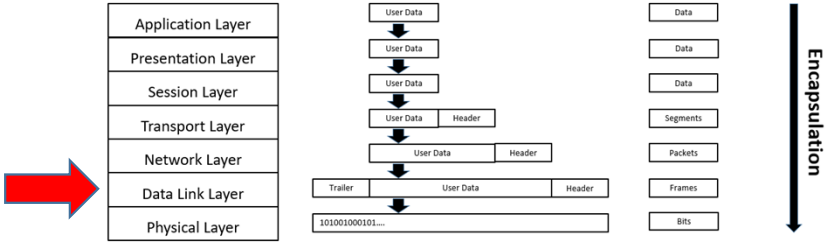
End-to-End Data Flow (Host A Source)

- The network layer deals with the whole segment, including its header, as data.
- The network layer does not read the segment header, but instead, it handles the segment and its header as a single data element.
- The data is then put into packets and headers are added to these packets.
- The network layer header contains information meant to reach the network layer on the other end.



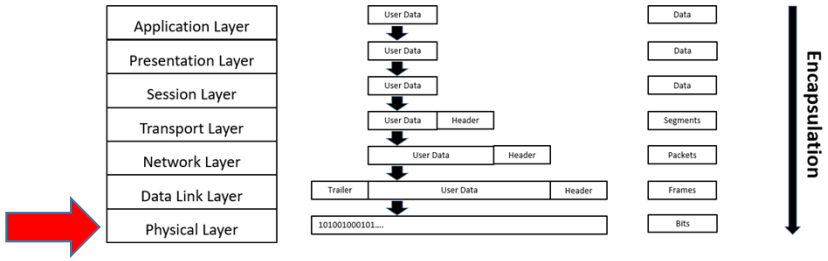
End-to-End Data Flow (Host A Source)

- This information includes a source and destination network-layer address along with few other parameters. These packets are sent down to the data-link layer.
- The data-link layer handles the packet and its header as a single data element, such that the network layer header is considered part of the data.
- The data-link layer puts the data into frames and adds a header and a trailer to each frame. eg. Ethernet frame



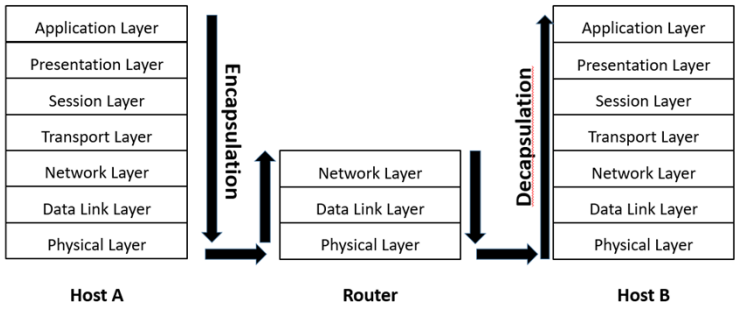
End-to-End Data Flow (Host A Source)

- The data-link header contains control information sent to the data-link layer on the other end, while the trailer usually contains error control information.
- The frames are then sent down to the physical layer where they are dealt with as raw bits. These bits are transferred through the physical channel to the router.



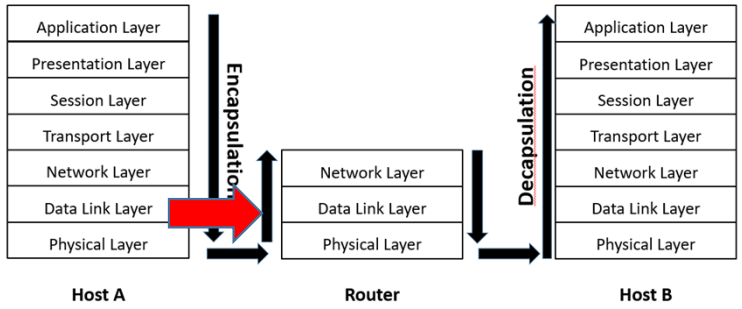
End-to-End Data Flow (Router)

- The router does a partial decapsulation because it does not need to read the data all the way up to its application-layer shape.
- The router needs only to read up to the network layer header to route the data to the destination.



