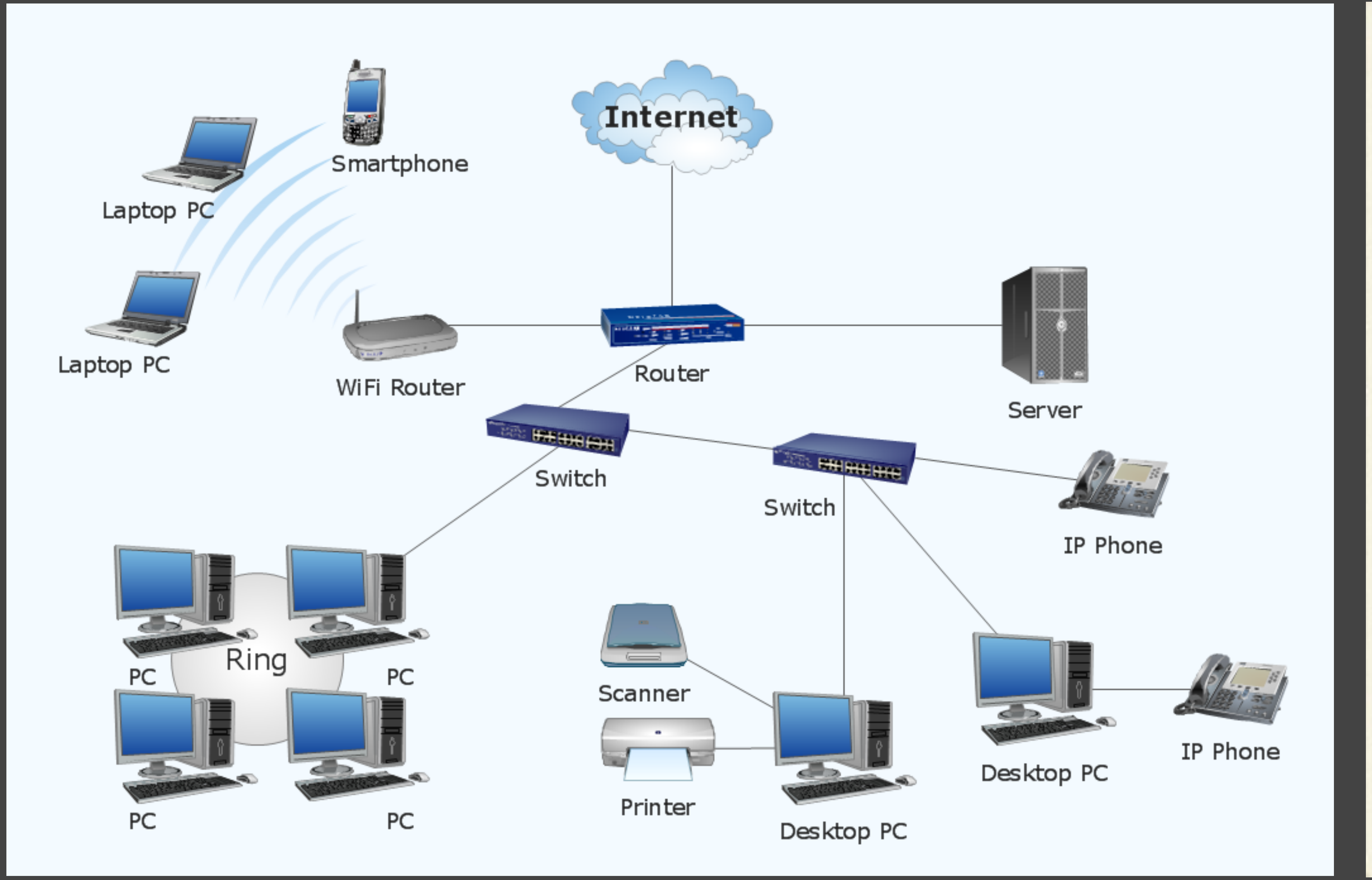
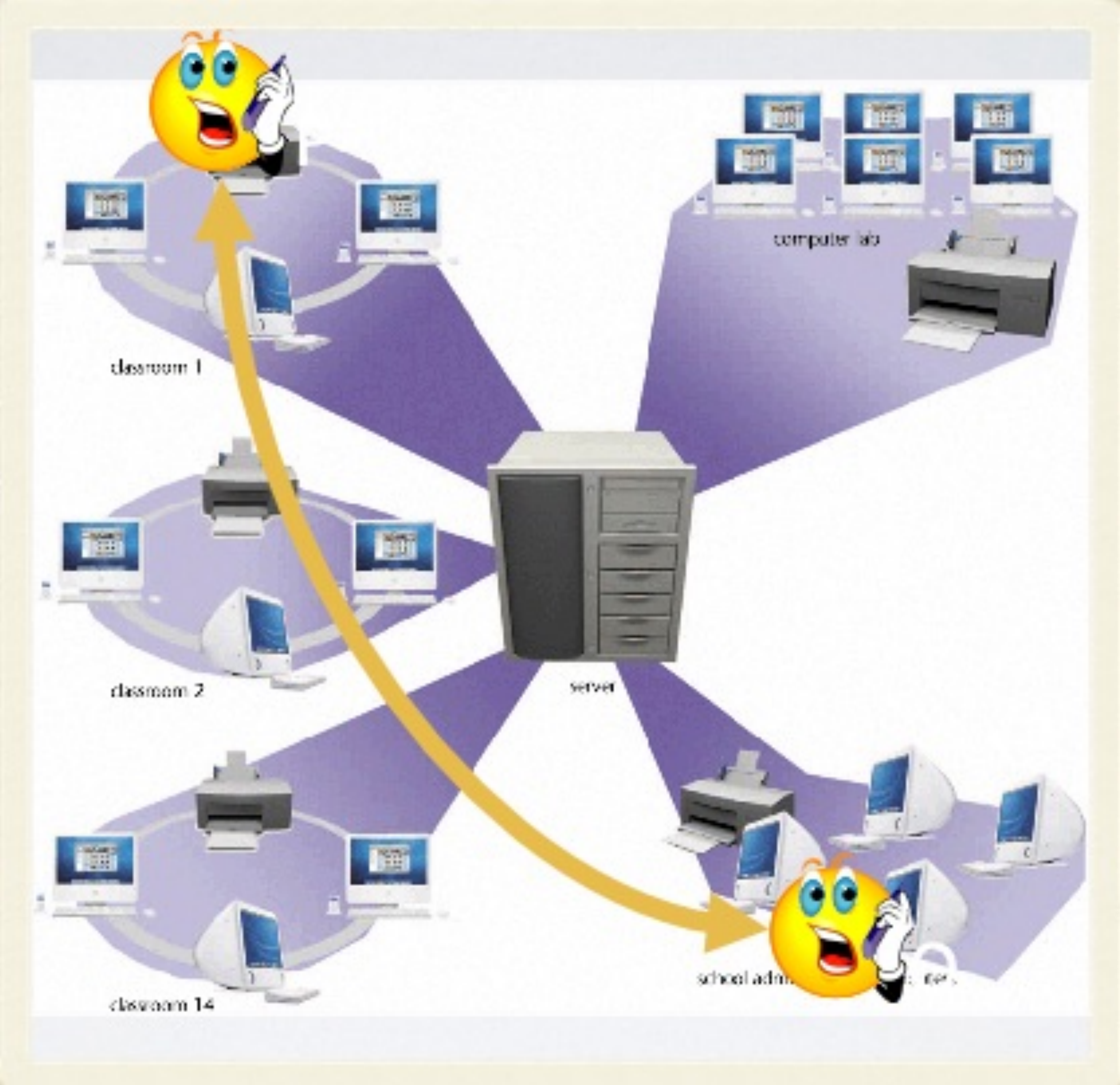
Computer Network Basic

