

Recap

Trust

- Trusted System, TCB, TPM

Access Control

- Access Control Matrix, Access Control List, Capability

Networking

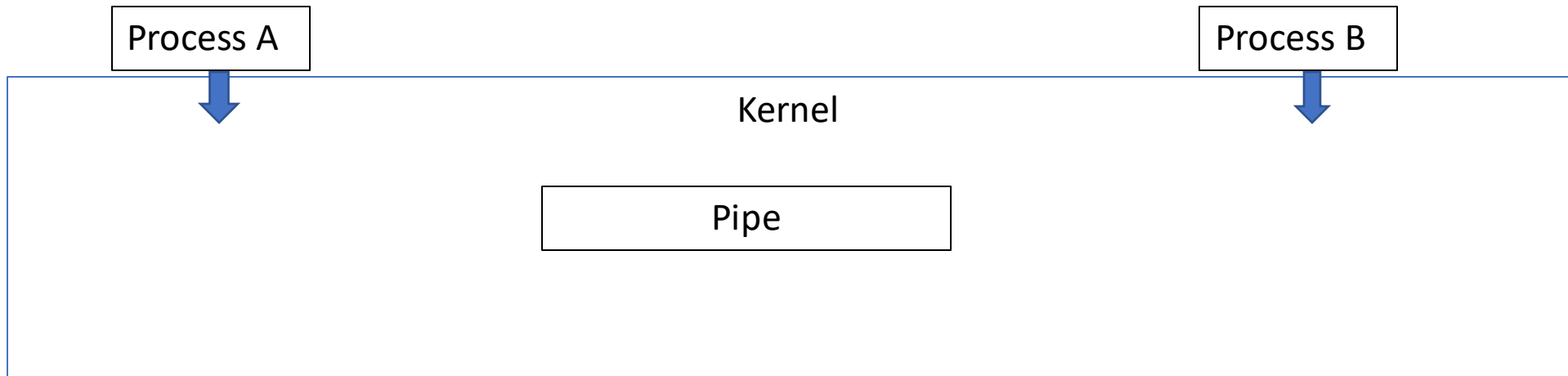
How is computer-to-computer communication implemented?

Recall: Process-to-process Communication

Logic View

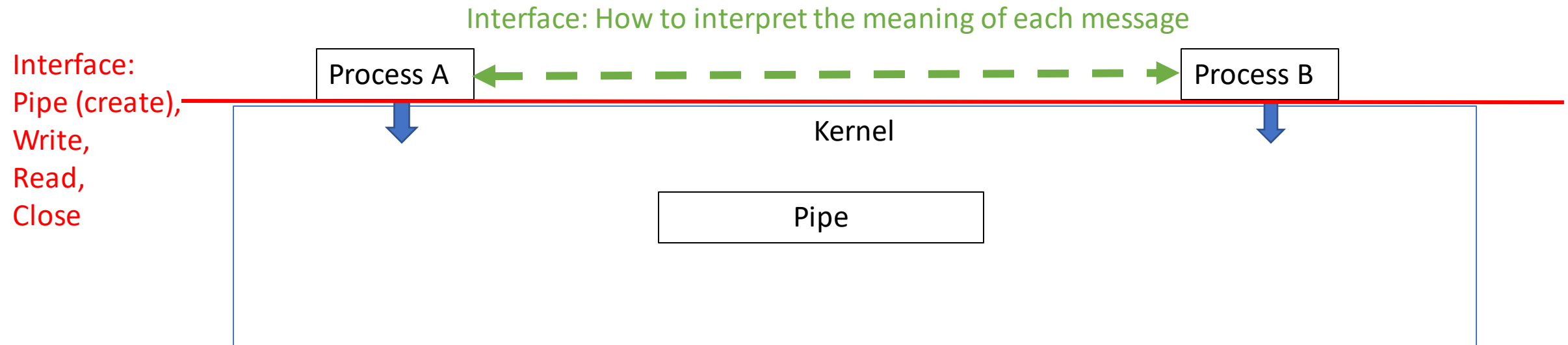


Implementation View



Recall: Process-to-process Communication

Implementation View



Take aways:

- Communication between **peers** (who don't directly interact with each other) are implemented based on **layers** and the bottom layers are able to directly interact.
- Need **between-layer interface** and **between-peer interface**.
- Communication between applications residing on different machines is similar though more complex!

