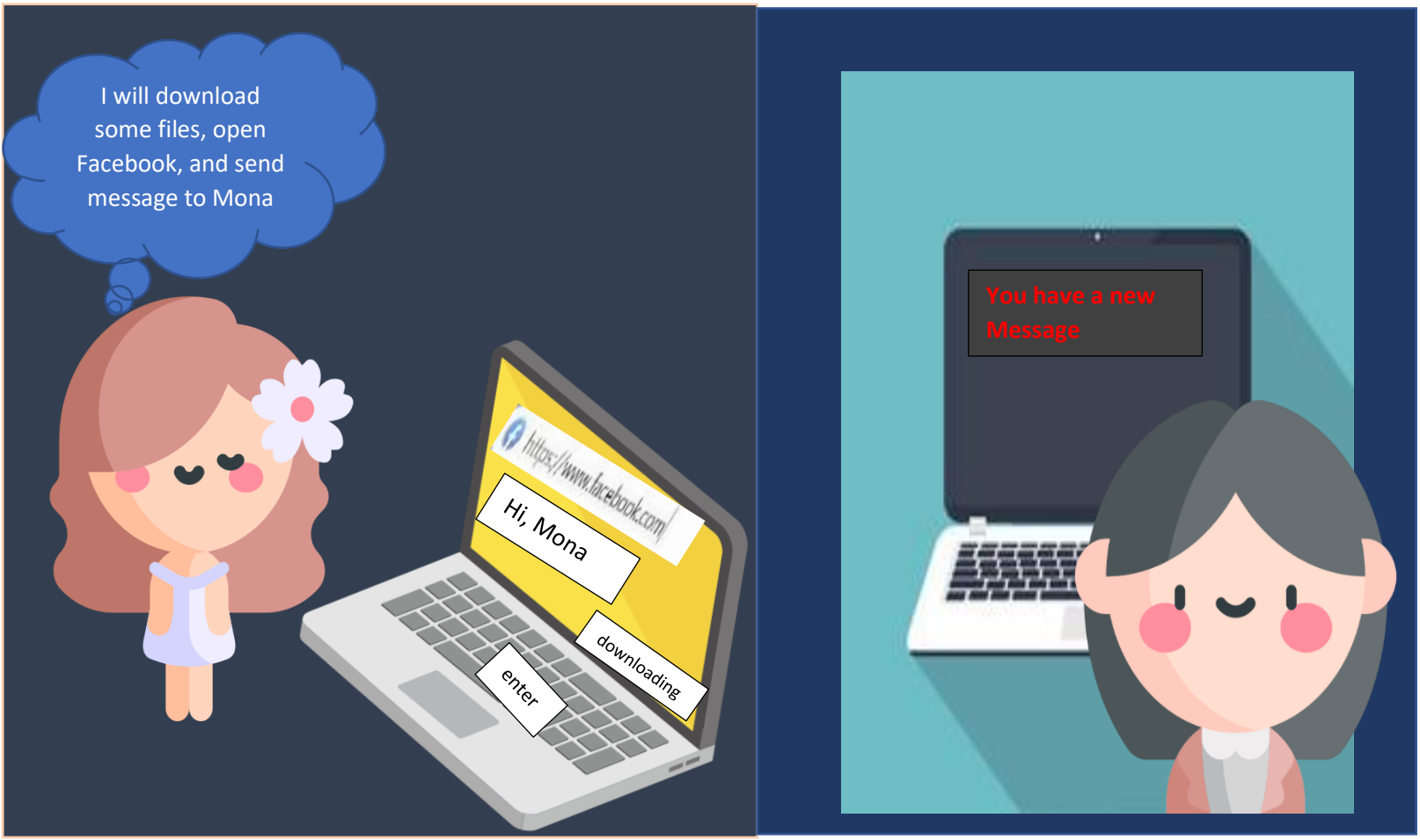
OSI Model


I will download some files, open Facebook, and send message to Mona
https://www.facebook.com/
Hi, Mona
downloading
enter
You have a new Message

on the other side

The Open Systems Interconnection model (OSI)

can be seen as a universal language for computer networking. It's based on the concept of splitting up a communication system into seven abstract layers

Hi layers, we will find the website, download files and send this message “Hi Mona”.

every layer will add her header.

OSI model

7 Application layer

6 Presentation layer

5 Session layer

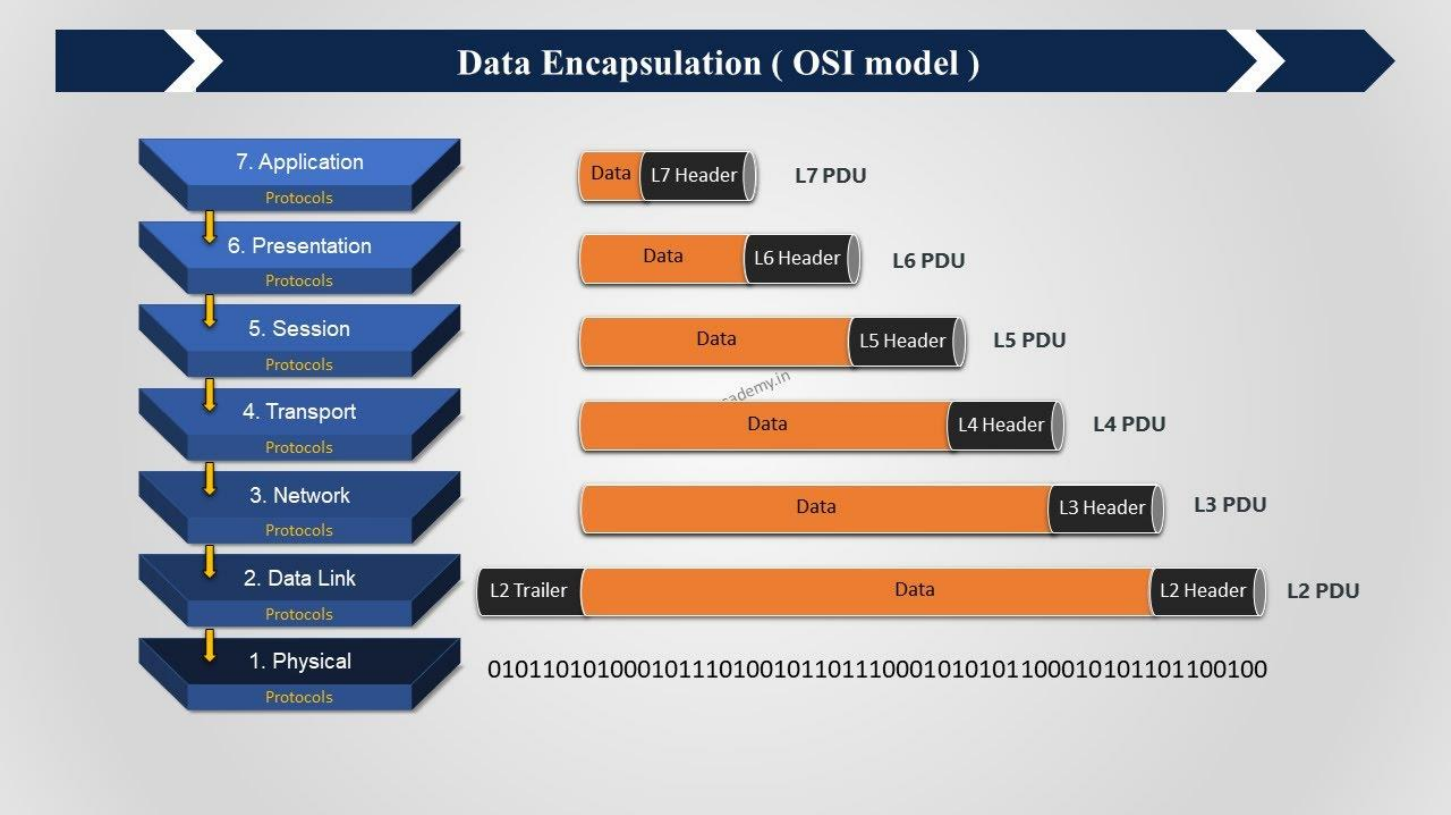
4 Transport layer

3 Network layer

But, I will add header and tailer!