Arthur Pai

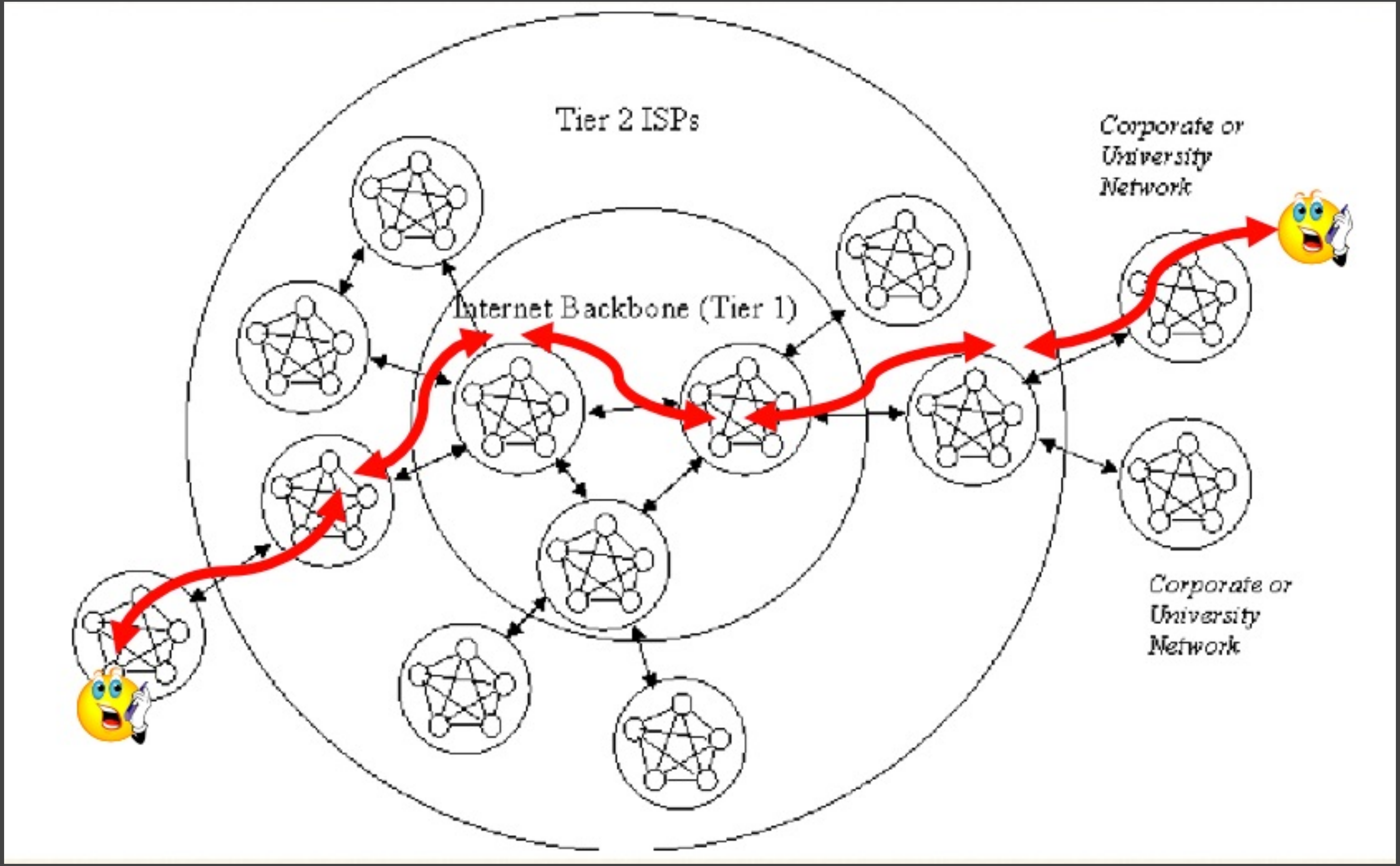
THE COMPUTER NETWORK IS...





- An interconnected collection of autonomous computers and devices
- Processes communicate with one another across the network
- Such communication is often transparent to end-users

THE COMPUTER NETWORK IS...


Tier 2 ISPs
Internet Backbone (Tier 1)
Corporate or University Network
Corporate or University Network

