Carrying Out God’s
Dominion Through IoT

CALVIN
INSTITUTE OF TECHNOLOGY

Computer Networks

(IEE2072 & IBDA2022 – Jaringan Komputer)

Lecture #03 -- Network Models

Erwin Anggadjaja, PhD
Department of IoT & Electrical Engineering
CALVIN INSTITUTE OF TECHNOLOGY

@eaFEB25

Agenda / Learning Objectives

- Understanding protocol layering.
- Understanding OSI Model and its layers.
- Understanding TCP/IP Protocol Suite

PART 1: Protocol Layering

Layering: One-layer concept

- ❑ In data communication and networking, a **protocol** defines the rules that both the sender and receiver, and all intermediate devices need to follow to be able to communicate effectively.
- ❑ When communication is simple, we may need only one simple protocol; when the communication is complex, we may need to divide the task between different layers, in which case we need a protocol at each layer, or protocol layering.

Daisy

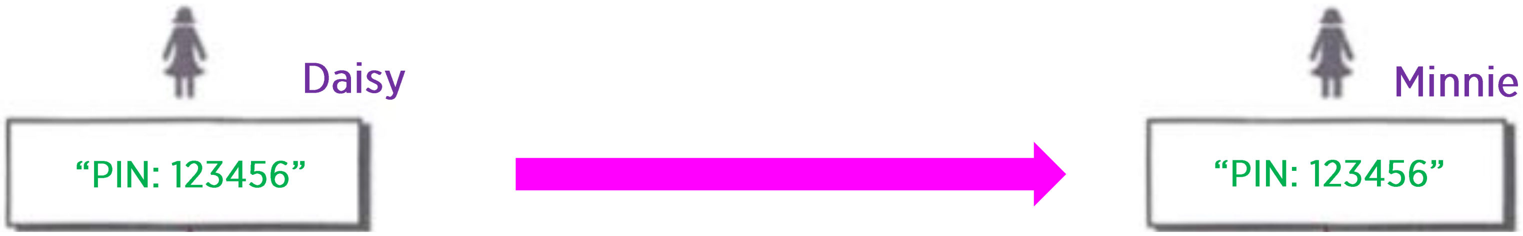
Minnie

The possible "protocol" between them:

Hallo, apa kabar?	Heiii!! Apa kabar juga!
This chair, can meh?	Can.. can.
Ada kompetisi MLBB nih	Gass lah
OK deh, see you yall!!	Sip. Bye.

What happens if the conversation is between a professor and a high-school student?

Layering: Example of three-layer concept



Daisy

“PIN: 123456”

Minnie

“PIN: 123456”

Layering: Advantage (1/3)

❑ With layering, complex task can be divided into several smaller and simpler tasks!

❑ In the previous example, YES, Daisy could just send Minnie about her PIN number...

- But..
  - Someone may hack the message easily *[security concern]*
  - The message could go to other party *[addressing concern]*
  - The message could be broken/disordered *[framing concern]*
  - etc . . .
- Then Daisy or Minnie need to resend the PIN number again.. with worrying the same concerns.

❑ Using layering, Daisy or Minnie just need to solve one of the concern!

Layering: Advantage (2/3)

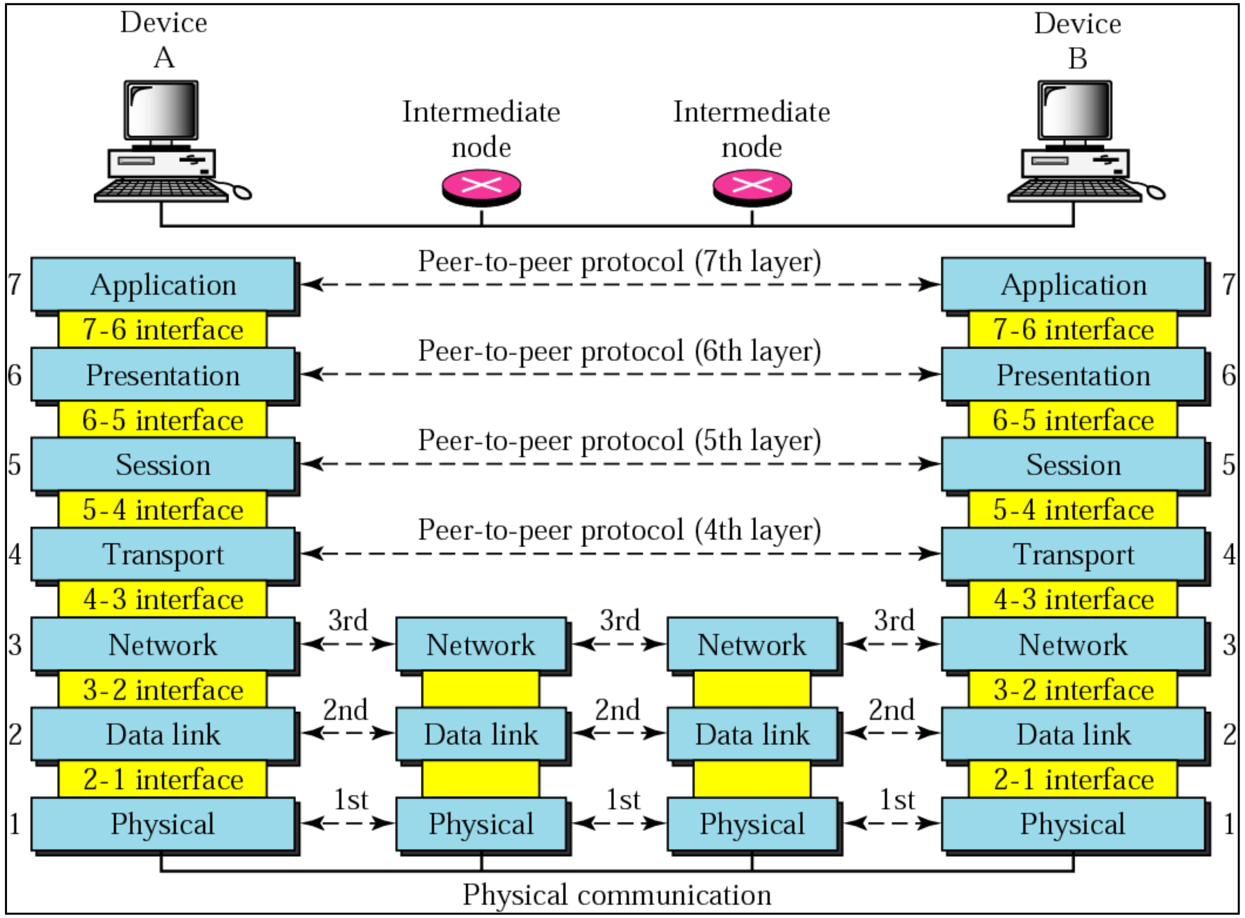
- ❑ With layering, it allows us to separate the services from the implementation.

- ❑ A layer needs to be able to receive a set of services from the lower layer and to give the services to the upper layer; we don't care about how the layer is implemented.
  - For example, in the case of layer 5 and 6 are "absent", as long as layer 4 can do the tasks provided by the 3rd layer, and 6th layer, the communication system still fine.

Layering: Advantage (3/3)

- ❑ With layering, communication does not always use only two end systems; there are intermediate systems that need only some layers, but not all layers!

- ❑ If we did not use protocol layering, we would have to make each intermediate system as complex as the end systems, which makes the whole system more expensive.



Layering: Principle (1/2)

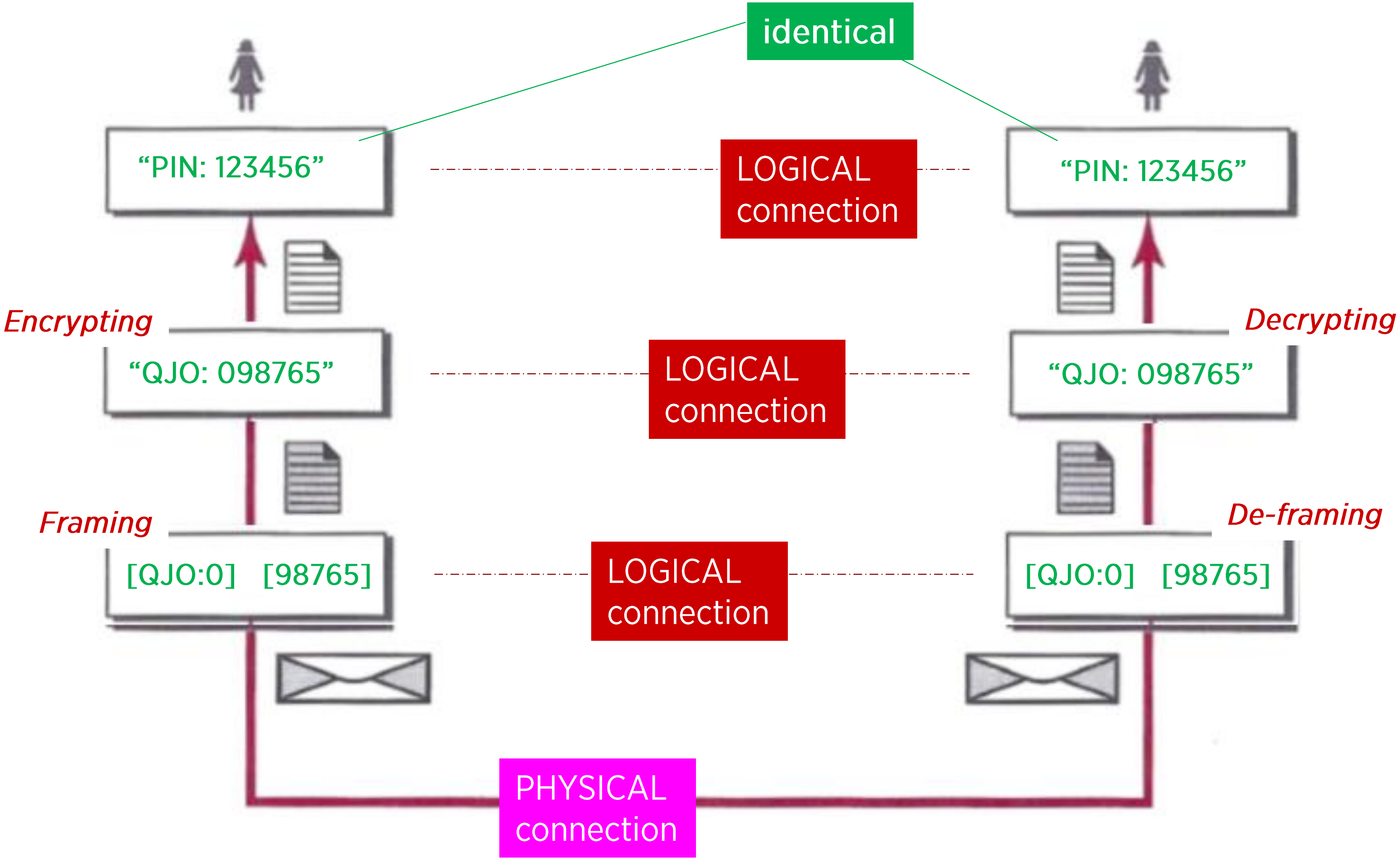
1st principle

- ❑ Each layer must able to perform two opposite tasks, one in each directions.
- ❑ For examples:
  - Layer 2 must be able to do *framing* and *de-framing*
  - Layer 1 must be able to *send* and *receive* data
  - Etc . .

2nd principle

- ❑ Each objects (the text) under each layer should be identical, with the connections are called logical connection.

Layering: Principle (2/2)



PART 2: OSI Model

OSI Model

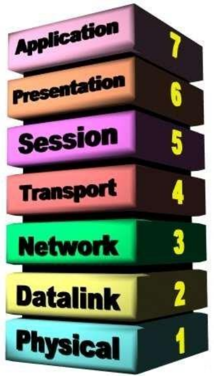
❑ Established in 1947, there was *International Standards Organization* (ISO), a multinational body dedicated to worldwide agreement on international standards.

❑ Emphasizing on network communication, the *Open Systems Interconnection* (*OSI*) model was introduced in the late 1970s.

- An *open system* is a set of protocols that allows any two different systems to communicate regardless of their underlying architecture.

- The purpose of the OSI model is to show how to facilitate communication between different systems without requiring changes to the logic of the underlying HW & SW.

- OSI model is not a protocol!! It is a model for understanding and designing a network architecture that is flexible, robust, and interoperable.