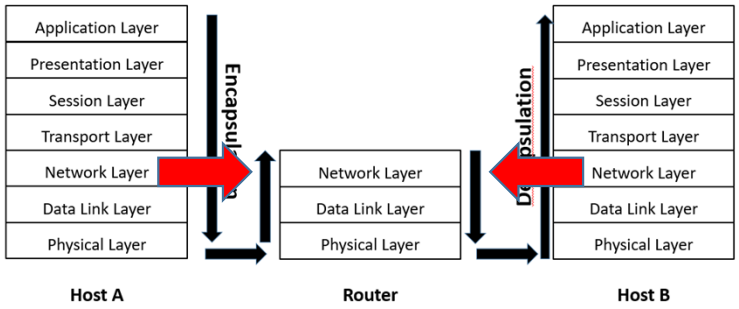
End-to-End Data Flow (Router)

- The process starts at the physical layer when the data is received as raw bits. These bits are then gathered into frames and sent over to the data-link layer.
- The data link layer reads the header and the trailer to know what to do with the data. The data-link layer then rips off the header and the trailer and sends the rest of the data as a packet to the network layer.



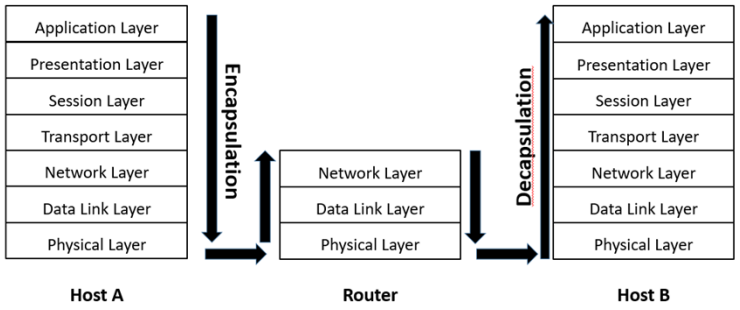
End-to-End Data Flow (Router)

- At the network layer, the network header is read to determine the destination network address. After knowing the destination network address, the router chooses the best route to send the data to the destination host.
- Starting from the network layer, the encapsulation starts again at the router. Data goes down the data-link and physical layers and all the way through the physical link to the destination host.



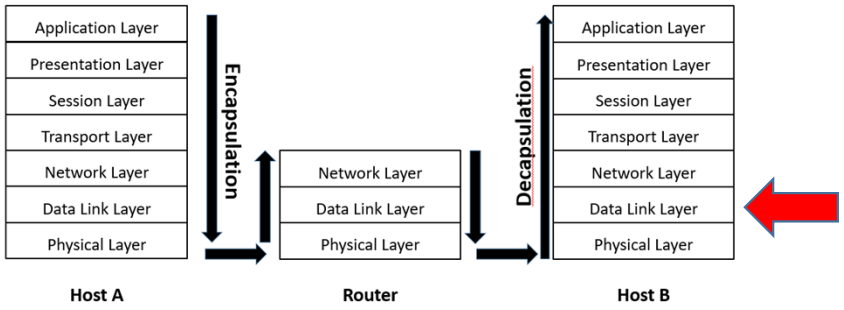
End-to-End Data Flow (Router)

- If there is another router in the way, the same process that took place in the first router is repeated until reaching the destination host.



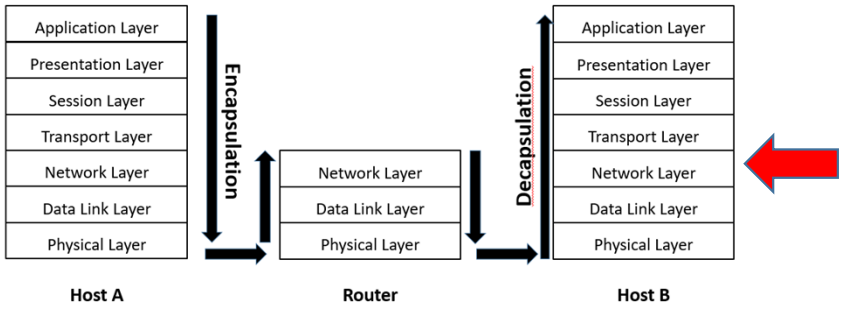
End-to-End Data Flow (Host B Destination)

- At the destination host, the raw bits are elevated as frames to the data-link layer.
- The header and trailer of each frame are read by the data-link layer and then removed. The rest of the data is elevated to the network layer as packets.



