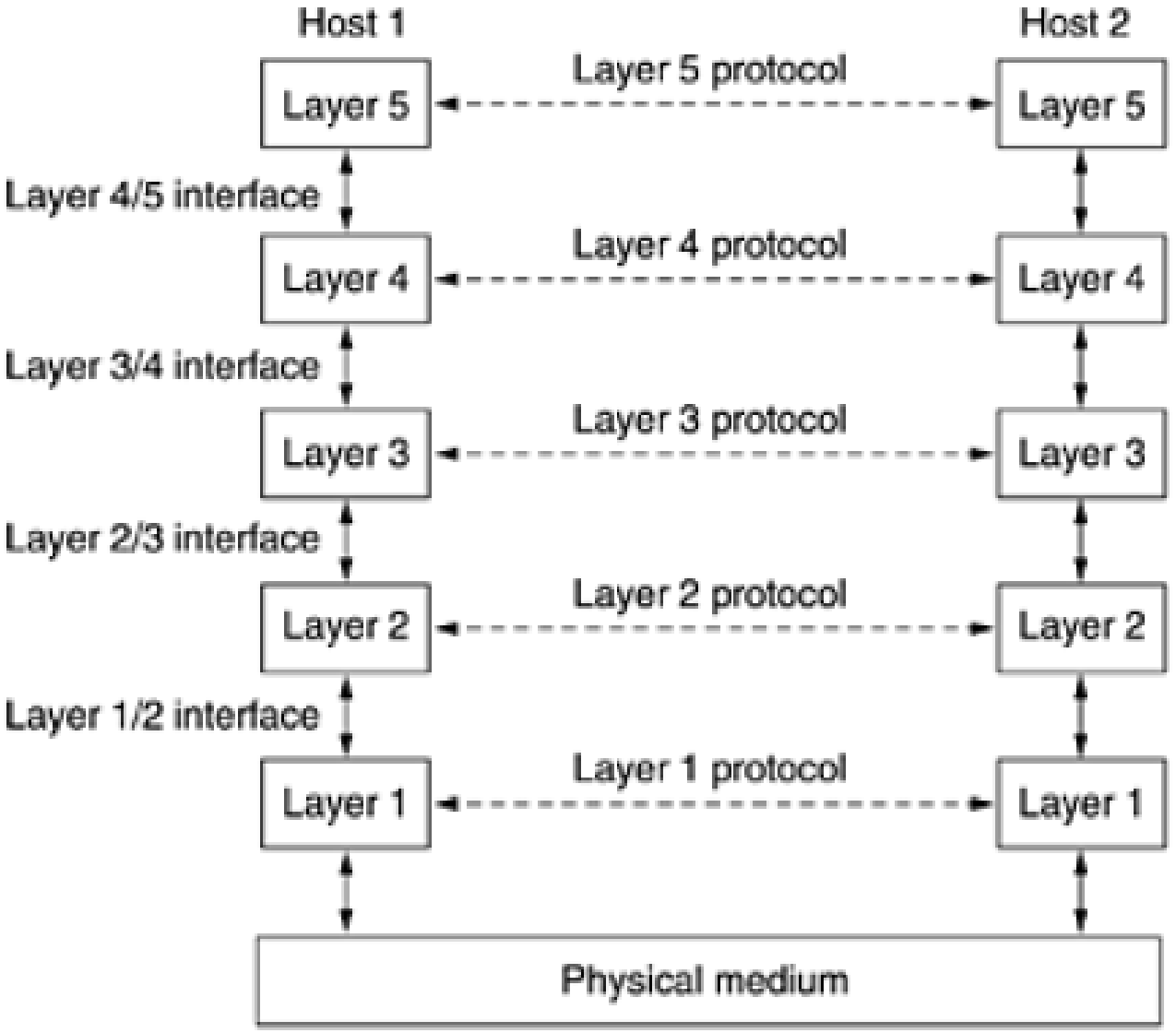
NETWORK SOFTWARE

Radhika Rani Chintala

- Protocol Hierarchies
- Design Issues for the Layers
- Connection-oriented Vs Connectionless Service
- Service Primitives
- Relationship of Services to Protocols
- OSI Layered Architecture
- TCP/IP Layered Architecture