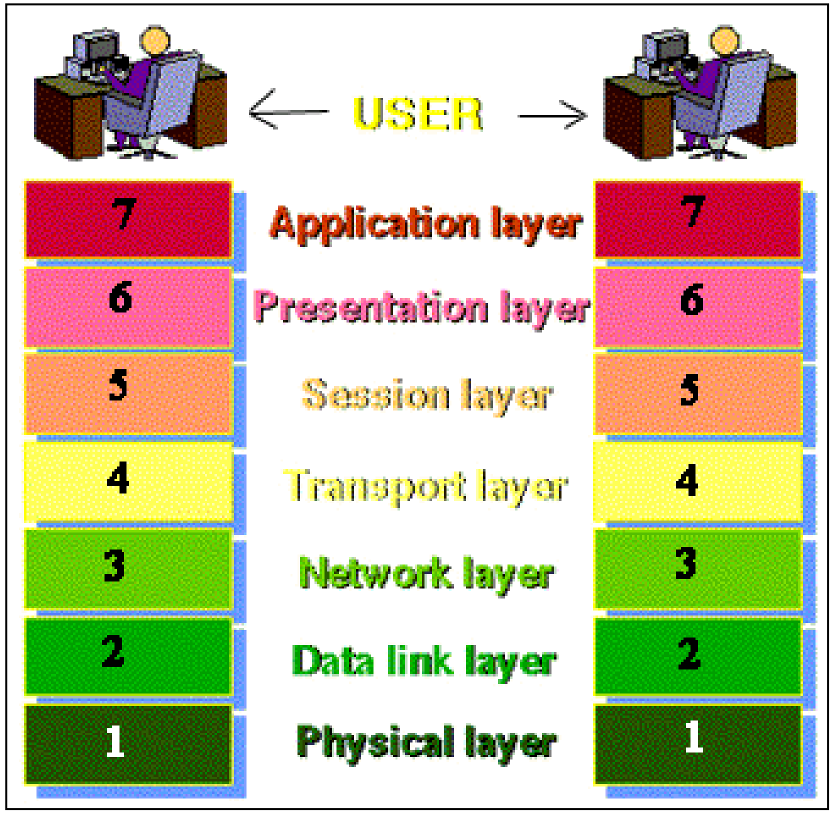


OSI Model

- ❑ The OSI model is a layered framework for the design of network systems that allows communication between all types of computer systems.

- ❑ It consists of seven *separated-but-related* layers, for each of them defines a part of the process of moving information across a network.

- ❑ Understanding the fundamentals of the OSI model provides a **solid basis** for exploring data communication.



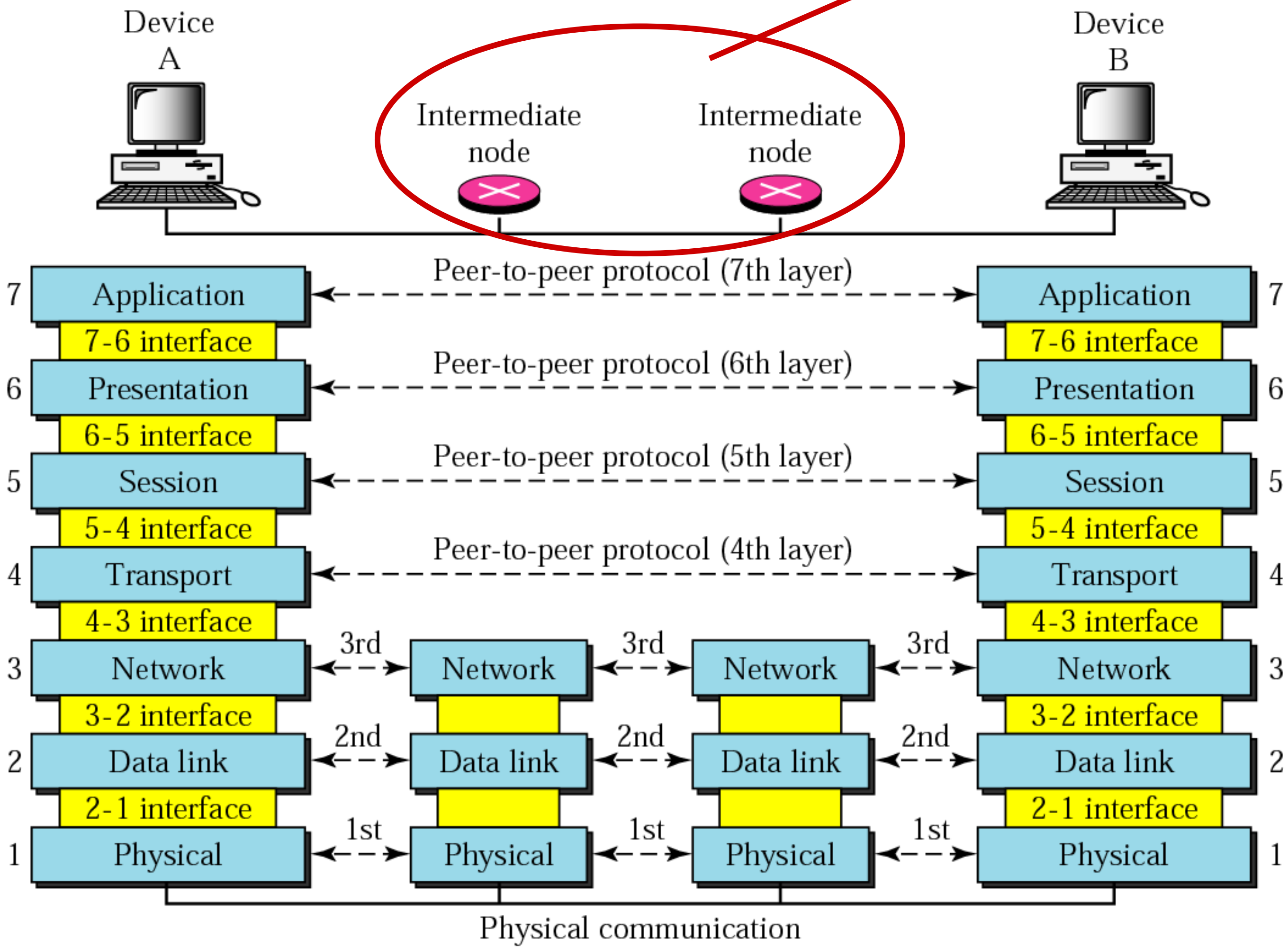
Layered Architecture

❑ When a message is sent from Device A to Device B, it may pass through many intermediate nodes (usually involve only the first three layers).

- Layer *X* on one machine logically communicates with layer *X* on another machine, by an agreed set of rules (protocols).

- Each layer calls upon the services of the layer just below it.

For example,
Layer 5 uses the services provided by layer 4, and provides services for layer 6.



CALVIN INSTITUTE OF TECHNOLOGY

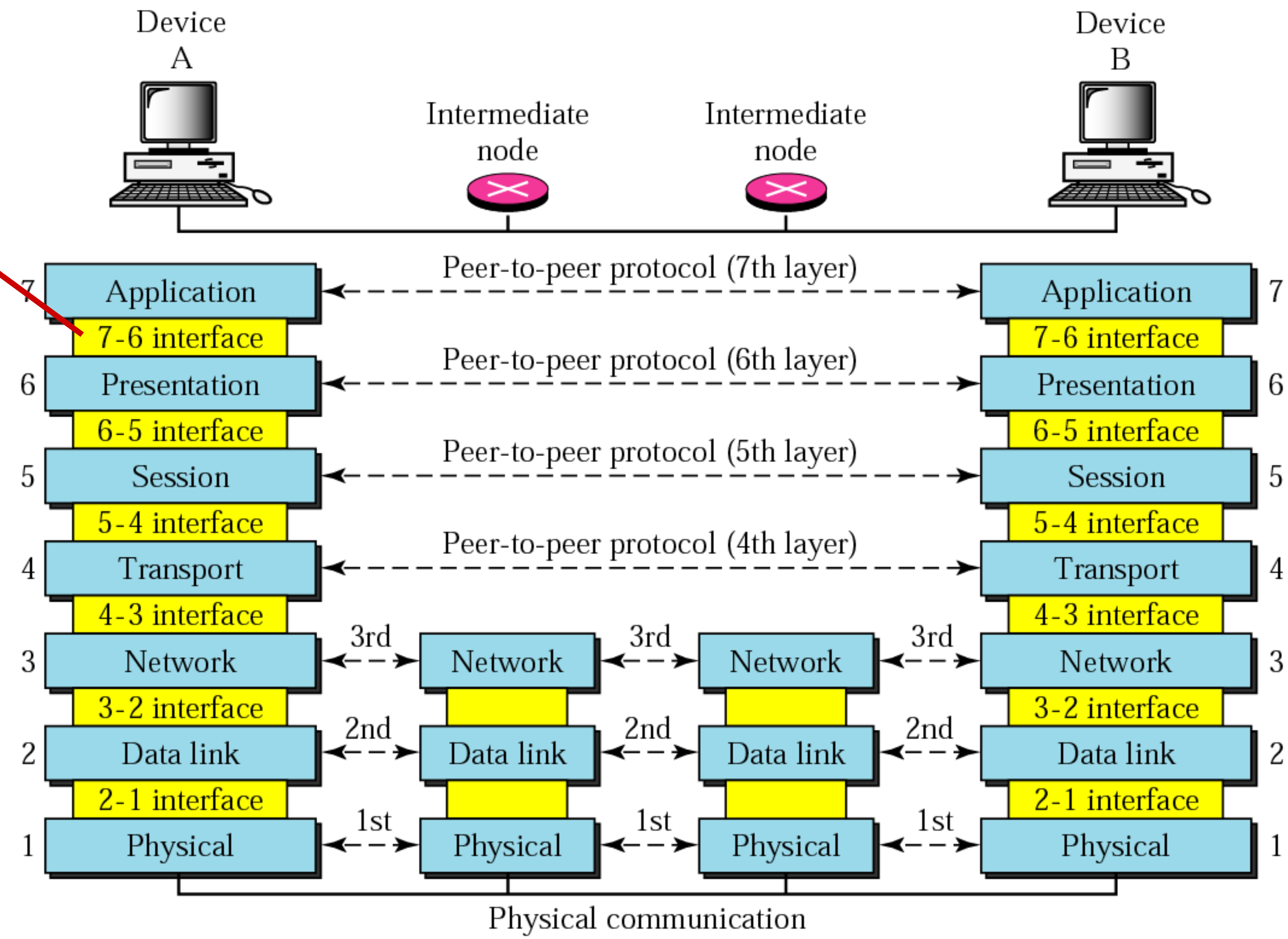
Interfaces Between Layers

- ❑ There is an interface between adjacent layers, which makes the passing of the data possible

- ❑ Each interface defines what information and services a layer must provide for the layer above it.

Notes

- ○ **Upper layers** are almost always implemented in software.
- ○ **Lower layers** are a combination of hardware and software.
- ○ **Physical layer** mostly hardware.



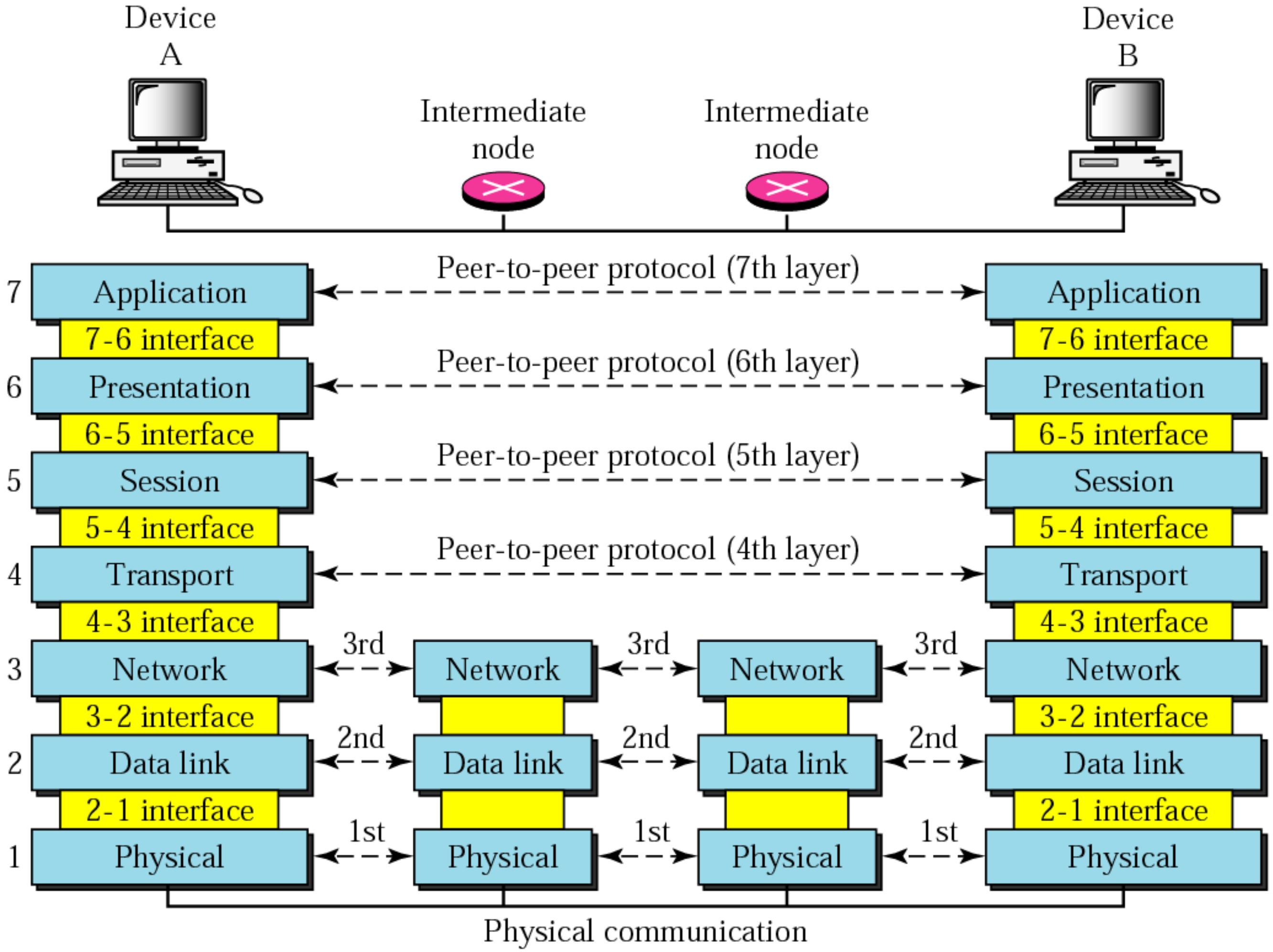
Organization of the layers

- ❑ Those seven layers can be classified as three sub-group:

Layers 5, 6, 7 ➔ **user support** layers, which allow interoperability among unrelated software systems.

Layer 4 is the **transport** layer, which linking the two sub-group to ensure the data is *understandable* and *usable*.

Layers 1 -3 ➔ **network support** layers, which take care of physical aspects of data (electrical specifications, physical connections, physical addressing, and transport timing and reliability).



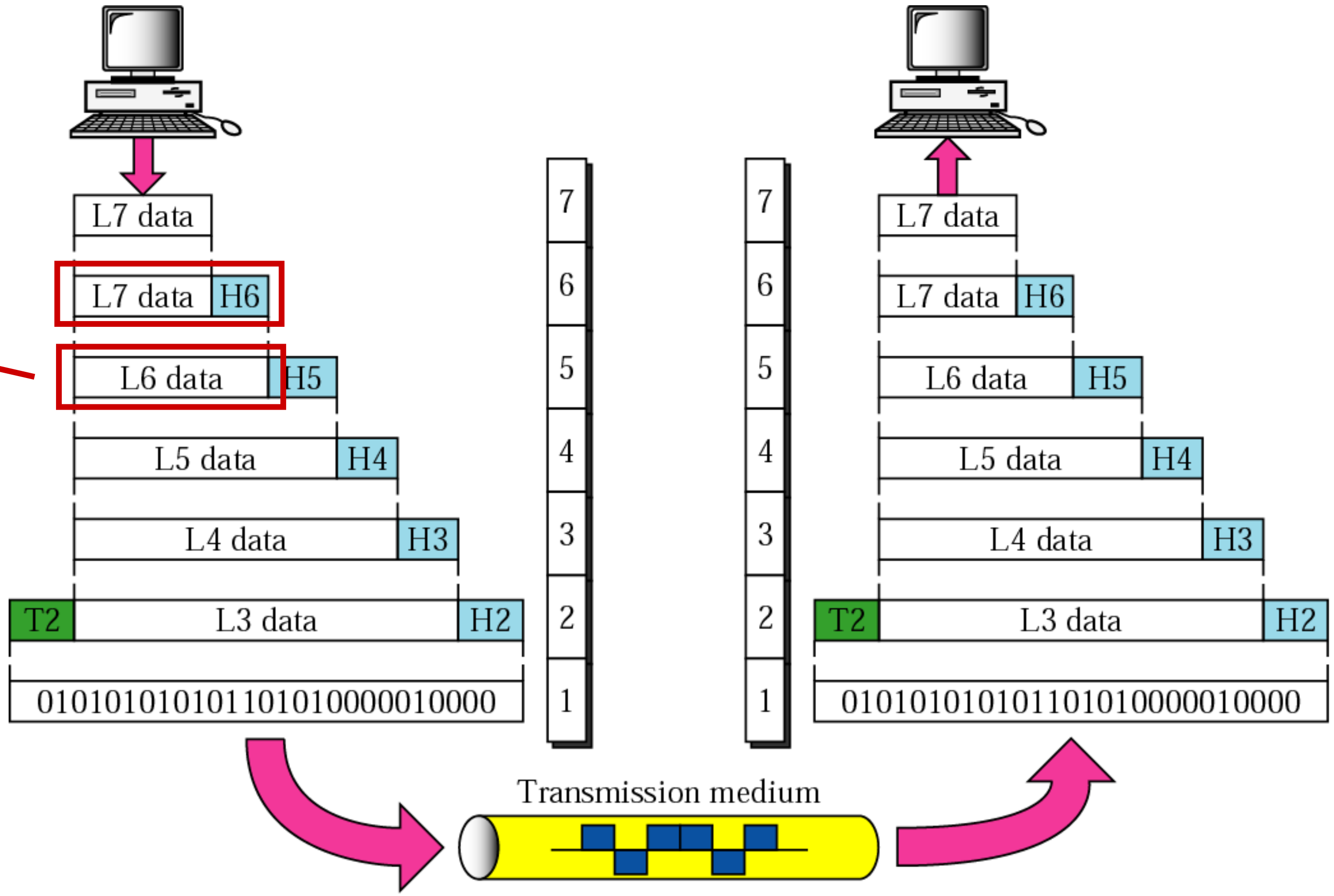
Encapsulation (1/2)

❑ Encapsulation is the packing of data and functions into a single component.

The data of a packet at level *N* is carrying the whole packet (data plus overhead) from level (*N+1*).

For level *N*, the whole packet coming from level *N+1* is treated as one integral unit.

Level *N* is not aware what part of the encapsulated packet is *data* and what part is the *header/trailer*.



Encapsulation (2/2)

❑ The process starts at layer 7, then descends in the sequential order.

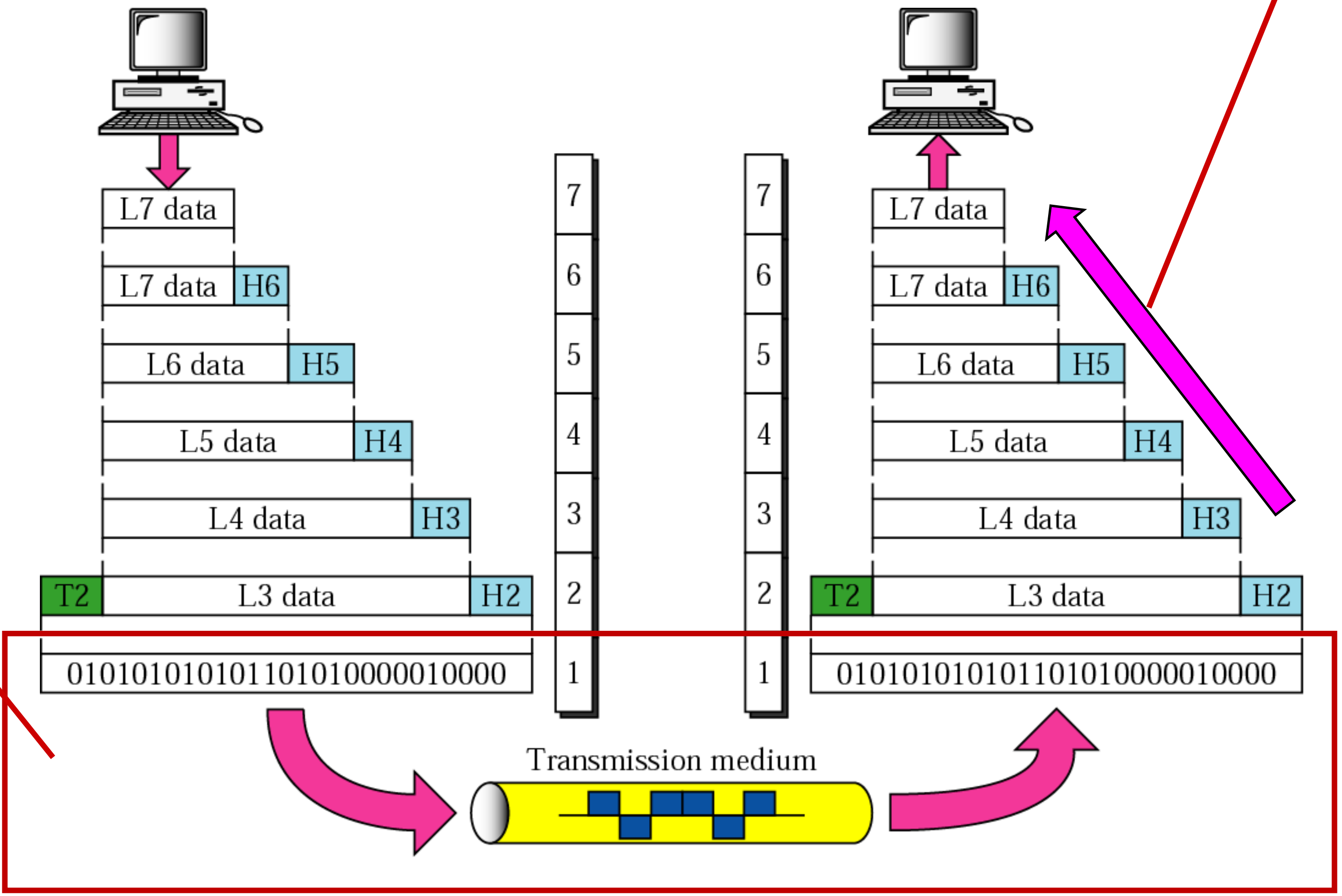
Later, at each layer of the recipient, the process of decapsulation are done.

At each layer

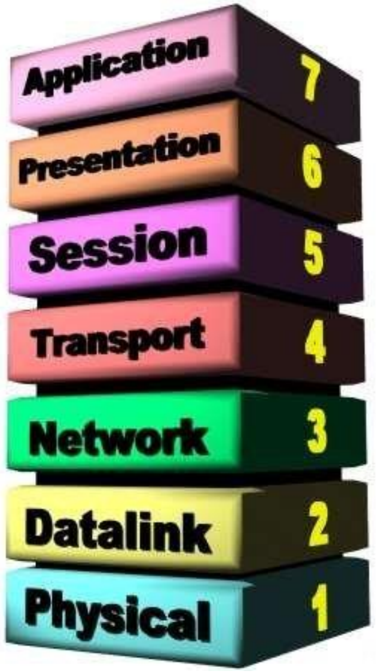
- a *header* (H) is added to the data unit, with exception an additional *trailer (T)* at layer 2 only.

At layer 1

- the data is changed into an electromagnetic signal and transported along a physical link.
- Upon reaching its destination, the signal is transformed back into digital form, then ascends through the layers.