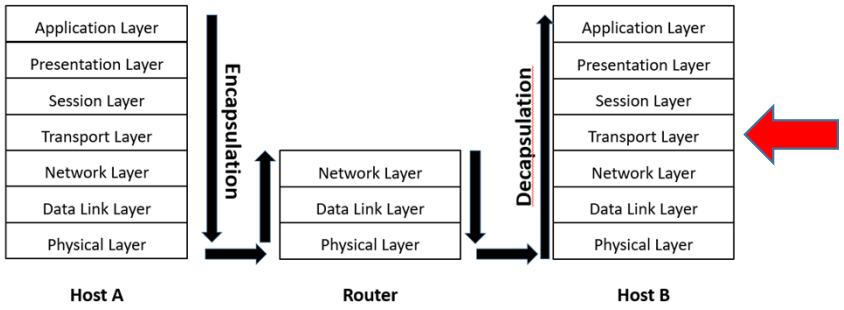
End-to-End Data Flow (Host B Destination)

- The header of the packet is read to determine if this is the correct destination for the packet and other network control information is also taken from the network layer header. The header is then removed and the rest of the data is elevated to the transport layer as segments.



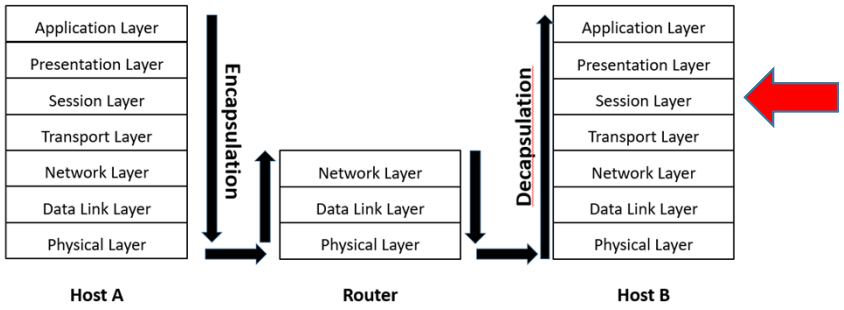
End-to-End Data Flow (Host B Destination)

- The header of each segment is read to determine the sequence number and arrange the segments in their correct order. Then, the header is also removed and the rest of the data is elevated to the session layer.
- The transport header also contains information of which session this data is going to. This information is passed to the session layer with the data.



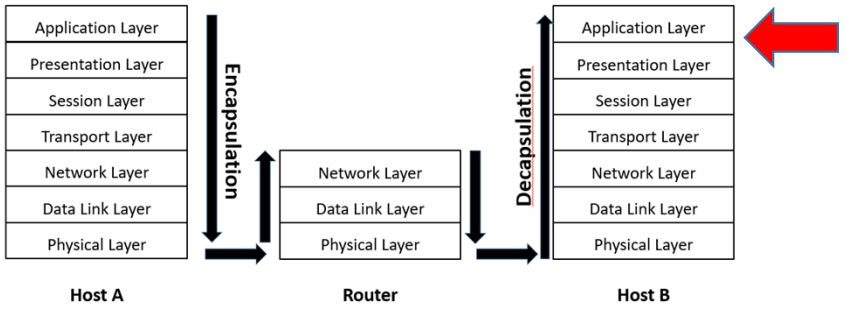
End-to-End Data Flow (Host B Destination)

- The session layer determines if this is the end of this session or not. If it is the last segment in the session, the session layer will wait for the request to end this session. If this is not the last segment in the session, the session layer waits for more data.



End-to-End Data Flow (Host B Destination)

- The data is then passed to the presentation layer to retransform the data into the shape that they were sent in by the sending-end application or to another form determined by the application. This might involve decompression, decryption, or translation.
- The data is then transferred to the application and received by the user.