2 Datalink layer

1 Physical layer


Data Encapsulation (OSI model)
7. Application
Protocols
6. Presentation
Protocols
5. Session
Protocols
4. Transport
Protocols
3. Network
Protocols
2. Data Link
Protocols
1. Physical
Protocols
Data
L7 Header
L7 PDU
Data
L6 Header
L6 PDU
Data
L5 Header
L5 PDU
Data
L4 Header
L4 PDU
Data
L3 Header
L3 PDU
L2 Trailer
Data
L2 Header
L2 PDU
0101101010001011101001011011100010101011000101011011001 00

I'm Application layer

Actually I provide an interface between software running on a computer and the network itself

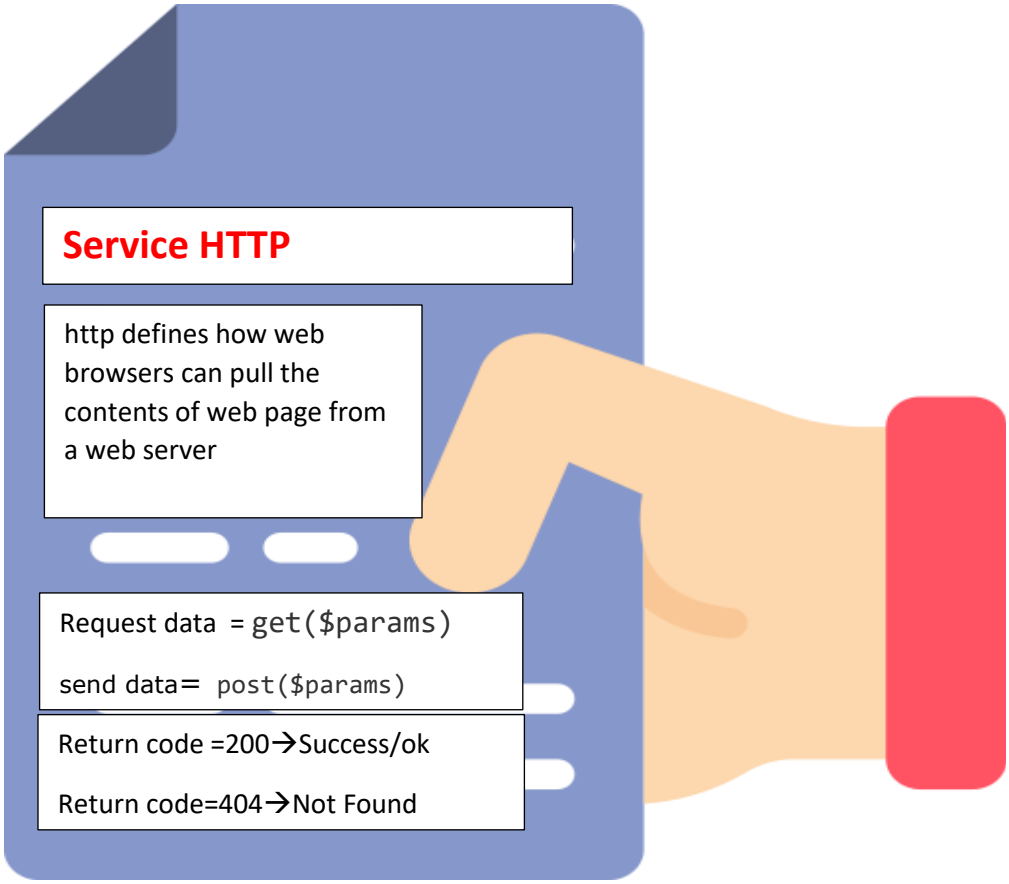
I provide services to the application software running on a computer

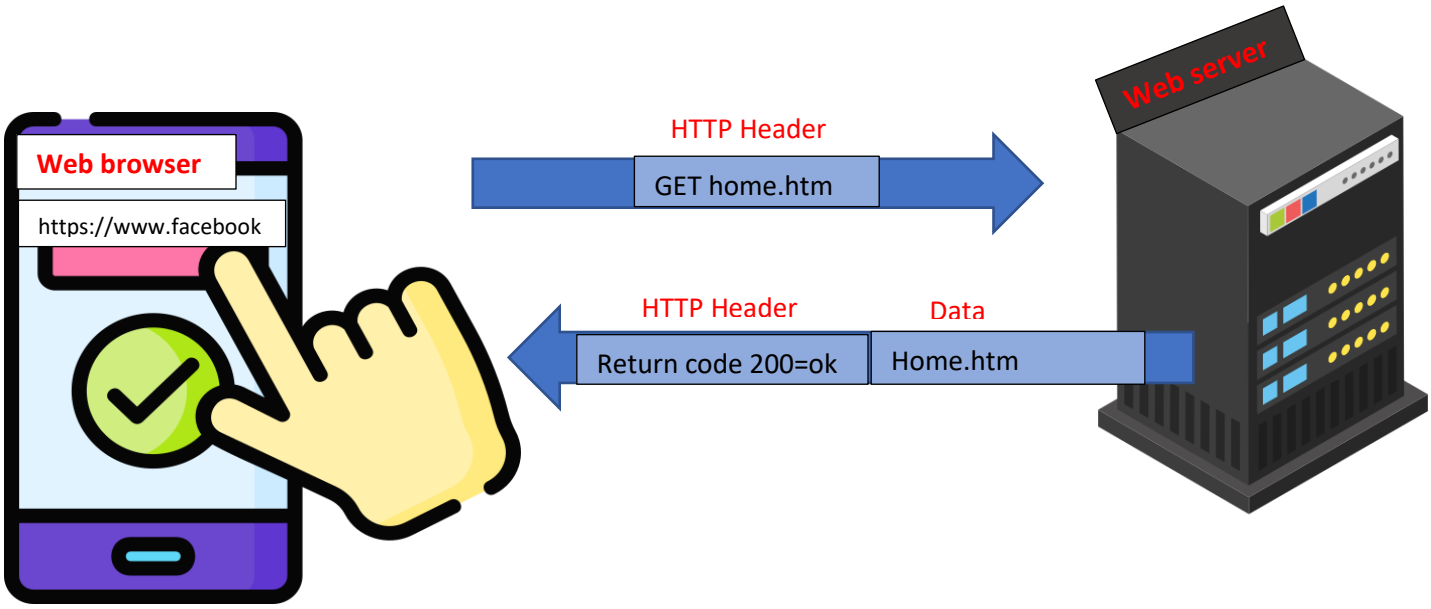
these are:

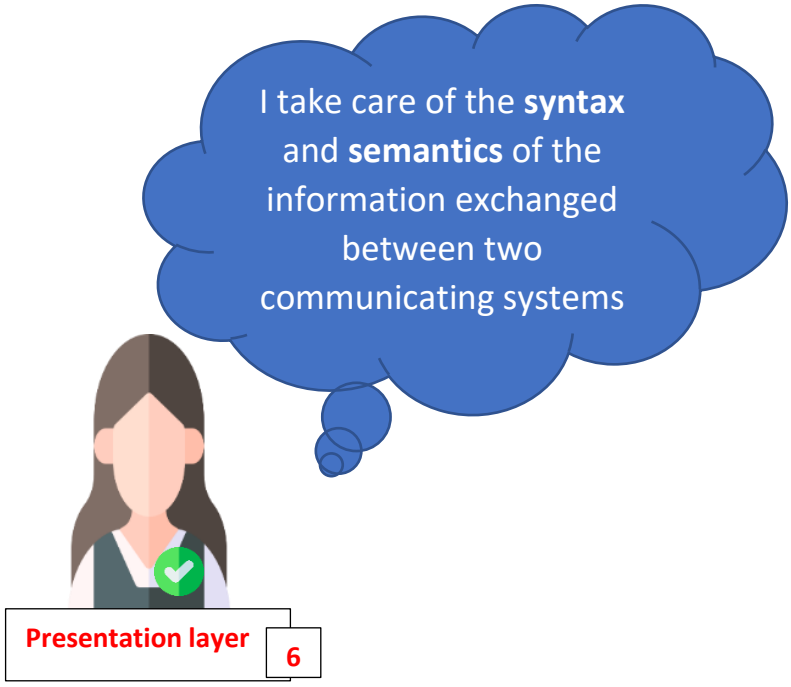
- **E-Mail service ➔POP, STMP**
- **Web applications ➔HTTP,HTTPS**
- **File transfer➔FTP,TFTP**
- **Directory services ➔DNS**
- **Host sessions ➔Telnet, SSH**
- **Network management ➔SNMP**

Application Software

Application layer


Service HTTP
http defines how web browsers can pull the contents of web page from a web server
Request data = get($params)
send data= post($params)
Return code =200→Success/ok
Return code=404→Not Found


Web browser
https://www.facebook
HTTP Header
GET home.htm
Web server
HTTP Header
Data
Return code 200=ok
Home.htm



I have three functions:

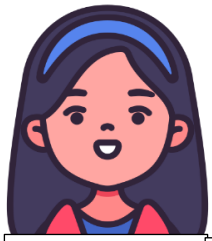
Encode: is a method of changing the way we represent data to standardize the data we are dealing with

Encryption : is a mathematical formula which, with the help of a key, changes plaintext into ciphertext.

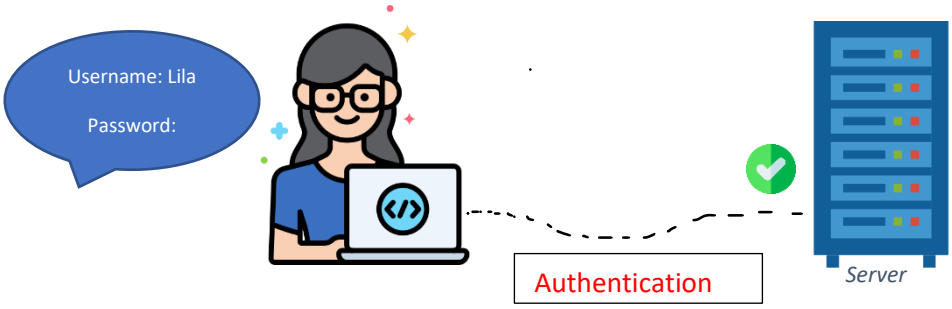
Compression: reduces the size of data

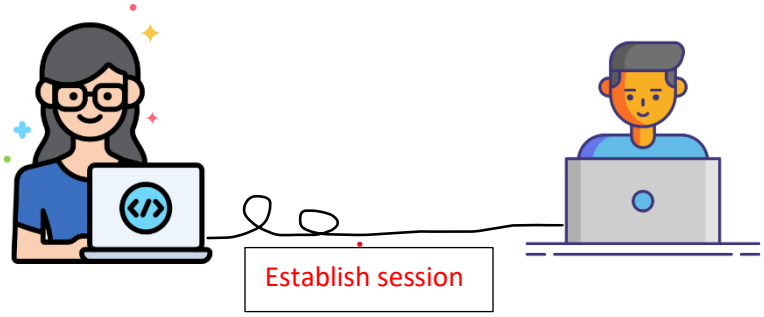
I control the dialogues (connections)between computers . I establish ,manage, and terminate Session , API, Sockets, Winsock

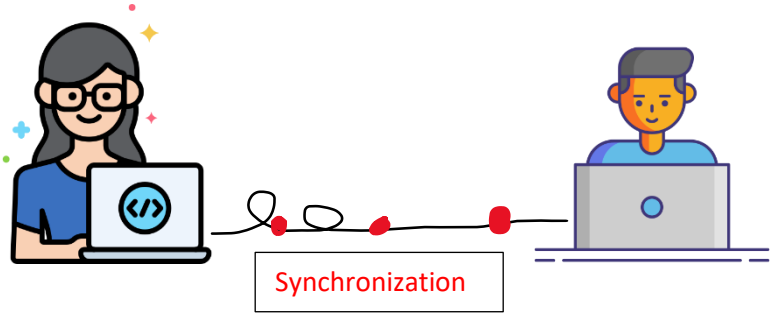
And allow a process to add checkpoints which are considered as synchronization points into stream of data if a crash happens return to last checkpoint.



Session layer 5



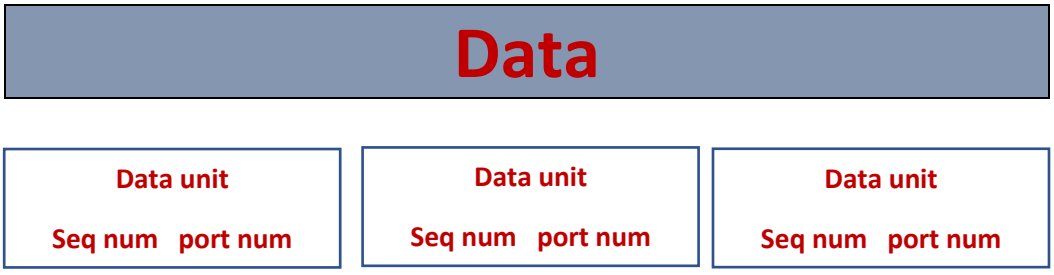




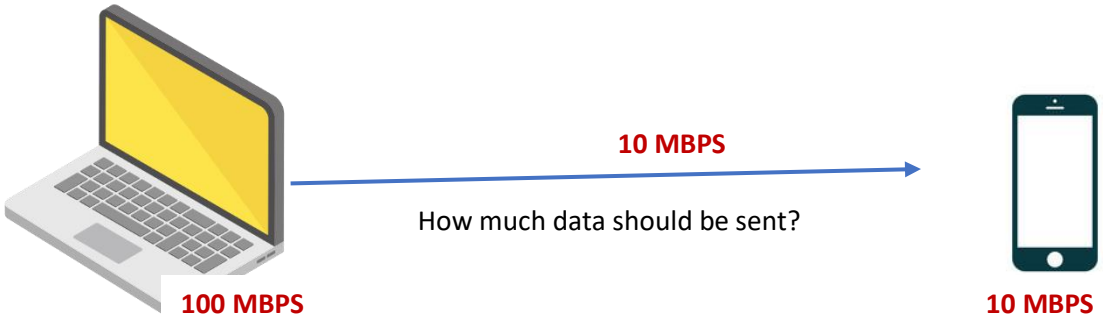
Accept data the layer above, split it up into smaller units. Pass these data units to the network layer, and ensure that all the pieces arrive correctly at the other end.