Analogy: Person-to-Person Communication in Traditional World

From: Alice, 1 Lincoln Way, Ames, IA To: Bob, 2 Michigan Ave, Chicago, IL

Interface: they understand the same language
to interpret each other's messages

But (suppose) they can't directly
(physically) reach out to each other

From: 1 Lincoln Way, Ames, IA To: 2 Michigan Ave, Chicago, IL

From: Ames, IA To: Chicago, IL

From: IA To: IL

Human/vehicular Carriers

Rely on
house
mailbox
to
communi
cate

Rely on
house
mailbox
to
communi
cate

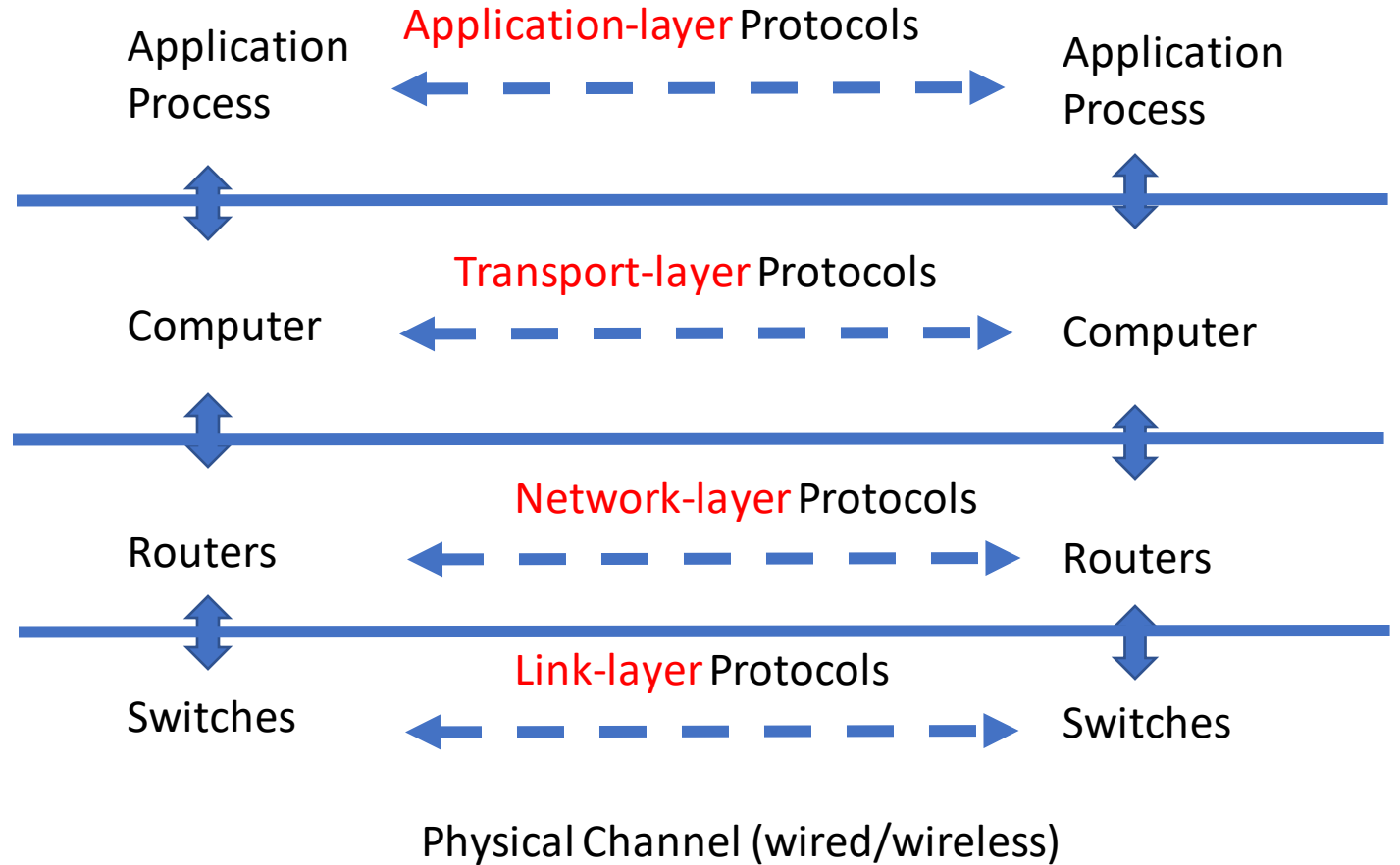
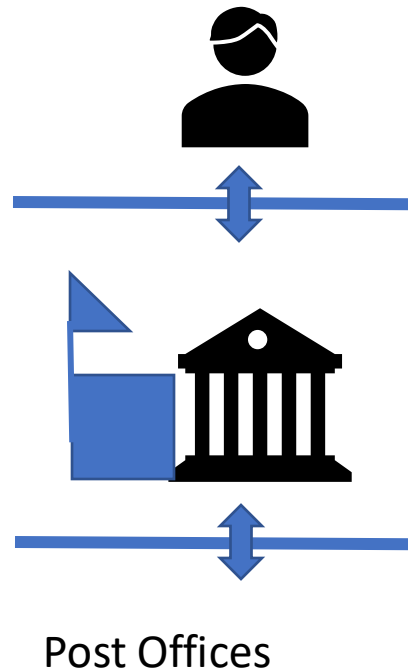
Rely on
post
office to
communi
cate