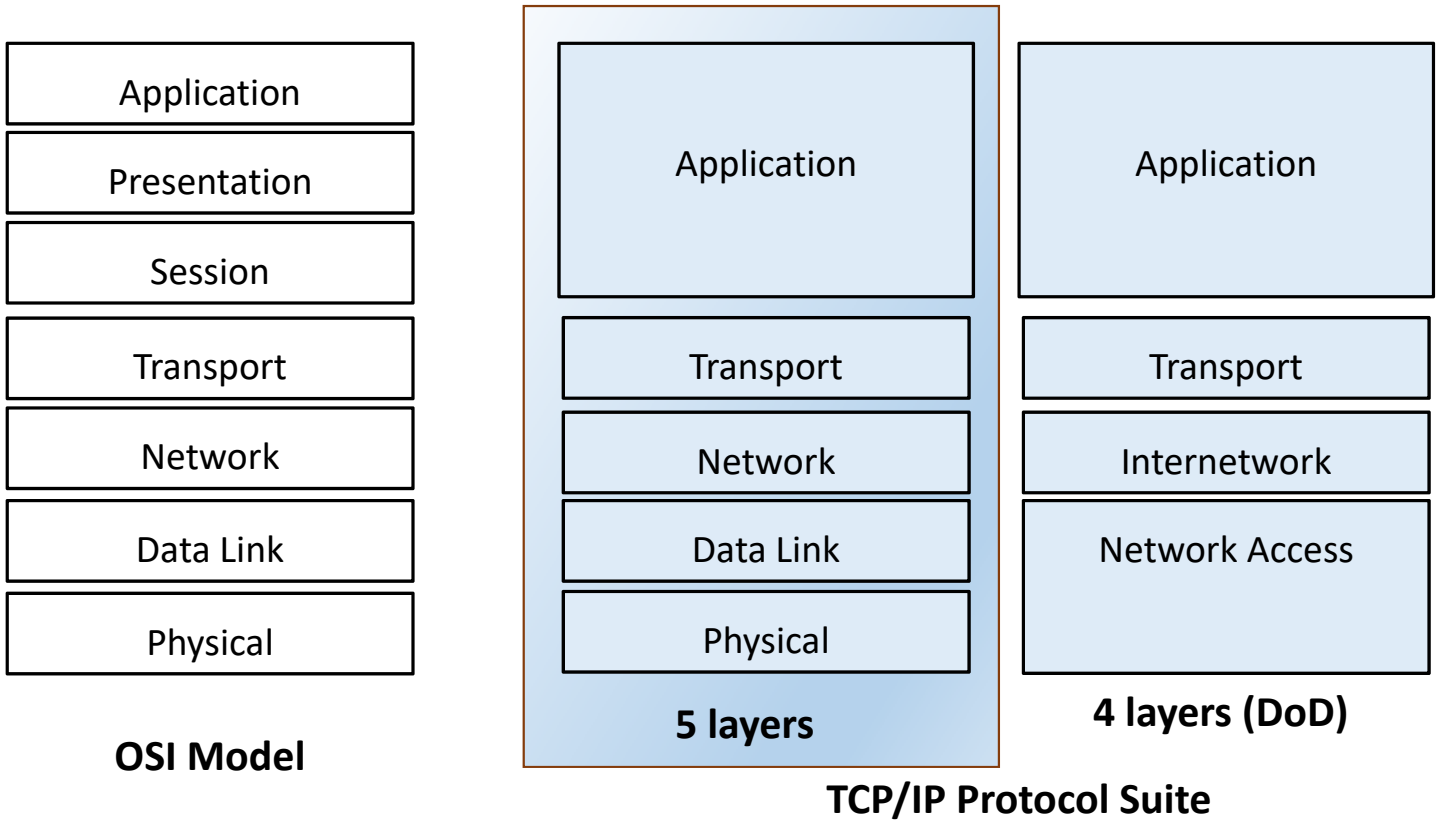


TCP/IP Protocol Suite

TCP/IP Protocol Suite/DoD Model

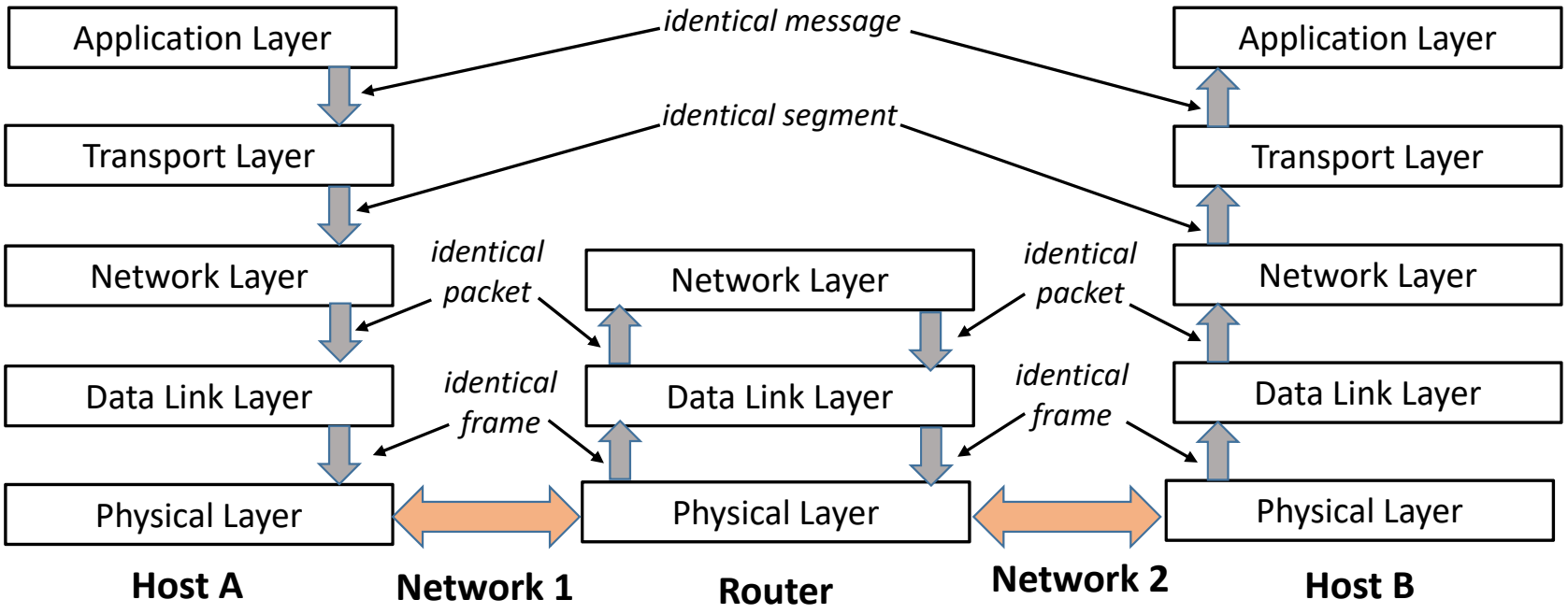
- Transmission Control Protocol/Internet Protocol (TCP/IP)
- It is a protocol suite –a set of protocols organized in different layers
- Each upper-level protocol is supported by the services provided by one or more lower-level protocols
- Developed prior to the OSI model –layers in the TCP/IP protocol suite do not match exactly with those in the OSI model.
- Developed using the Department of Defense (DoD) reference model.