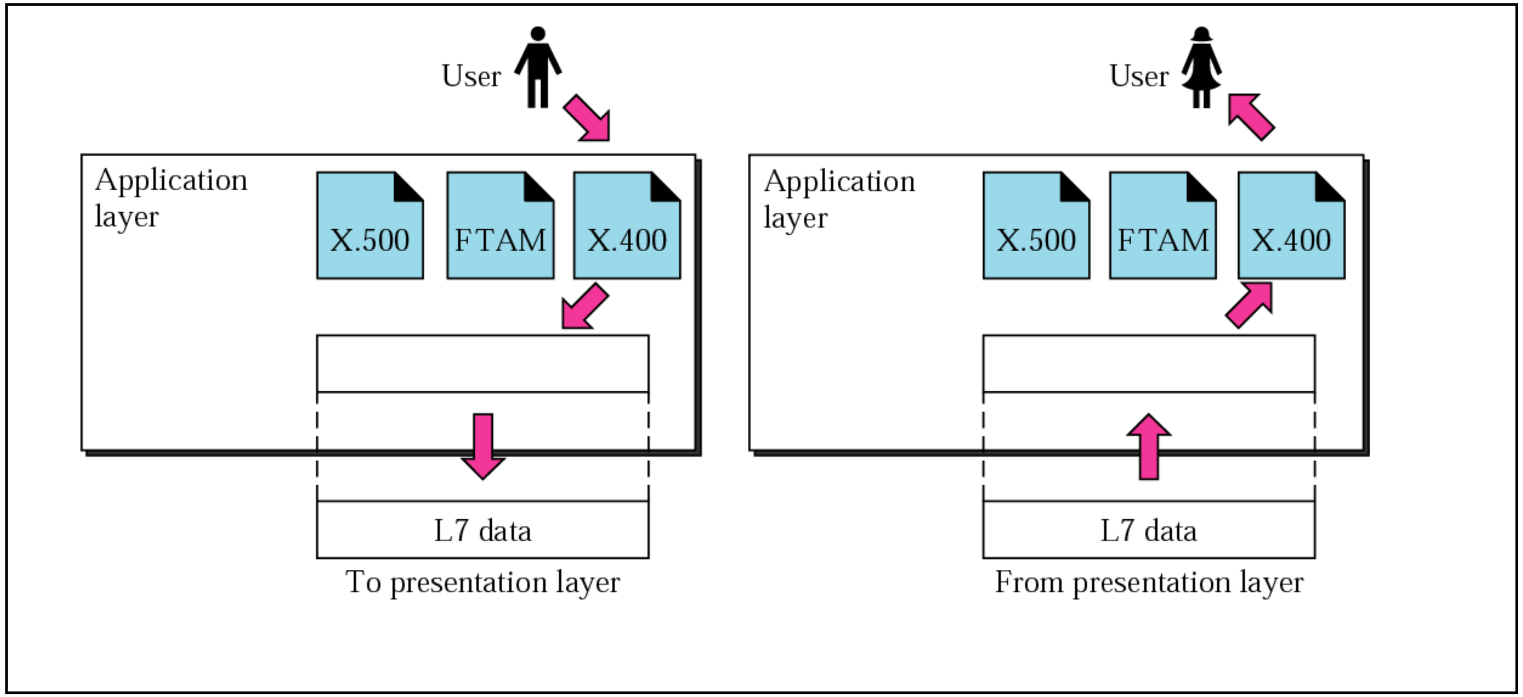
Layers in OSI Model



APPLICATION Layer (1/2)

- ❑ The application layer enables the user (human or software) to access the network.

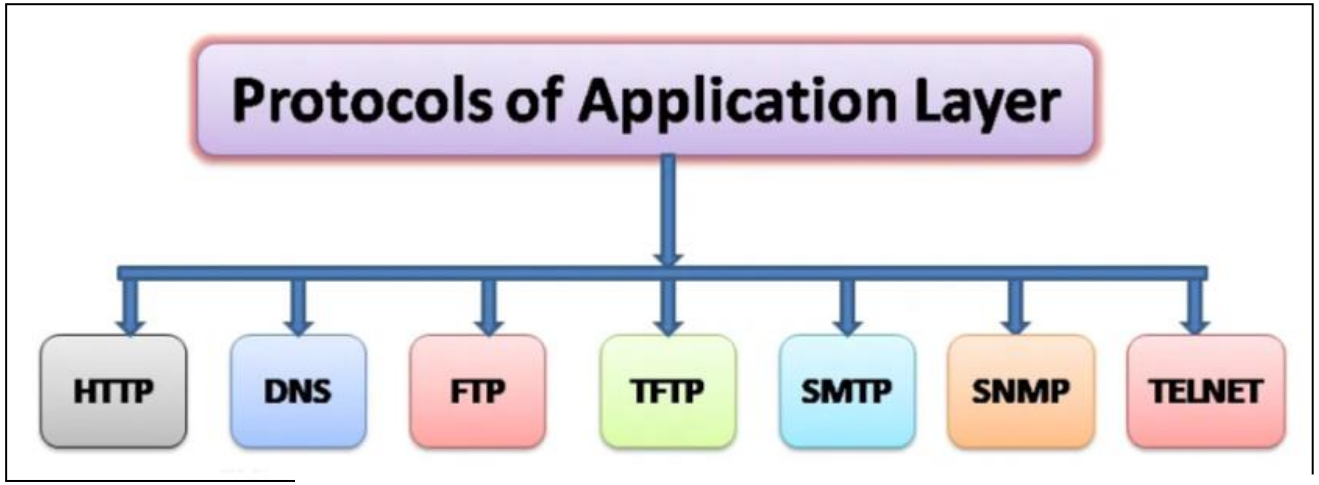
- ❑ It provides user interfaces and support for services such as e-mail, Network Virtual Terminal (NVT), remote file access and transfer (File Transfer, Access, and Management (FTAM)) , shared database management (services directory), and other types of distributed information services.



CALVIN INSTITUTE OF TECHNOLOGY

APPLICATION Layer (2/2)

- ❑ More of application layer protocols:

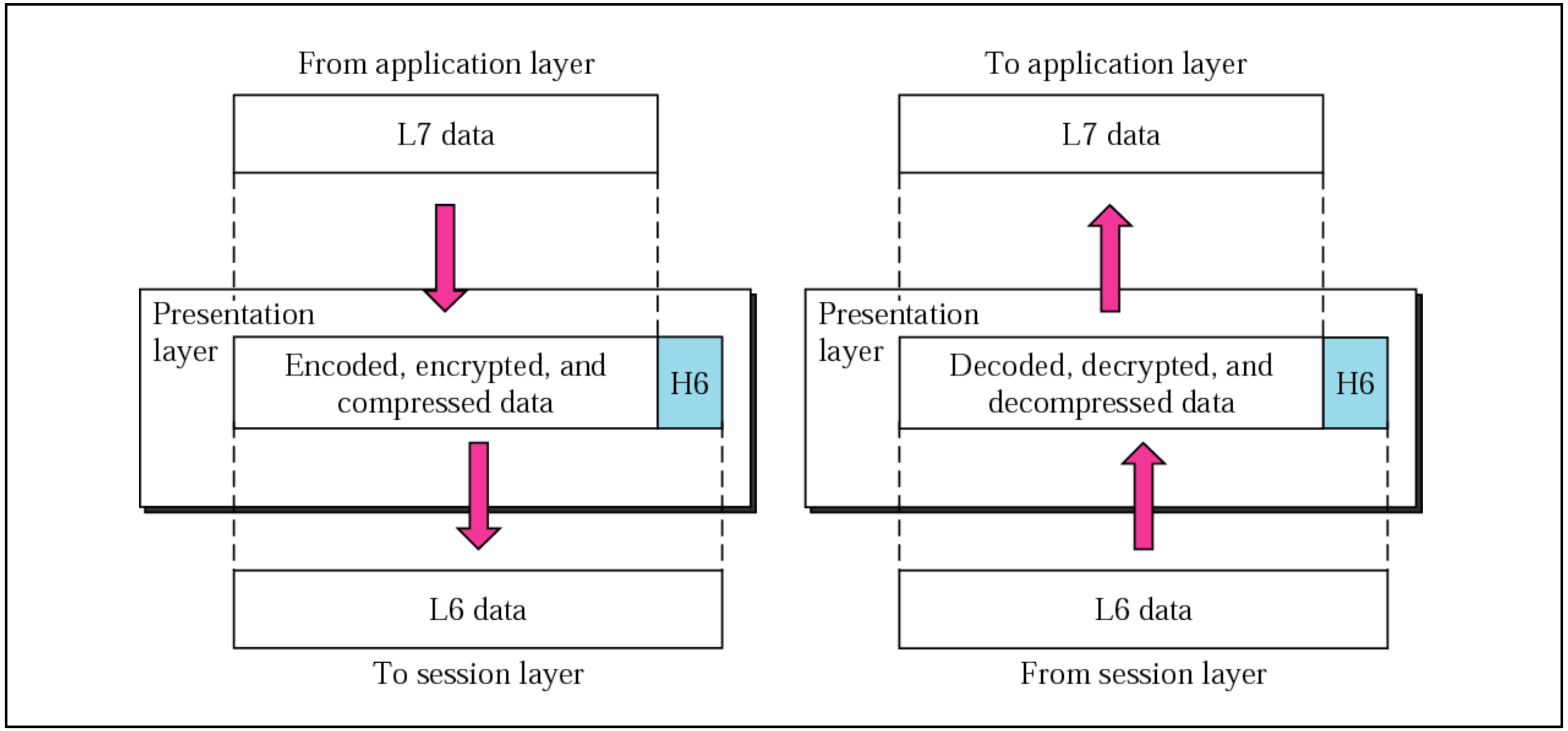


Please check, fyi. .
https://rohan-katkar19.medium.com/applications-layer-protocols-4820df9bd52f

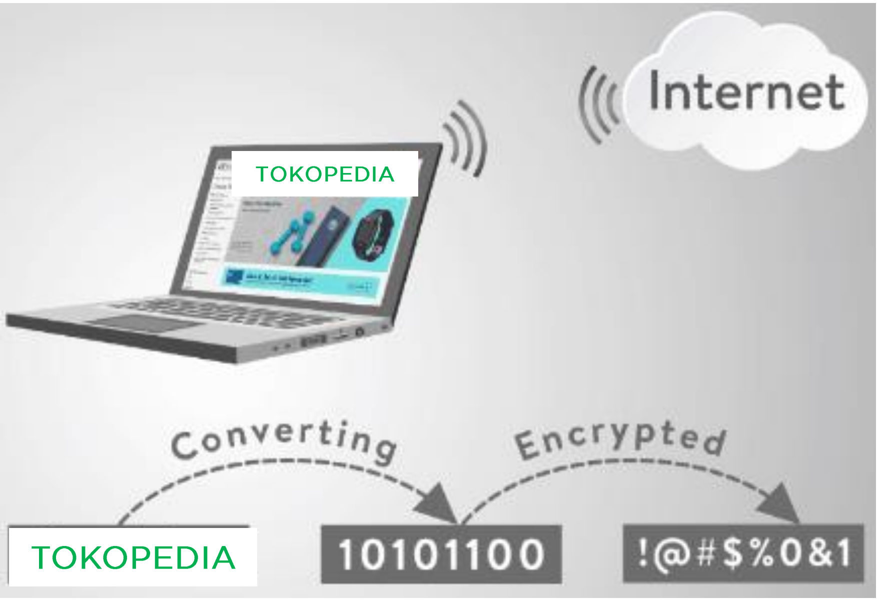
PRESENTATION Layer (1/2)

- ❑ The presentation layer is concerned with the syntax and semantics of the information exchanged between two systems.

- ❑ Specific responsibilities of the presentation layer include translation (Encoding methods), encryption (security), and data compression.



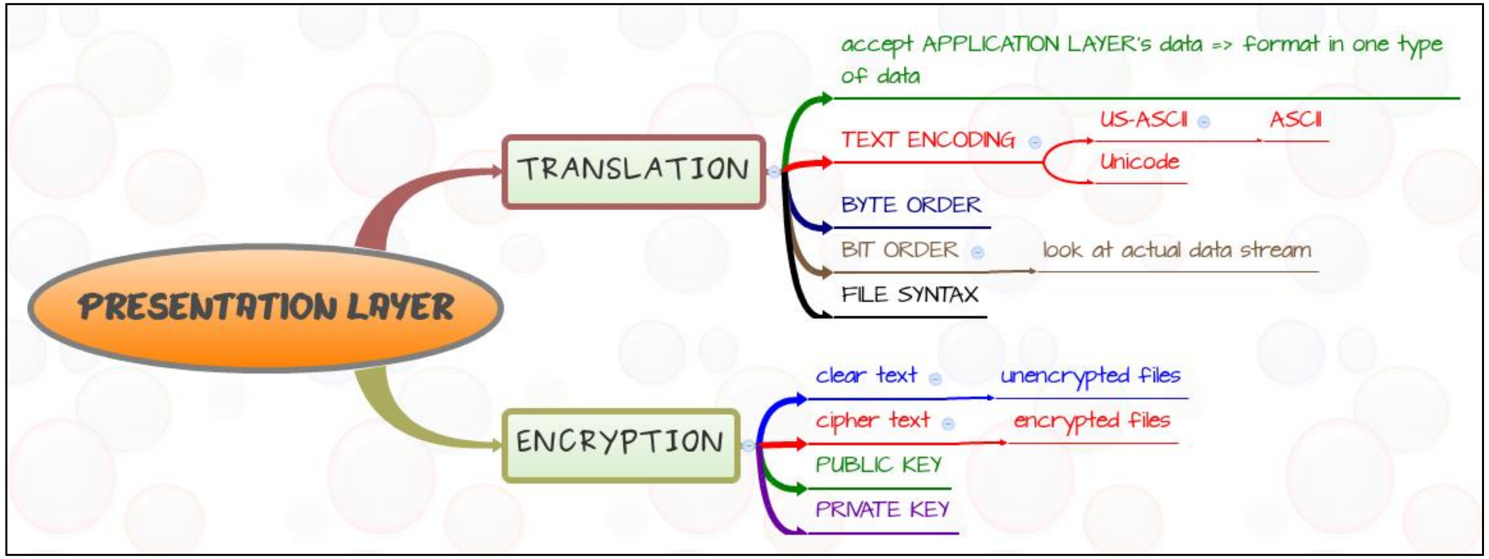
PRESENTATION Layer (2/2)



- Let's just say we are ordering something from *tokope*ia*. These transactions are handled in a secure transmission, i.e. the data passing between the "store" (the Website Application) will transmit encrypted data to the Presentation Layer that will need to be decrypted and processed.