Rely on
post
office to
communi
cate

Post Offices
(local, regional,
national,
international)

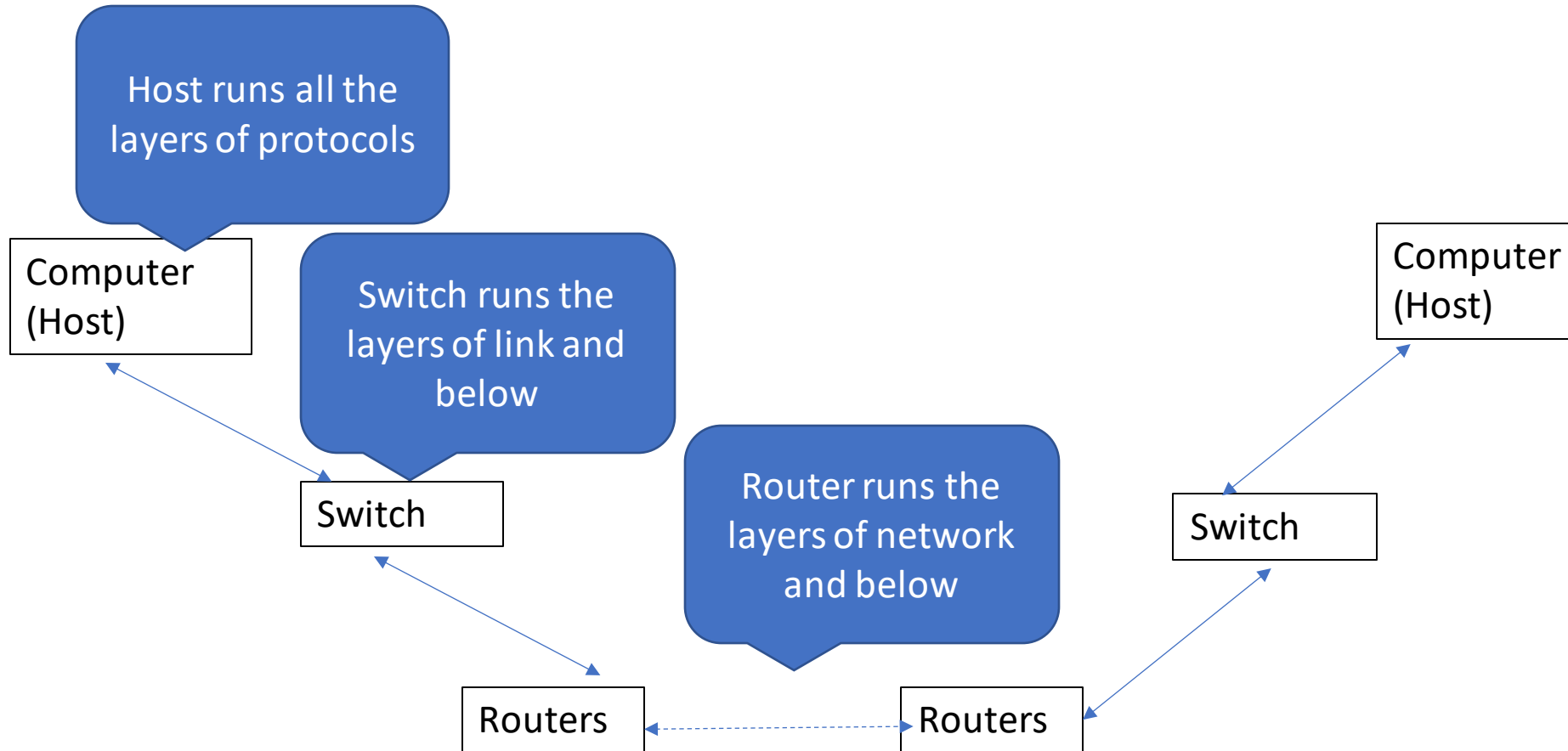
Post Offices
(local, regional,
national,
international)