LAYERED ARCHITECTURE

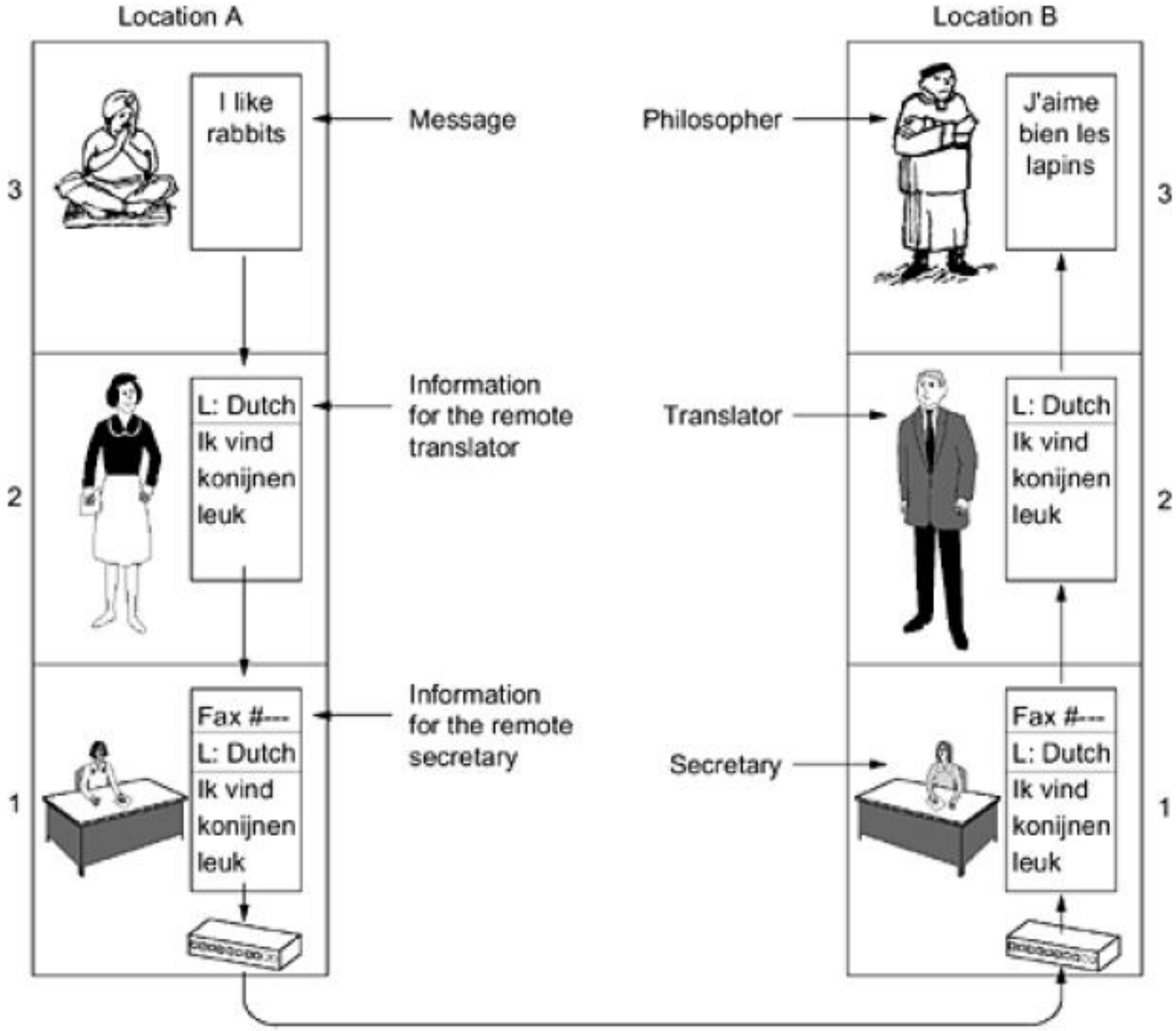
Radhika Rani Chintala

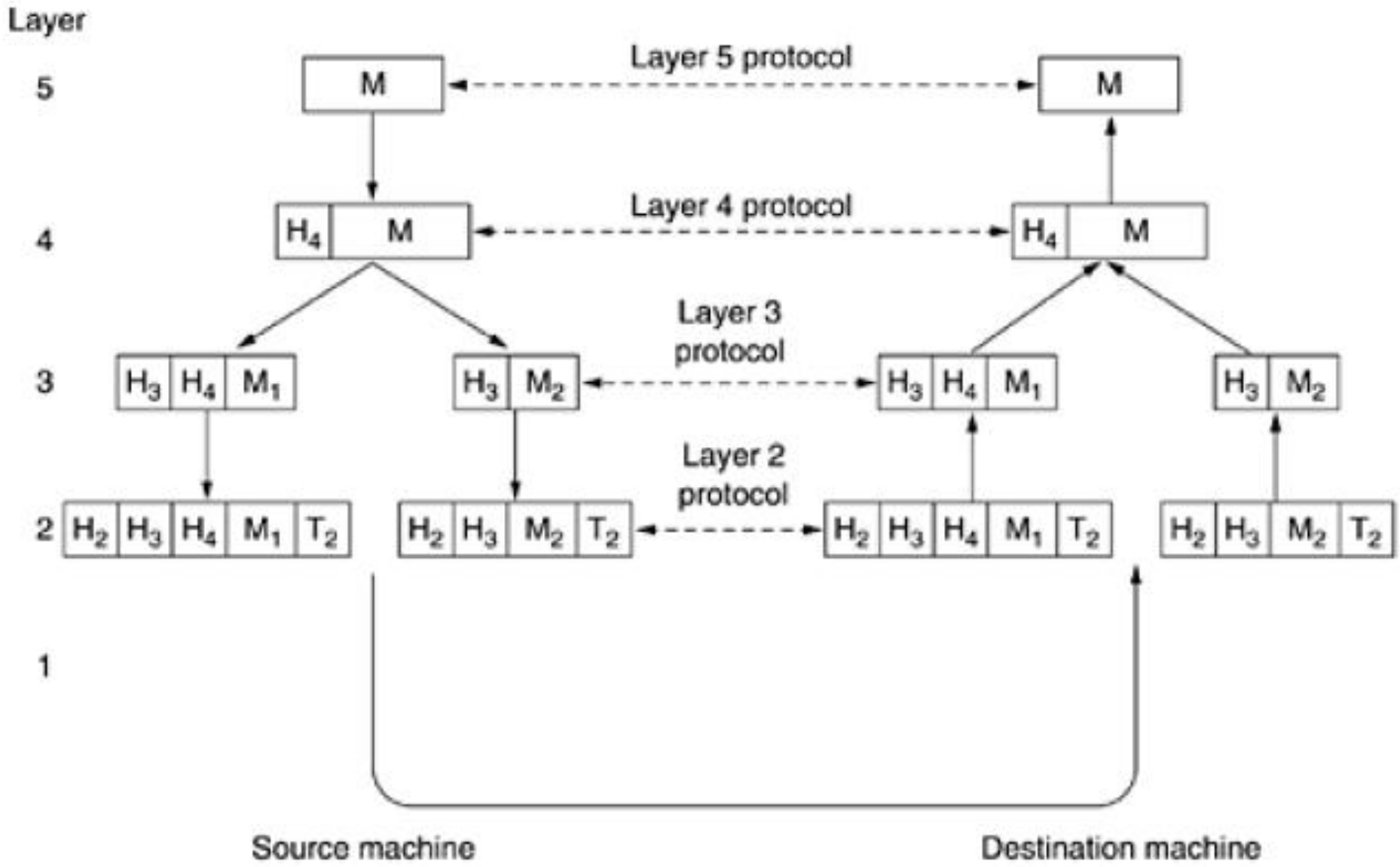


Protocol Hierarchies

- To reduce design complexity, networks are organized as a stack of layers, each one built upon the one below it.
- The no. of layers, the name, content, and the function of each layer differ from network to network.
- The purpose of each layer is to offer certain services to the higher layers.
- Layer n on one machine carries on a conversation with layer n on another machine.
- The rules and conventions used in this conversation are collectively known as the layer n protocol.
- **Protocol:** An agreement between the communicating parties on how communication is to proceed.
- Violating the protocol will make communication more difficult, if not completely impossible.
- The entities comprising the corresponding layers on different machines are called peers.
- In reality, no data are directly transferred from layer n on one machine to layer n on another machine.
- **Interface** defines which primitive operations and services the lower layer makes available to the upper one.

The Philosopher-Translator-Secretary Architecture




Layer
5
4
3
2
1
M
M
Layer 5 protocol
H4 M
H4 M
Layer 4 protocol
H3 H4 M1
H3 M2
H3 H4 M1
H3 M2
Layer 3 protocol
H2 H3 H4 M1 T2
H2 H3 M2 T2
H2 H3 H4 M1 T2
H2 H3 M2 T2
Layer 2 protocol
Source machine
Destination machine

Design Issues for the Layers

- Every layer needs a mechanism for identifying senders and receivers **(Addressing or naming)**.
- The rules for data transfer **(Protocols)**.
- **Error control** is an important issue because physical communication circuits are not perfect.
- How to keep a fast sender from swamping a slow receiver with data **(Flow Control)**.
- Another problem that must be solved at several levels is the inability of all processes to accept arbitrarily long messages **(disassembling, transmitting, and then reassembling messages)**.
- When there are multiple paths between source and destination, a route must be chosen **(Routing)**.
- Designs that continue to work well when the network gets large are said to be scalable.
- The last major design issue is to secure the network by defending it against different kinds of threats **(Security)**.

Connection-oriented Vs Connectionless Service

- Layers can offer two different types of service to the layers above them : **Connection-oriented, Connectionless**
- **Connection-oriented** service is modeled after the telephone system.
- To use a connection-oriented network service, the service user first establishes a connection, uses the connection, and then releases the connection.
- In some cases when a connection is established, the sender, receiver, and subnet conduct a **negotiation** about the parameters to be used, such as maximum message size, quality of service required, and other issues.
- Typically, one side makes a proposal and the other side can accept it, reject it, or make a counterproposal.