- This Presentation layer handles translating the data from that top ("Application") layer to the below ("Session") layer.

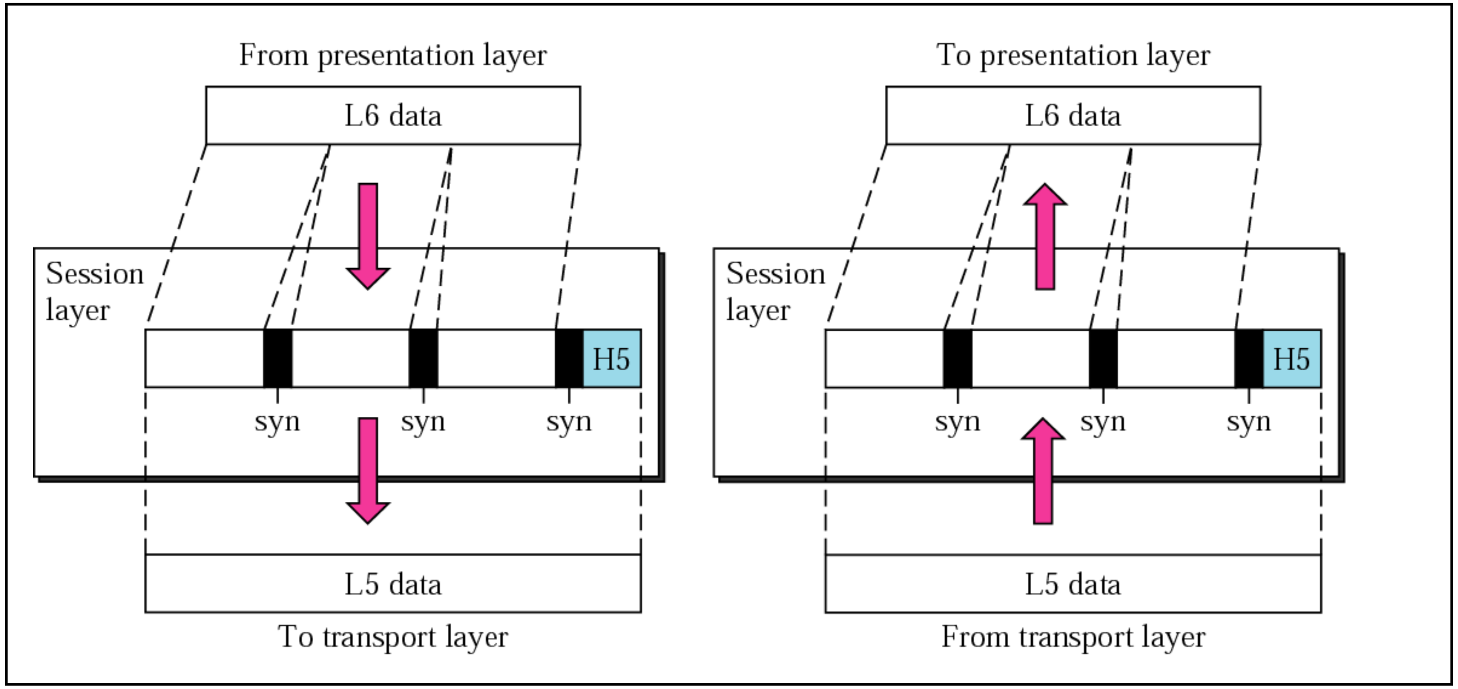
- And the other way around.

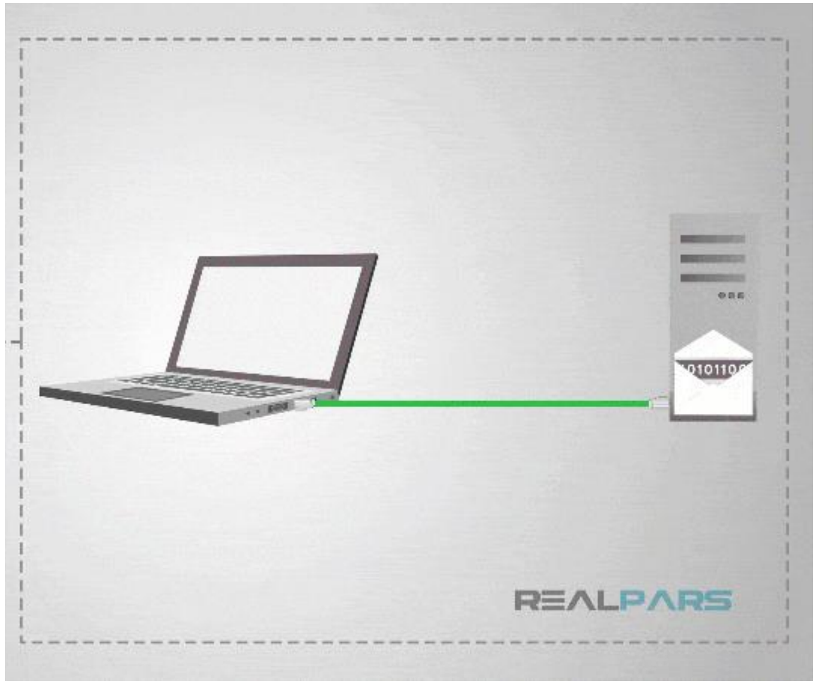


SESSION Layer (1/2)

- ❑ The session layer is the network dialog controller which establishes, maintains, and synchronizes the interaction between communicating systems.

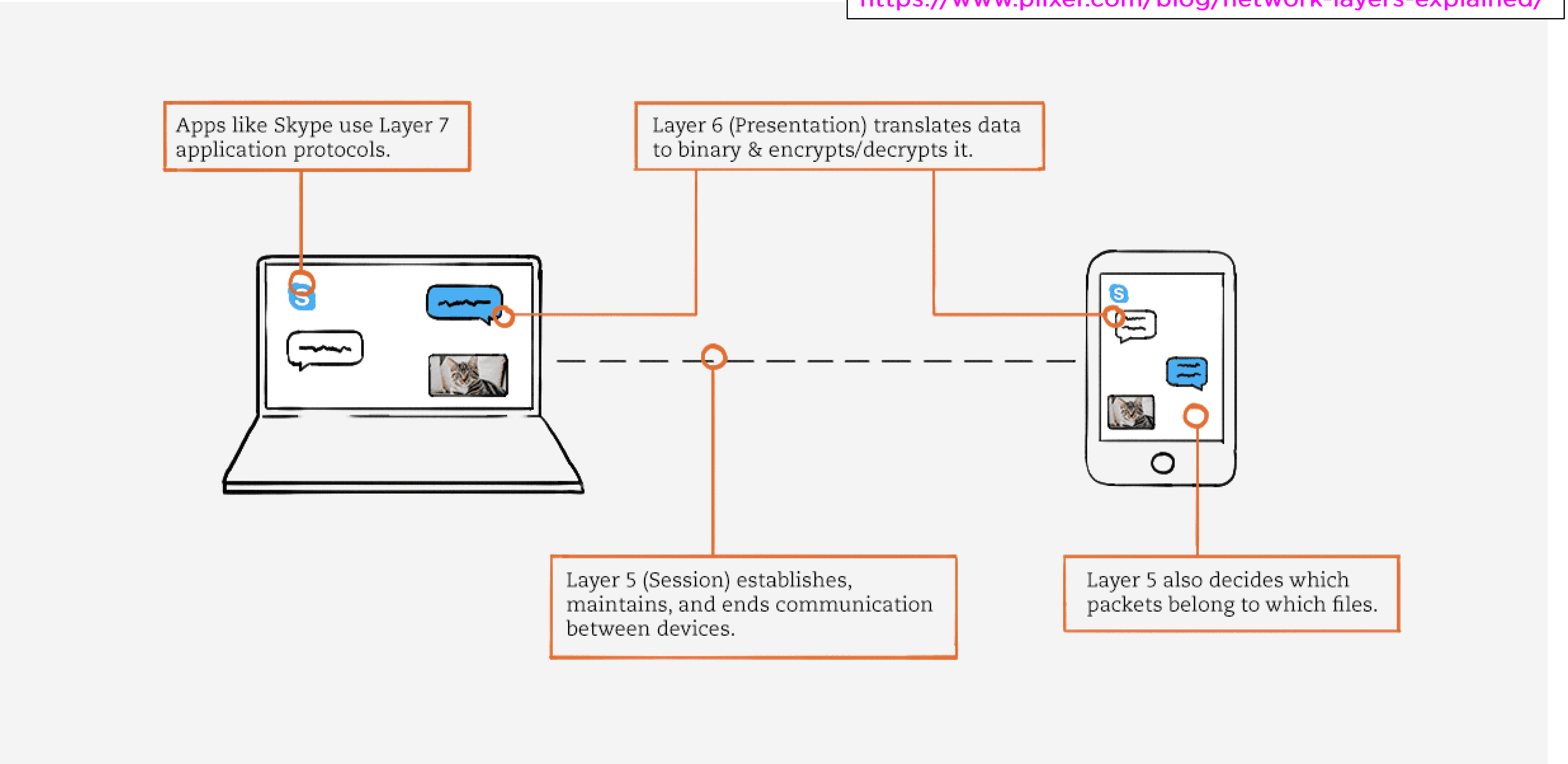
- ❑ Specific responsibilities of the session layer include the following: dialog control (half/full duplex) and synchronization (checkpoints of data)
  - Example: if a system is sending a 2.000 pages file, it is advisable to insert checkpoints after every 100 pages





SESSION Layer (2/2)

https://www.plixer.com/blog/network-layers-explained/



Apps like Skype use Layer 7 application protocols.

Layer 6 (Presentation) translates data to binary & encrypts/decrypts it.

Layer 5 (Session) establishes, maintains, and ends communication between devices.

Layer 5 also decides which packets belong to which files.

TRANSPORT Layer (1/4)

❑ The Transport Layer is responsible for process-to-process delivery of the entire message.

- A process is an application program running on the host

❑ Responsibilities:

Port/Service-point Addressing

- In transport layer, source-to-destination delivery means delivering *processes* (running program) between one computer to another computer.
- The transport layer header must therefore include a type of address called a *service-point address* (or *port address*).
- In other words, Transport Layer gets the entire message to the correct process on that computer; while Network Layer gets each packet to the correct computer.

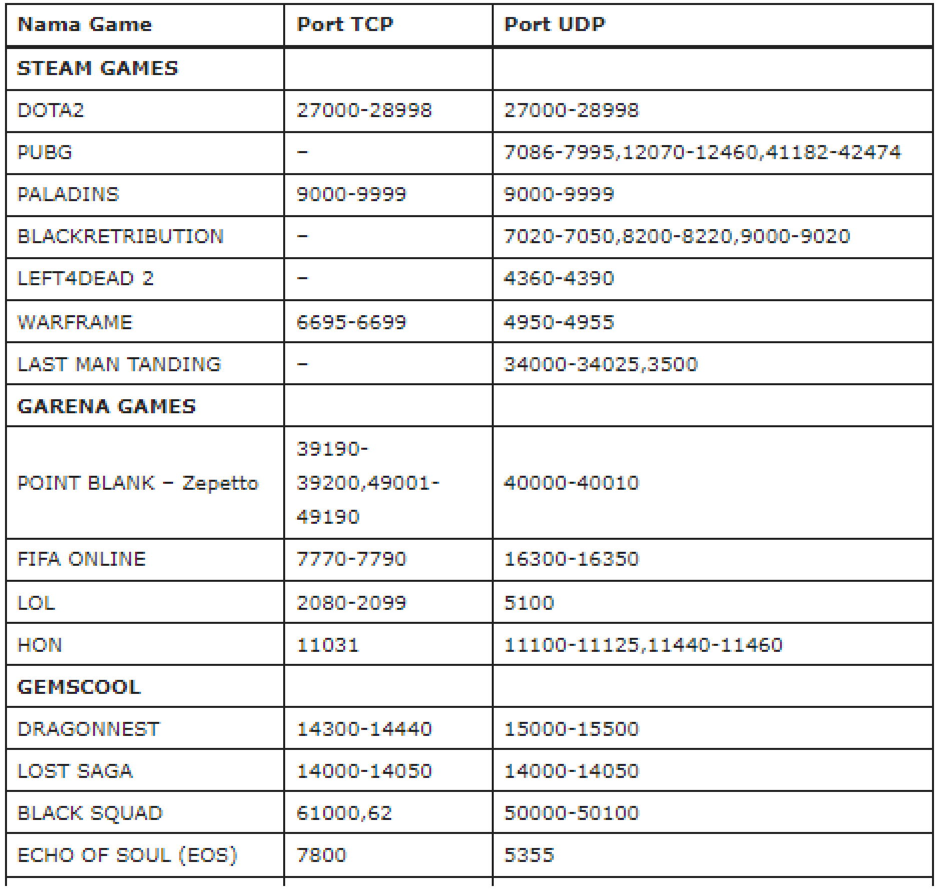
IMPORTANT!