Mapping to Internet

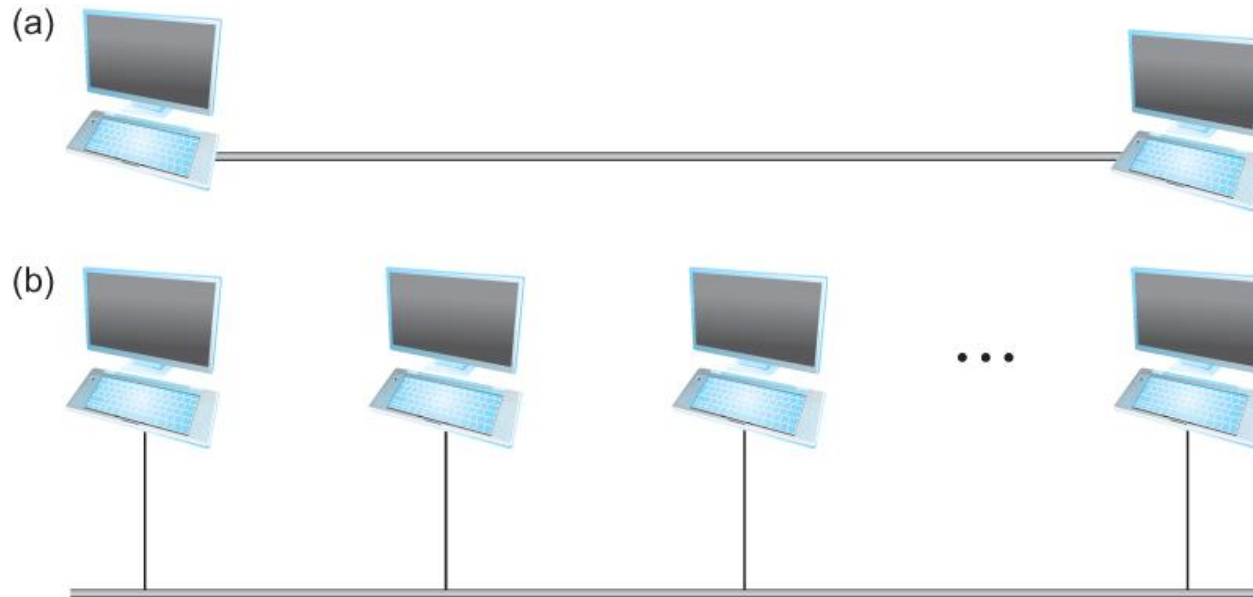


Note: interface is called protocol

Simplified Topology of Internet



Connectivity



- (a) Point-to-point
(b) Multiple access

Connectivity terminology

Nodes – computers, devices, routers...