I have Six Functions

1-Segmentation

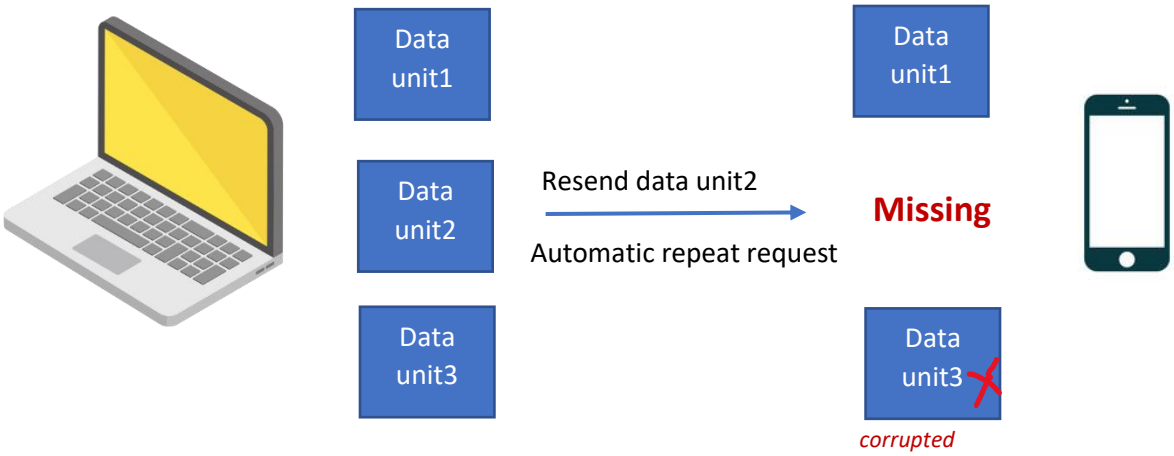


2-Flow control



3-Error Control

How can errors be detected &corrected



4- checksum (hashing): used for error detection in transport layer header

5- control connection services

Connection oriented transmission service ➔transmission control protocol(TCP)

Before delivering packets, the connection is made with the transport layer at the destination machine Three-way handshake.

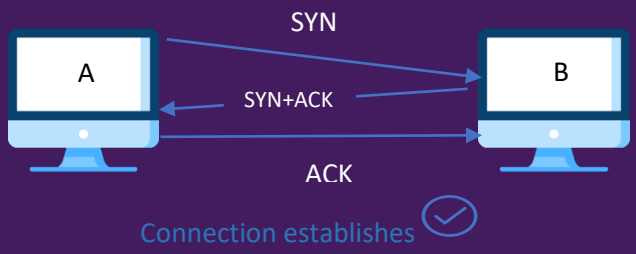
Connectionless service➔user datagram protocol(UDP)

Each segment is considered as an independent packet and delivered to the transport layer at the destination machine.

TCP

Required: send data from A to B

Step 1 : establish connection from A to B by 3 way handshake



Connection establishes

Step 2: send data as and wait acknowledge

Step 3: end connection

Close connection

Slow

unicast

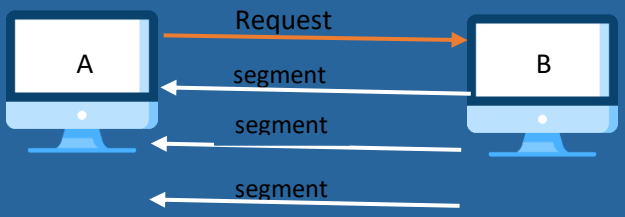
Reliable➔resend, Seq, Ack, Timing, Flow control (sliding window).

DNS, HTTP, HTTPS, SSH, SMTP

Control bits flags➔SYN, ACK, FIN, RST, PSH, Urgent *

UDP

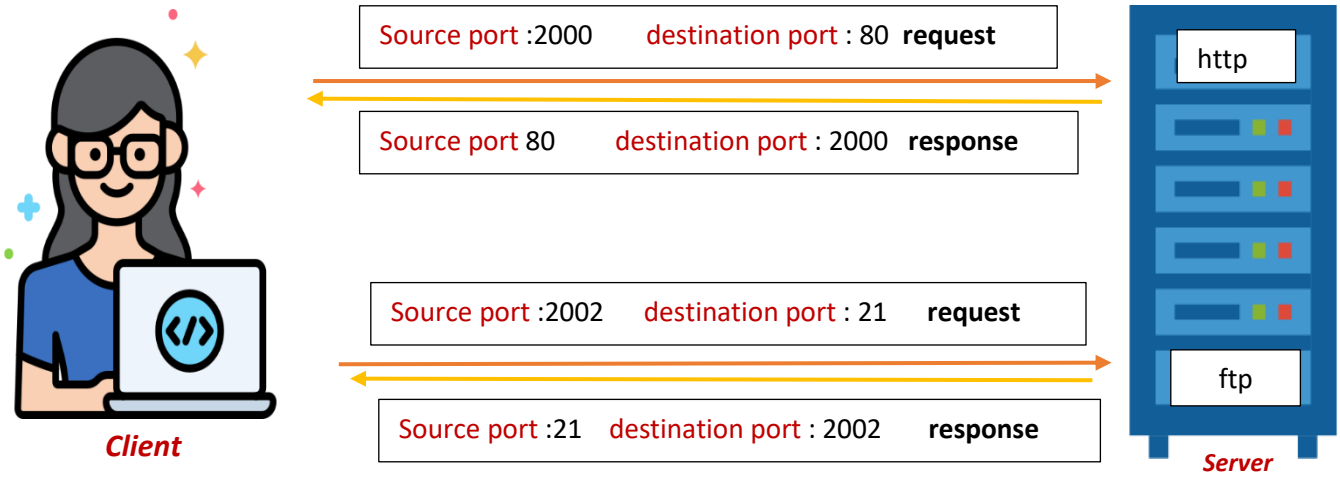
Required: send data from A to B



- Fast use in real time application
- Multicast, broadcast
- UDP performs a very limited amount of error checking.
- This minimum number of overheads

6-Service Point Addressing (port number): Transport Layer header includes port number, this layer gets the message to the correct process on the computer.

1. Source port → process number on the computer.
2. Destination port → process number in another or same computer.
   Note → Socket = IP + Port number



Port number= 16 bit

Number of ports= 2^16=65536 port

Well known = service or applications = 0:1023

Registered = user service or application =1024 :4951

Private port =4952:65535

Some well known

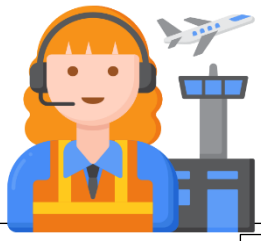
HTTP:80

HTTPS:443

DNS:53

FTP: 20,21

I assign addresses, transfer network packets from the source to the destination, decide the root to be taken by the packets to travel from the source to the destination among the multiple roots available in a network …routing