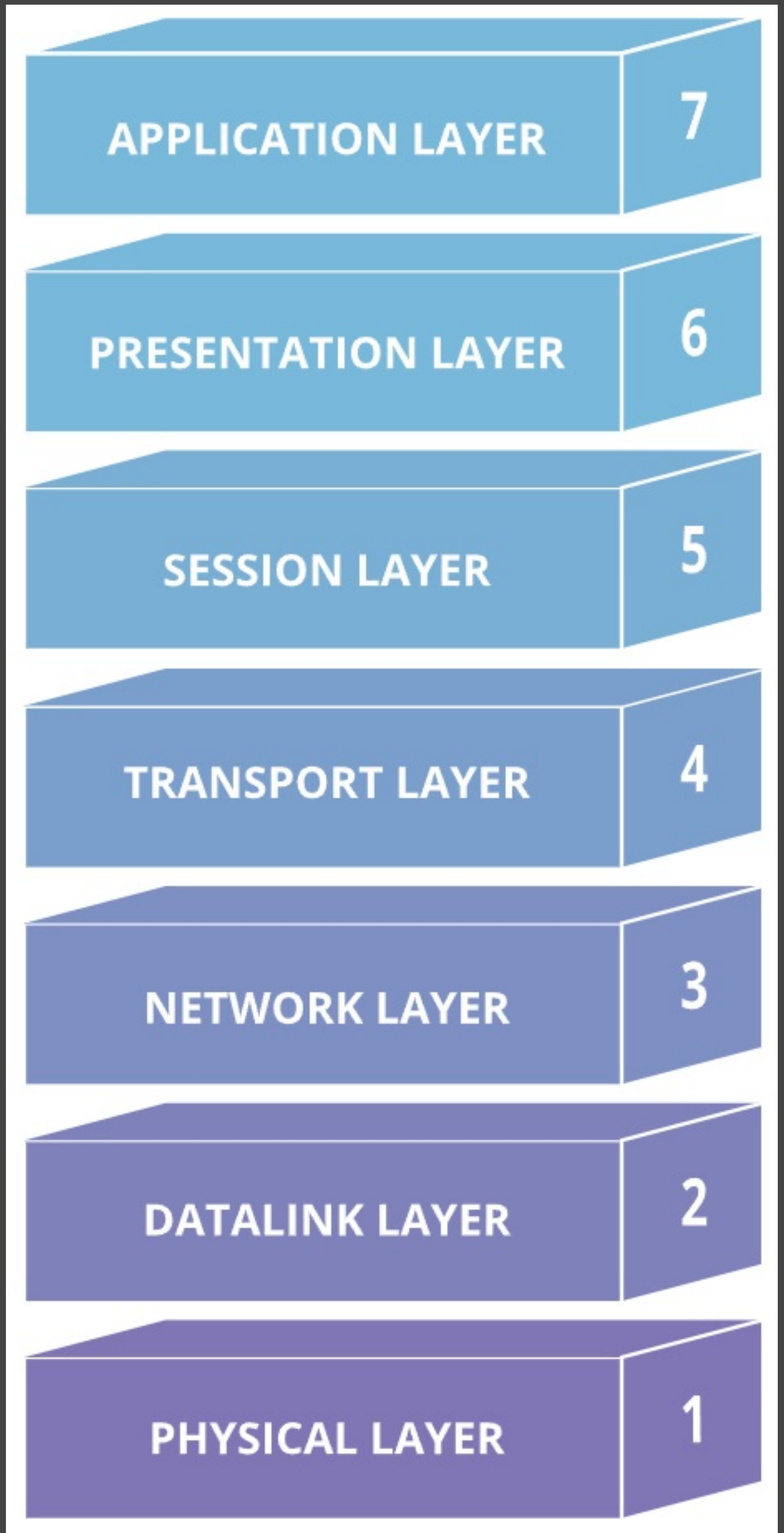
BASICS

The OSI Reference Model 網路七層

- Open System Interconnection

- The International Standards Organization (ISO) proposal for the standardization of the various protocols used in computer networks