Connection-oriented Service

- **Connection-oriented** service is modeled after the telephone system.
- To use a connection-oriented network service, the service user:
  - first establishes a connection,
  - uses the connection,
  - releases the connection.
- In some cases when a connection is established, the sender, receiver, and subnet conduct a **negotiation** about the parameters to be used such as:
  - maximum message size
  - quality of service required,
  - other issues.
- Typically, one side makes a proposal and the other side can accept it, reject it, or make a counterproposal.
- The order of messages are followed.

Connectionless Service

- Connectionless service is modeled after the postal system.
- Each message (letter) carries the full destination address, and each one is routed through the intermediate nodes inside the system.
- There are different names for messages in different contexts; a packet is a message at the network layer.
- **Store-and-forward switching:** Intermediate nodes receive a message in full before sending it on to the next node.
- **Cut-through switching:** Transmission of a message at a node starts before it is completely received by the node.
- Chances of unordered messages.

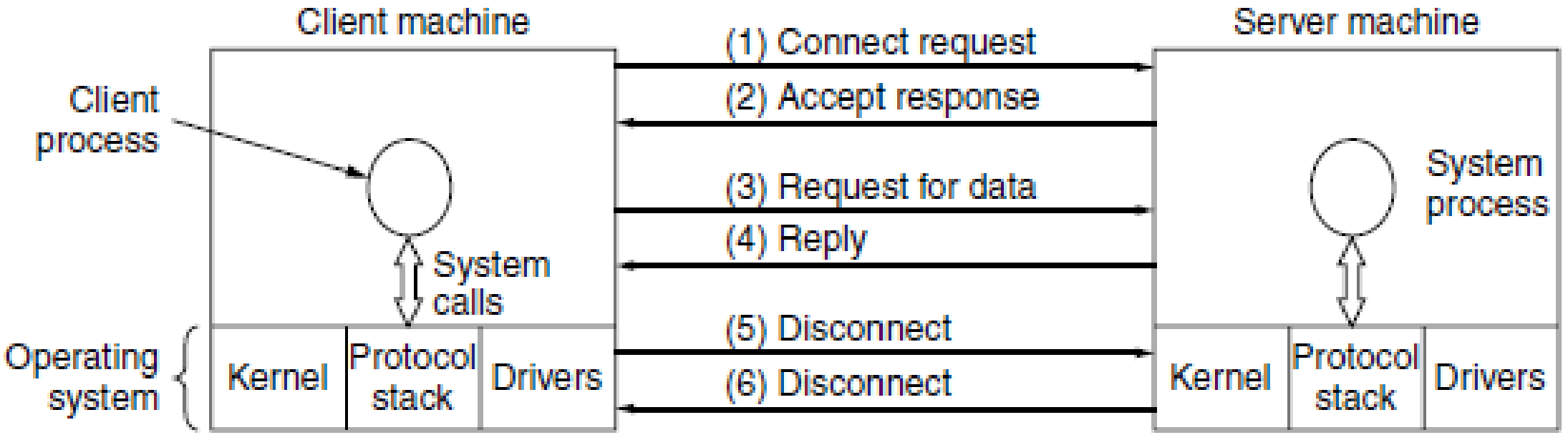
Service Primitives

- A **Service** is formally specified by a **set of primitives** (operations) available to user processes to access the service.
- These primitives tell the service to perform some action or report on an action taken by a peer entity.
- The protocol stack is located in the operating system and the primitives are normally system calls.
- The set of primitives available depends on the nature of the service being provided.
- The primitives for connection-oriented service are different from those of connectionless service.

Service Primitives for Connection-oriented service

Primitive	Meaning
LISTEN	Block waiting for an incoming connection
CONNECT	Establish a connection with a waiting peer
ACCEPT	Accept an incoming connection from a peer
RECEIVE	Block waiting for an incoming message
SEND	Send a message to the peer
DISCONNECT	Terminate a connection

- These primitives might be used for a request-reply interaction in a client-server environment.


Client machine
Server machine
Client process
System calls
System process
(1) Connect request
(2) Accept response
(3) Request for data
(4) Reply
(5) Disconnect
(6) Disconnect
Operating system
Kernel
Protocol stack
Drivers
Kernel
Protocol stack
Drivers

The Relationship of Services to Protocols

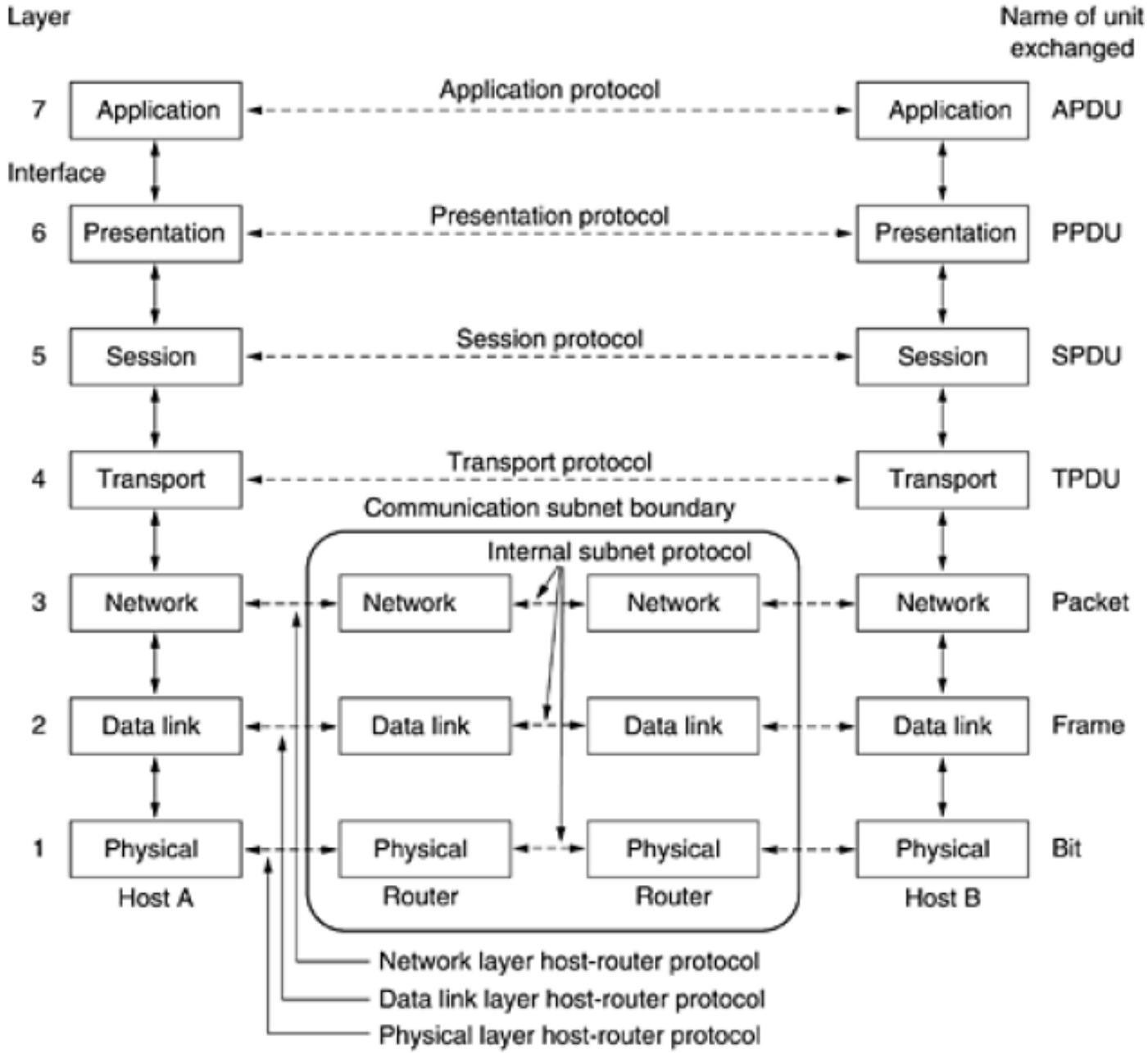
- Services and protocols are distinct concepts.
- A **Service** is a set of primitives (operations) that a layer provides to the layer above it. But it says nothing at all about how these operations are implemented.
- A service relates to an interface between two layers, with the lower layer being the service provider and the upper layer being the service user.
- A **Protocol,** in contrast, is a set of rules governing the format and meaning of the packets, or messages that are exchanged by the peer entities within a layer.
- Entities use protocols to implement their service definitions.
- **Services relate to the interfaces between layers, In contrast, protocols relate to the packets sent between peer entities on different machines.**

