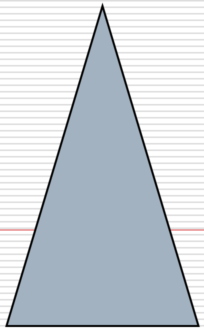
# The OSI Model

# OSI or ISO

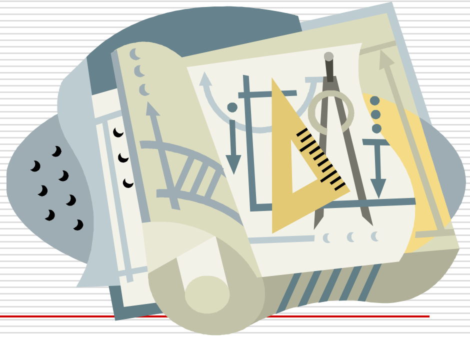
- The layering model is OSI, which stands for **O**pen **S**ystem **I**nterconnection.
- The organization that developed the model is ISO, **I**nternational **O**rganization for **S**tandardization.
  - ANSI (American National Standards Institute) is a member of ISO.
    - ANSI farms out much of the hardware standardization to the IEEE.

# ISO

- <u>**I**</u>nternational <u>**O**</u>rganization for <u>**S**</u>tandardization, shouldn't that be IOS???
- No.  ISO is not really an acronym.
- "ISO" was chosen because it means equal as in an <u>iso</u>sceles triangle that has two equal sides.



# Standards



- “**Standards** are **documented agreements** containing **technical specifications** or other **precise criteria** to be used consistently as rules, guidelines, or definitions of characteristics, to ensure that materials, products, processes and services are fit for their purpose.”
- From the ISO website

# Open versus Proprietary


- **Open**: specifications are public.
  - Requires committee agreement, can be slow.
- **Proprietary**: Privately owned and controlled, specifications are kept private so that other companies cannot duplicate.
  - No outside agreement needed, can be reactive (faster).
- Consumers prefer open and standardized systems, as these allow them to use the products of various manufacturers.
  - But sometimes it takes too long to come up with them.

# Layering/Interfacing

- Layering is an important concept in computing.
- When a user runs an application,
  - The user interfaces with the application.
  - The application interfaces with the operating system.
  - The operating system interfaces with the BIOS.
  - The BIOS interfaces with the input/output hardware.

# To what purpose?

- The user need only know something about the application and doesn’t need to know any details about the operating system, the BIOS or the hardware.
- Similarly the application needs information about the user and the operating system but not the BIOS or hardware.
- For layering to be successful there must be a “standard” interface into a layer.
  - The operating system is said to provide an Application Programming Interface (API).

# More than just a way of thinking about it

- These layers can be seen in the industry, there are
  - Companies that specialize in hardware
    - Seagate, Maxtor, Western Digital
  - Companies that specialize in BIOS
    - AMI, Phoenix
  - Companies that specialize in operating systems
    - Microsoft, Sun, B
  - Companies that specialize in applications
    - Oracle, Microsoft, AOL

# Communications Layering

- The communication between two (or more) users on a network similarly is broken down into layers.
- A possible model for this layering was laid out by the ISO.

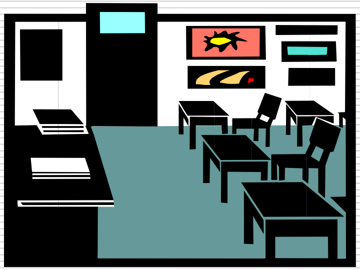
# What is OSI?

- OSI serves as a "reference model" for how information is transmitted between any two points in a network.
- It guides manufacturers (a.k.a. vendors) so they can make products that will work with other products.
- The OSI model is comprised of seven layers that are involved in communicating between two nodes of a network.

# It’s not the law

- The OSI model was never adopted as a rigorous set of specifications.
- The lower layers (physical, data link, network and transport) are for the most part adhered to.
  - Recall however that we have already mentioned a “brouter” which serves as both bridge (data link layer) and router (network layer).

# Classroom not boardroom



- Many vendors agreed at one time or another to adhere to OSI, however, the system proved to be too loosely defined, overly broad and the companies had much invested in their proprietary standards.
- Some products like X.400 (an e-mail system used mainly in Europe and Canada, an alternative to SMTP) adhere to OSI.
- But mainly OSI has become a teaching tool.

# OSI's Purpose

- Even if manufacturers do not restrict themselves to a particular OSI layer, they usually think of and describe their products in relationship to the OSI model.
- It provides a language and framework for discussion of networks, so it persists and continues to be taught.

# TCP/IP model

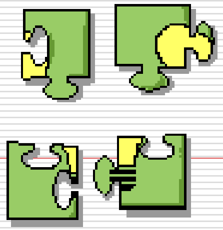
- Another model for understanding communications networks is the TCP/IP model.
- It also is broken into layers
  - It has no equivalent of the OSI model's physical layer.
  - The next three layers of the OSI model and the TCP/IP model are roughly equivalent.
  - The top three layers of the OSI model are combined into one layer in the TCP/IP model.

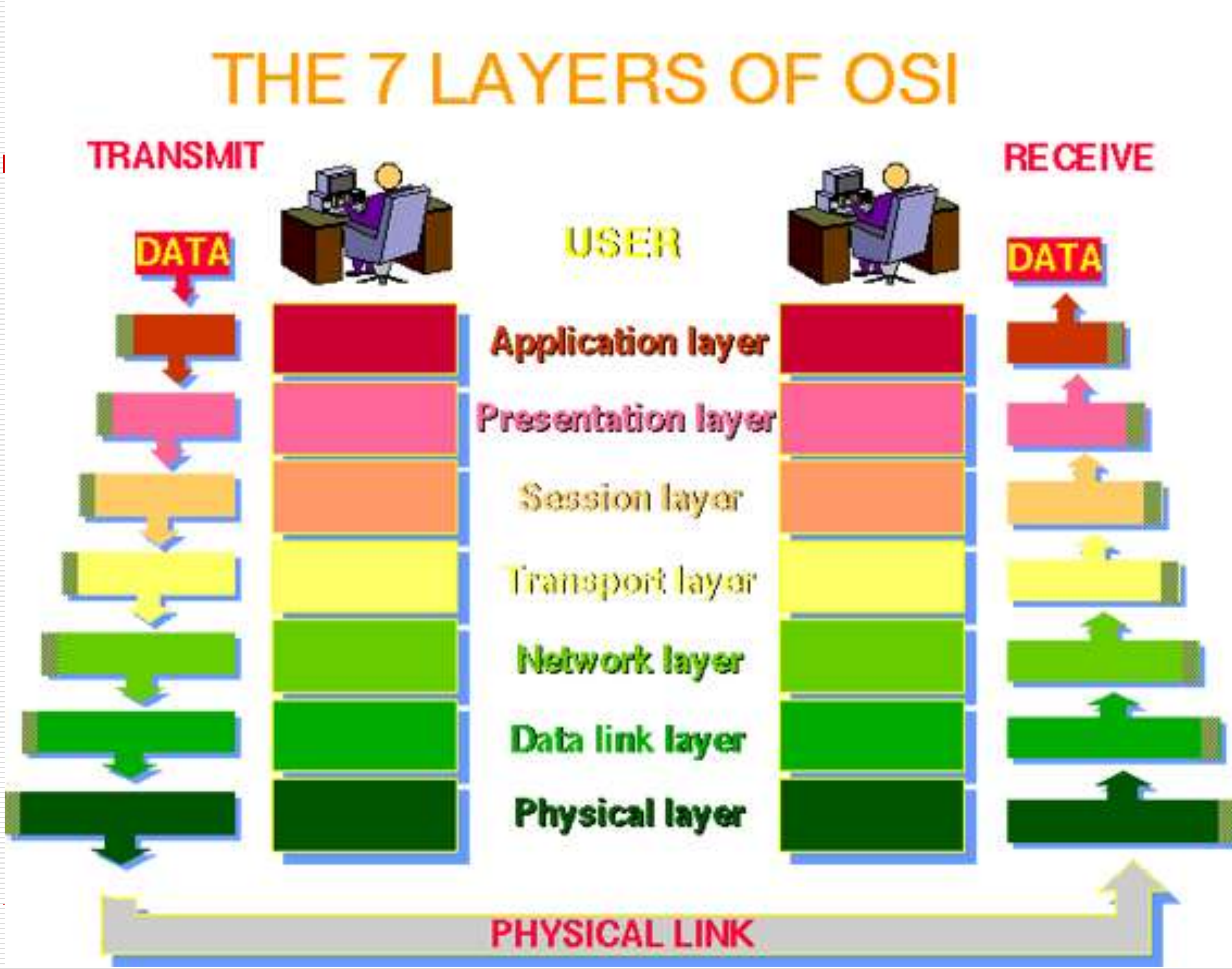
# The main idea

- The communication between computers can be broken down into layers.
  - Each layer is characterized by its functions and how it interfaces with the adjacent layers.
- Within the source's computer and starting at the user level, the communication is passed down through the layers to the lowest layer where it is sent through some transmission medium.

# The main idea (Cont.)

- The communication travels at the lowest layer (physical), occasionally rising up to the second layer (bridge) or third layer (router) until it reaches the destination.
- The communication now passes up through the layers.