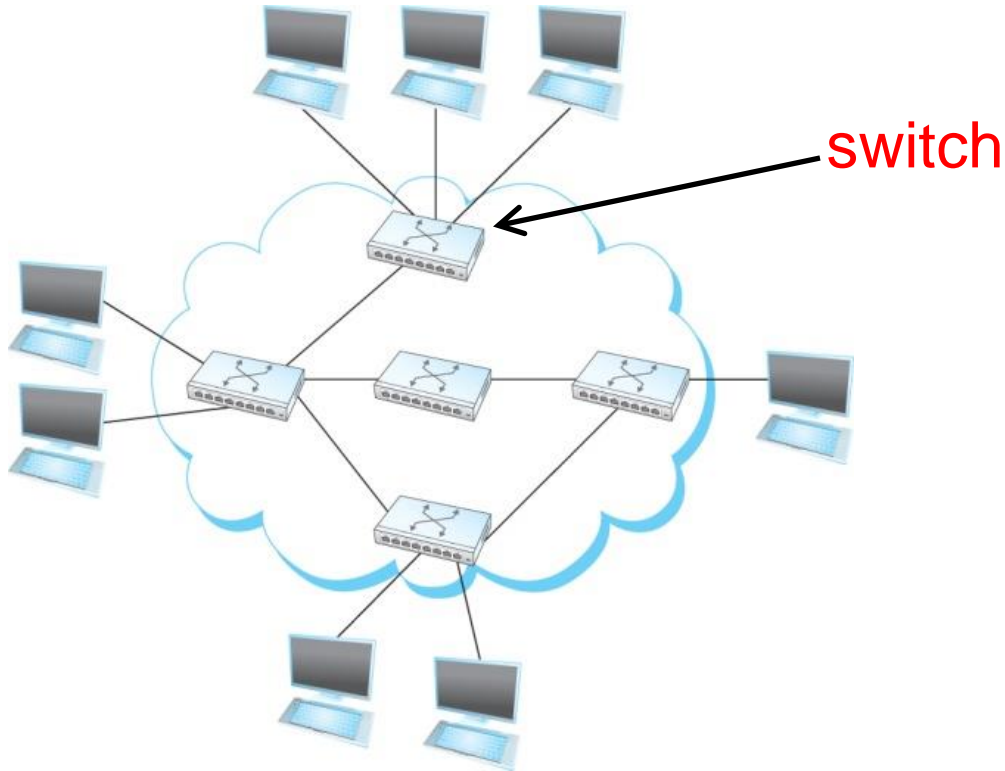
Link – physical connection between nodes

Point-to-point – 2 nodes on 1 link

Multiple access – multiple nodes on 1 link

Connectivity (cont.)

Switched Network



Terminology (contd.)

Packet, message: block of data being communicated

Switches – nodes with multiple links that forward data/packets from one link to another

Switched Network: Switches provide connectivity across a network by forwarding data between links

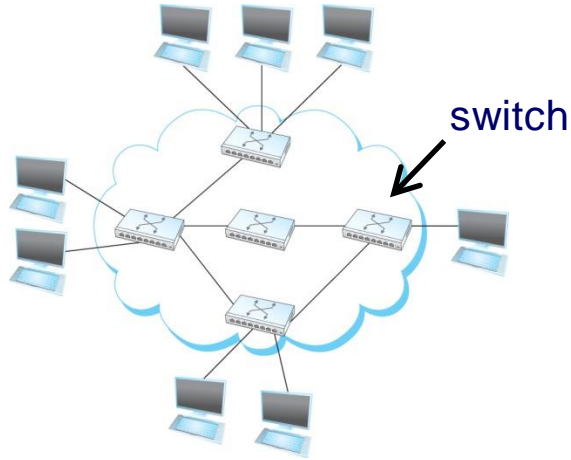
- Circuit Switched: physically connect two links
- Packet Switched: repeat a packet of data from one link to another

Store-and-forward: incoming packets are buffered (e.g., FIFO queue) and forwarded