Network layer 3

I have Three Functions

1- Logical addressing

Logical address (IP) = 32 bit = Network part +host part= ➔ipv4

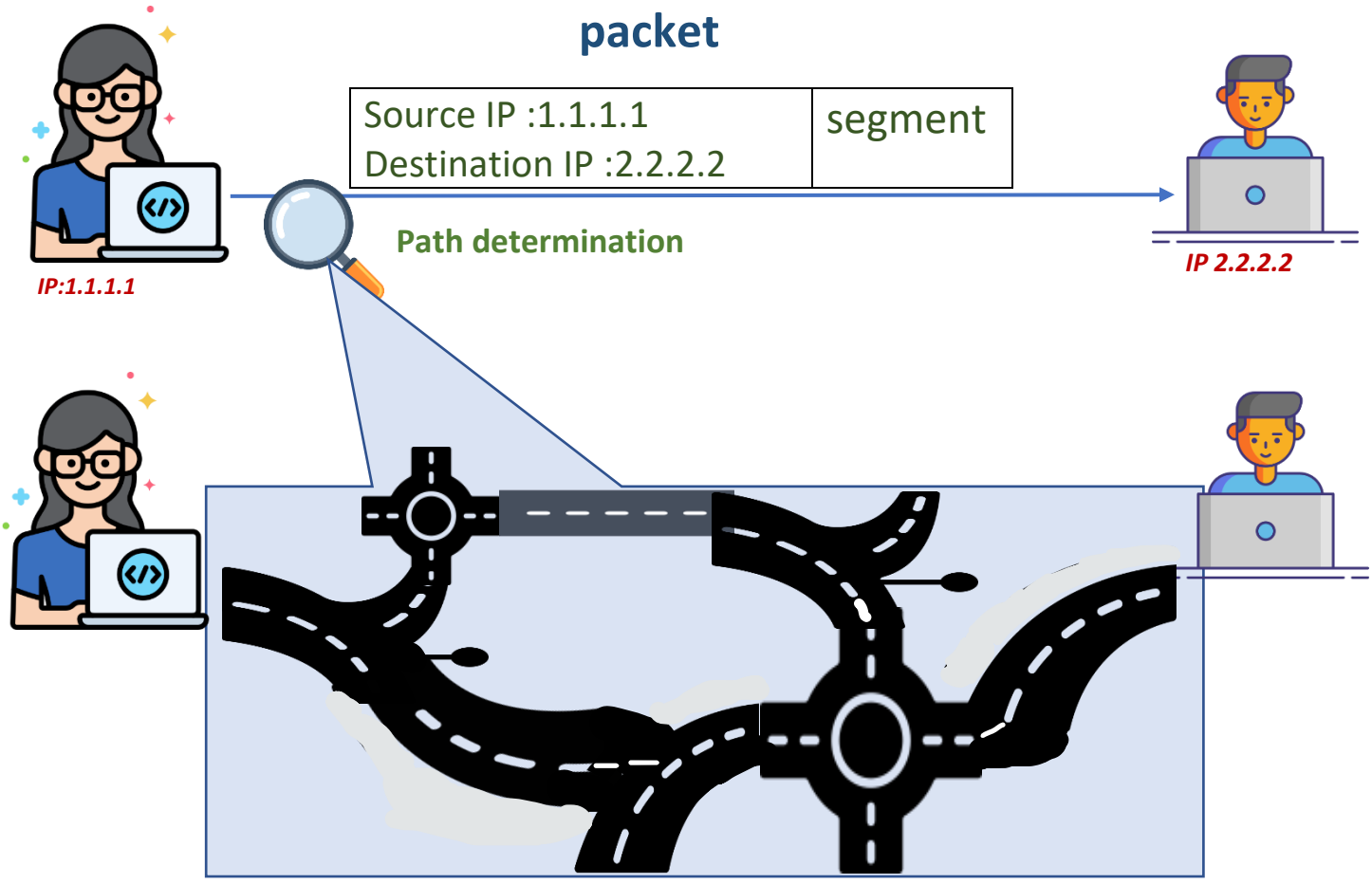
Network	Host

32

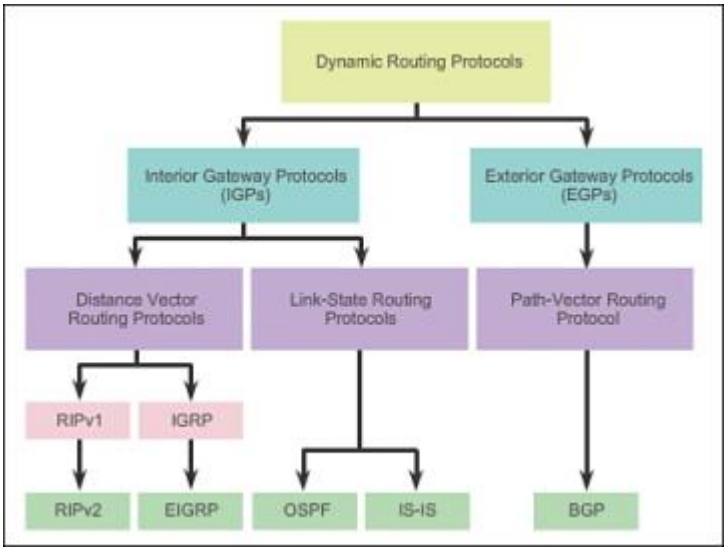
Packet = segment +IP

packet

IP	segment



2- Path determination: decide the root to be taken by the packets to travel from the source to the destination among the multiple roots available in a network by static or dynamic routing protocol



3- IP fragmentation

Different Networks may have different maximum transmission unit (MTU).When one network wants to transmit datagrams to a network with a smaller MTU.

Routers split a packet that is too larger:

-Typically break into large pieces

-Copy IP header to pieces

-Adjust length on pieces

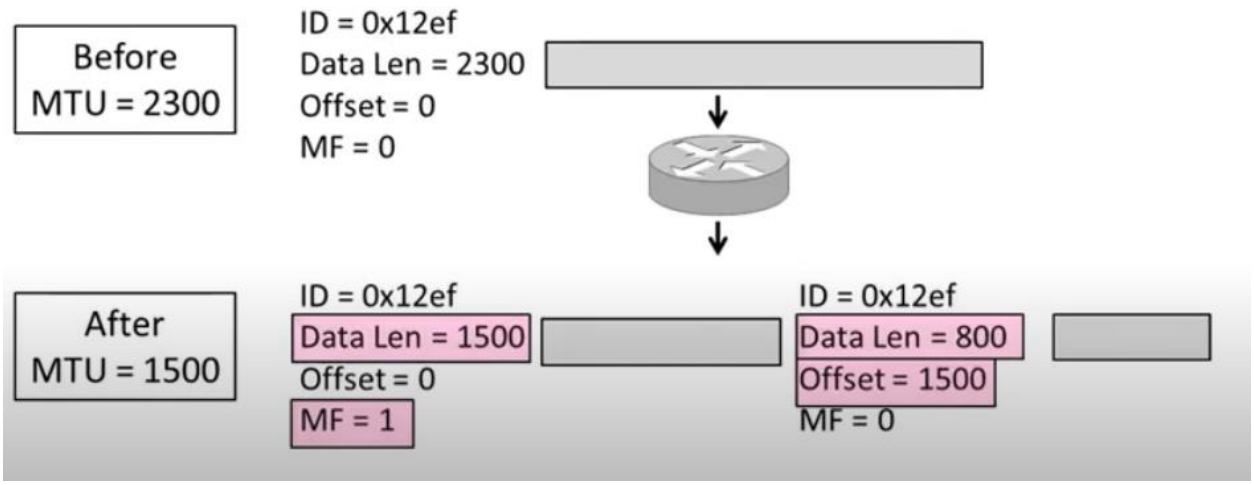
-Set offset to indicate position

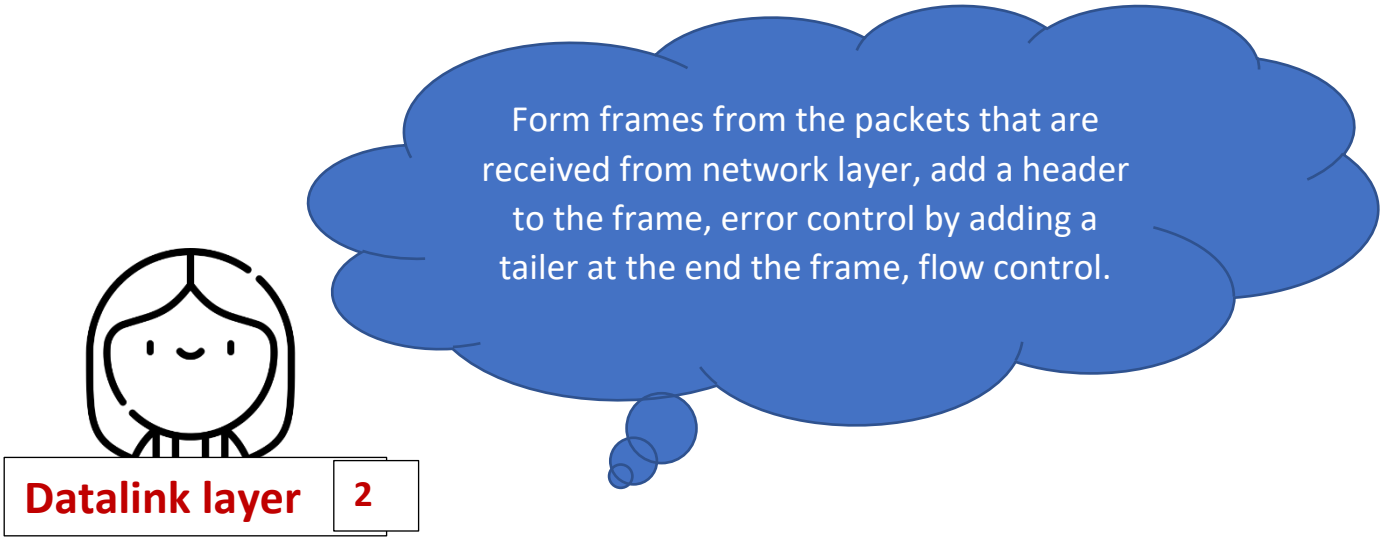
-Set MF (more fragments) on all pieces except last

Fields in IPv4 header for fragmentation

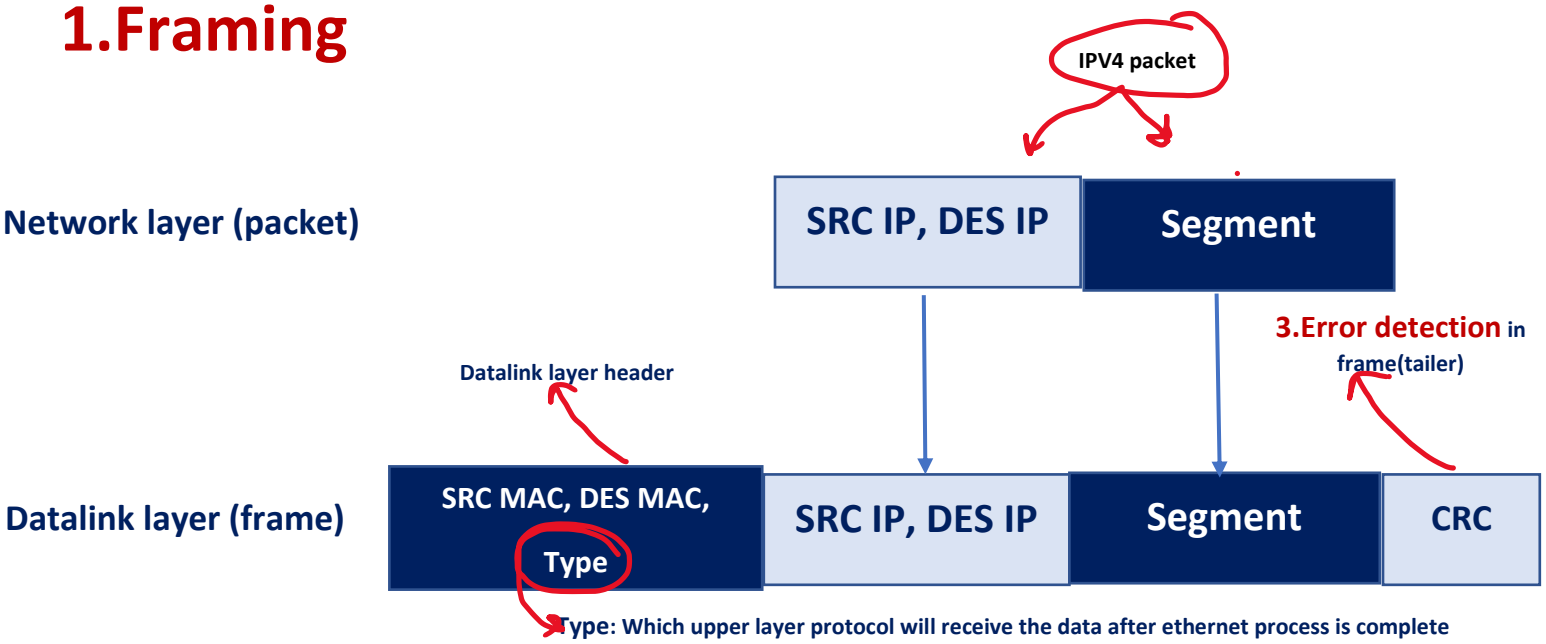
- **More fragments (MF = 1 bit) –** tells if more fragments are ahead of this fragment i.e. if MF = 1, more fragments are ahead of this fragment and if MF = 0, it is the last fragment.
- **Don't fragment (DF = 1 bit) –** if we don't want the packet to be fragmented