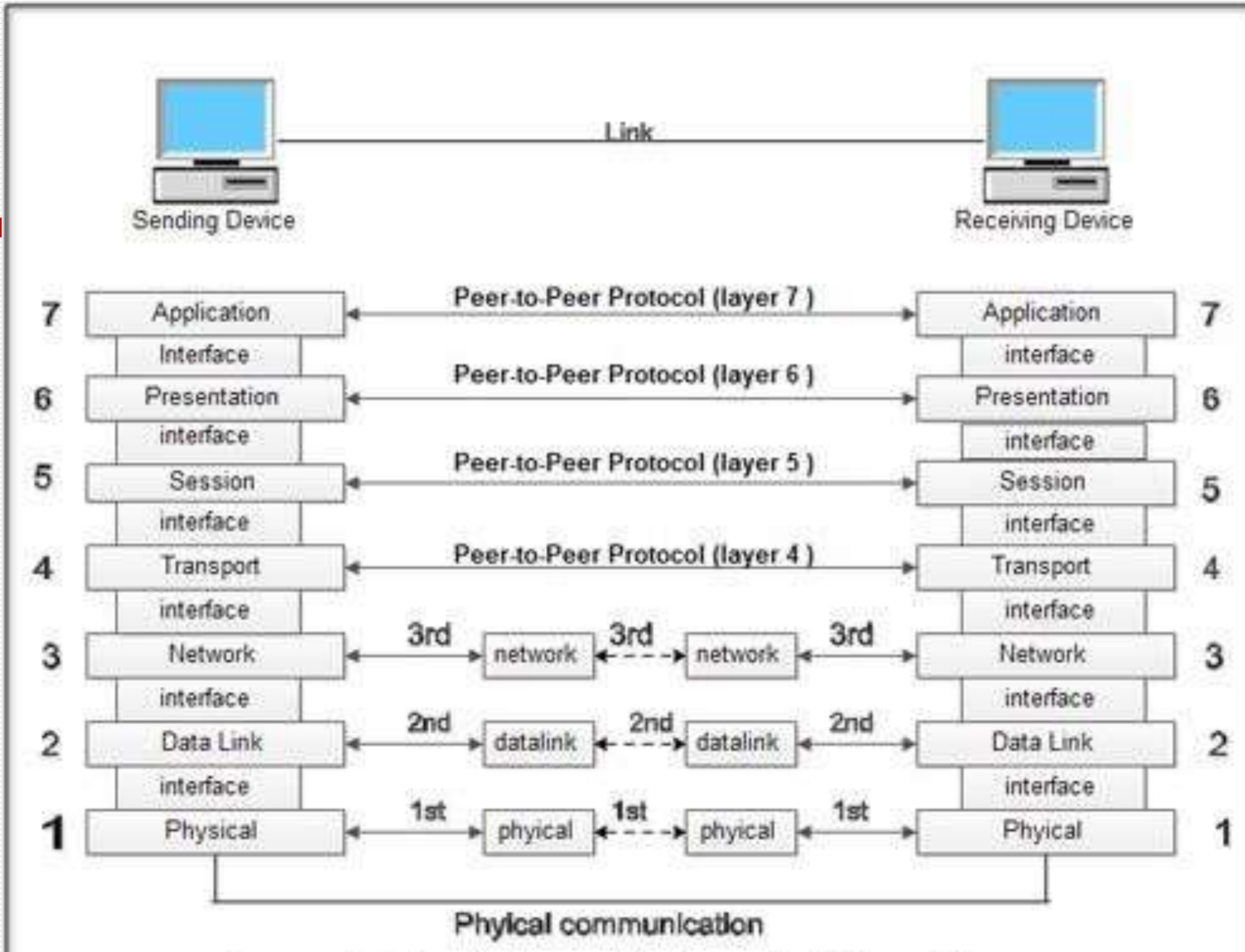

THE 7 LAYERS OF OSI
TRANSMIT
RECEIVE
DATA
USER
DATA
Application layer
Presentation layer
Session layer
Transport layer
Network layer
Data link layer
Physical layer
PHYSICAL LINK

# The main idea (Cont.)

- The layers are realized by a combination of the operating system, applications (for instance a browser), network protocols such as TCP/IP, and the software and hardware that actually place a physical signal onto a transmission line (NIC, modem, etc)
  - Don't confuse TCP/IP (the stack, model) with layer protocols TCP (layer 4) and IP (layer 3).

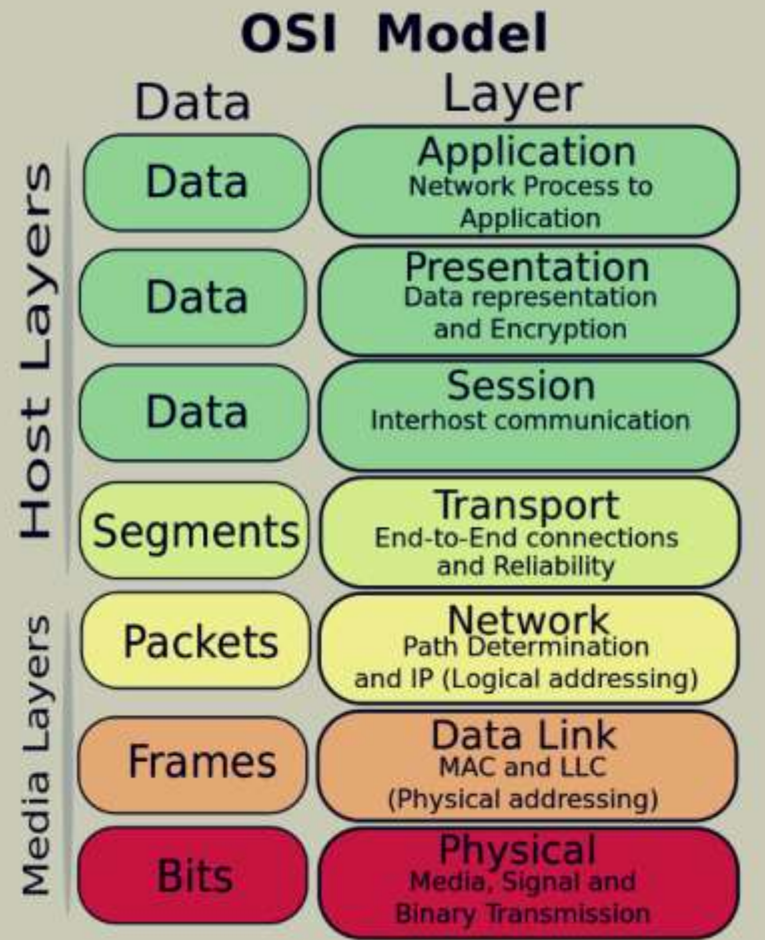
# OSI: Network / Computer

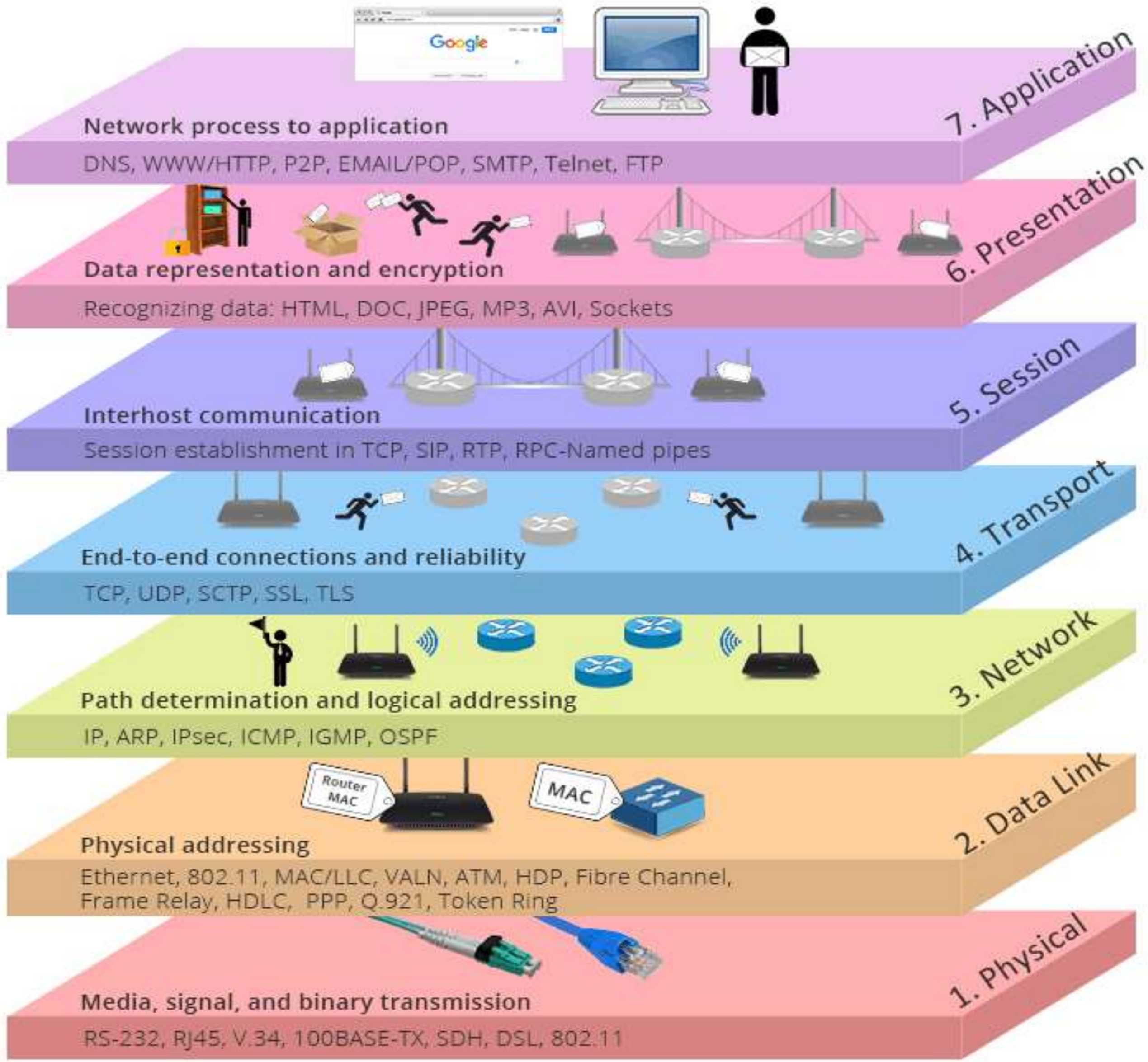
- The Transport layer can be viewed as the division between the computer and the network.
- The lowest three layers (physical, data link and network) deal with "a best effort" to get information to the desired computer.
- The Transport layer verifies the effort, and hides the network from the layers above.
- The remaining layers treat the data as if it was local.



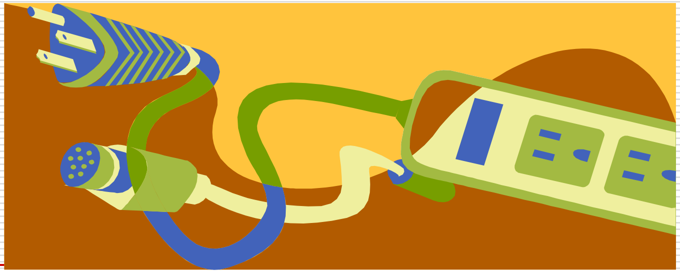

Communication between the layers in OSI model

# An Exchange Using the OSI Model




7. Application
Network process to application
DNS, WWW/HTTP, P2P, EMAIL/POP, SMTP, Telnet, FTP
6. Presentation
Data representation and encryption
Recognizing data: HTML, DOC, JPEG, MP3, AVI, Sockets
5. Session
Interhost communication
Session establishment in TCP, SIP, RTP, RPC-Named pipes
4. Transport
End-to-end connections and reliability
TCP, UDP, SCTP, SSL, TLS
3. Network
Path determination and logical addressing
IP, ARP, IPsec, ICMP, IGMP, OSPF
2. Data Link
Router MAC
MAC
Physical addressing
Ethernet, 802.11, MAC/LLC, VALN, ATM, HDP, Fibre Channel, Frame Relay, HDLC, PPP, Q.921, Token Ring
1. Physical
Media, signal, and binary transmission
RS-232, RJ45, V.34, 100BASE-TX, SDH, DSL, 802.11