TCP/IP Protocol Suite vs OSI Model



Layering Principle

- Layered protocols are designed so that layer *n* at the destination receives exactly the same object sent by layer *n* at the source.



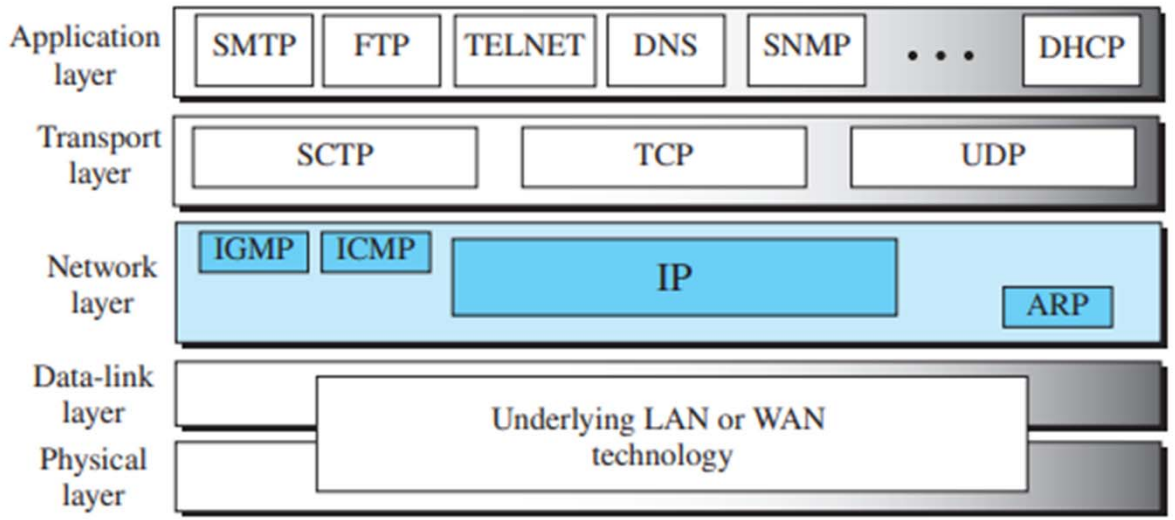
Benefits for Layering

- Modularity
  - Each layer focuses on a specific function (e.g., routing, encryption, data formatting), making the system easier to design, understand, and troubleshoot.
- Interoperability
  - By standardizing layer functions and interfaces, devices and software from different vendors can work together as long as they follow the same protocols for each layer.