- 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.

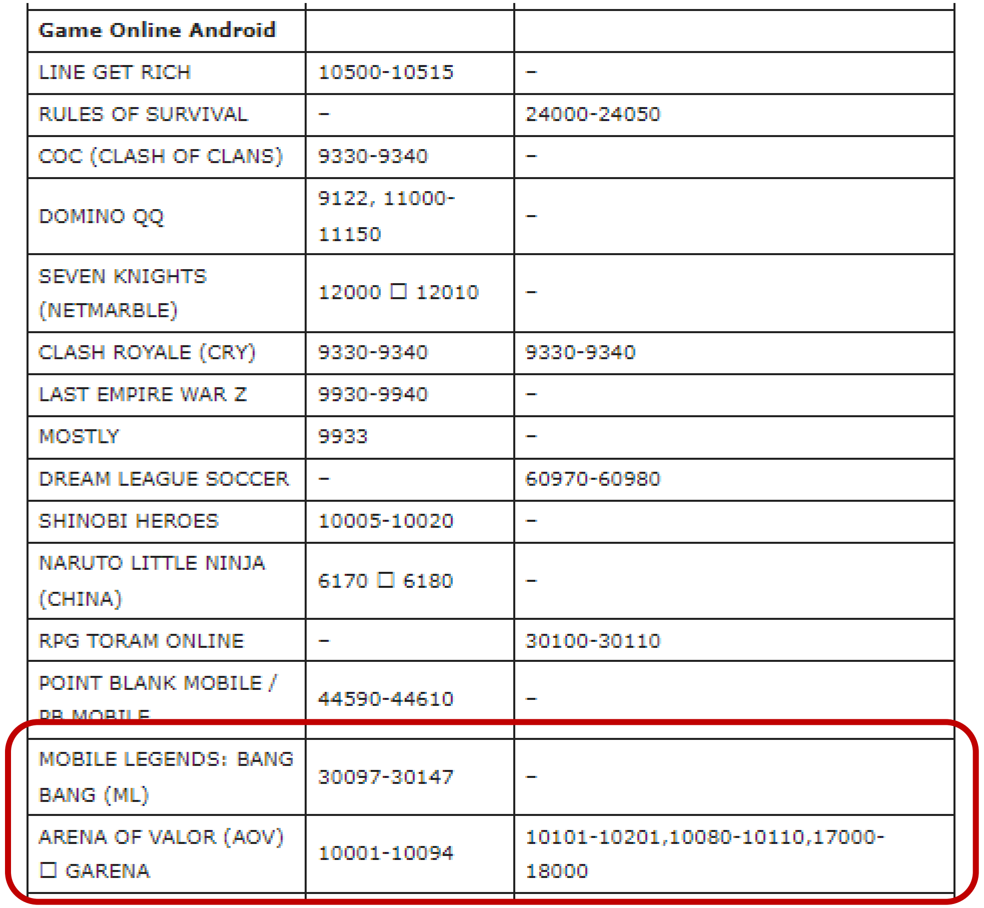
- NETWORK Layer, the other hand, also oversees source-to-destination delivery of individual packets, BUT it does not recognize any relationship between those packets.

upss..

“..Daftar port game online dibawah ini bisa digunakan untuk membuat rule agar koneksi ke game online tersebut bisa lancar, bisa juga untuk memisahkan antara koneksi download, browsing, streaming dan game online. Selain itu kita juga bisa membatasi atau memblokir game online dari mikrotik dengan menggunakan daftar port dibawah ini dengan menggunakan Firewall Filter Rules atau Firewall Raw..”

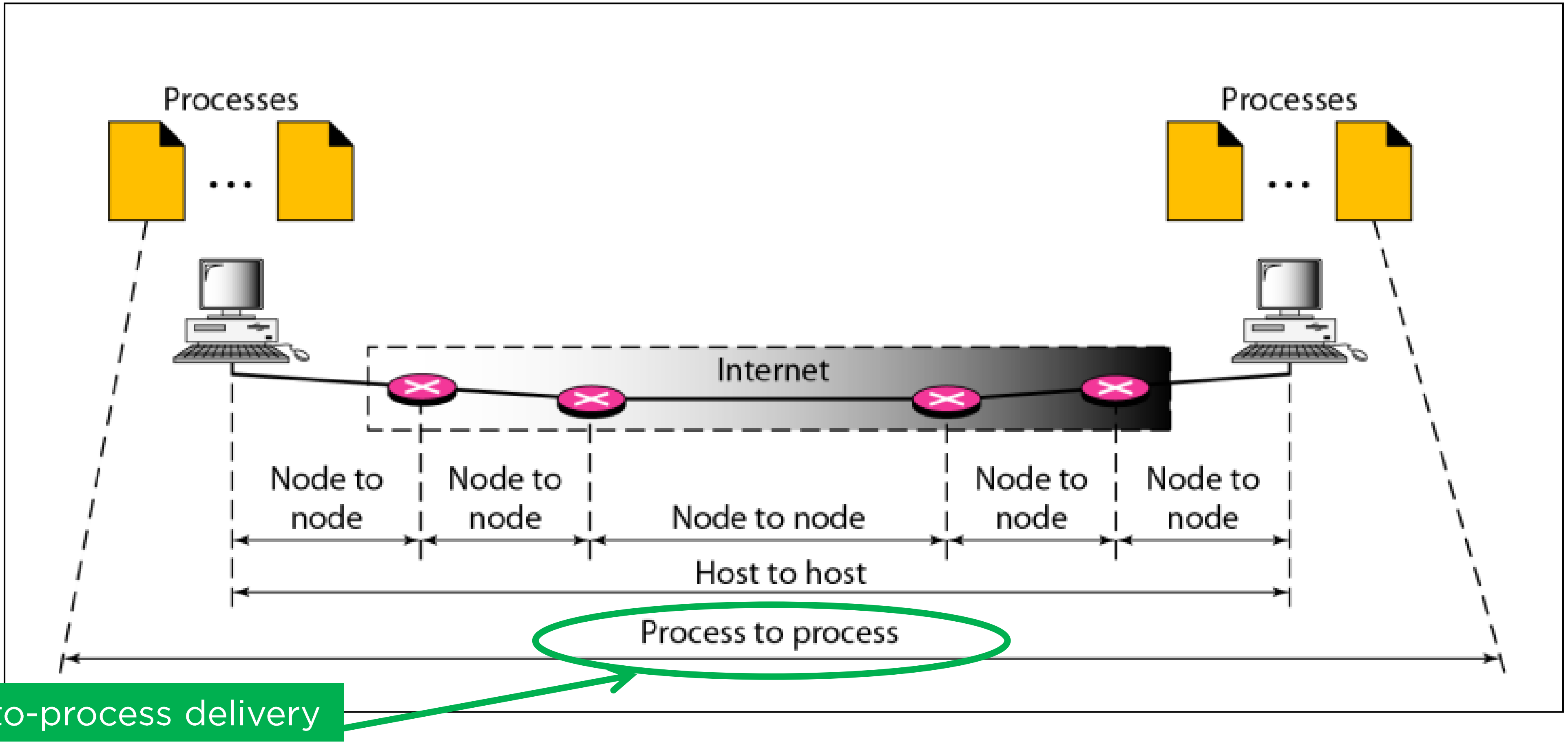


Nama Game	Port TCP	Port UDP
STEAM GAMES		
DOTA2	27000-28998	27000-28998
PUBG	–	7086-7995,12070-12460,41182-42474
PALADINS	9000-9999	9000-9999
BLACKRETRIBUTION	–	7020-7050,8200-8220,9000-9020
LEFT4DEAD 2	–	4360-4390
WARFRAME	6695-6699	4950-4955
LAST MAN TANDING	–	34000-34025,3500
GARENA GAMES		
POINT BLANK – Zepetto	39190-39200,49001-49190	40000-40010
FIFA ONLINE	7770-7790	16300-16350
LOL	2080-2099	5100
HON	11031	11100-11125,11440-11460
GEMSCOOL		
DRAGONNEST	14300-14440	15000-15500
LOST SAGA	14000-14050	14000-14050
BLACK SQUAD	61000,62	50000-50100
ECHO OF SOUL (EOS)	7800	5355



Game Online Android		
LINE GET RICH	10500-10515	–
RULES OF SURVIVAL	–	24000-24050
COC (CLASH OF CLANS)	9330-9340	–
DOMINO QQ	9122, 11000-11150	–
SEVEN KNIGHTS (NETMARBLE)	12000 □ 12010	–
CLASH ROYALE (CRY)	9330-9340	9330-9340
LAST EMPIRE WAR Z	9930-9940	–
MOSTLY	9933	–
DREAM LEAGUE SOCCER	–	60970-60980
SHINOBI HEROES	10005-10020	–
NARUTO LITTLE NINJA (CHINA)	6170 □ 6180	–
RPG TORAM ONLINE	–	30100-30110
POINT BLANK MOBILE / PB MOBILE	44590-44610	–
MOBILE LEGENDS: BANG BANG (ML)	30097-30147	–
ARENA OF VALOR (AOV) □ GARENA	10001-10094	10101-10201,10080-10110,17000-18000

TRANSPORT Layer (2/4)

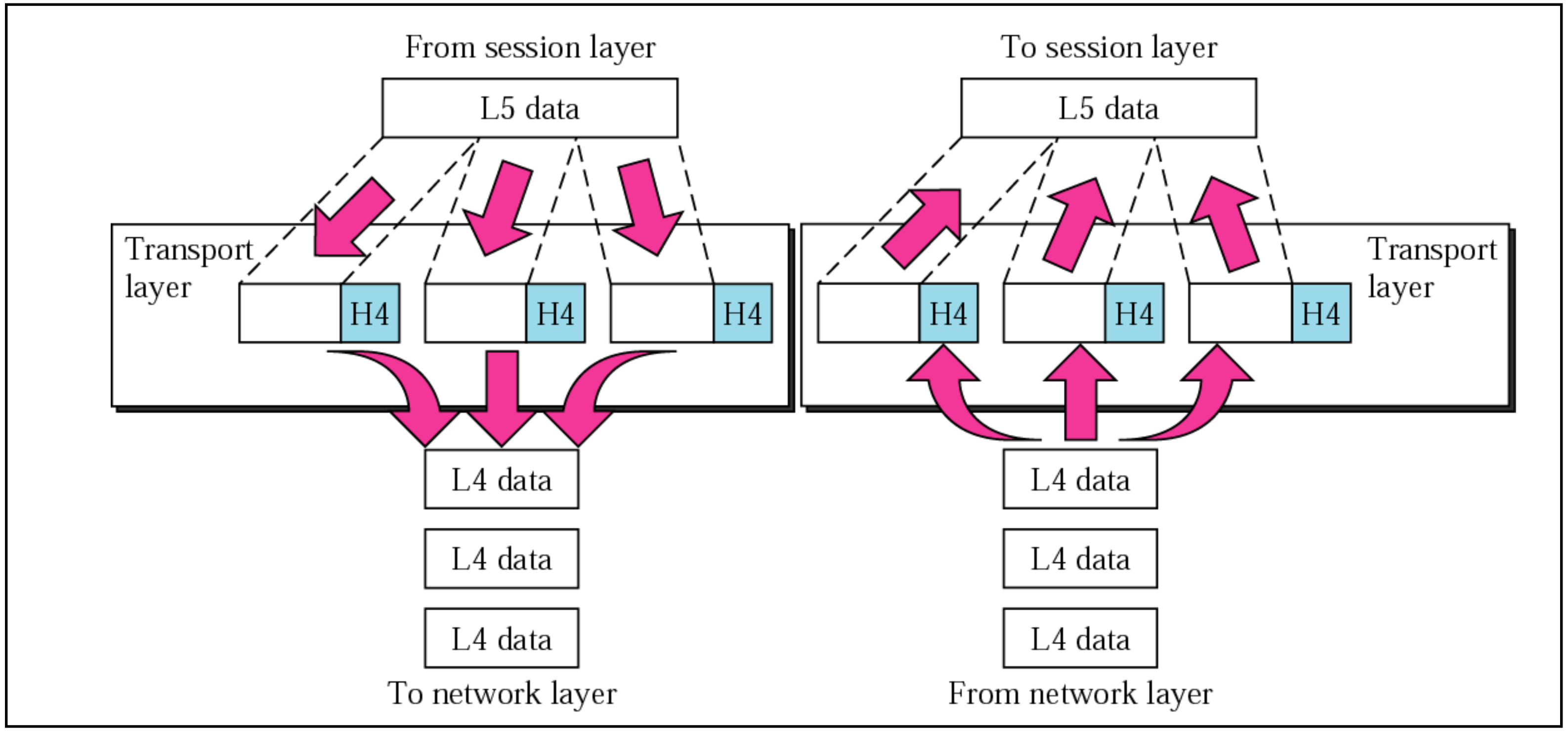


Process-to-process delivery

TRANSPORT Layer (3/4)

Segmentation and Reassembly

- A message is divided into transmittable segments (=*segmentation*=), with each segment containing a *sequence number*. These numbers enable the transport layer to re-assemble the message correctly (=*reassembly*=) upon arriving at the destination and to identify and replace packets that were lost in transmission.