# Layer 1: Physical

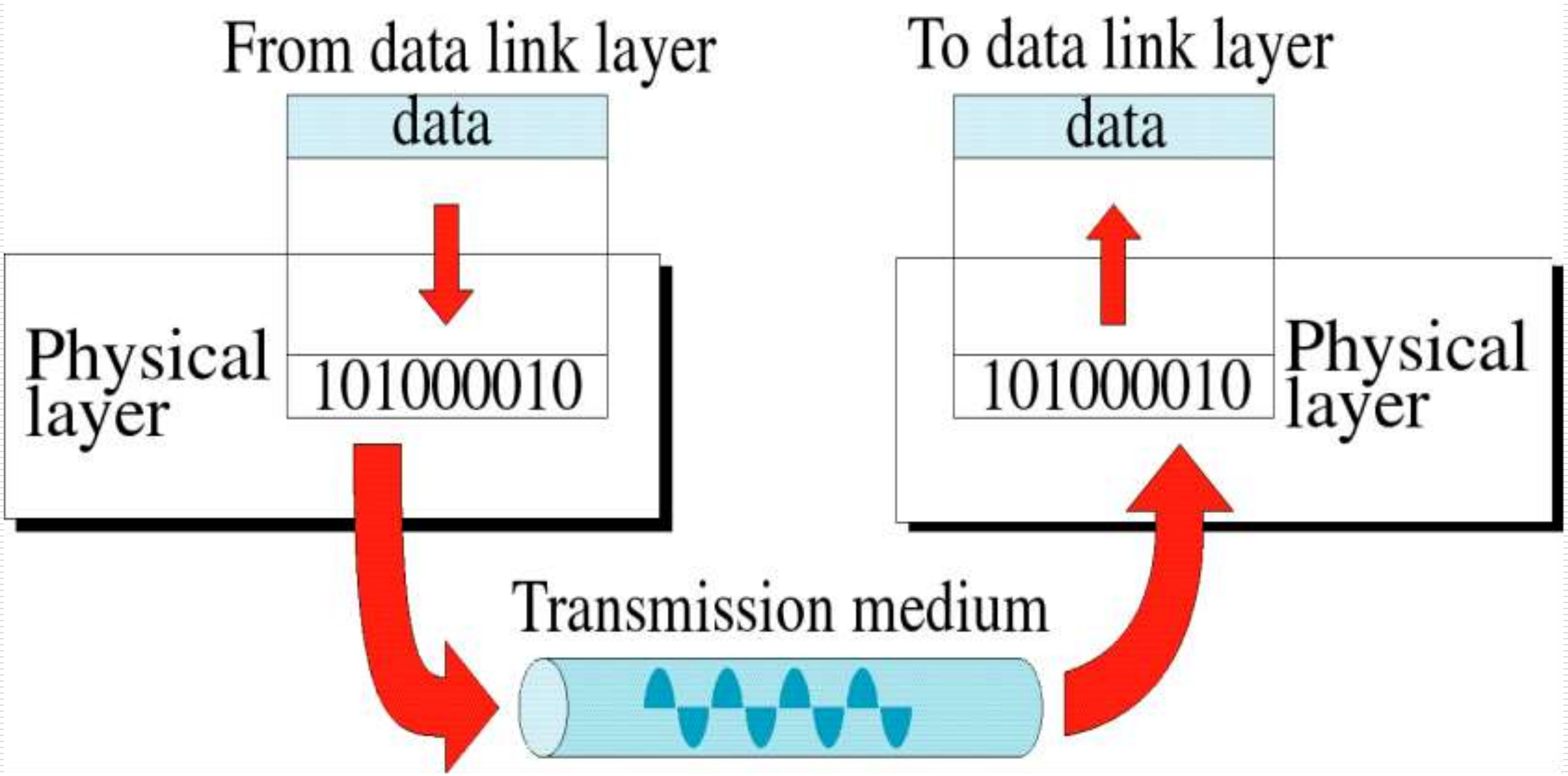


- There is no interpretation at this level, a stream of 1's and 0's are put into a form convenient for transmission.
  - Waves (with little regard for their information content) are sent and received.
- This level is the most hardware oriented. It includes specifications about
  - NIC card speeds
  - Types and lengths of cable
  - Voltage characteristics (range, level or edge)
  - Etc.

# Physical Layer (Cont.)

- The physical layer involves protocols for actual transmission
  - Ethernet
  - FDDI
  - RS232
  - ATM
- These protocols also involve the interface with the next higher layer.

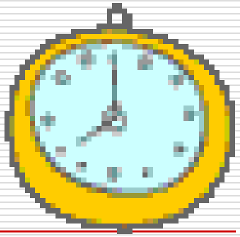
# Physical Layer (Cont.)



# Layer 2: The data-link layer

- At this layer one begins to consider bytes instead of just bits, one examines some of the information content of the signal (at least the address and some of the error detection sequencing)
- Recall that bridges operate at this level
  - They know where a packet is headed.
  - They know whether or not it has been involved in a collision.
  - Bit stuffing occurs at this level.

# Data Link (Cont.)

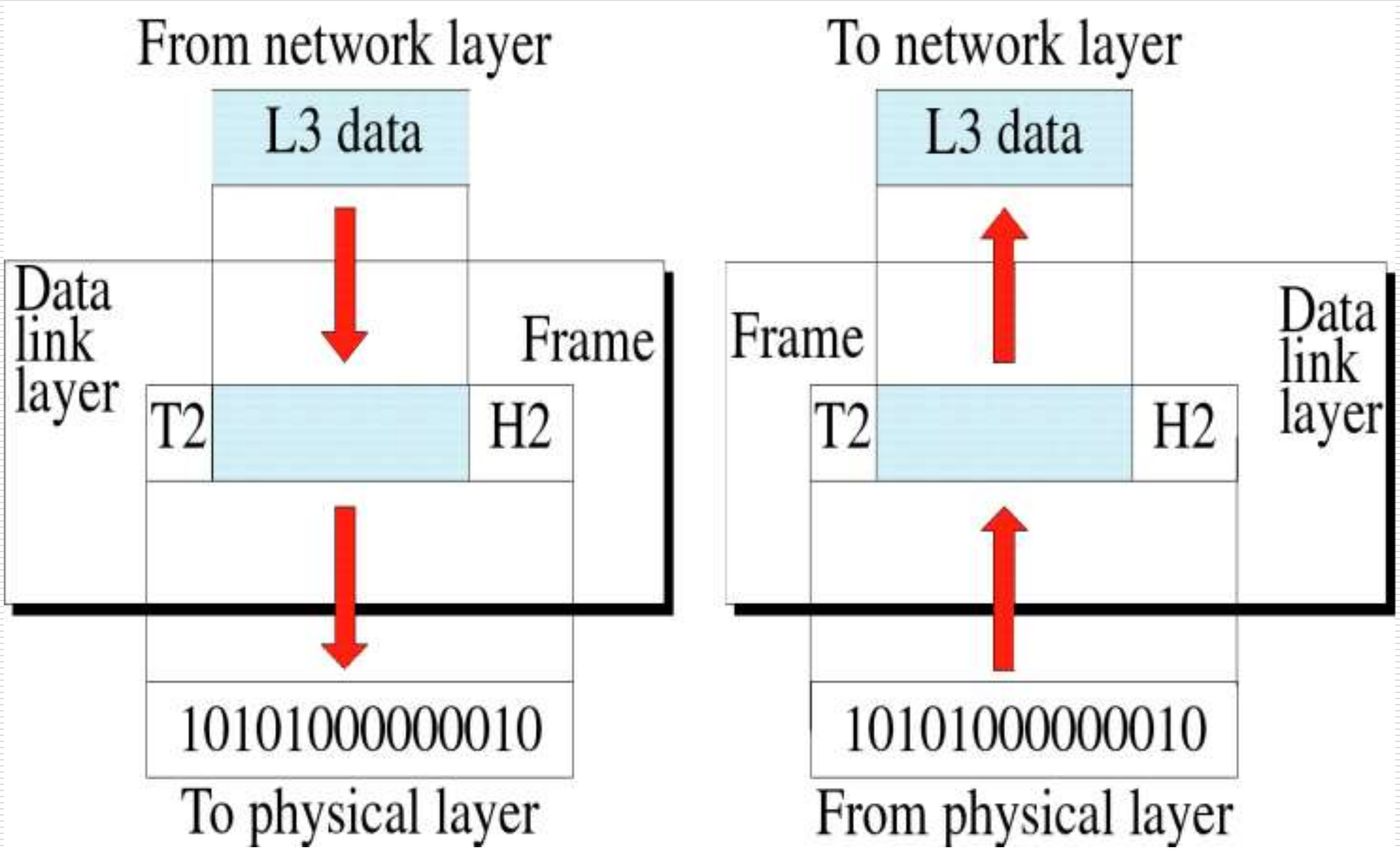


- Data packets are encoded and decoded into bits.
- It directs packets and handles errors from the physical layer.
- It handles synchronization (timing).
  - It must know where one bit ends and the next one begins.
  - It must know where one byte ends and the next one begins.