Reference Models

- OSI Reference Model
- TCP/IP Reference Model

OSI Reference Model

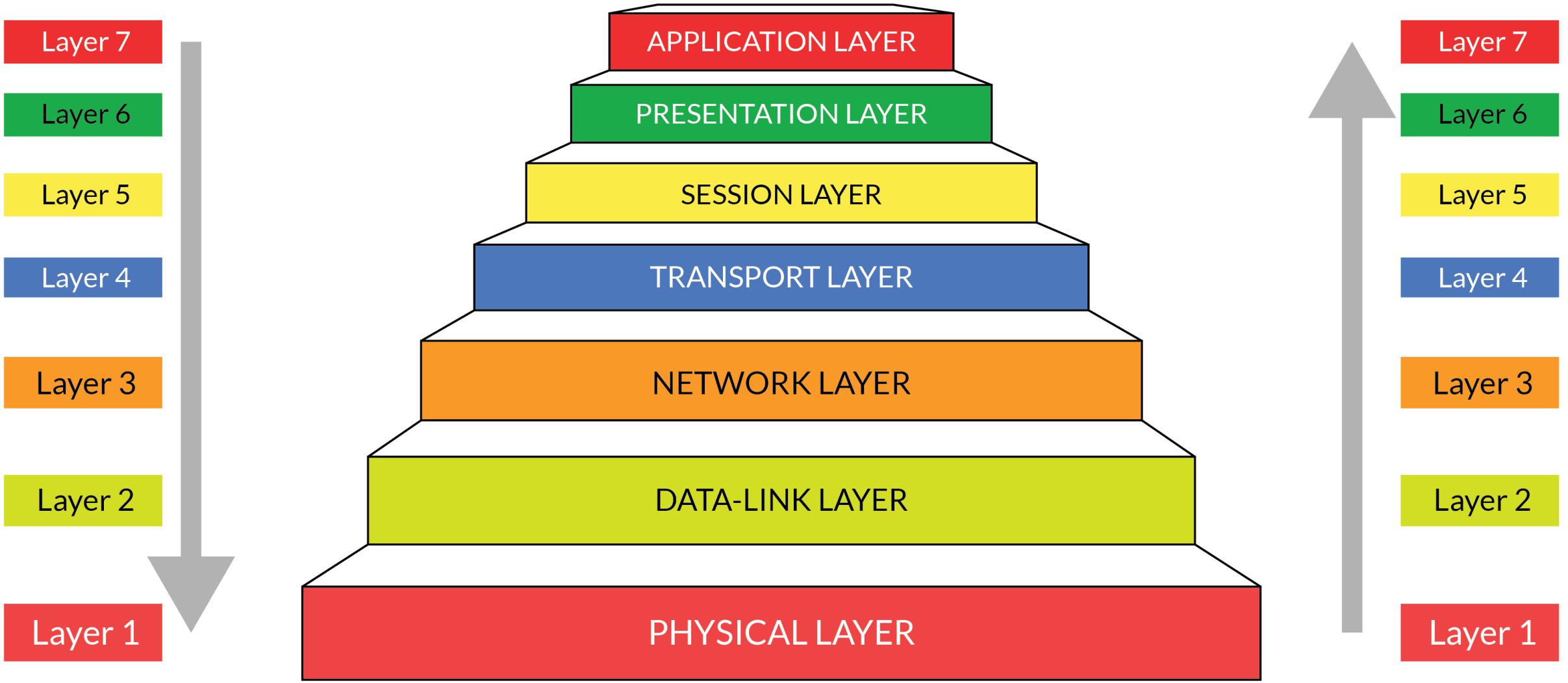
- OSI: Open Systems Interconnection
- Proposed by the International Standards Organization (ISO) as a first step toward international standardization of the protocols used in the various layers.
- The model is called the ISO OSI (Open Systems Interconnection) Reference Model because it deals with connecting open systems—that is, systems that are open for communication with other systems.