Example





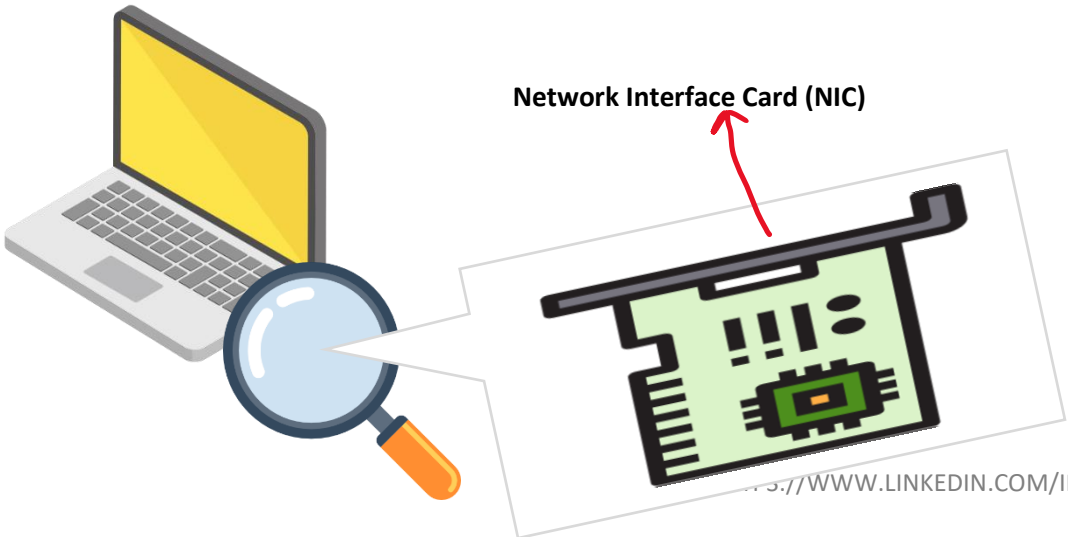
1.Framing

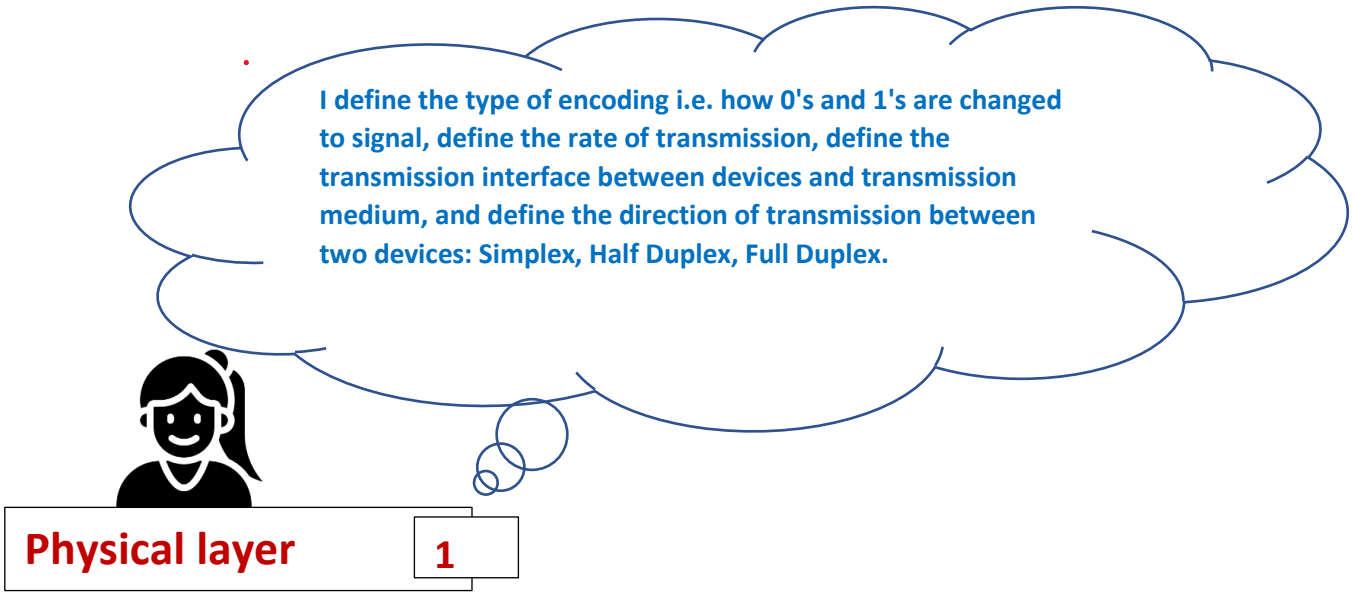


2.Physical Addressing

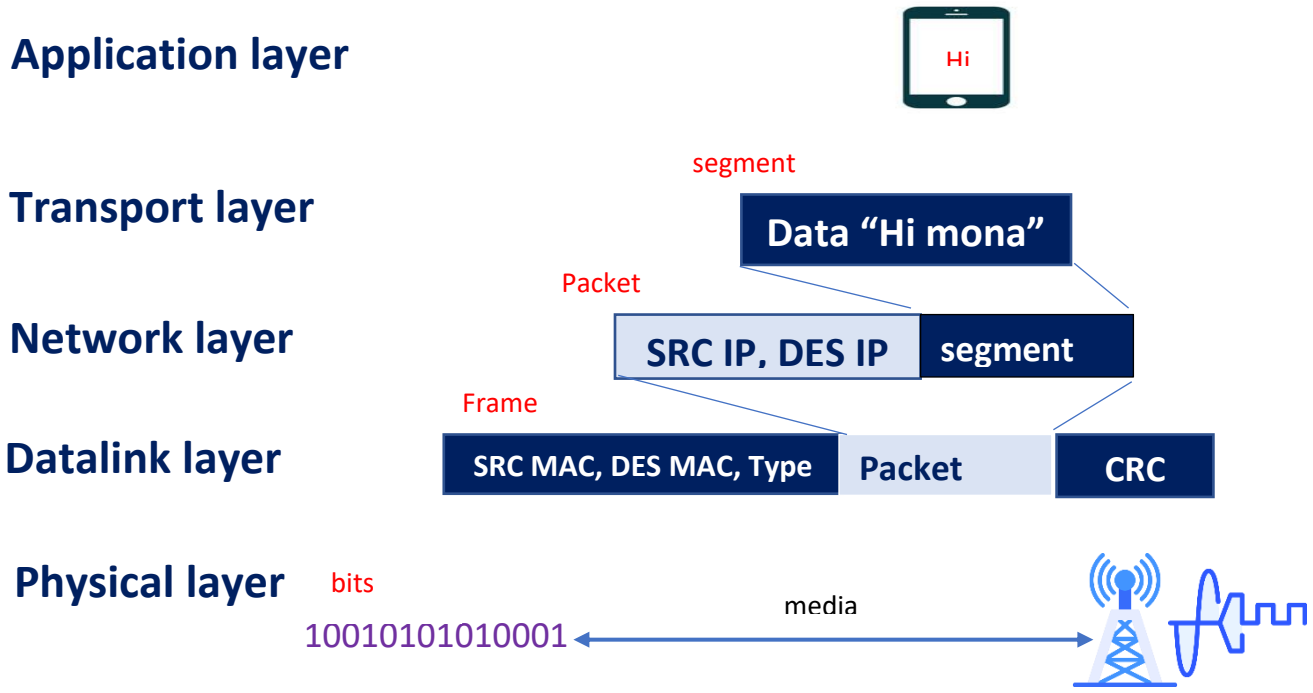
adds a header to the frame source MAC and destination MAC address

MAC Address (hardware address): is a unique identifier assigned to a network interface controller (NIC) is 48 bits.