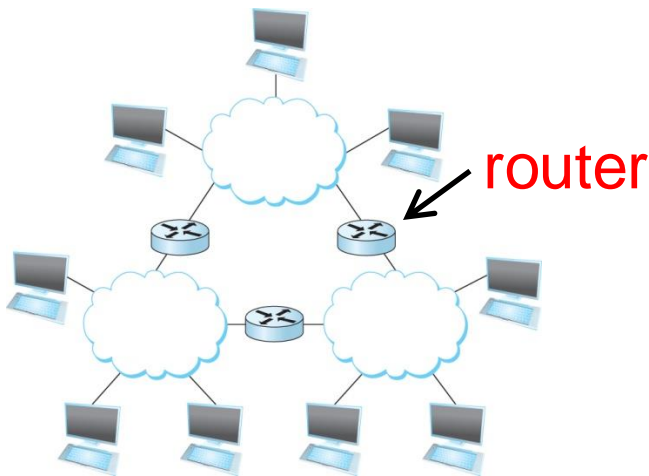
Connectivity (cont.)

Terminology (contd.)

Switched Network



Interconnection of networks



Cloud – abstract view of a network

Hosts – Computers/devices connected to the network

Router/gateway – forwards data between networks

Routing – forwarding of data over a path from one host to another

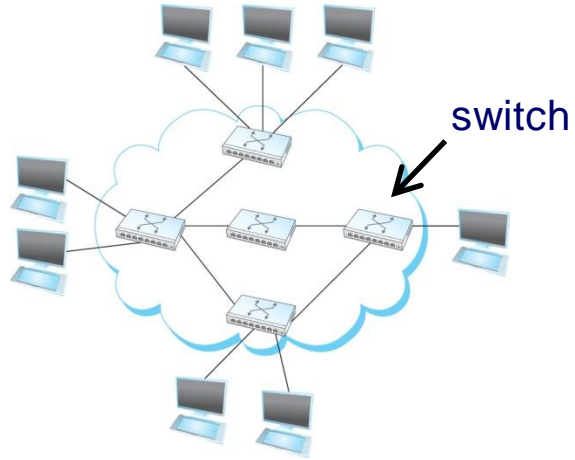
Host-to-host connectivity – two hosts have a route between them

Address – a number that describes the location of a host or router

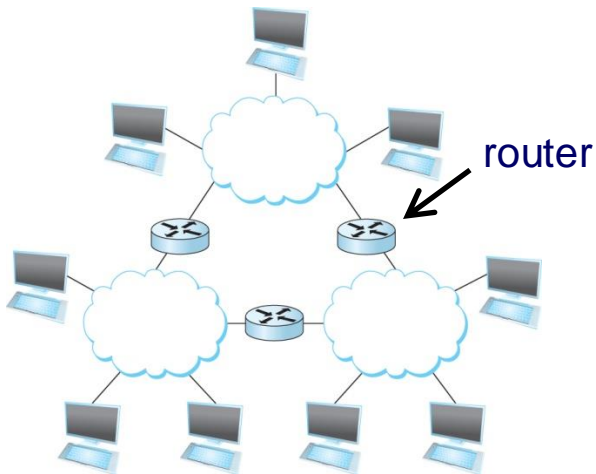
Unicast/broadcast/multicast – unicast is routing a message to one host, broadcast is forwarding a message to all hosts

Connectivity (cont.)

Switched Network - LAN



Interconnection of networks - WAN



Terminology (contd.)