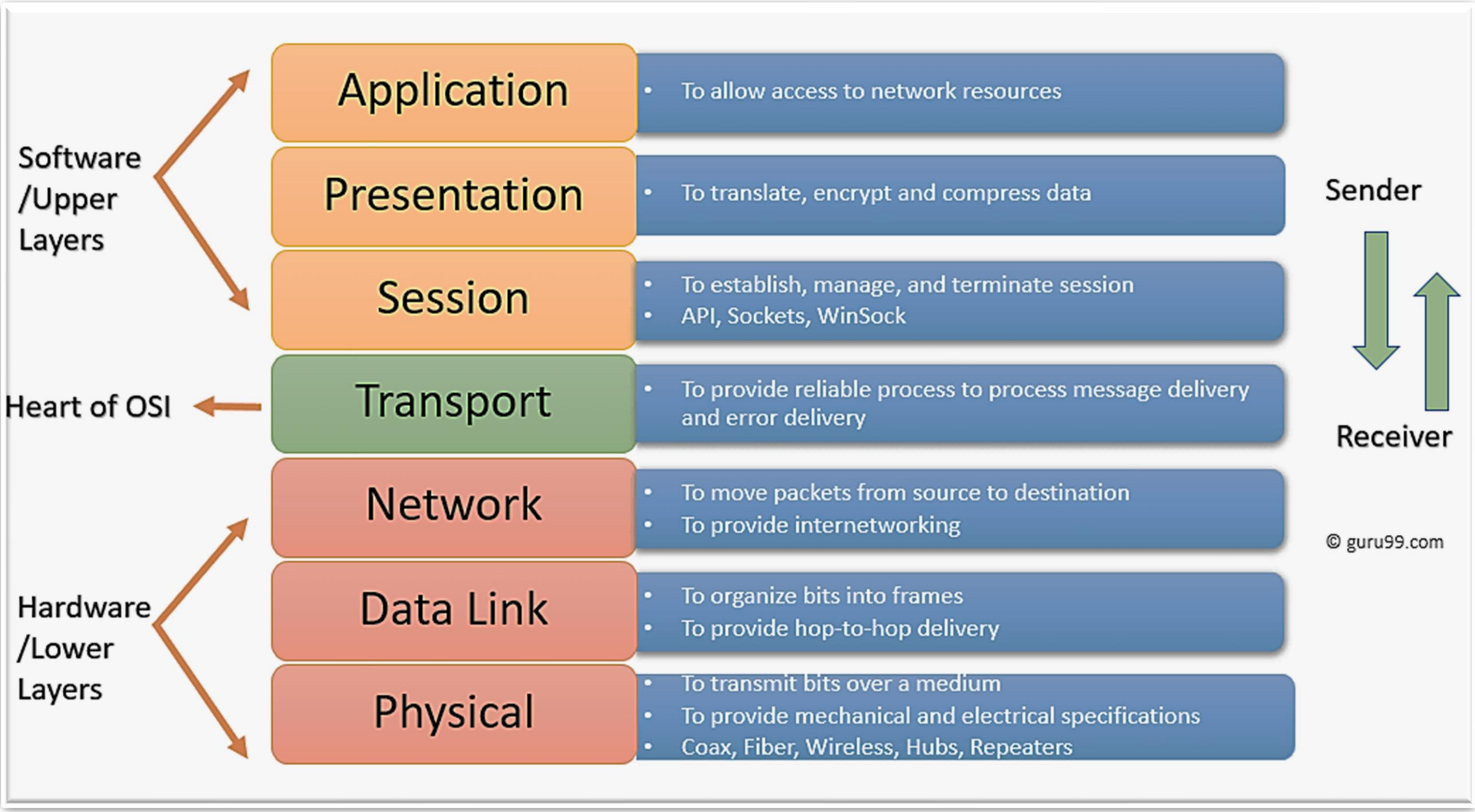


OSI MODEL

Client Side

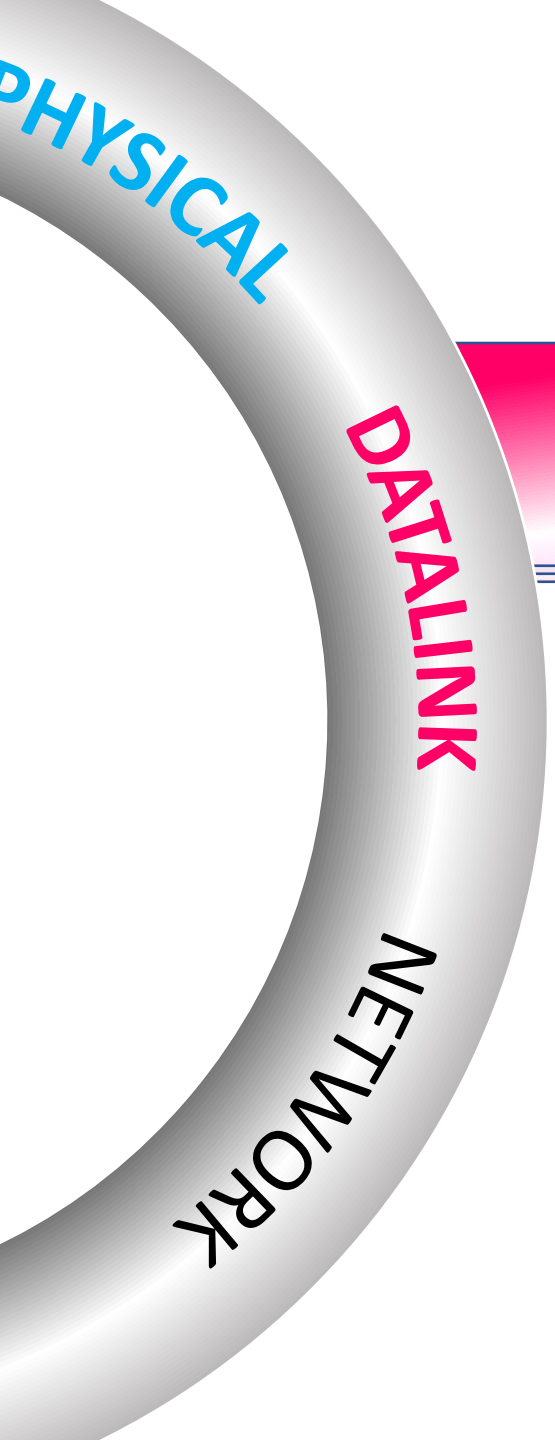
Server Side




Application
To allow access to network resources
Presentation
To translate, encrypt and compress data
Session
To establish, manage, and terminate session
API, Sockets, WinSock
Transport
To provide reliable process to process message delivery and error delivery
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
Coax, Fiber, Wireless, Hubs, Repeaters
Software /Upper Layers
Heart of OSI
Hardware /Lower Layers
Sender
Receiver
© guru99.com

PHYSICAL LAYER

- The physical layer is concerned with transmitting raw bits over a communication channel.
- When one side sends a 1 bit, it is received by the other side as a 1 bit.
- Whether transmission may proceed simultaneously in both directions.
- How initial connection is established and how it is torn down when both sides are finished.
- The design issues here largely deal with
  - Mechanical, electrical, and timing interfaces,
  - Physical transmission medium, which lies below the physical layer.