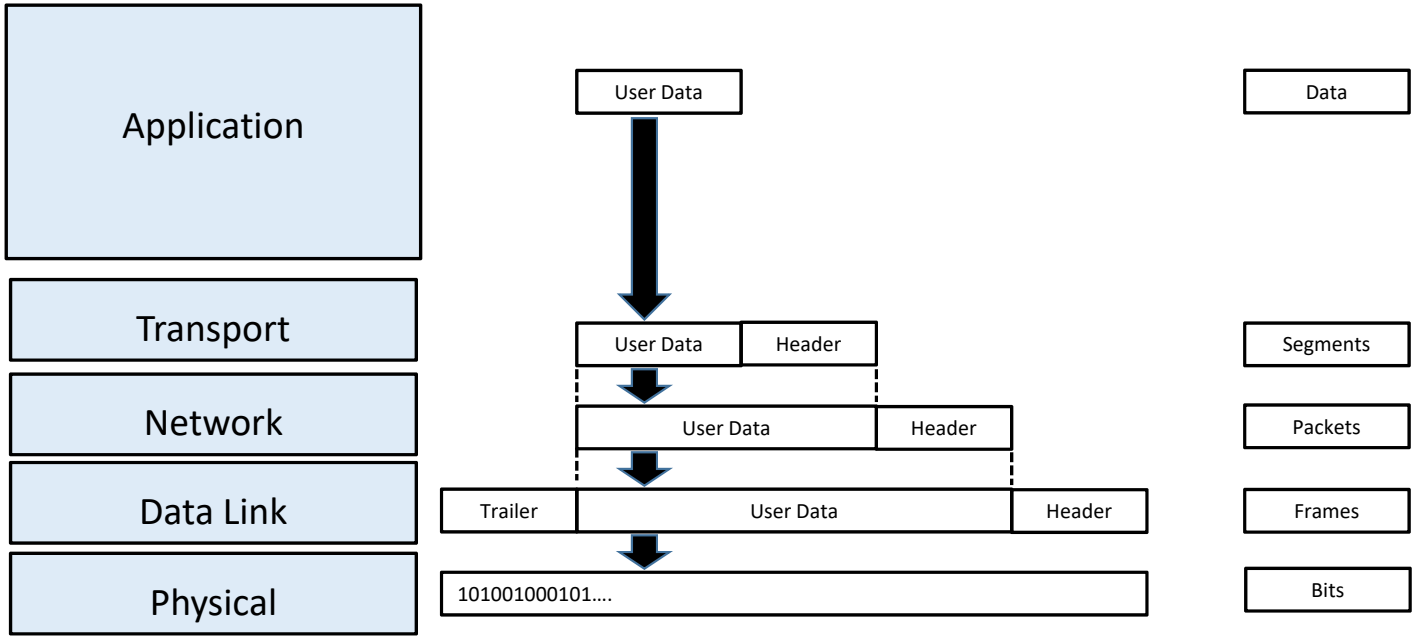
Benefits for Layering

- Flexibility and Abstraction
  - Changes or improvements can be made in one layer (e.g., upgrading a physical cable or switching encryption algorithms) without affecting other layers, due to clearly defined interfaces.
- Simplified Troubleshooting
  - Network issues can be diagnosed layer by layer. For example, if a message is received incorrectly, engineers can check whether the problem lies in transmission, formatting, or routing.
- Standardization
  - The OSI model provides a common language and framework for implementing networking protocols.

TCP/IP Protocol Suite

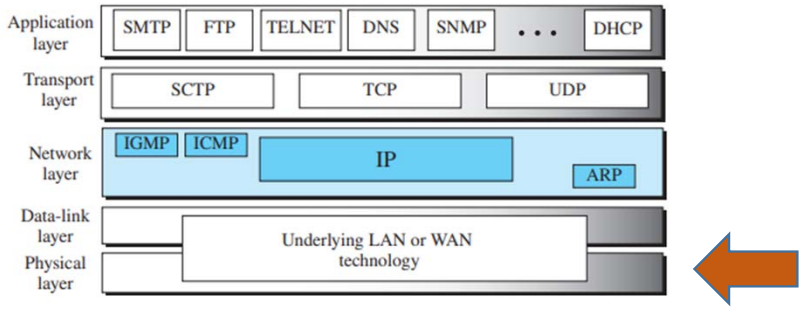


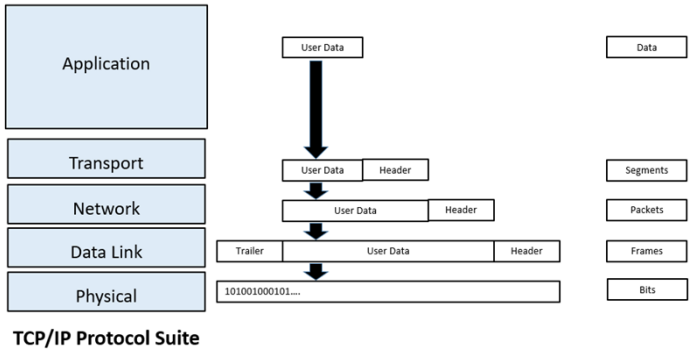
TCP/IP Protocol Suite - Data Encapsulation



TCP/IP Protocol Suite - Physical layer

- TCP/IP **does not define** any specific protocol for the physical layer.
- It supports all of the standard and proprietary protocols.
- At this level, the communication is between two nodes, either a computer or router.
- The unit of communication is a **single bit**. When the connection is established between the two nodes, a stream of bits is flowing between them.