TRANSPORT Layer (4/4)

Flow control & Error control

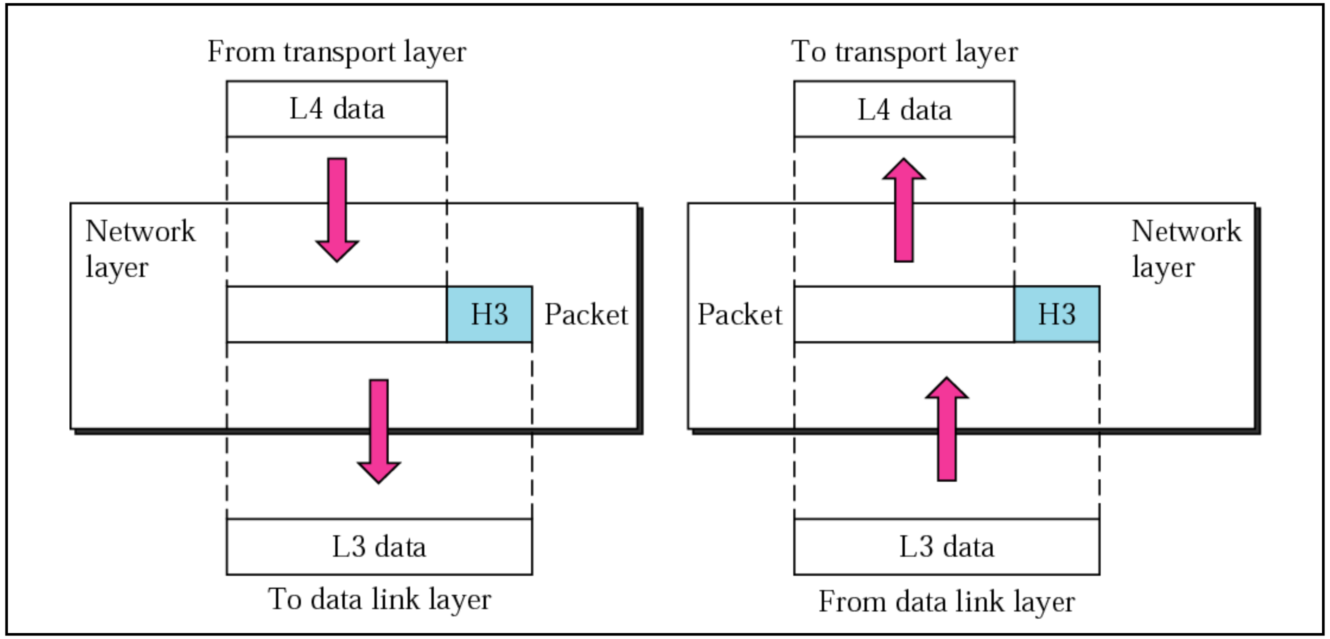
- In this layer, flow control is performed *end-to-end* (unlike Datalink Layer), while error control is performed *process-to-process* (unlike Datalink Layer)
- The sender makes sure the entire message arrives at the receiver without error (damage, loss, or duplication).
- For error correction, it is usually achieved through retransmission.

Connection control

- The transport layer can be either connection-less or connection-oriented.
- A connection-less Transport Layer treats each segment as an ***independent*** packet, while a connection-oriented transport layer makes a connection ***dependent*** between segments.

NETWORK Layer (1/3)

- ❑ The Network Layer is responsible for the source-to-destination delivery of a packet, possibly across multiple networks (links). In other words, this layer ensures that each packet gets from its point of origin to its final destination.
  - If two systems are attached to different networks (links) with connecting devices, there is often a need for the Network layer.
  - On the other hand, if two systems are connected to the same network (link), there is usually no need for a Network Layer (➔ this is for Datalink Layer).



NETWORK Layer (2/3)

- Other responsibilities:

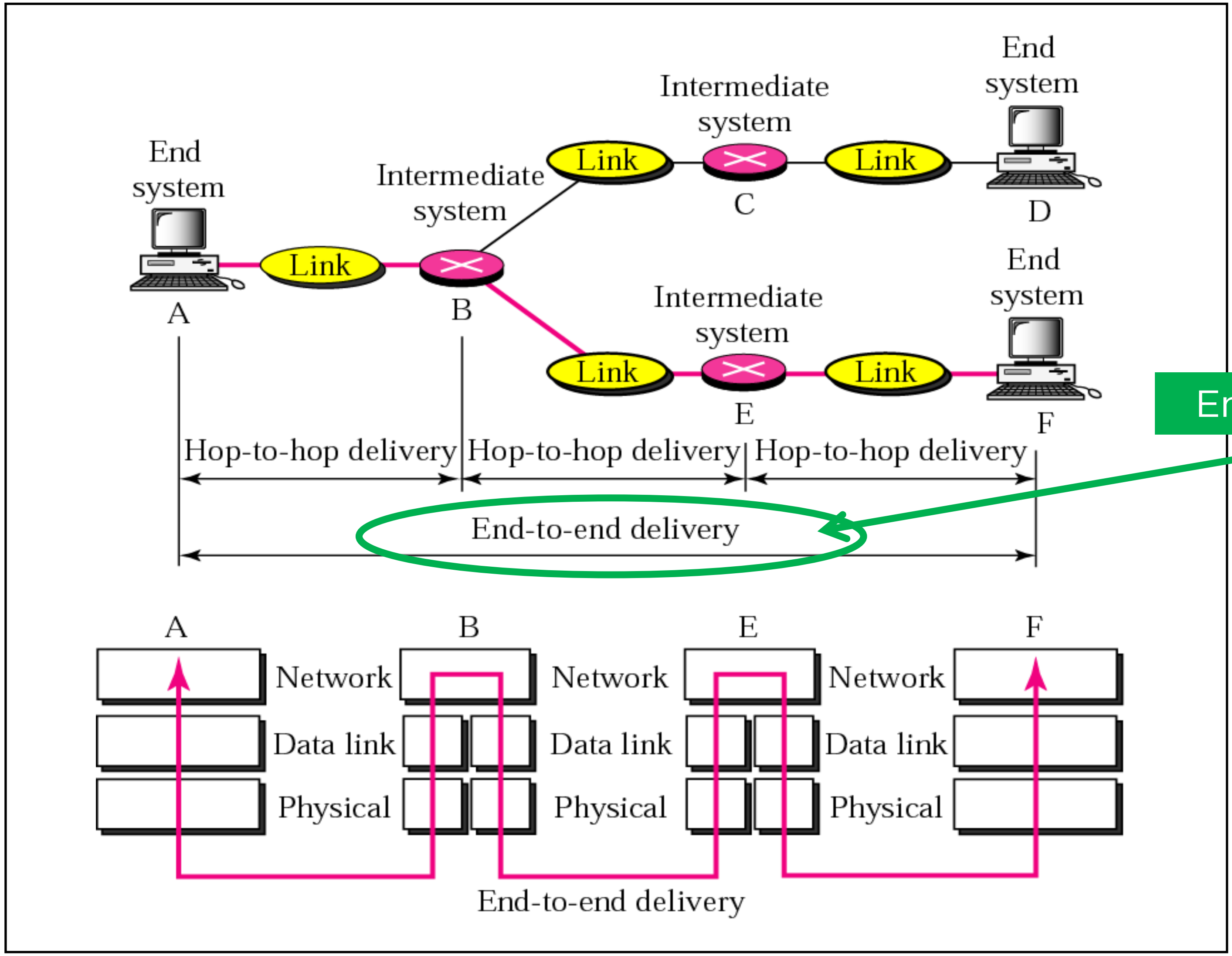
Logical addressing

- The network layer adds a header to the upcoming packet which includes the logical addresses of the sender and receiver.
- Logical vs. Physical addressing
    - The physical address is implemented by the **Data Link Layer** handles the addressing problem locally/defined by its LAN/WAN (intra-network).
    - If a packet passes the network boundary, we need another addressing system to help distinguish the source and destination systems, it is called logical address, or also referred as IP (Internet Protocol) address.

Routing

- When independent networks or links are connected together creating a large network, the connecting devices (such routers/switches) will route/switch the packets to their final destination. This is the important function of **Network Layer**.

NETWORK Layer (3/3)



End-to-end delivery

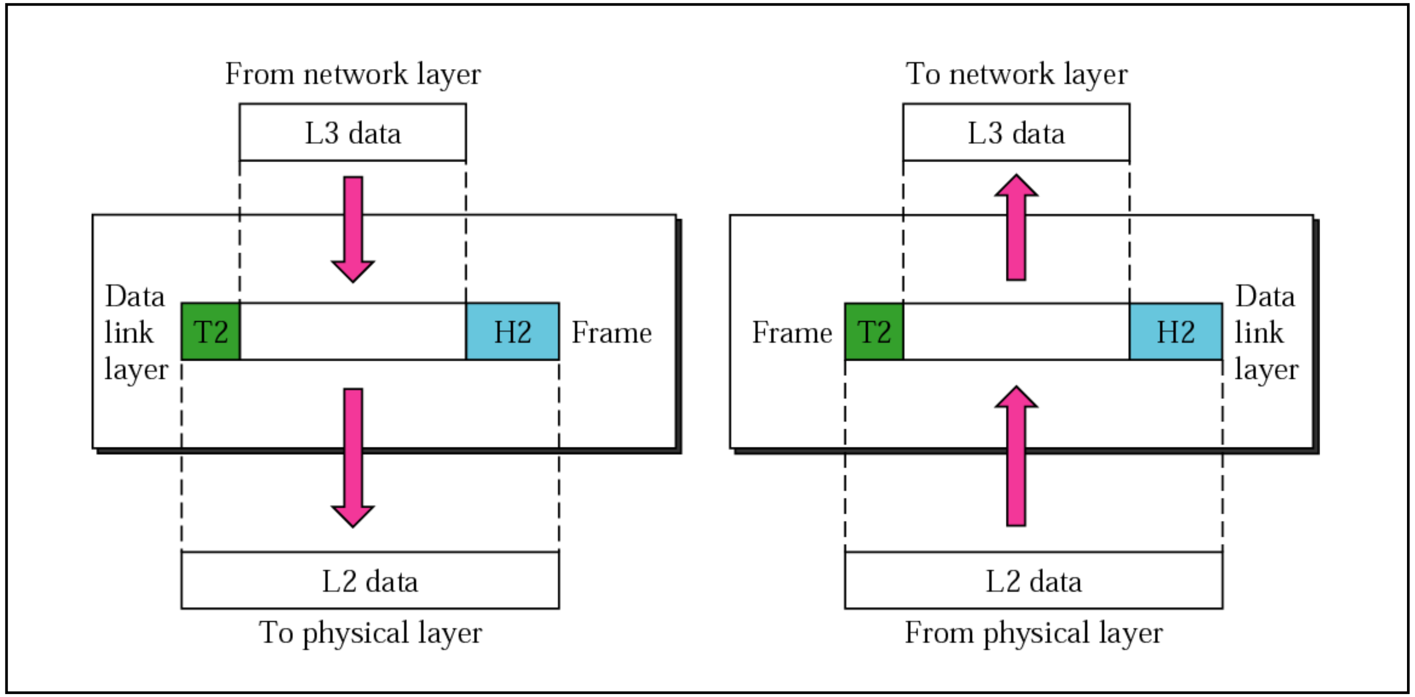
DATA LINK Layer (1/3)

- The data link layer transforms the physical layer (a raw transmission facility) to a reliable link.
  - It makes the physical layer “appear” error-free to the upper layer (Network layer).