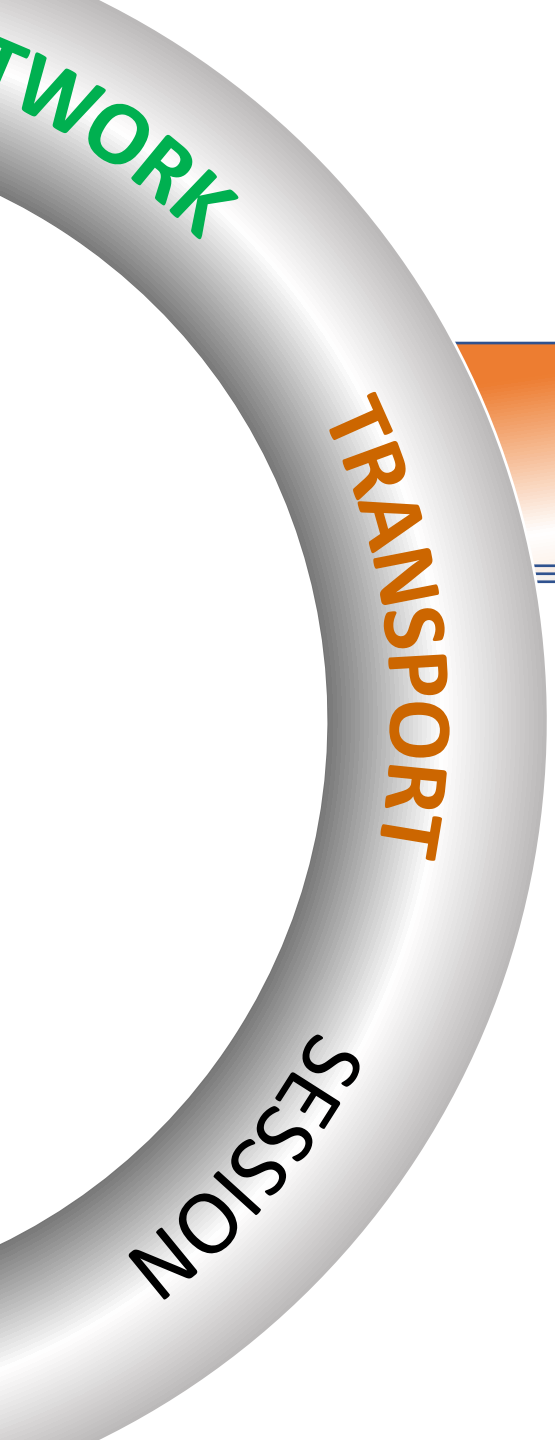
DATALINK LAYER

- The main task of the data link layer is to transform a raw transmission facility into a line that appears free of undetected transmission errors to the network layer.
- Divides data into frames and transmits them sequentially.
- Frame: typically, a few hundred or a few thousand bytes.
- If the service is reliable, the receiver confirms correct receipt of each frame by sending back an acknowledgement frame.
- Flow control and Error control.

NETWORK LAYER

- A key design issues are
  - how packets are routed from source to destination.
  - Control of congestion
  - The quality of service provided (delay, transit time, jitter, etc.).

- Network layer has to overcome all problems to allow heterogeneous networks to be interconnected.

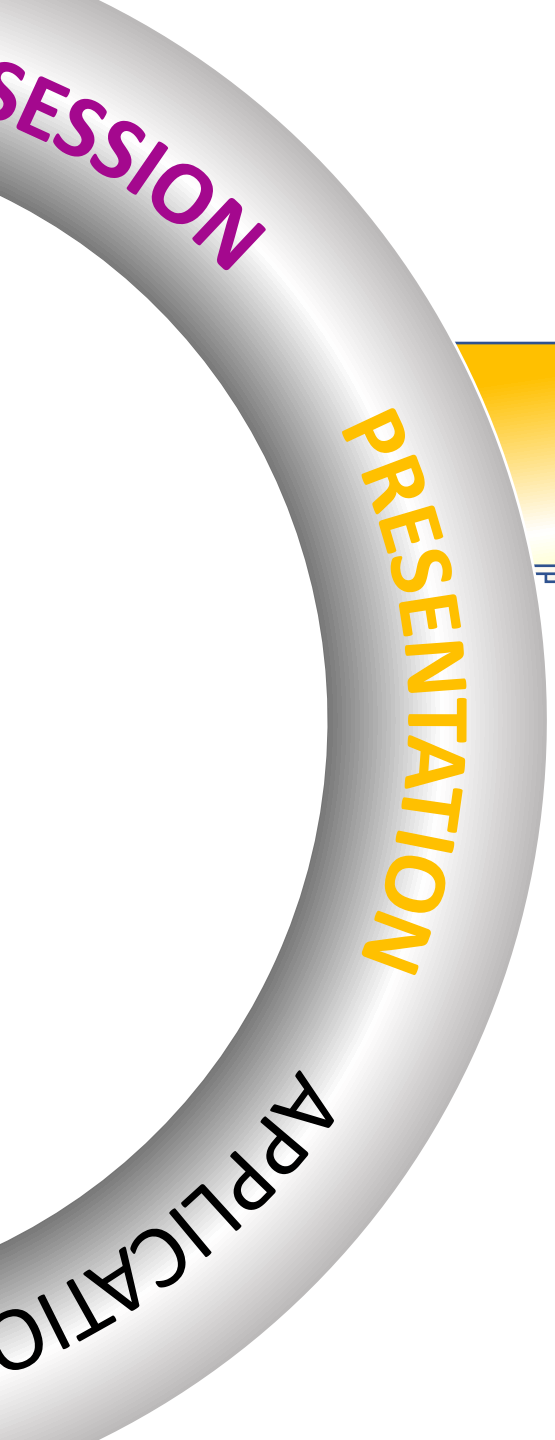


TRANSPORT LAYER

- The basic function of the **transport layer** is to accept data from above it, split it up into smaller units if need be, pass these to the network layer, and ensure that the pieces all arrive correctly at the other end.
- Determines what type of service to provide to the session layer, and, ultimately, to the users of the network.
- Provides connection-oriented and connectionless services.

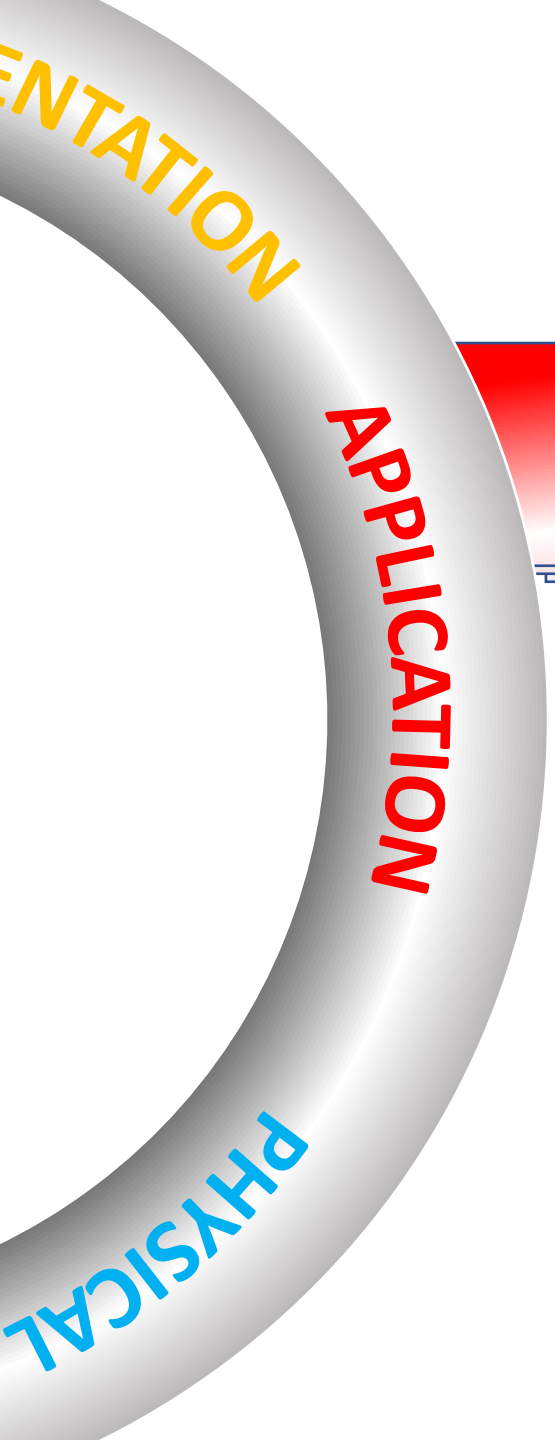
SESSION LAYER

- The session layer allows users on different machines to establish sessions between them.
- Sessions offer various services including:
  - **Dialog control** (keeping track of whose turn it is to transmit)
  - **Token management** (preventing two parties from attempting the same critical operation at the same time)
  - **Synchronization** (checkpointing long transmissions to allow them to continue from where they were after a crash).