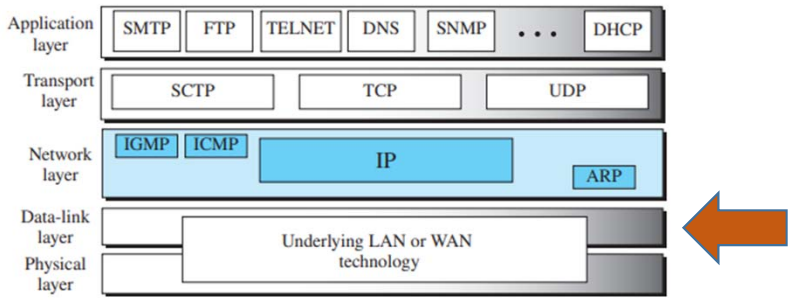


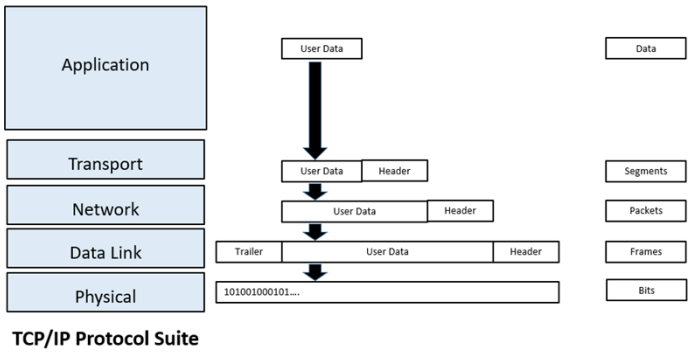


TCP/IP Protocol Suite

TCP/IP Protocol Suite - Data Link layer

- TCP/IP **does not define** any specific protocol for the data link layer.
- It supports all of the standard and proprietary protocols.
- At this level, the communication is also between two nodes.
- The unit of communication is a packet called a **frame**.
- A **frame** is a packet that encapsulates the data received from the network layer with an added header and a trailer (optional).





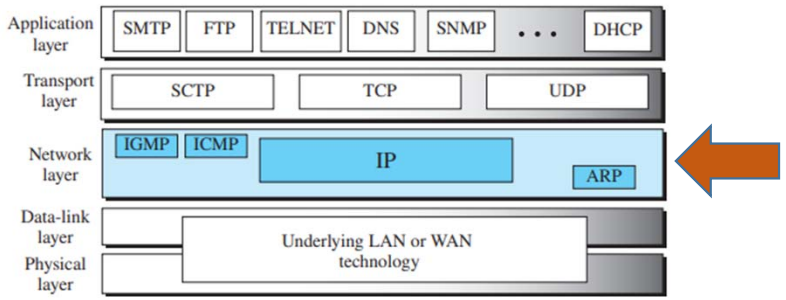
TCP/IP Protocol Suite

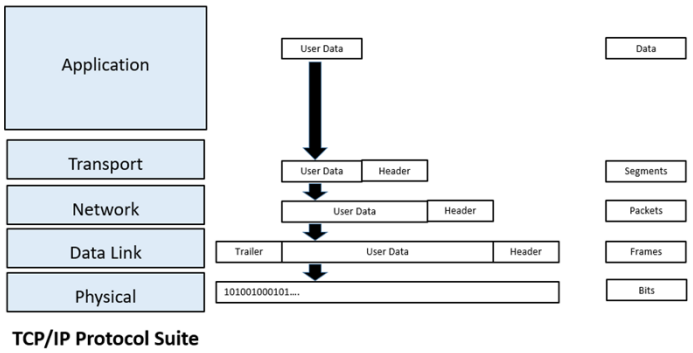
TCP/IP Protocol Suite – Layer 1 & 2

- The TCP/IP protocol suite was designed to work over any underlying network infrastructure.
- By not tying itself to specific hardware standards, TCP/IP can run over Ethernet, Wi-Fi, optical fiber, or even legacy systems like token ring.
- This makes it flexible, encouraging broad adoption across different types of networks.