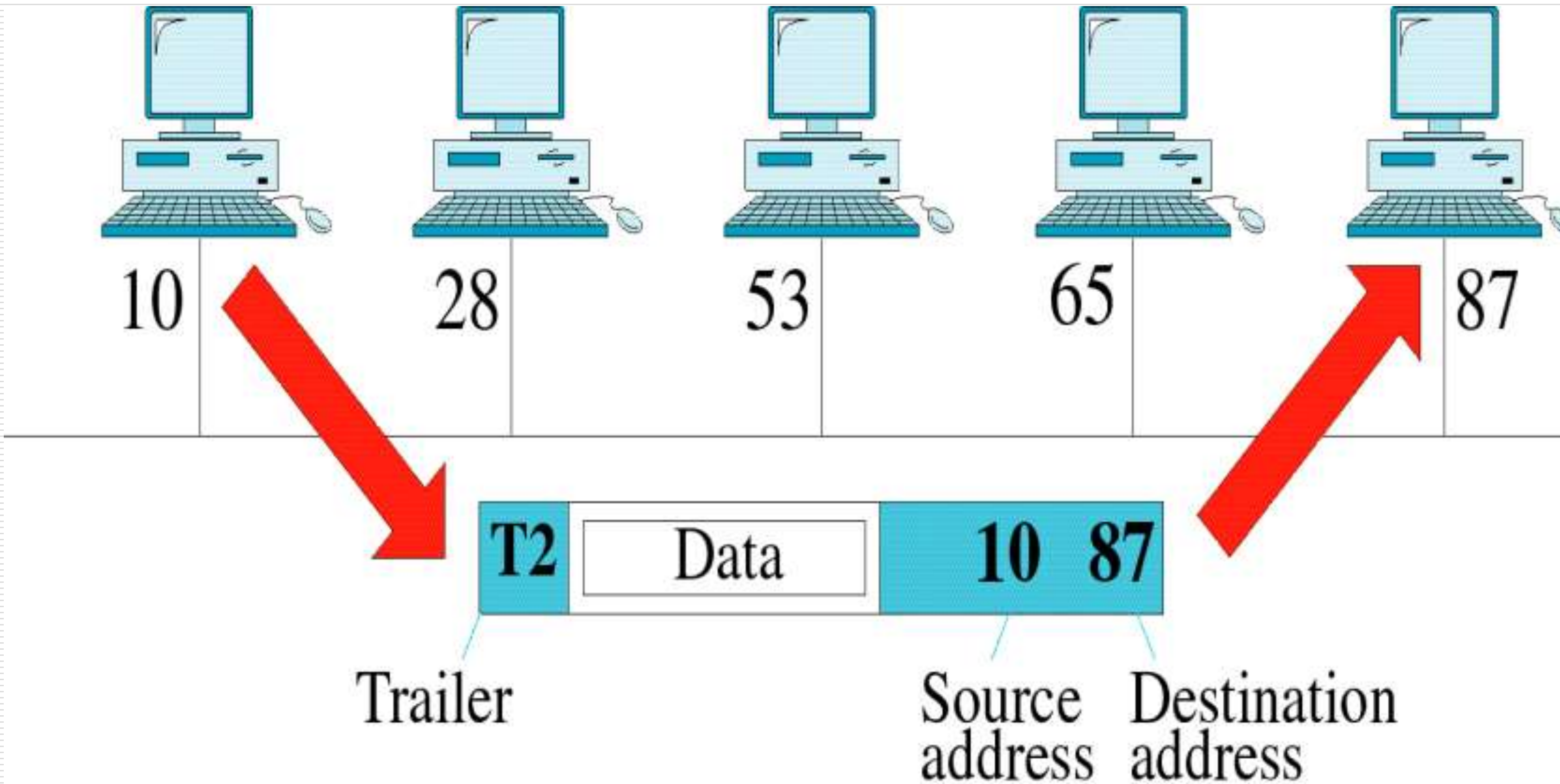
# Data link sublayers

- The data link layer is divided into two sublayers:
  - The **MAC** (Media Access Control) sublayer: takes the signal from or puts the signal onto the transmission line ("touches" physical layer).
  - The LLC (Logical Link Control) sublayer: starts to interpret the signal as data, includes timing (synchronization) and error checking.

# Data Link (Cont.)

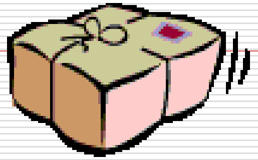


# Data Link (Cont.)



# Layer 3: The network layer

- The router acted at this layer.
- One of the main functions of the layer is routing. Store and forward are network layer functions.
- In a connection-oriented scheme, the virtual circuit is established at the network layer.
- Building the routing tables, troubleshooting the routing tables when there is a lot of traffic or if a connection goes down.
- The network layer also gathers related packets (packet sequencing).



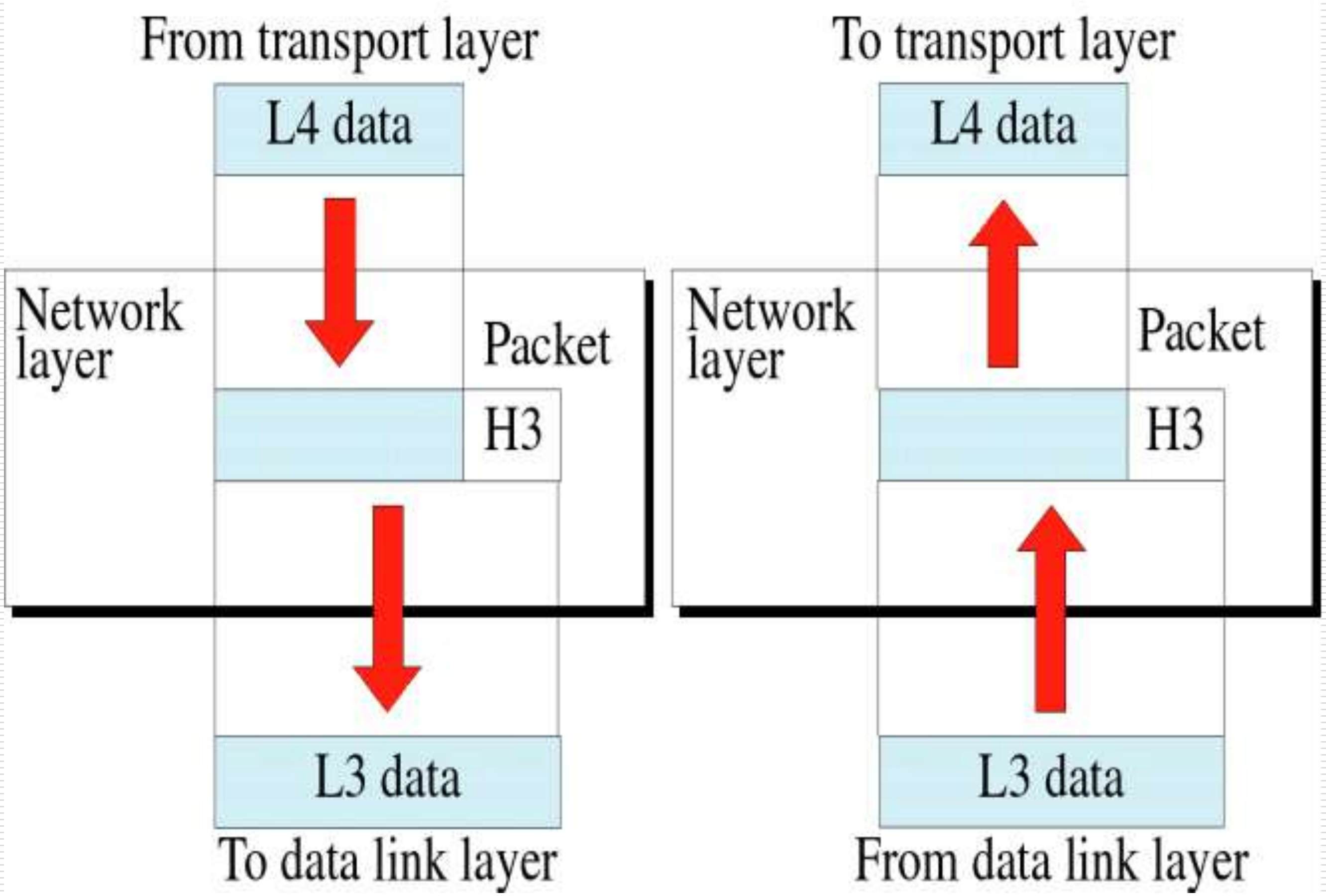
# Functions of Network Layer

- **Internetworking:** It provides Internetworking.

- **Logical Addressing:** When packet is sent outside the network, N/W layer adds Logical (network) address of the sender & receiver to each packet.
- Network addresses are assigned to local devices by n/w administrator and assigned dynamically by special server called DHCP (Dynamic Host Configuration Protocol)

- **Routing:** When independent n/w are connected to create internetwork several routes are available to send the data from S to D. These n/w are interconnected by routers & gateways that route the packet to final destination.

# The network layer



# Layer 4: The transport layer

- As stated before, Layer 4 is the dividing line between inter-computer transactions and intra-computer transactions.
- Layer 4 manages end-to-end verification.
  - The lower layers make a "best effort" but if data is lost so be it. Layer 4 must ensure that the information was received intact.
- It does a higher-order error-checking.
- The transfer should be "transparent." The higher layers do not know the data came from another computer.

# Whose job is it?

- At a node Layer 3 collects associated packets if one was dropped it may throw them all away.
- It is the responsibility of the source's Layer 4 to look for some acknowledgement that all packets arrived.  If no acknowledgment is received, it should retransmit.

# Functions of Transport Layer



- **Segmentation of message into packet & reassembly of packets into message.**
- **Port addressing:** Computers run several processes. TL header include a port address with each process.
- **Flow Control:** Flow control facility prevents the source form sending data packets faster than the destination can handle.
- **Error control**: TL ensures that the entire message arrives at the receiving TL without error.