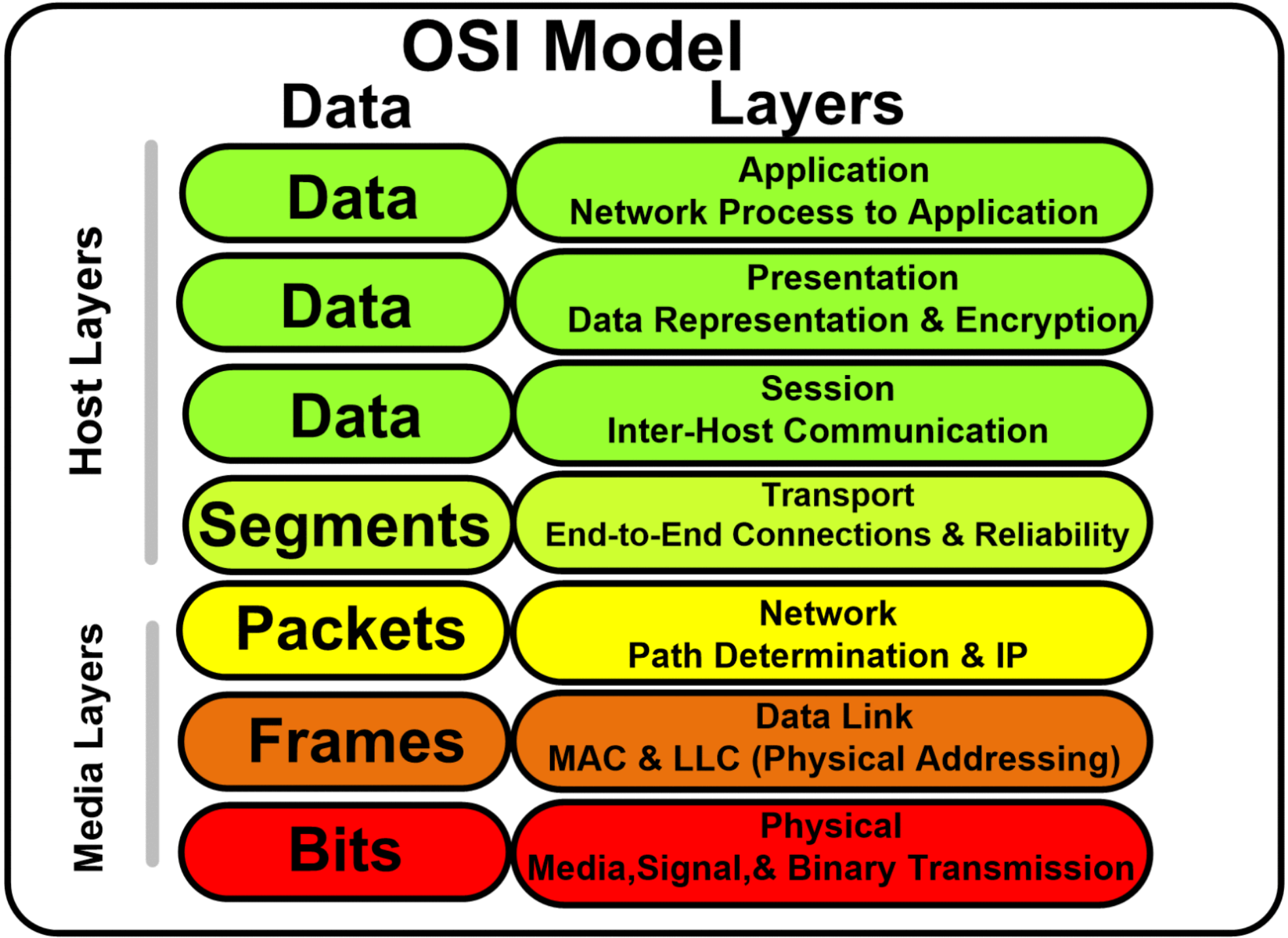
PRESENTATION LAYER

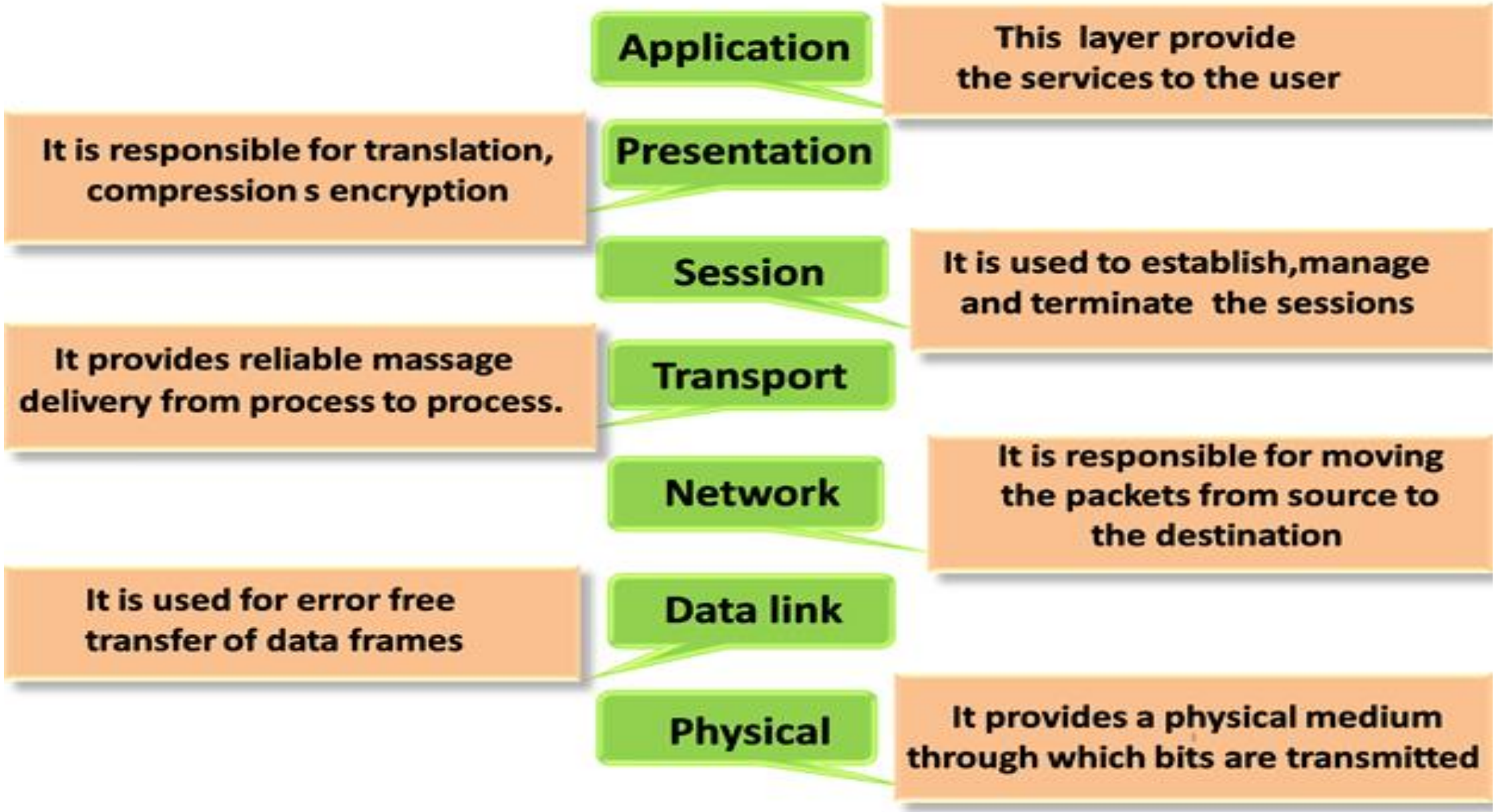
- The presentation layer is concerned with the syntax and semantics of the information transmitted.
- In order to make it possible for computers with different data representations to communicate, the data structures to be exchanged can be defined in an abstract way, along with a standard encoding to be used "on the wire.‘’
- The presentation layer manages these abstract data structures and allows higher-level data structures (e.g., banking records), to be defined and exchanged.