- The Seven-Layer protocol stack



BASICS

7. 應用層: HTTP, HTTPS, FTP, Telnet, SSH, SMTP, POP3, Socket Programing

6. 表達層: 把數據轉換為能與接收者的系統格式相容並適合傳輸的格式

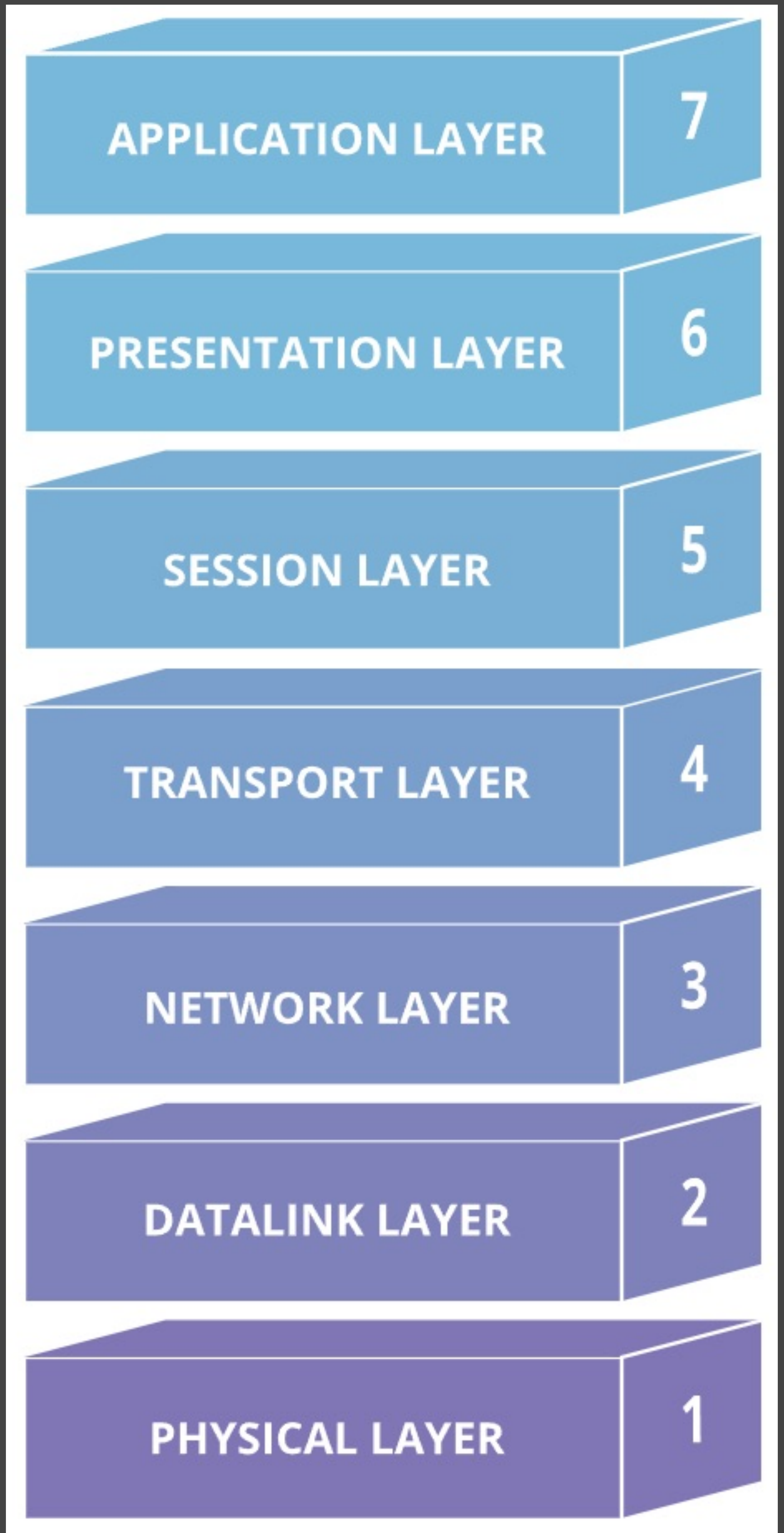
5. 會議層: 負責在數據傳輸中設定和維護電腦網路中兩台電腦之間的通訊連接

4. 傳輸層: 把傳輸表頭(TH)加至資料以形成封包。傳輸表頭包含了所使用的協定等傳送資訊。***TCP, UDP***

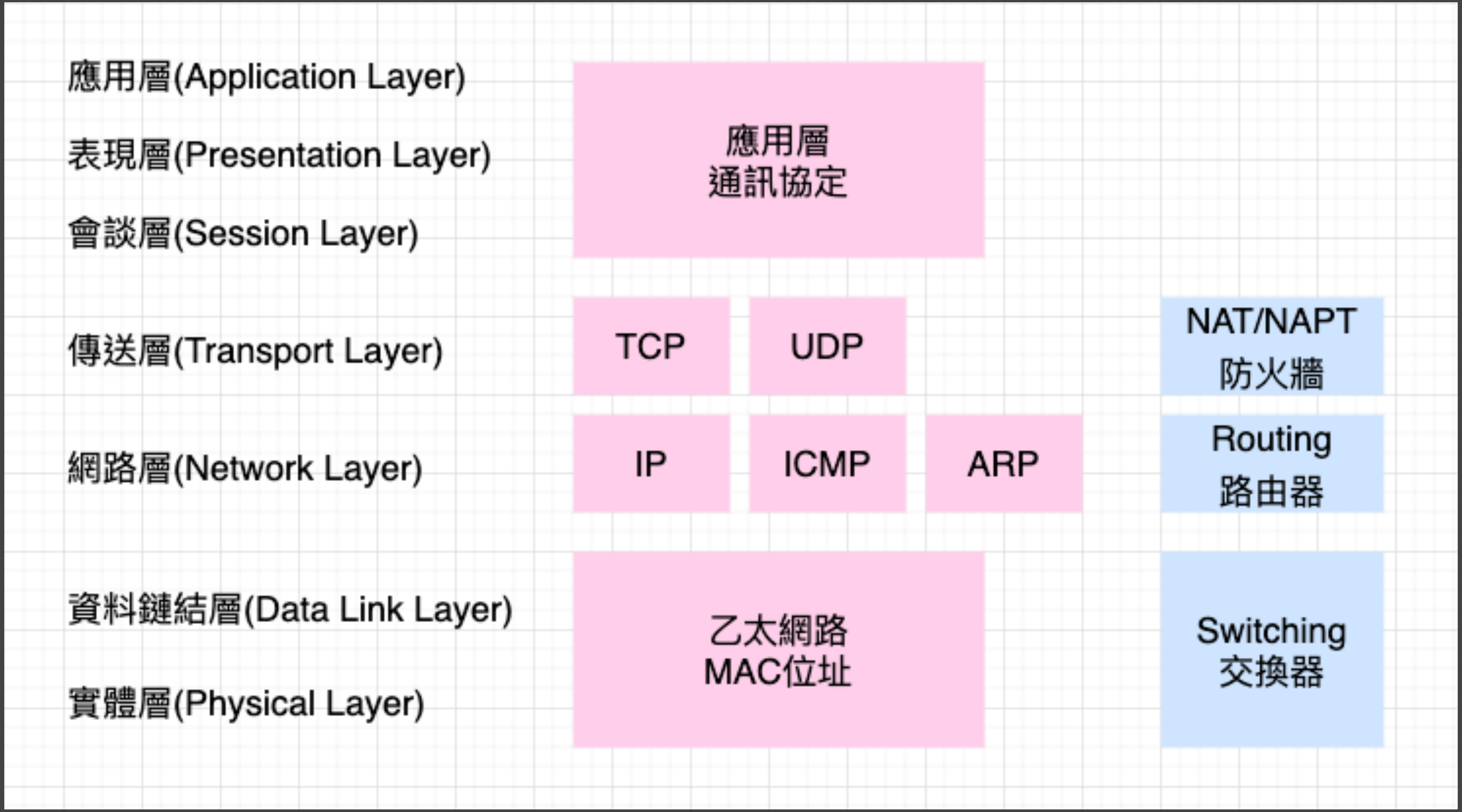
3. 網路層: 決定數據的路徑選擇和轉寄，將網路表頭(NH)加至數據包，以形成封包。網路表頭包含了網路資料。**網際網路協定*(IP)***

2. 資料連結層: 負責網路尋址、錯誤偵測和改錯。**乙太網路, *MAC*位址**

1. 實體層: 用來定義網路裝置之間的位元資料傳輸，也就是在電線或其他物理線材上，傳遞0與1電子訊號



BASICS



應用層(Application Layer)

表現層(Presentation Layer)

會談層(Session Layer)

傳送層(Transport Layer)

網路層(Network Layer)

資料鏈結層(Data Link Layer)

實體層(Physical Layer)