**The transport layer is responsible for process-to-process delivery.**

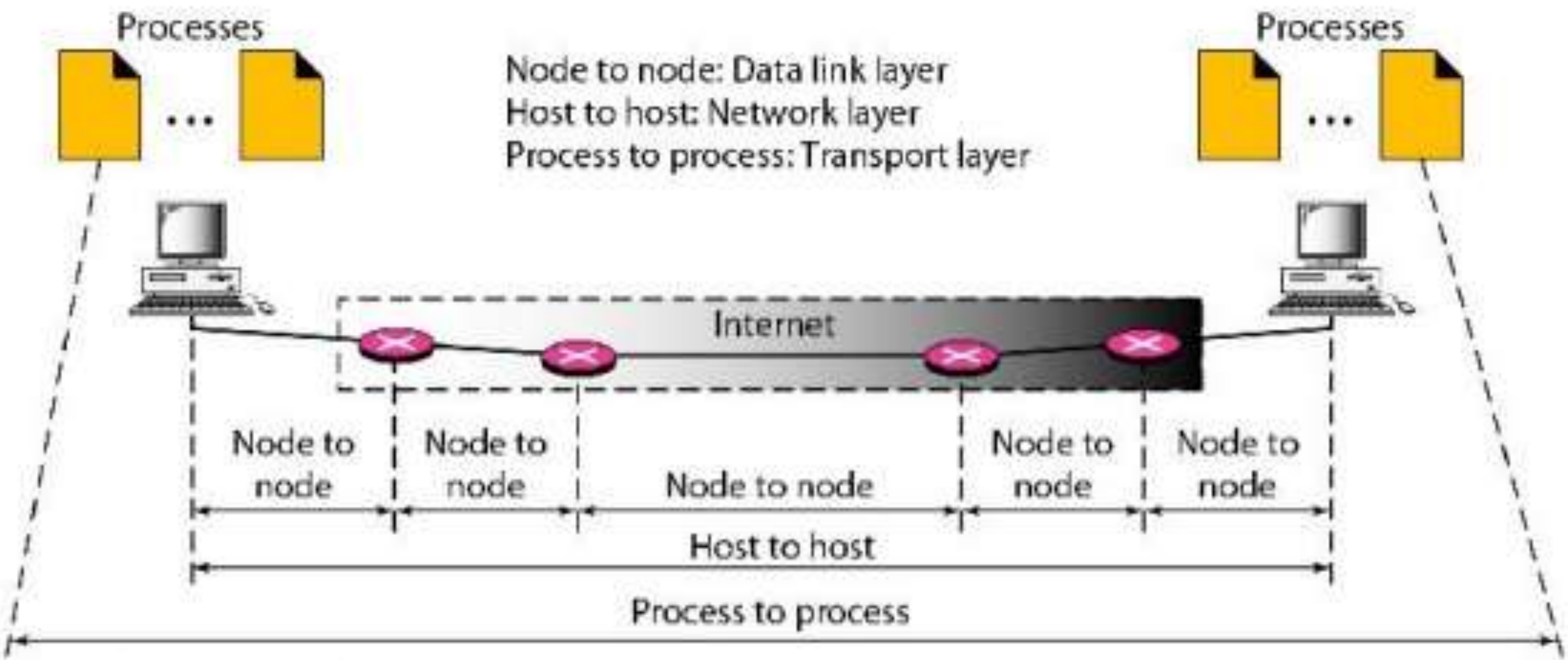
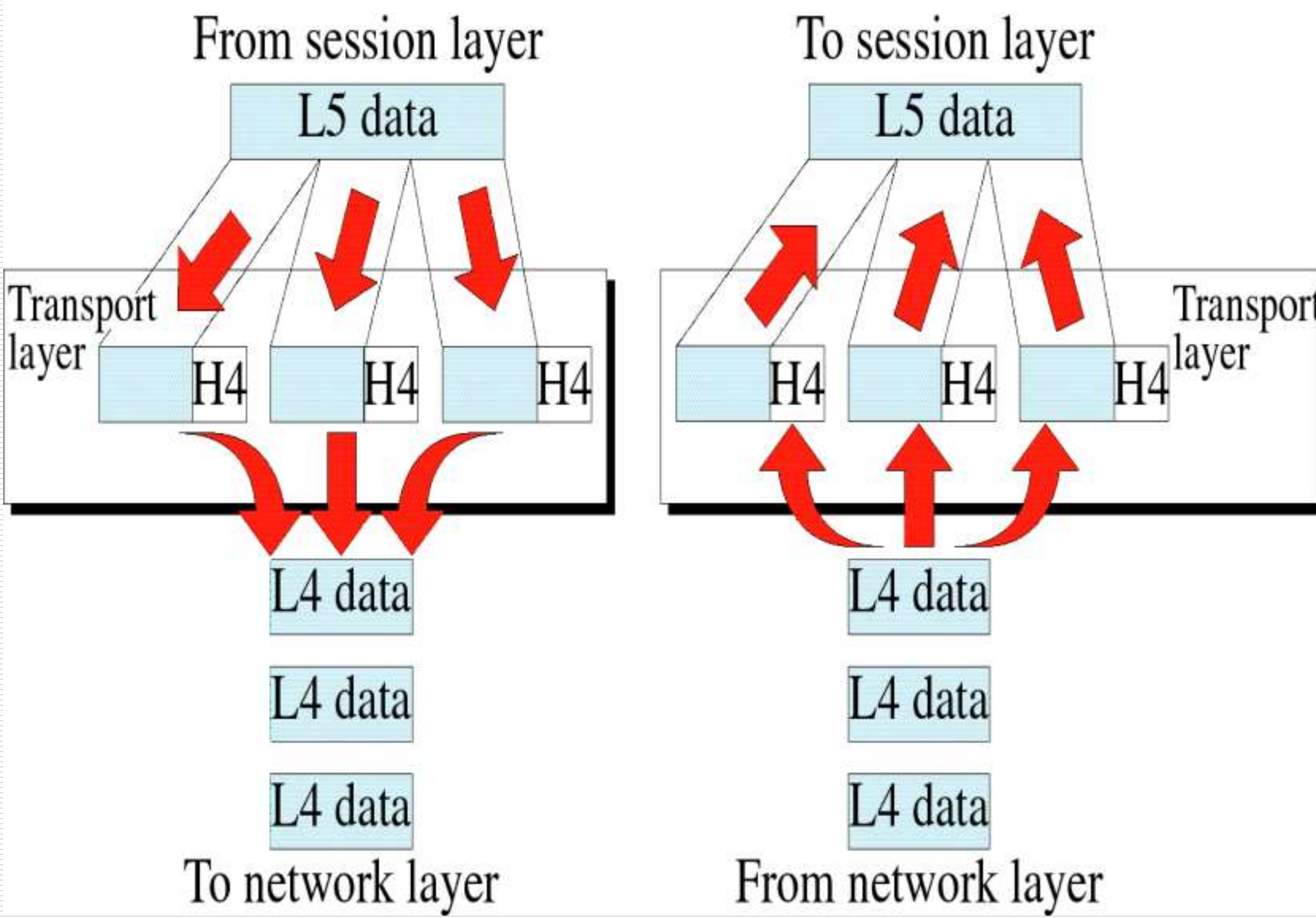


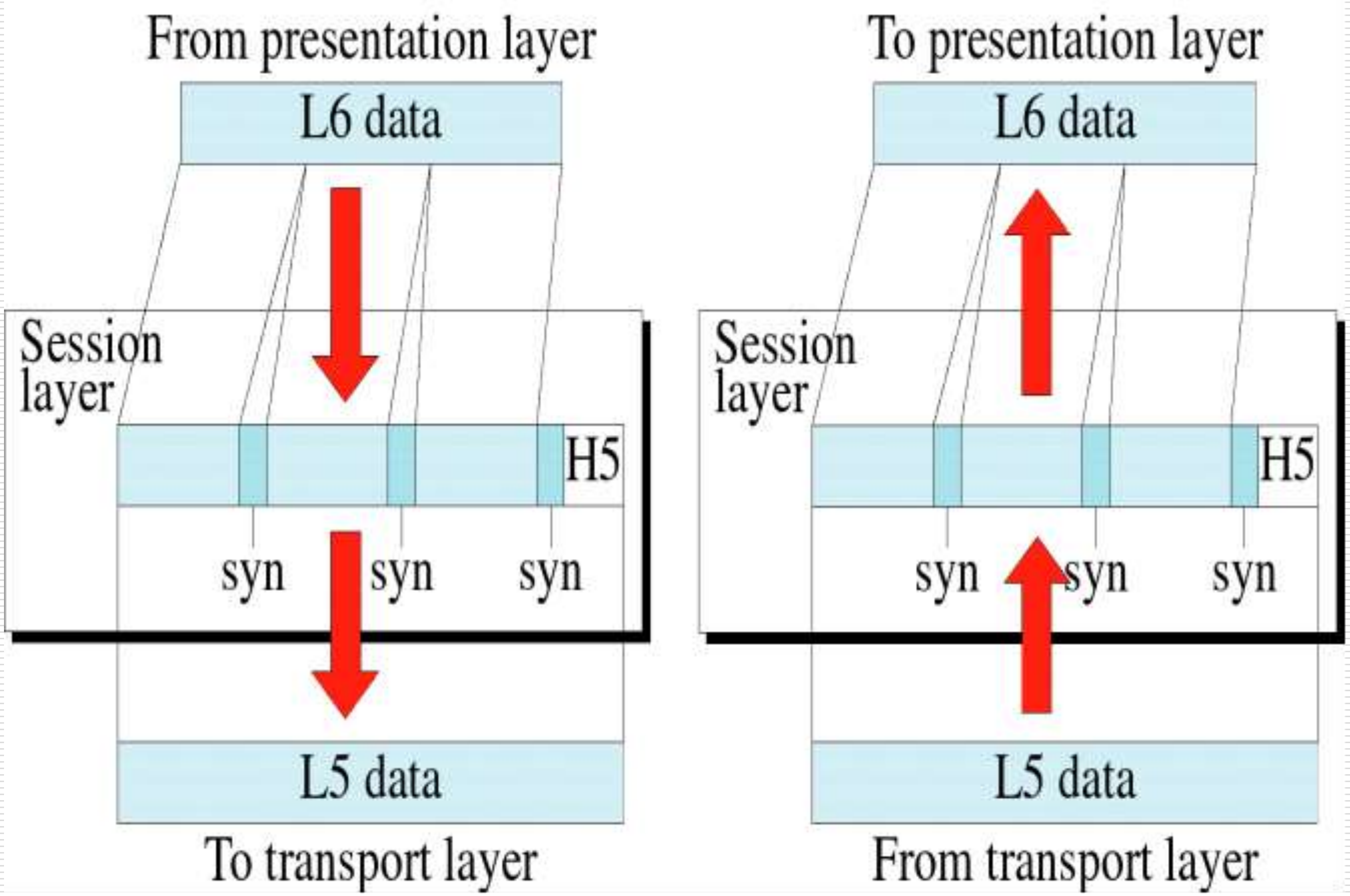

**Figure 4.1 Types of data deliveries**


From session layer
L5 data
Transport layer
H4
H4
H4
L4 data
L4 data
L4 data
To network layer
To session layer
L5 data
Transport layer
H4
H4
H4
L4 data
L4 data
L4 data
From network layer

# Layer 5: The session layer

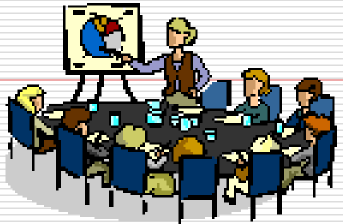
- Recall when discussing connection-oriented schemes, we mentioned the idea of a "session."
- It is an agreement between a source and destination to communicate.
- This layer establishes, manages and terminates sessions between applications (they could be on the same computer or on different computers).