- Responsibilities:

Framing

To divide the stream of *bits* from the network layer into *frames.*



DATA LINK Layer (2/3)

❑ Other responsibilities:

Physical addressing

To add a header contains the sender/receiver address (within or outside its network).

Flow control.

- A mechanism in the receiver to prevent overwhelming rate from the sender.
- Ex. STOP-AND-WAIT

Error control.

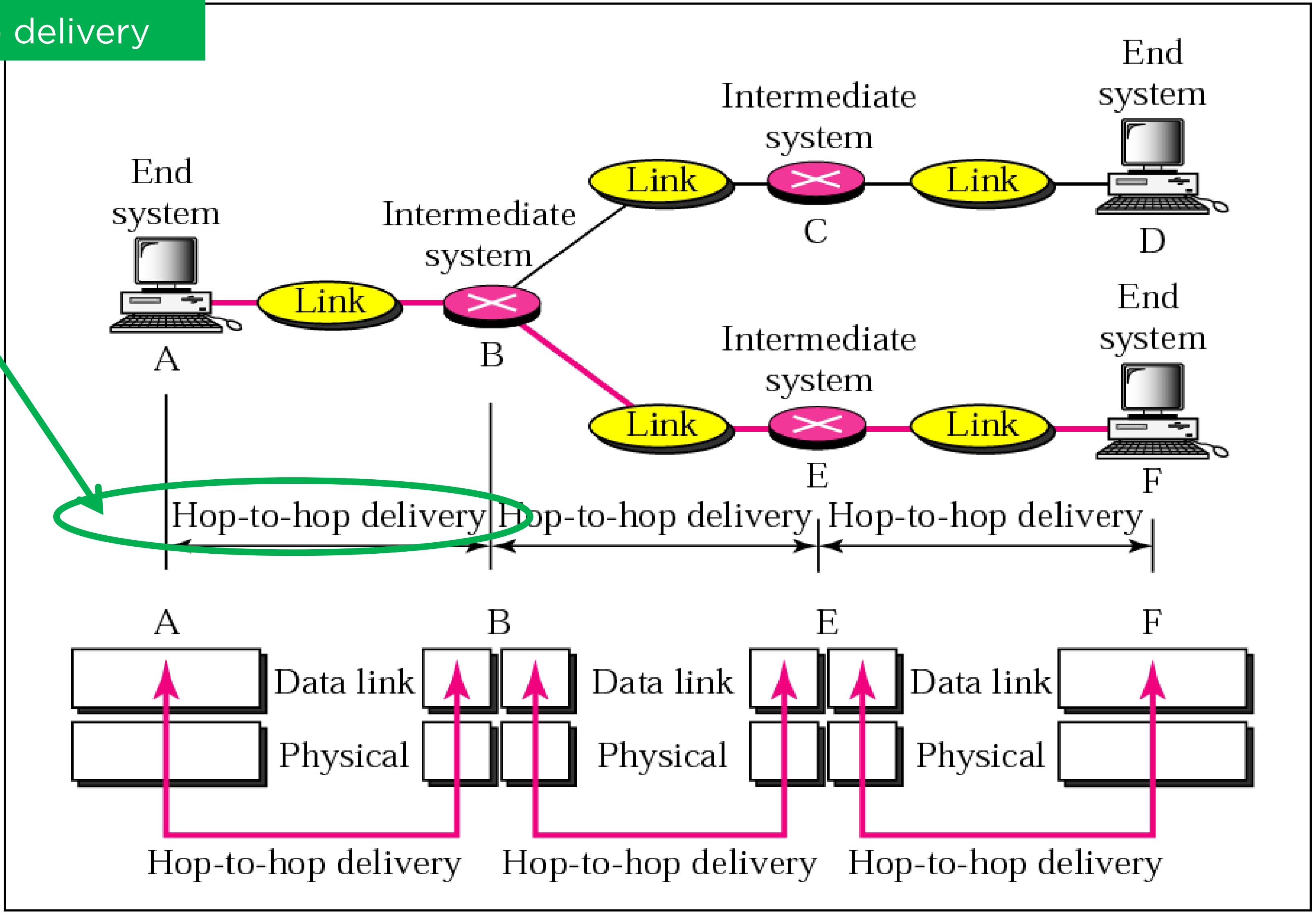
- Some mechanism: error detection, retransmission of data, …

Access control.

- When two or more devices are connected to the same link, data link layer protocols are necessary to determine which device has control over the link at any given time
- Ex. Random Access (ALOHA), Multiple Access (TDMA, FDMA, CDMA)

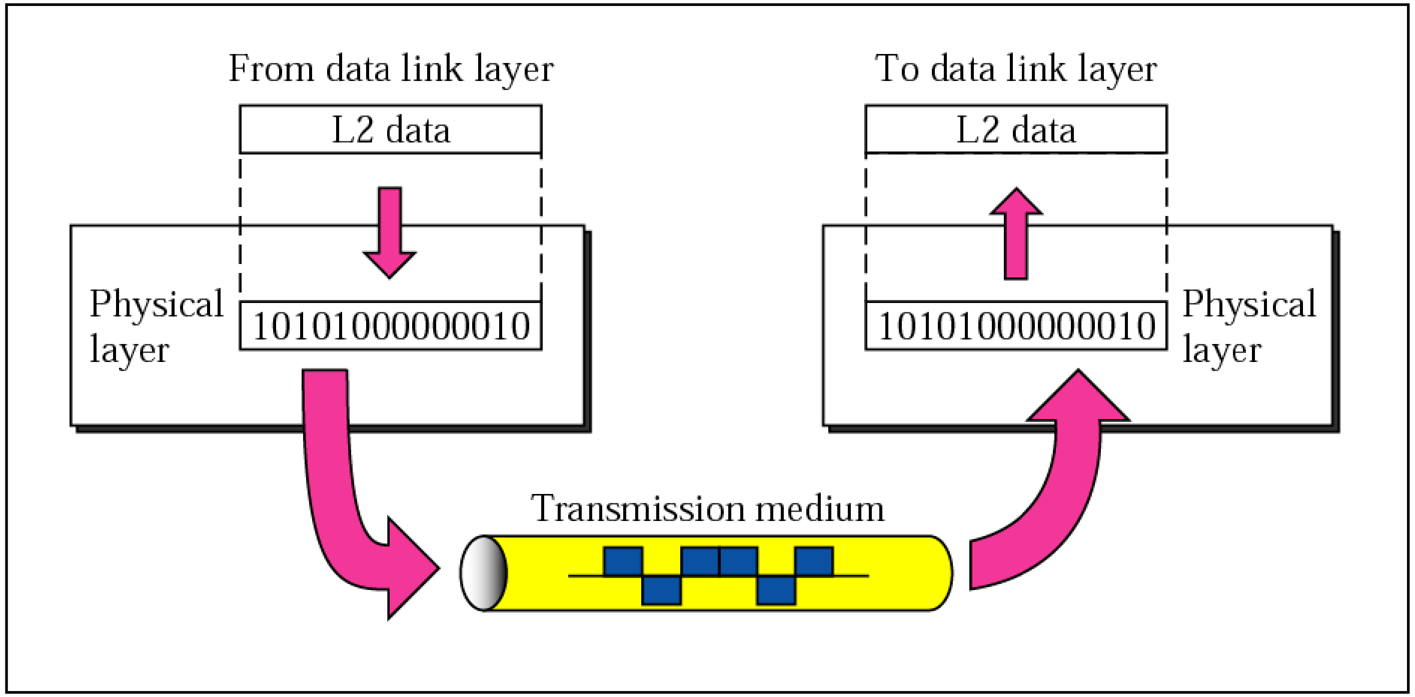
DATA LINK Layer (3/3)

Hop-to-hop/
Node-to-node delivery



PHYSICAL Layer (1/2)

- ❑ The physical layer coordinates the functions required to carry a bit stream over a physical medium.
- ❑ It deals with the mechanical and electrical specifications of the interface and transmission media.
- ❑ It also defines the procedures and functions that physical devices and interfaces have to perform for transmission to occur.



PHYSICAL Layer (2/2)

- ❑ The physical layer is also concerned with the following:

  Physical characteristics of interfaces and media.
  Ex. wired or wireless (LAN/PAN/...)

  Representation of bits.
  To be transmitted, bits must be encoded into signals -- electrical or optical.

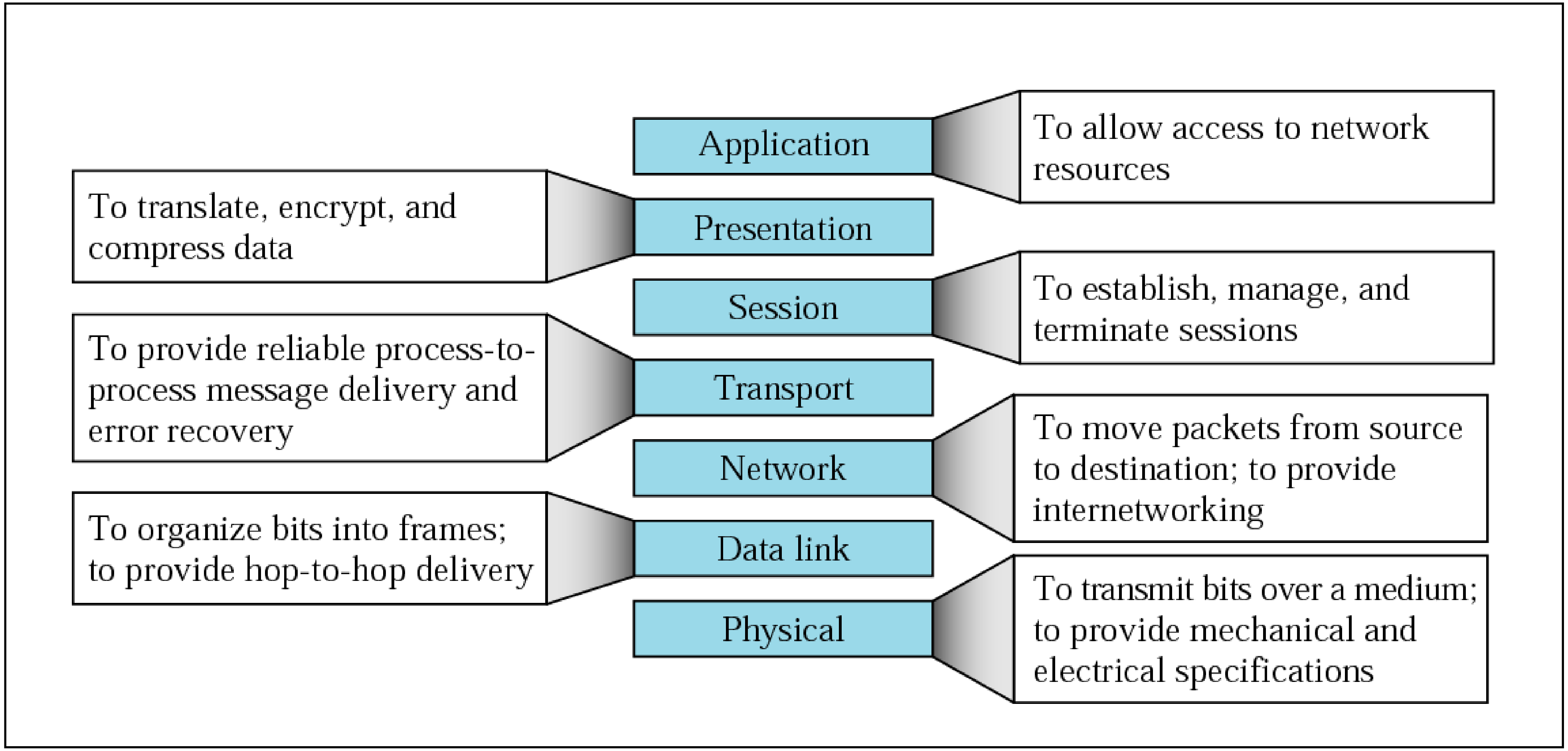
  Data rate.
  The transmission rate: how the bits are sent for some duration of time

  Synchronization of bits.
  Defines how the bit rate and clock rate are synchronized

  Line configuration/Physical topology
  Example: point-to-point/multipoint configuration; mesh/star/ring/bus/ . . .

  Transmission mode
  Defines the direction of transmission between two devices, like: simplex/half-duplex/full-duplex.

OSI Layer: Summary



Application: To allow access to network resources

Presentation: To translate, encrypt, and compress data

Session: To establish, manage, and terminate sessions

Transport: To provide reliable process-to-process message delivery and error recovery

Network: To move packets from source to destination; to provide internetworking

Data link: To organize bits into frames; to provide hop-to-hop delivery

Physical: To transmit bits over a medium; to provide mechanical and electrical specifications

to be continued

TCP/IP Protocol Suite