應用層
通訊協定

TCP

UDP

NAT/NAPT
防火牆

IP

ICMP

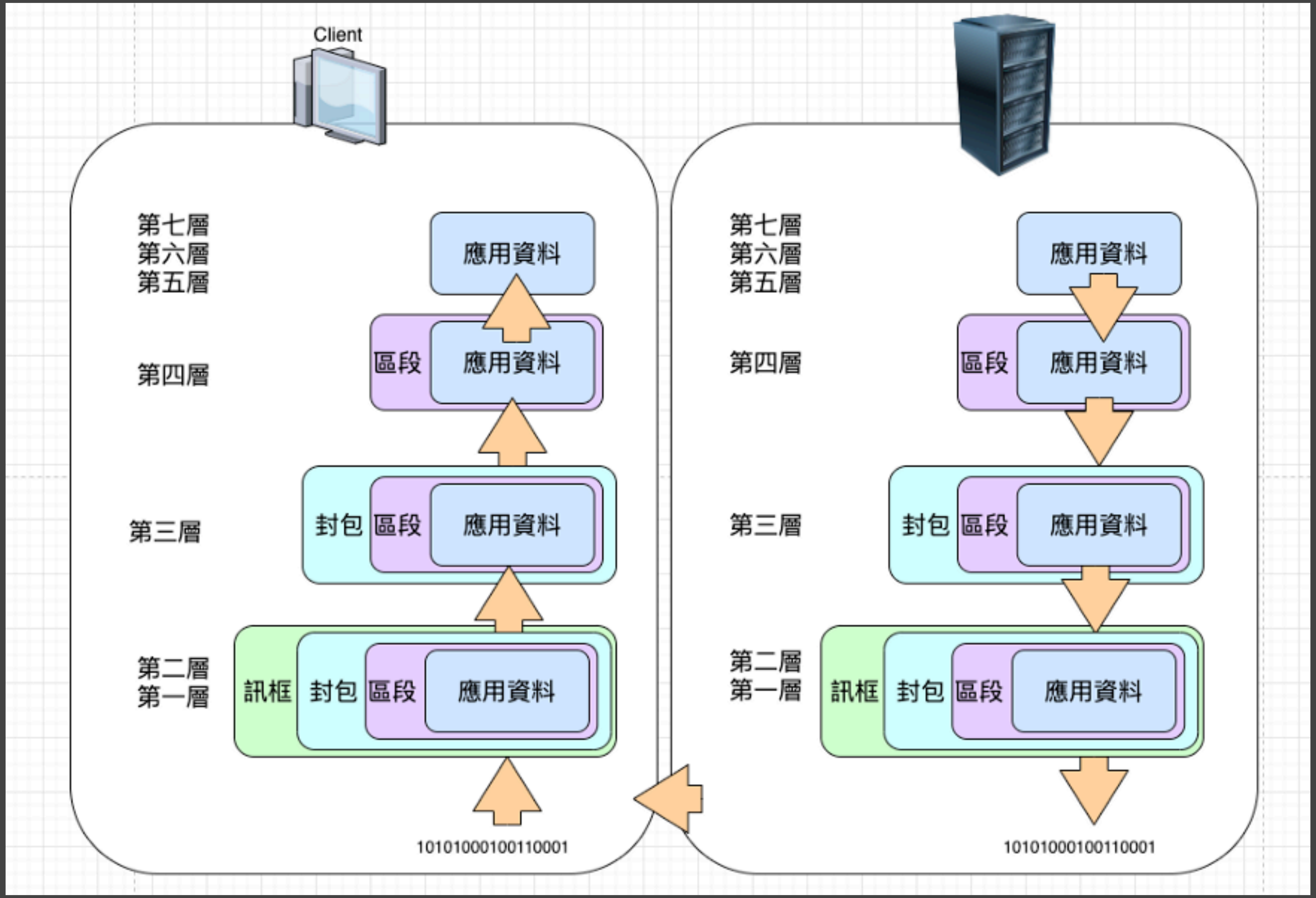
ARP

Routing
路由器

乙太網路
MAC位址

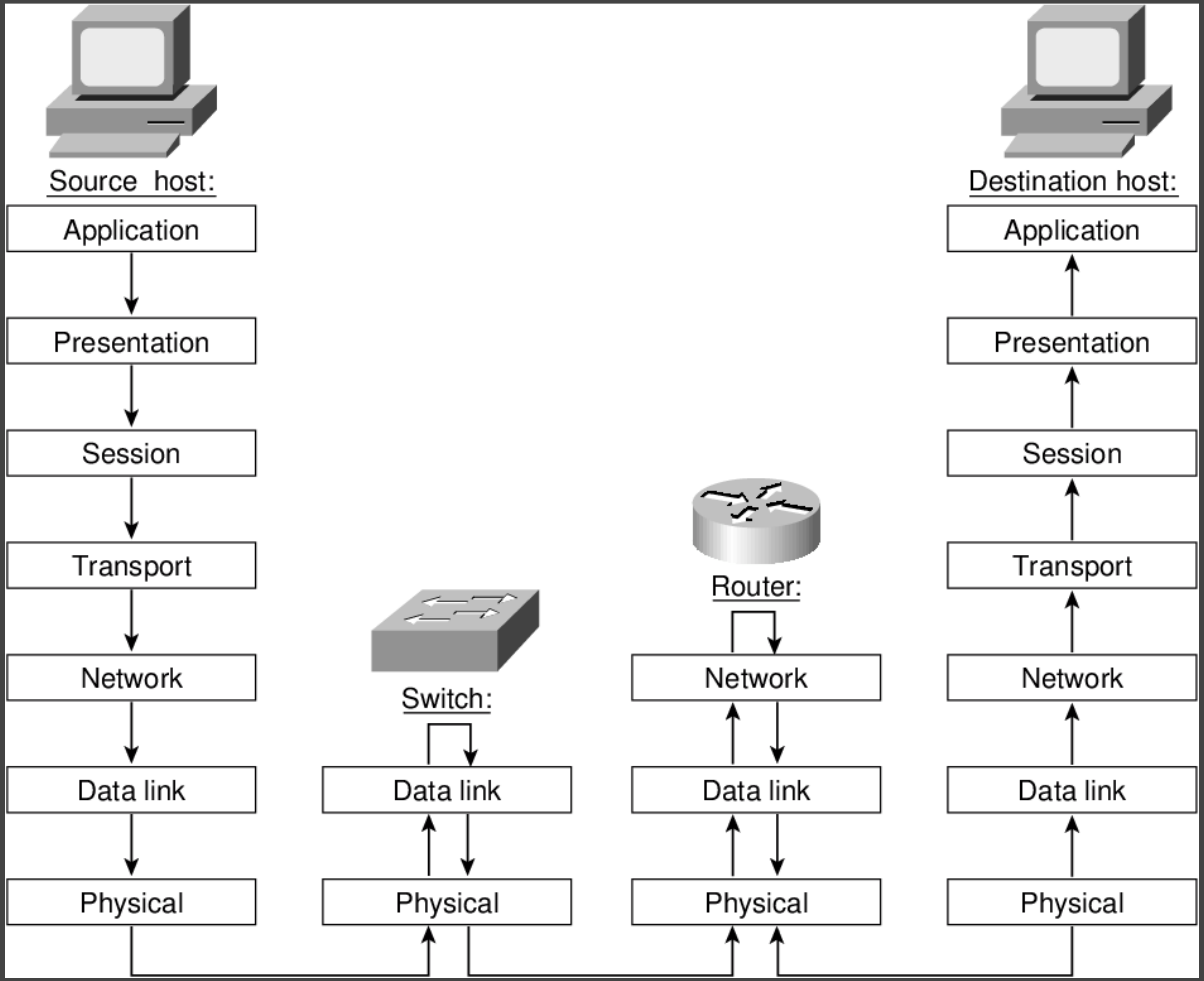
Switching
交換器

MESSAGE PACKING



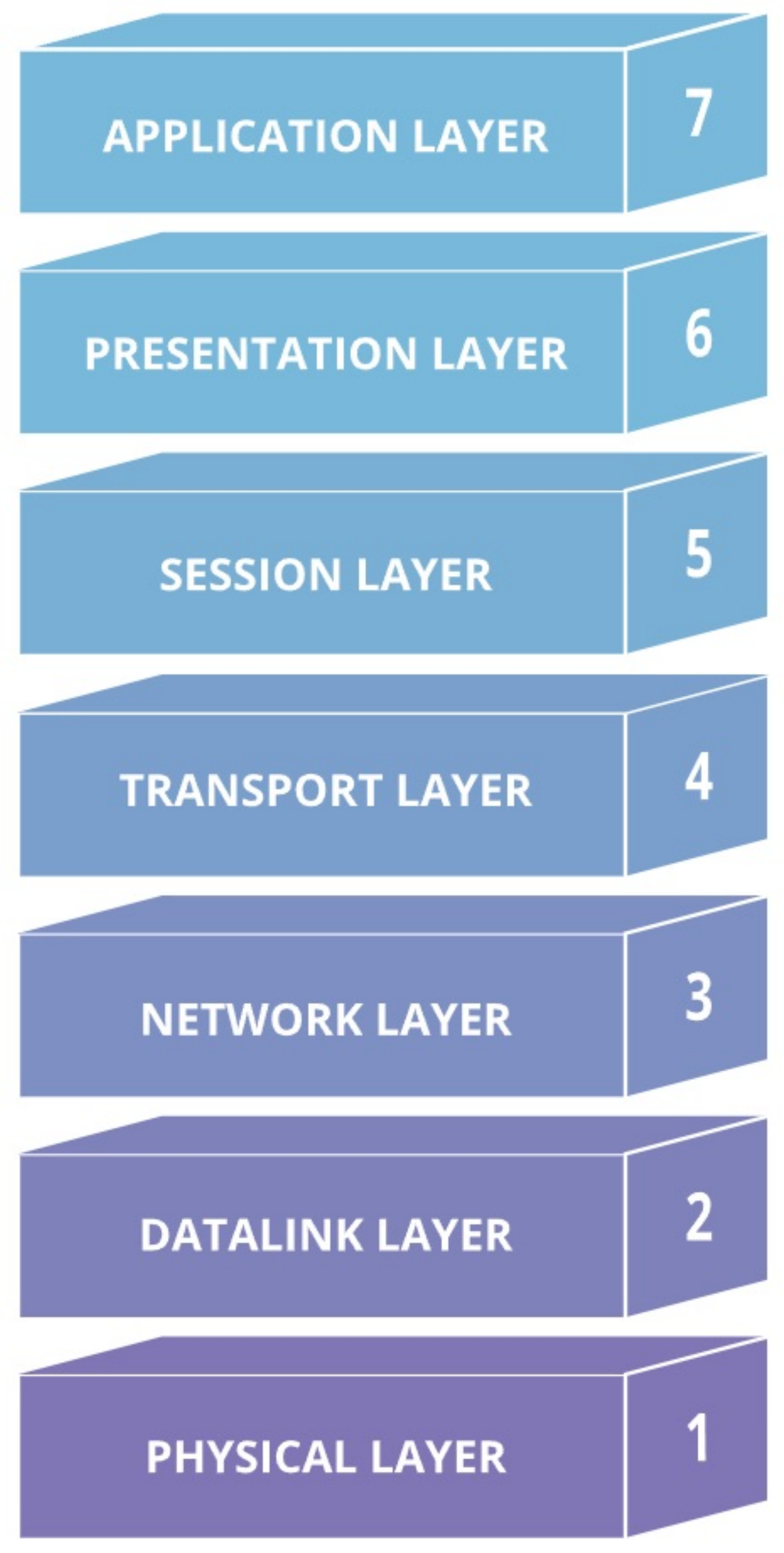
Each layer prepends or appends its information in a ***header*** or trailer