# The session layer

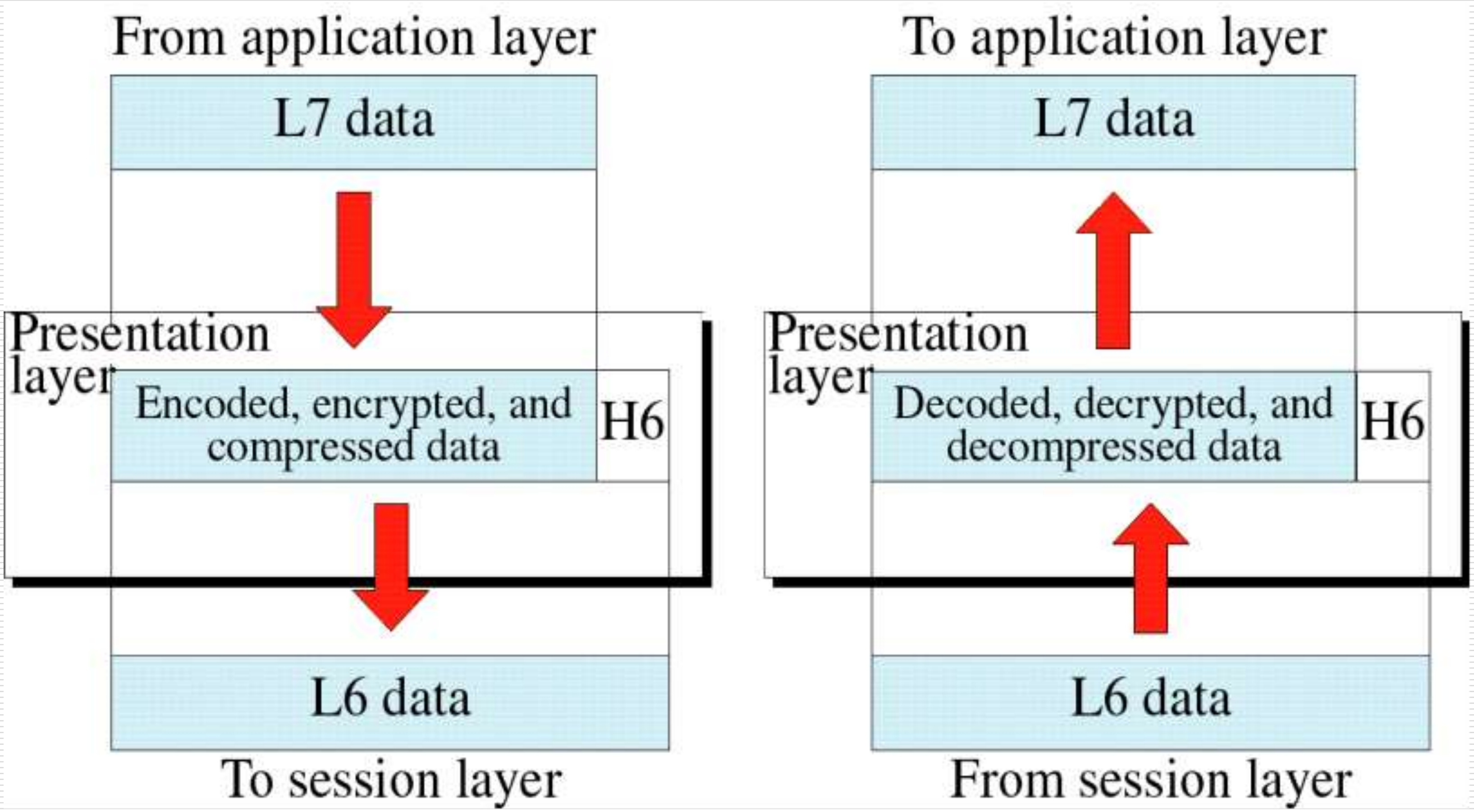


# Layer 6: The presentation layer

- This layer provides independence from differences in data representation (e.g., encryption) by translating from application to network format, and vice versa.
- The presentation layer works to transform data into the form that the application layer can accept. This layer formats and encrypts data to be sent across a network, providing freedom from compatibility problems. It is sometimes called the "syntax layer."



# The presentation layer



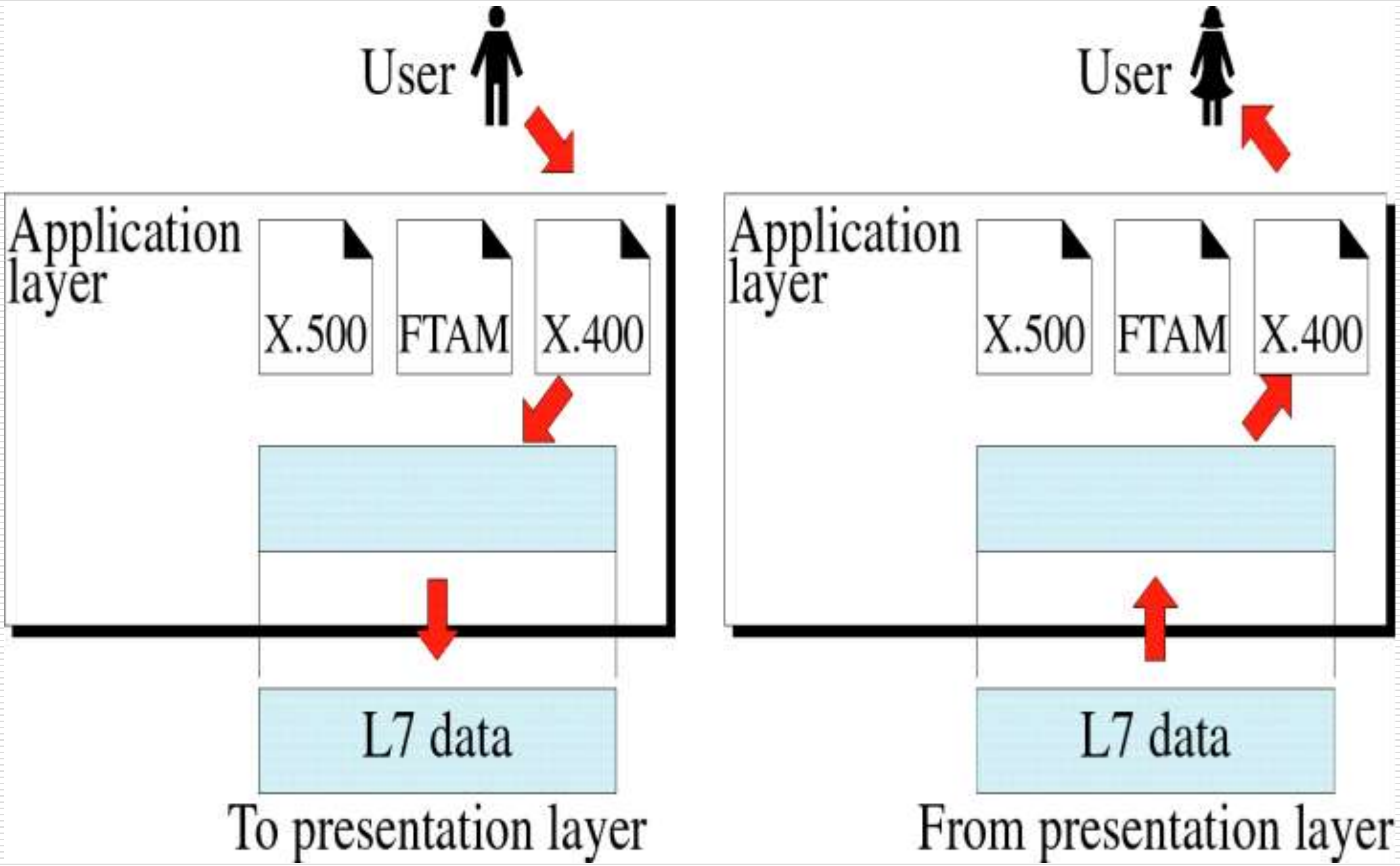
# Layer 7: The application layer

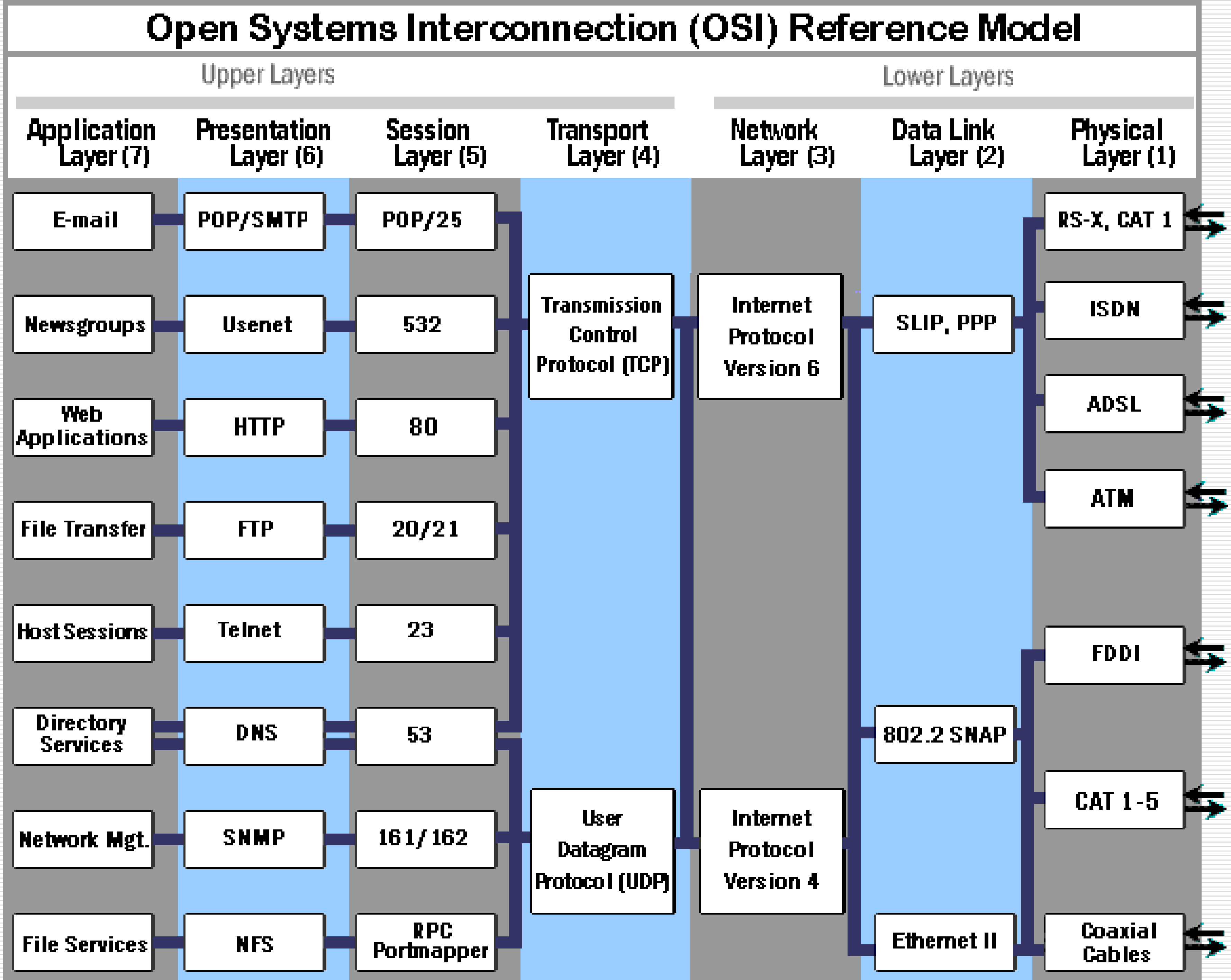
- This layer supports application and end-user processes.
- Communication partners are identified, quality of service is identified, user authentication and privacy are considered, and any constraints on data syntax are identified.

# Layer 7: The application layer

- Everything at this layer is application-specific. This layer provides application services for file transfers, e-mail, and other network software services. Telnet and FTP are applications that exist entirely in the application level.
- These are **not applications** (like Word and Excel) but **services for** such **applications**!