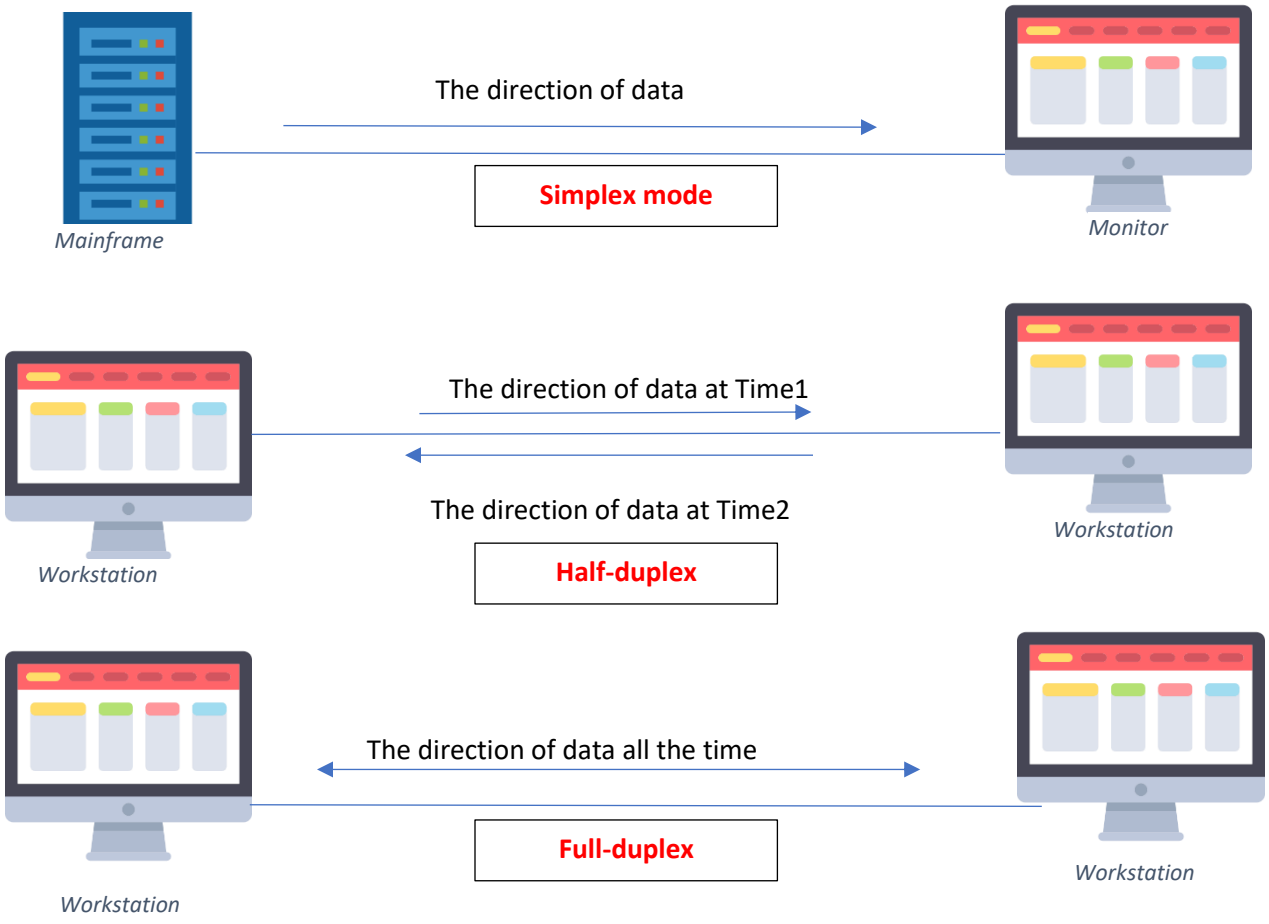




1. **Representation of Bits:** Data in this layer consists of stream of bits. The bits must be encoded into signals for transmission. It defines the type of encoding i.e. how 0's and 1's are changed to signal.



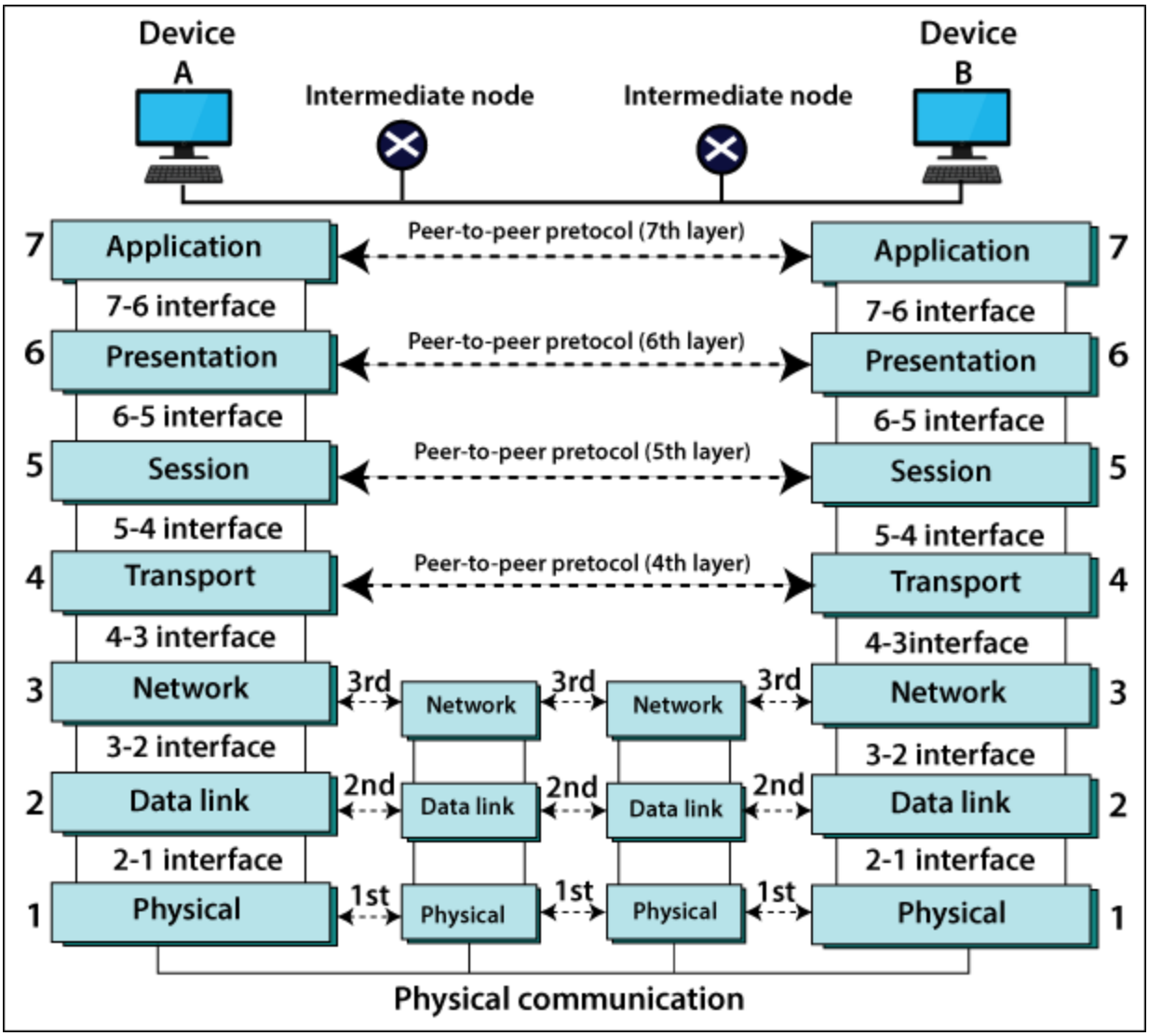
2. **Transmission Modes:** Physical Layer defines the direction of transmission between two devices: Simplex, Half Duplex, Full Duplex.



OSI Model

7. Application	**User interface**	**Data**	**Gateways**
6. Presentation	**Data Presentation, encryption, compression, encoding.**	**Data**	**Gateways**
5. Session	**Maintaining sessions**	**Data**	**Gateways**
4. Transport	**Process to process communication, port nos.**	**Segments**	**Gateways**
3. Network	**Logical addressing, source to destination delivery.**	**Packets**	**Routers**
2. Data-Link	**Physical addressing, Node to Node delivery.**	**Frames**	**Switches, Bridges**
1. Physical	**Moves bits between devices**	**Bits**	**Hubs, Repeaters**

TCP		UDP	
FTP	20,21	DNS	53
SSH	22	BooTPS/DHCP	67
Telnet	23	TFTP	69
SMTP	25	NTP	123
DNS	53	SNMP	161
HTTP	80		
POP3	110		
IMAP4	143		
HTTPS	443		


Device A
Intermediate node
Intermediate node
Device B
7
Application
Peer-to-peer pretocol (7th layer)
Application
7
7-6 interface
7-6 interface
6
Presentation
Peer-to-peer pretocol (6th layer)
Presentation
6
6-5 interface
6-5 interface
5
Session
Peer-to-peer pretocol (5th layer)
Session
5
5-4 interface
5-4 interface
4
Transport
Peer-to-peer pretocol (4th layer)
Transport
4
4-3 interface
4-3interface
3
Network
3rd
Network
3rd
Network
3rd
Network
3
3-2 interface
3-2 interface
2
Data link
2nd
Data link
2nd
Data link
2nd
Data link
2
2-1 interface
2-1 interface
1
Physical
1st
Physical
1st
Physical
1st
Physical
1
Physical communication

https://www.youtube.com/watch?v=LeaEmOUVEn0/

https://www.encryptionconsulting.com/education-center/

https://www.flaticon.com/

https://www.studytonight.com/computer-networks/