TCP/IP Protocol Suite - Network layer

- The main purpose of this layer is to select the best path for the data to travel through from the source to the destination, i.e., routing.
- The leading protocol operating in this layer is Internet Protocol (IP).
- There is a group of supporting protocols that support IP in doing its job, such as the Internet Control Message Protocol (ICMP), Address Resolution Protocol (ARP), and Reverse Address Resolution Protocol (RARP).

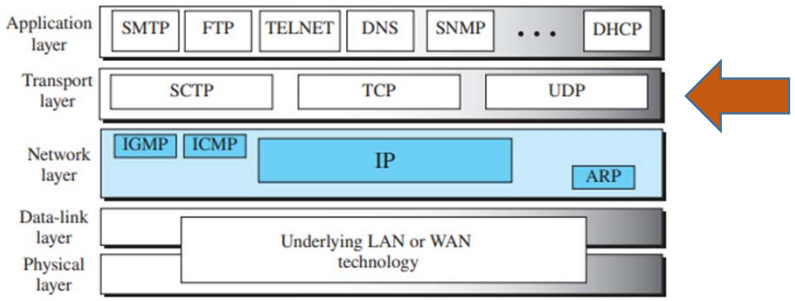


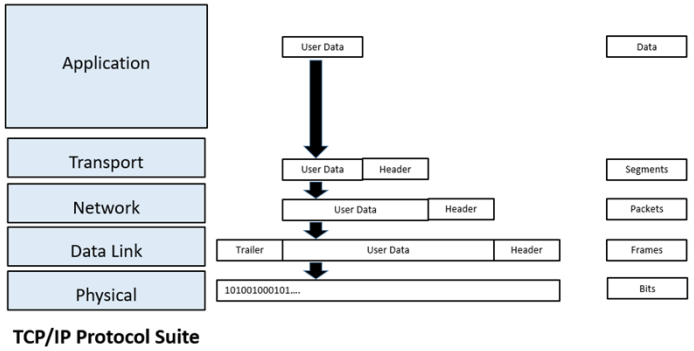


TCP/IP Protocol Suite

TCP/IP Protocol Suite - Transport layer

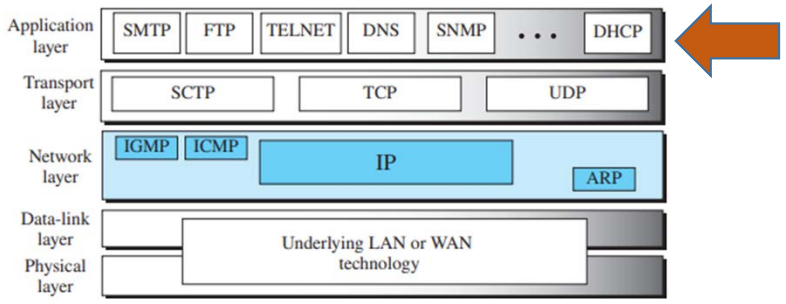
- The transport layer in the TCP/IP model has similar purposes to that of the OSI model. It is designed to give the source and destination the ability to have end-to-end conversation.
- There are two defined protocols that can operate in this layer; Transmission Control Protocol (TCP) and User Datagram Protocol (UDP). These two protocols provide connection-oriented and connectionless communications.

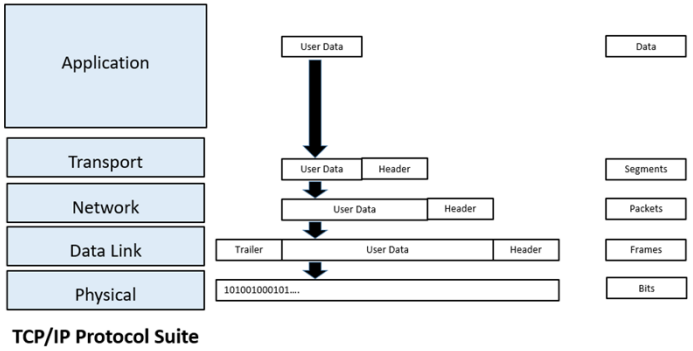




TCP/IP Protocol Suite - Application layer

- The main duty of this layer is to take data from the applications and deliver it to the transport layer and collect data from the transport layer and deliver it to the correct applications. It contains a huge group of high-level protocols that cover a wide range of applications.
- The most common is the Hyper Text Transfer Protocol (HTTP). Other widely used application layer protocols are File Transfer Protocol (FTP), Simple Mail Transfer Protocol (SMTP), Post Office Protocol 3 (POP3), Telnet, and Domain Name Service (DNS).





TCP/IP Protocol Suite

End