# The application layer




Open Systems Interconnection (OSI) Reference Model
Upper Layers
Lower Layers
Application Layer (7)
Presentation Layer (6)
Session Layer (5)
Transport Layer (4)
Network Layer (3)
Data Link Layer (2)
Physical Layer (1)
E-mail
POP/SMTP
POP/25
Newsgroups
Usenet
532
Web Applications
HTTP
80
File Transfer
FTP
20/21
Host Sessions
Telnet
23
Directory Services
DNS
53
Network Mgt.
SNMP
161/162
File Services
NFS
RPC Portmapper
Transmission Control Protocol (TCP)
User Datagram Protocol (UDP)
Internet Protocol Version 6
Internet Protocol Version 4
SLIP, PPP
802.2 SNAP
Ethernet II
RS-X, CAT 1
ISDN
ADSL
ATM
FDDI
CAT 1-5
Coaxial Cables

# OSI in real life

- For example, TCP/IP is usually packaged with other Internet programs as a suite of products that support communication over the Internet. This suite includes the File Transfer Protocol (FTP), Telnet, the Hypertext Transfer Protocol (HTTP), e-mail protocols, and sometimes others.
- Although TCP fits well into the Transport layer of OSI and IP into the Network layer, the other programs fit rather loosely (but not neatly within a layer) into the Session, Presentation, and Application layers.

# Layered Software

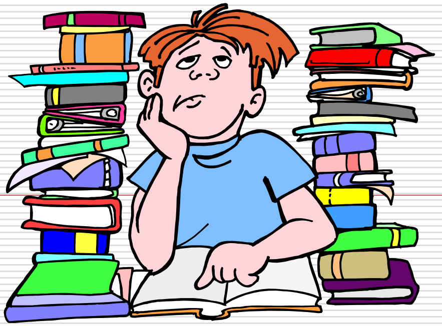
- In order to communicate on a network, each computer must have software that passes packets from layer to layer. Each layer only communicates with its neighbors.
- The entire package of software for moving through layers may be called a "suite" or a "stack."
  - "What stack are you running?"
- The most widely used communication stack is TCP/IP.

# Fig. 16.3: Stacks and Vendors

| Vendor | Stack |
| --- | --- |
| Novell Corporation | Netware |
| Banyan System Corporation | VINES |
| Apple Computer Corporation | AppleTalk |
| Digital Equipment Corporation | DECNET |
| IBM | SNA |
| (many vendors) | TCP/IP |

# Multiple protocol stacks

- One computer may communicate with different computers using different protocols across the SAME physical medium.
- The information that identifies which protocol is being used are contained within the frames.



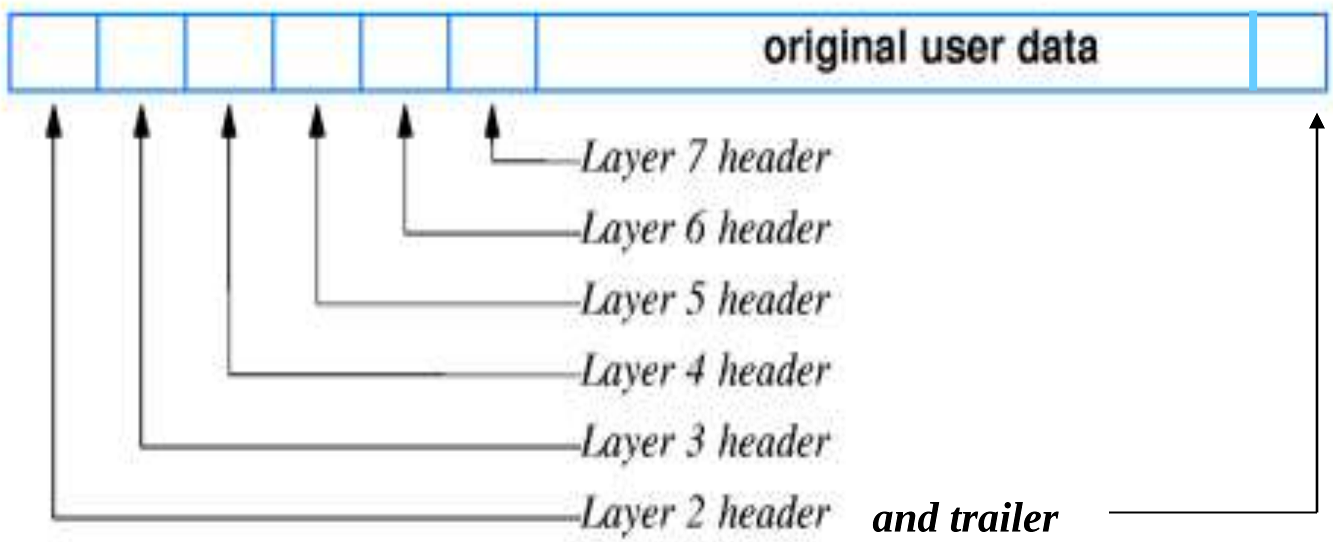
# Nested Headers

- As a source computer passes a message down through the protocol stack, each layer adds a **header**.
- The header added at a particular layer is intended for the corresponding layer on the destination computer.
- Thus the message passed down from Layer X to Layer Y on the source computer should be the same as the message passed up from Layer Y to Layer X on the destination computer.

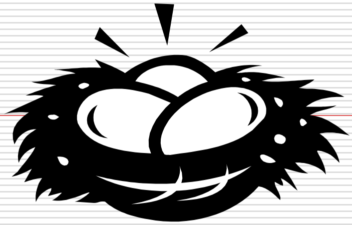
# Layer-to-Layer

- Because source Layer Y added the header that destination Layer Y strips off, this is effectively a Layer-Y-to-Layer-Y communication (even though it passes through other lower layers).
- For instance, the source and destination presentation layers have to negotiate the format of the data
  - Will it be text?   Will it be encrypted?
- There is no direct link between presentation layers, the information passes down through the physical layer; nevertheless, the two layers do communicate.

# Nested Headers



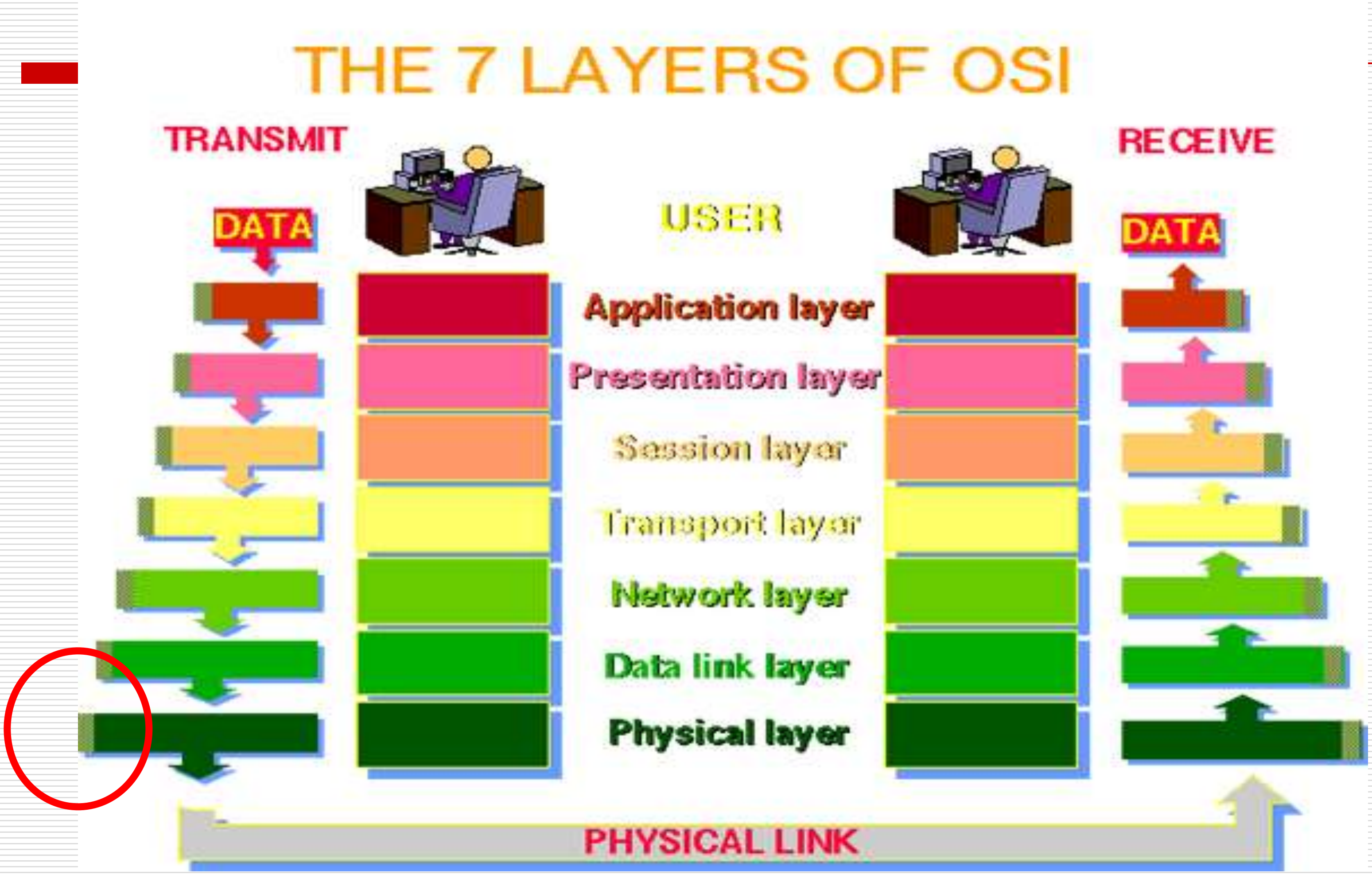

**Layer 2 also adds a trailer**



# No physical layer header

- The physical layer does not add a header.
- Headers contain information needed by the layers.
- The physical layer does not operate at an “information” level.