NETWORK ROUTING



ETHERNET

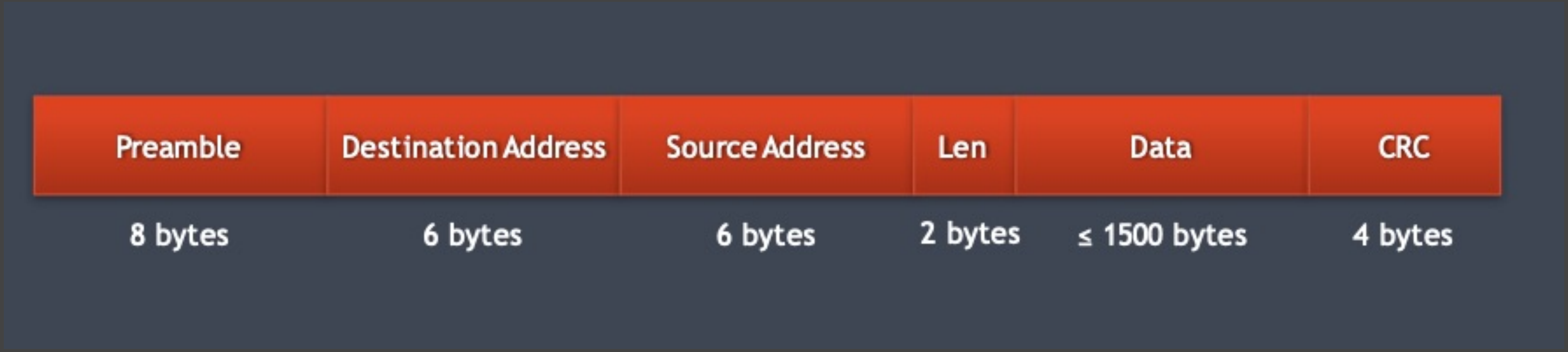
- **Physical layer**
  - Ethernet physical infrastructure
  - Ethernet encompasses both
    - the data link layer
    - the physical layer
  - IEEE 802.3 standard



ETHERNET

- Every Ethernet interface is assigned a unique 48-bit (hardware) address
- Represented as a sequence of 6 hexadecimal bytes delimited by ':' characters
  - 00:50:56:9F:27:3E
  - **ipconfig** to see network interfaces in windows system
- Also known as ***MAC address*** or ***physical address***
- Ethernet addresses are assigned to vendors by a central authority

ETHERNET



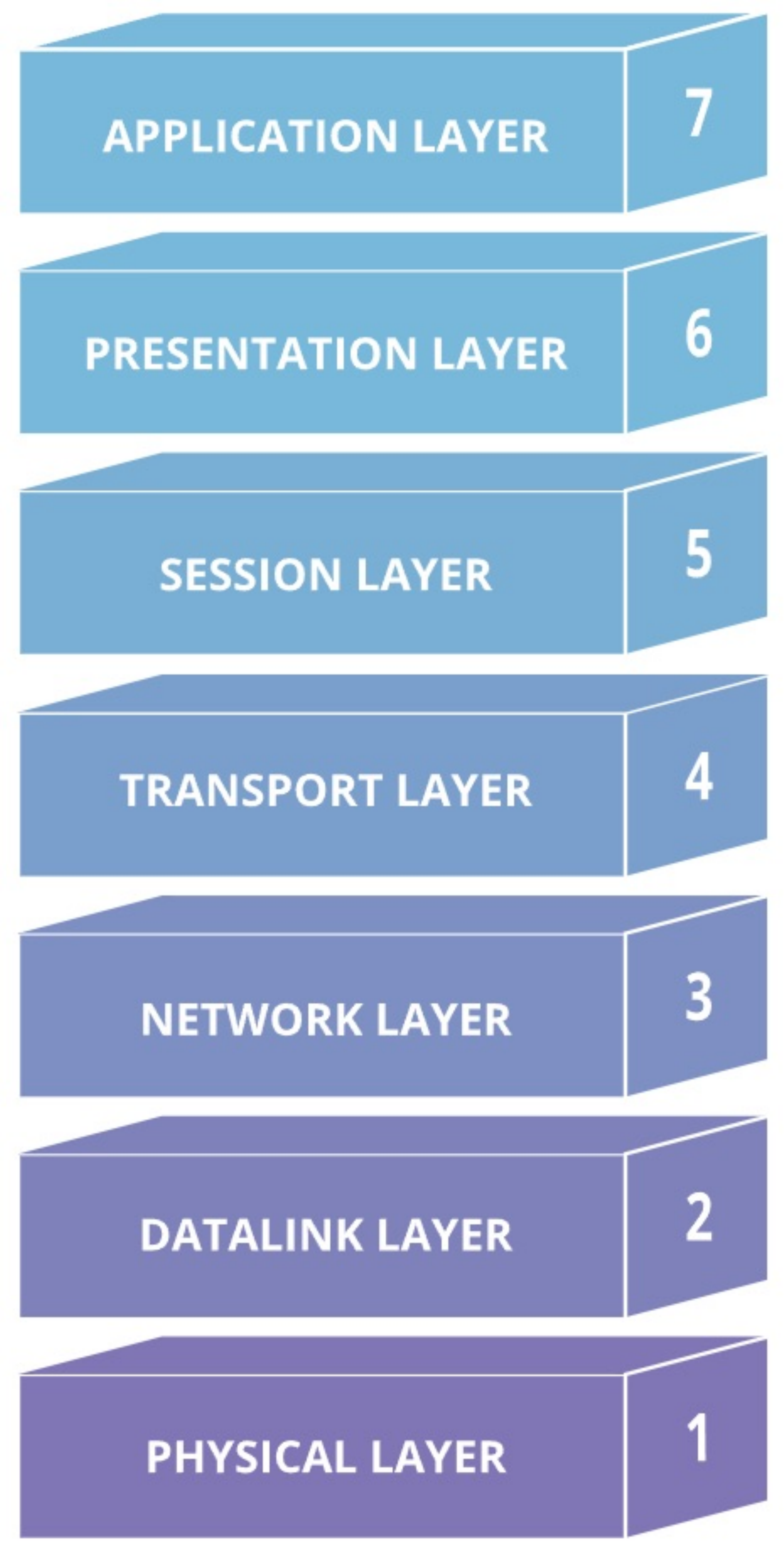
Preamble	Destination Address	Source Address	Len	Data	CRC
8 bytes	6 bytes	6 bytes	2 bytes	≤ 1500 bytes	4 bytes

Messages are sent over Ethernet via ***frames***:

The ***Preamble*** is a sequence of alternating `1`s and `0`s for synchronization
(In hex: AA:AA:AA:AA:AA:AA:AA:AB)

IP ADDRESS

- **Network layer**
  - IP address type
  - IP provides host-to-host delivery service of packets (called datagrams)
  - IP is connectionless
  - IP is unreliable
  - IP also provides translation between different data link protocols