APPLICATION LAYER

- The application layer contains a variety of protocols that are commonly needed by users.
- One widely-used application protocol is HTTP (HyperText Transfer Protocol), which is the basis for the World Wide Web.
- Other application protocols are used for file transfer, electronic mail, and network news.

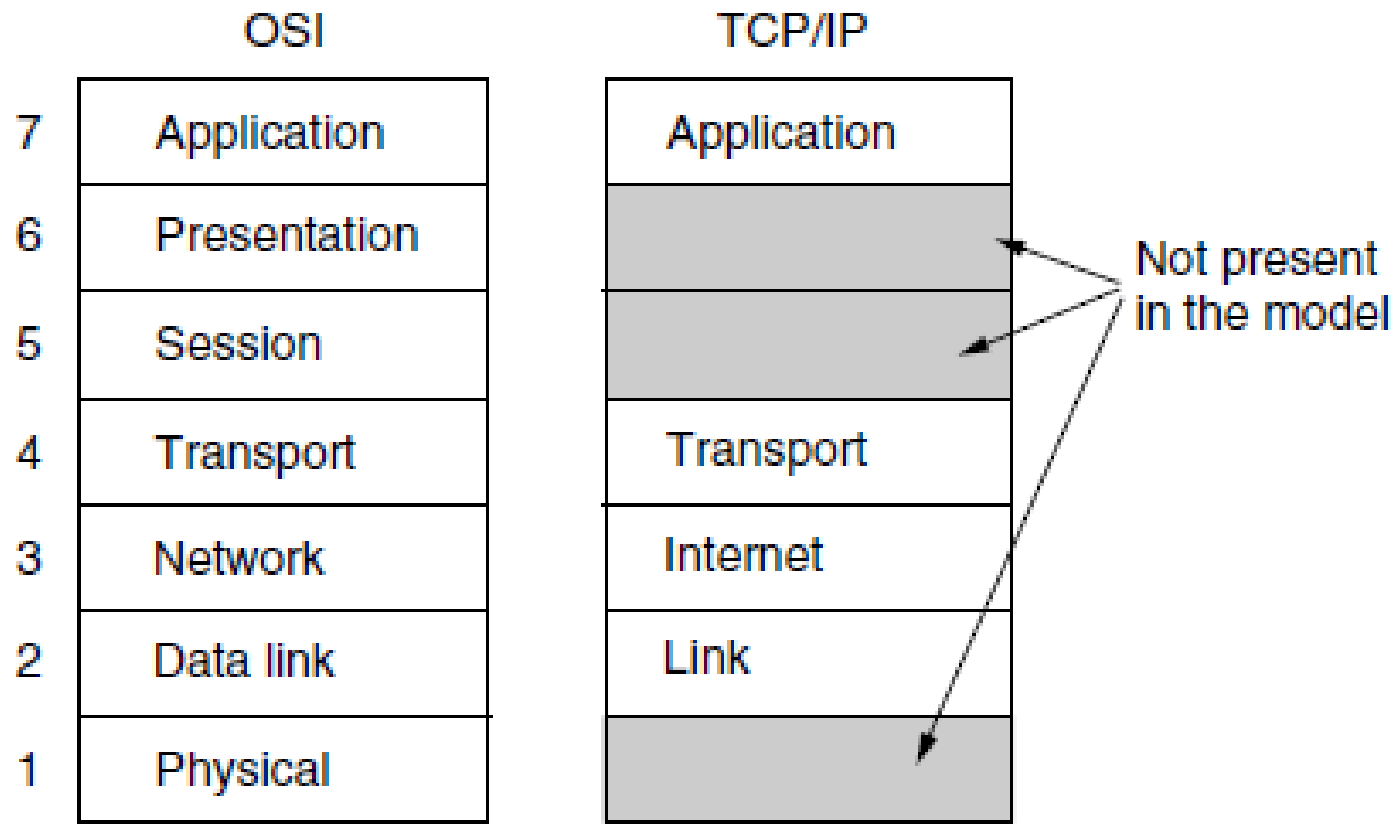

OSI Model
Data
Layers
Host Layers
Media Layers
Data
Application
Network Process to Application
Data
Presentation
Data Representation & Encryption
Data
Session
Inter-Host Communication
Segments
Transport
End-to-End Connections & Reliability
Packets
Network
Path Determination & IP
Frames
Data Link
MAC & LLC (Physical Addressing)
Bits
Physical
Media,Signal,& Binary Transmission


Application
This layer provide the services to the user
It is responsible for translation, compression s encryption
Presentation
Session
It is used to establish,manage and terminate the sessions
It provides reliable massage delivery from process to process.
Transport
Network
It is responsible for moving the packets from source to the destination
It is used for error free transfer of data frames
Data link
Physical
It provides a physical medium through which bits are transmitted

TCP/IP Reference Model

- Used in the grandparent of all wide area computer networks, the ARPANET, and its successor, the worldwide Internet.
- It eventually connected hundreds of universities and government installations, using leased telephone lines.
- When satellite and radio networks were added later, the existing protocols had trouble interworking with them, so a new reference architecture was needed.
- Since applications with divergent requirements were envisioned, ranging from transferring files to real-time speech transmission, a flexible architecture was needed.

TCP/IP Reference Model..



- Protocol Hierarchies
- Design Issues for the Layers
- Connection-oriented Vs Connectionless Service
- Service Primitives
- Relationship of Services to Protocols
- OSI Layered Architecture
- TCP/IP Layered Architecture


THANK YOU