# What's wrong with this picture?



# The OSI Model - Drawbacks

- Late appearance (after widespread application of another models like TCP/IP)
- Heavy implementation
  - ignores less reliable but prompt connectionless services
  - multiplicate the layer functions throughout several layers
- Result: slow protocols

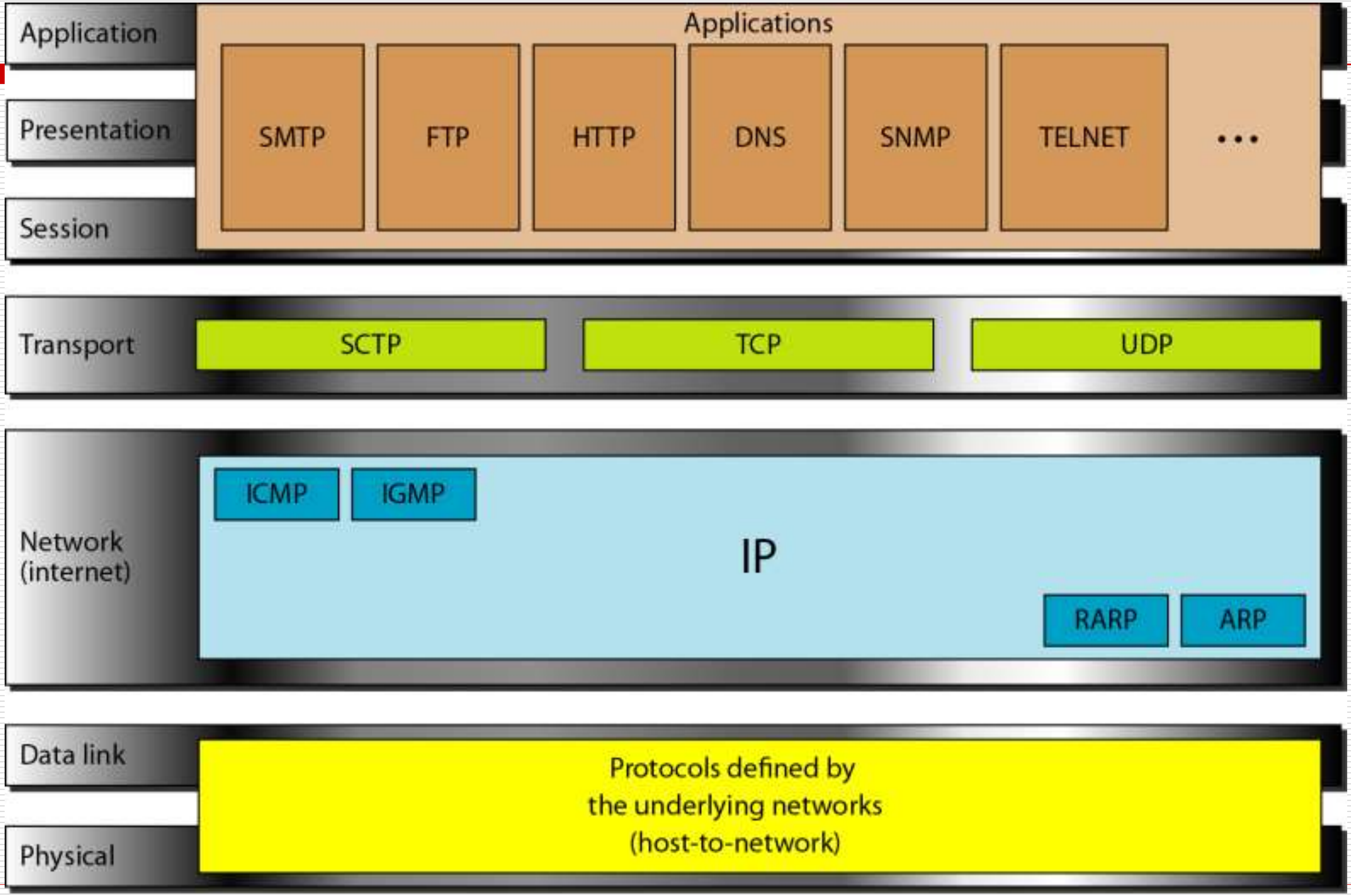
Table 18: OSI Reference Model Layer Summary

| Group | # | Layer Name | Key Responsibilities | Data Type Handled | Scope | Common Protocols and Technologies |
|---|---|---|---|---|---|---|
| Lower Layers | 1 | **Physical** | Encoding and Signaling; Physical Data Transmission; Hardware Specifications; Topology and Design | Bits | Electrical or light signals sent between local devices | (Physical layers of most of the technologies listed for the data link layer) |
| | 2 | **Data Link** | Logical Link Control; Media Access Control; Data Framing; Addressing; Error Detection and Handling; Defining Requirements of Physical Layer | Frames | Low-level data messages between local devices | IEEE 802.2 LLC, Ethernet Family; Token Ring; FDDI and CDDI; IEEE 802.11 (WLAN, Wi-Fi); HomePNA; HomeRF; ATM; SLIP and PPP |
| | 3 | **Network** | Logical Addressing; Routing; Datagram Encapsulation; Fragmentation and Reassembly; Error Handling and Diagnostics | Datagrams / Packets | Messages between local or remote devices | IP; IPv6; IP NAT; IPsec; Mobile IP; ICMP; IPX; DLC; PLP; Routing protocols such as RIP and BGP |
| | 4 | **Transport** | Process-Level Addressing; Multiplexing/Demultiplexing; Connections; Segmentation and Reassembly; Acknowledgments and Retransmissions; Flow Control | Datagrams / Segments | Communication between software processes | TCP and UDP; SPX; NetBEUI/NBF |
| Upper Layers | 5 | **Session** | Session Establishment, Management and Termination | Sessions | Sessions between local or remote devices | NetBIOS, Sockets, Named Pipes, RPC |
| | 6 | **Presentation** | Data Translation; Compression and Encryption | Encoded User Data | Application data representations | SSL; Shells and Redirectors; MIME |
| | 7 | **Application** | User Application Services | User Data | Application data | DNS; NFS; BOOTP; DHCP; SNMP; RMON; FTP; TFTP; SMTP; POP3; IMAP; NNTP; HTTP; Telnet |

# Reference Models - the TCP/IP Model

- Developed for ARPANET (70ties US national military network) and inherited in the Internet
- Features:
  - flexible routing - tolerant to loss of network nodes, subnets, route[r]s, connections, etc.
  - flexible architecture - tolerant to different throughput and application services (off-line, on-line, real-time)
  - 4-layer structure

*TCP/IP and OSI model*



# Protocol stacks

## TCP/IP stack

- Internet networks are based on TCP/IP protocols, so the TCP/IP model and protocol stack have a growing importance.
- TCP/IP is based on **five** protocol layers instead of seven. The OSI model session and presentation layers can be considered empty in TCP/IP context.
- TCP/IP stack with example protocols is shown below:

| | | |
|---|---|---|
| *7* | Application | Telnet, FTP, SMTP, SNMP, HTTP |
| *4* | Transport | TCP, UDP |
| *3* | Network | IP |
| *2* | Data link | HDLC or LAN frames |
| *1* | Physical | Voltage levels |

# The TCP/IP Model - The "Host/Network Layer"

- Corresponds to OSI Physical+Data Link Layers
- Unspecified strictly as protocol
- implementations vary in different networks and even hosts
- only restriction: serving upper (internet) layer in transmission of data packets

# The TCP/IP Model - The Internet Layer

- Connectionless layer (in order to provide the flexibility needed)
- Implementation: **IP**
- free independent exchange of packets (IP datagrams) transparently to the sender and receiver
  ➲**routing** is a key issue in IP
- standard **packet format** (strictly supported) for proper routing
- corresponds to OSI Network Layer

# The TCP/IP Model - The Transport Layer

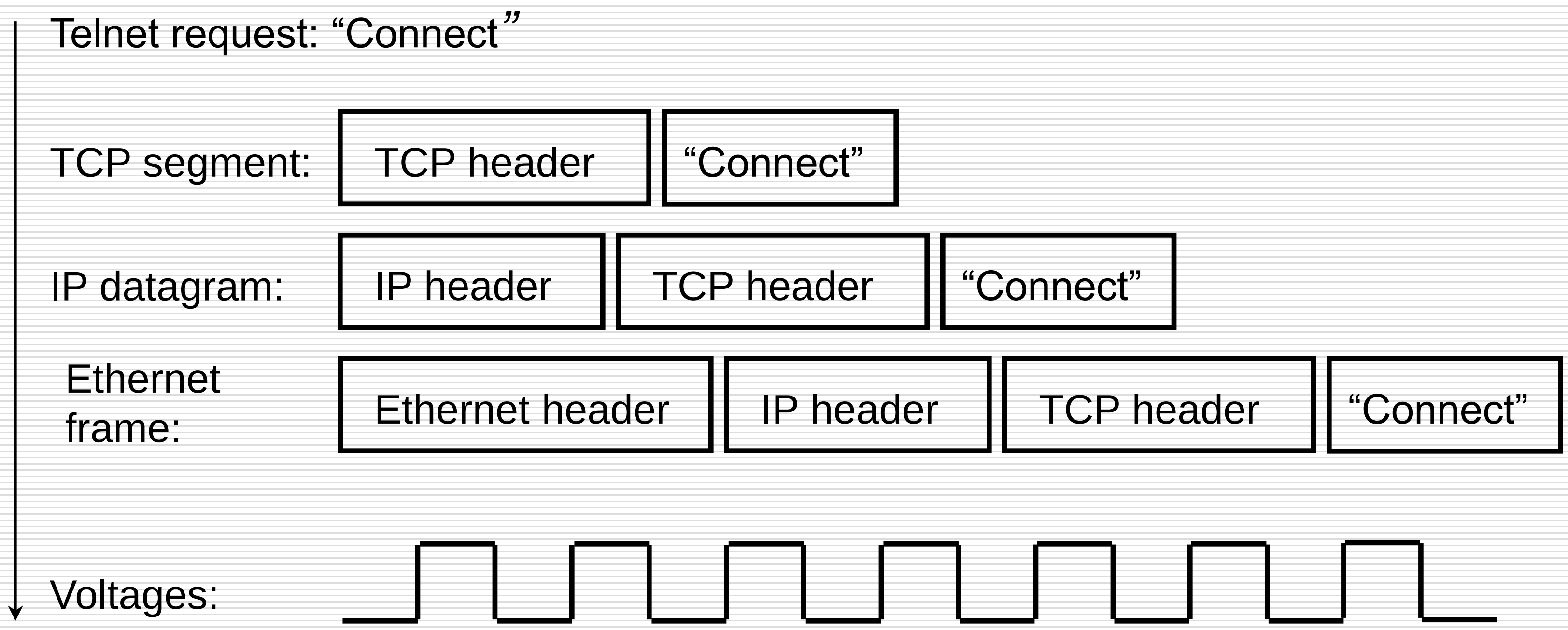
- Supports "point-to-point" connectivity between the source and destination (like OSI transport layer)
- Implemented by two protocols:
  - **TCP** (Transmission Control Protocol) - connection oriented, delivers the byte stream from source to destination by fragmentation into discrete messages for transmission by IP. Receiving TCP assembles the incoming messages to output stream
  - **UDP** (User Datagram Protocol) - connectionless, unreliable, non-sequential, for prompt delivery (multimedia applications)

# The TCP/IP Model - The Application Layer

- Top level protocols (session and presentation layer functions are performed by the application when needed) like:
  - TELNET
  - FTP
  - SMTP
  - DNS
  - HTTP
  - ………

# Service to protocol -mapping

Encapsulation between protocol layers



# The TCP/IP Model - Drawbacks

- Tightly specified, non-general model, oriented to the suspected set of protocols
- The lowest "host/network" layer is practically unspecified and this makes difficulties applying new communication media and technologies
- freeware protocols:
  - wide application but bad documentation,
  - bad quality of some and
  - security problems (big possibilities for hackers)

# Reference Models - OSI vs. TCP/IP

- Similarities:
  - **structure**: stack of protocols
  - **functionality**: routing + point-to-point connectivity + application supporting functions
- Dissimilarities (OSI)/(TCP):
  - conceptuality/applicability
  - hidden, transparent, replaceable protocols / conservative, non-conceptual approach
  - mostly connection oriented / pure connectionless oriented
  - 7 layers / 4 layers