APPLICATION LAYER 7

PRESENTATION LAYER 6

SESSION LAYER 5

TRANSPORT LAYER 4

NETWORK LAYER 3

DATALINK LAYER 2

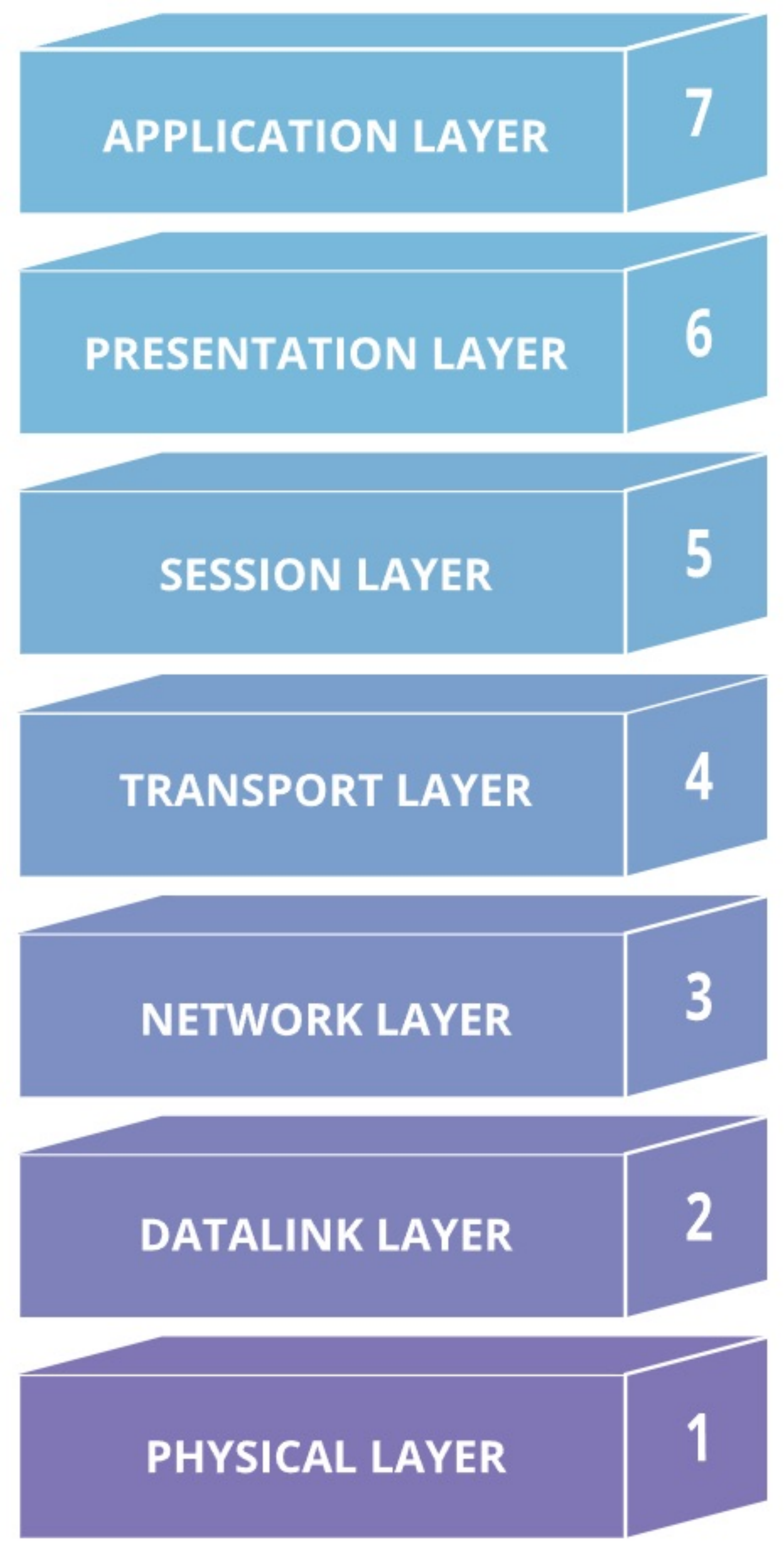
PHYSICAL LAYER 1

IP ADDRESS

- IP address types
- IPv4 is a series of 4 octets -> 32 bits
  - 192.0.2.33
- IPv6 is a series of 16 octets -> 128 bits
  - 2001:0db8:c9d2:aee5:73e3:934a:a5ae:9551
- Each IP address contains information that identifies
  - network ID and host ID
- ***Routing tables*** rely on IP address *classes*:
  - class A -> 192.0.2.33
  - class B -> 192.0.2.33
  - class C -> 192.0.2.33

TCP & UDP

- **Transport layer**
  - Two protocols at the Transport Layer:
    - TCP (Transmission Control Protocol)
    - UDP (User Datagram Protocol)



TCP & UDP

- TCP (Transmission Control Protocol)
  - A connection-based protocol that provides a reliable
  - flow of data between two ends
  - like the phone-call processing
    - dial number
    - connection established
    - speaking to one another
    - the data sent from one end of connection actually gets to the other end

TCP & UDP

- TCP (Transmission Control Protocol)
    - Guarantee send to the other end ***in order***
    - Designed for applications that require reliable communications, such as:
        - The HypertextTransfer Protocol (HTTP)
        - File Transfer Protocol (FTP)
        - Telnet
        - so on…

TCP & UDP

- UDP (User Datagram Protocol)
  - Non connection-based protocol that sends independent packets of data called - datagram from one end to another with no guarantees about arrival
- Like the mail-server processing
  - Sent a letter through the postal service
  - The order of delivery is not guaranteed…
  - Each datagram is independent

TCP & UDP

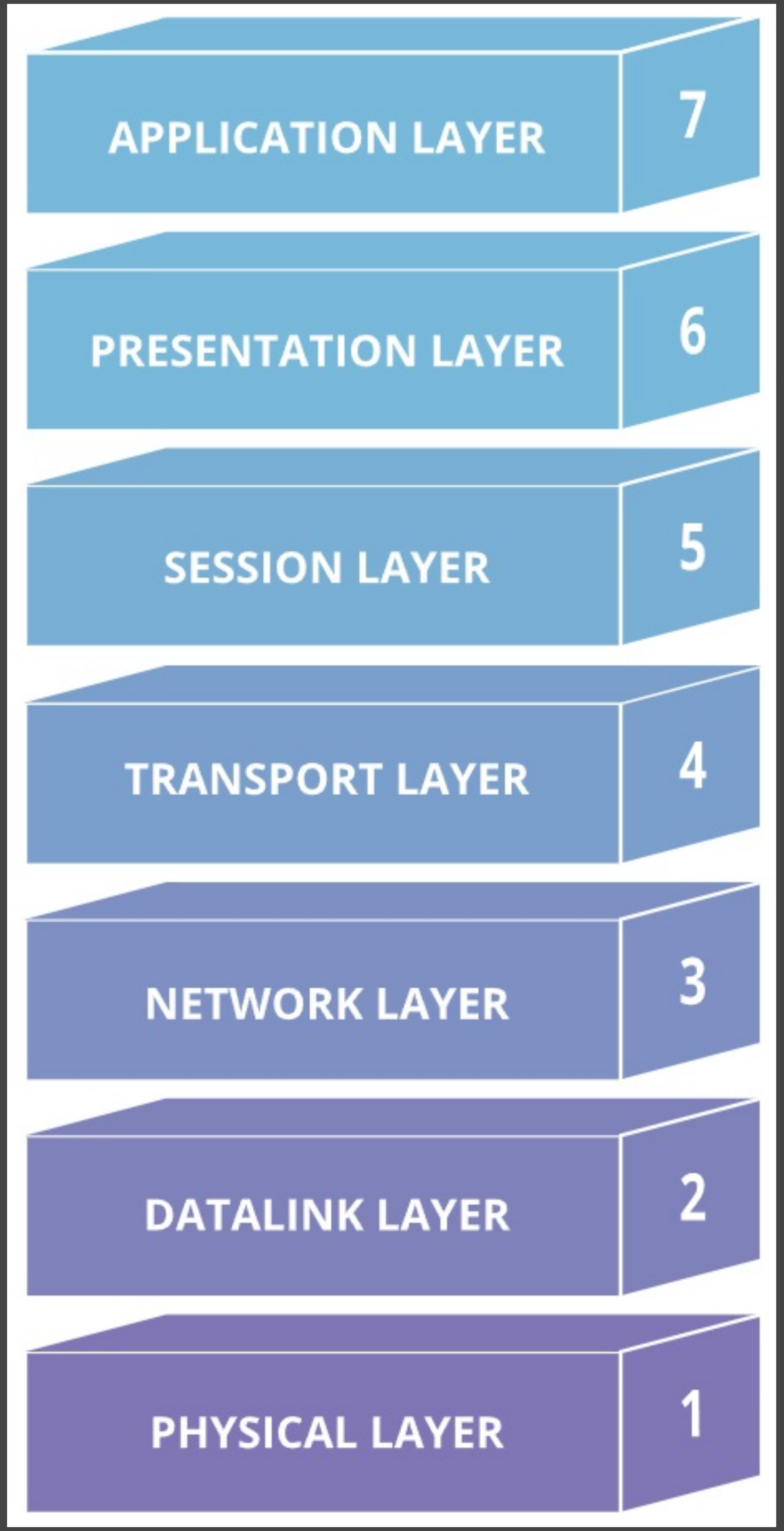
- UDP (User Datagram Protocol)
  - Designed for applications don’t require such strict standards, in fact this standard may be slowed down by the extra overhead or the reliable connection
    - Audio or video streaming
    - Part of game genres
    - ...
  - Many firewalls and routers do not allowing UDP packets pass through by default configured settings

TCP & UDP

- Port number
  - the computer is identified by its 32-bit IP address
    - which IP used to deliver data to the right computer on the network
  - ports are identified by a 16-bit number, which TCP and UDP use to deliver the data to the right application
  - 0~1023 are reserved by well-known services such as HTTP(80) and FTP and other system services

SOCKET PROGRAMING

- Application layer
  - Network APIs provide the bridge between applications and protocol software
  - Services are made available (often by the OS)



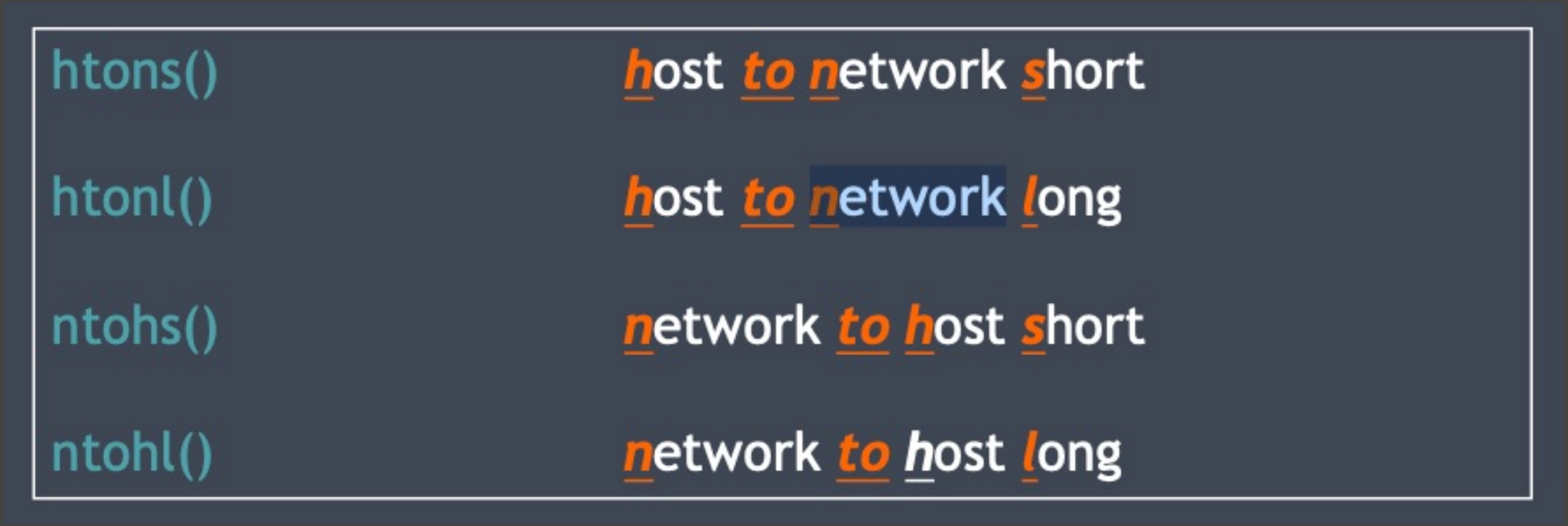
SOCKET PROGRAMING

- Network API often provides a generic programming interface:
  - support for multiple communication protocol suites/families (e.g.TCP, UDP, IP)
  - endpoint address representation independence
  - network data types (for portability)
    - With from-host and to-host conversion functions
    - e.g. htons(), ntohs(), htonl(), ntohl(), etc.

SOCKET PROGRAMING

- Byte Order
  - Take data b34f for example:
    - Big-Endian -> b34f
    - Little-Endian -> 4fb3
- Network Byte Order is Big-Endian
  - Means b34f for every protocol
- Host Byte Order is
  - Intel processor is Little-Endian -> 4fb3
  - Motorola processor is Big-Endian -> b34f

SOCKET PROGRAMING



htons()	host to network short
htonl()	host to network long
ntohs()	network to host short
ntohl()	network to host long

Use convert function before transferring data (integer type)

Good news is,

modern network SDK tackles all of this...

SOCKET PROGRAMING

What is a socket?

- A way to speak to other programs using standard Unix file descriptors
- Everything in Unix is a file!
  - a FIFO, a terminal, a real on-the-disk file…etc
  - a network connection is still a file
- A file descriptor is simply an integer associated with an opened file
  - Like FILE object in C language
  - When Unix programs do any sort of I/O, they do it by reading or writing to a file descriptor

SOCKET PROGRAMING

- There are two types of internet Sockets
  - Stream socket
    - connection sockets
    - reliable type
      - 1, 2, 3 -> 1, 2, 3
    - TCP protocol
  - Datagram socket
    - connectionless sockets
    - un-reliable type
      - 1,2,3 -> maybe 1,3,2 or 2,1,3 or nothing
    - UDP protocol

References

- MIS — 網路概論- OSI、MAC、IP
- 什麼是OSI的7層架構？和常聽到的Layer 7有關？
- WIKI - OSI模型
- Cloudflare - 什麼是 OSI 模型？
- J.Day, Zimmermann H, “The OSIreference model” published in IEEEvol.71,issue 12, pages:1334-1340,1983.