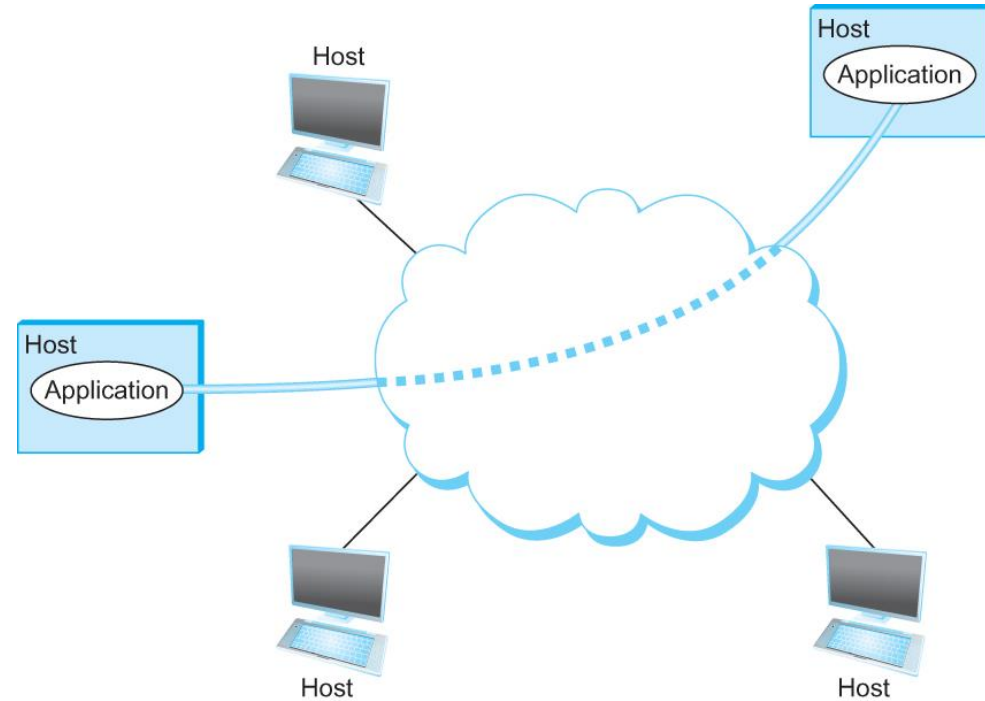
LAN (local area network) – connects computers within a limited area, uses switches

WAN (wide area network) – extends over a large geographic area, uses routers

Network Protocols and Architecture

Goal is to provide applications transparent communication across hosts

Abstraction of a channel or “pipe”



We have seen pipes on local machine, how to provide similar abstraction across machines?

Communication Patterns

Client/Server

Two types of communication channel

- Request/Reply Channels (reliable, asynchronous)
- Message Stream Channels (synchronous, fault-tolerant)
- Others ...

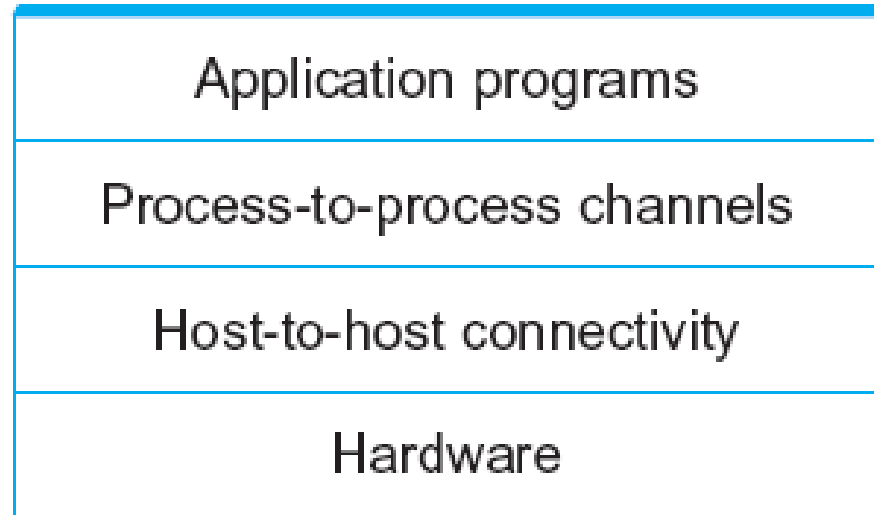
Reliability

Application has expectation that messages are delivered without corruption or loss

Many sources of errors, network should hide them

- Bits are lost
 - Bit errors (1 to a 0, and vice versa)
 - Burst errors – several consecutive errors
- Packets are lost (Congestion)
- Links and Node failures
- Messages are delayed
- Messages are delivered out-of-order
- Third parties eavesdrop

Network Architecture



Example of a layers of abstraction in network system

Network Architecture Alternatives

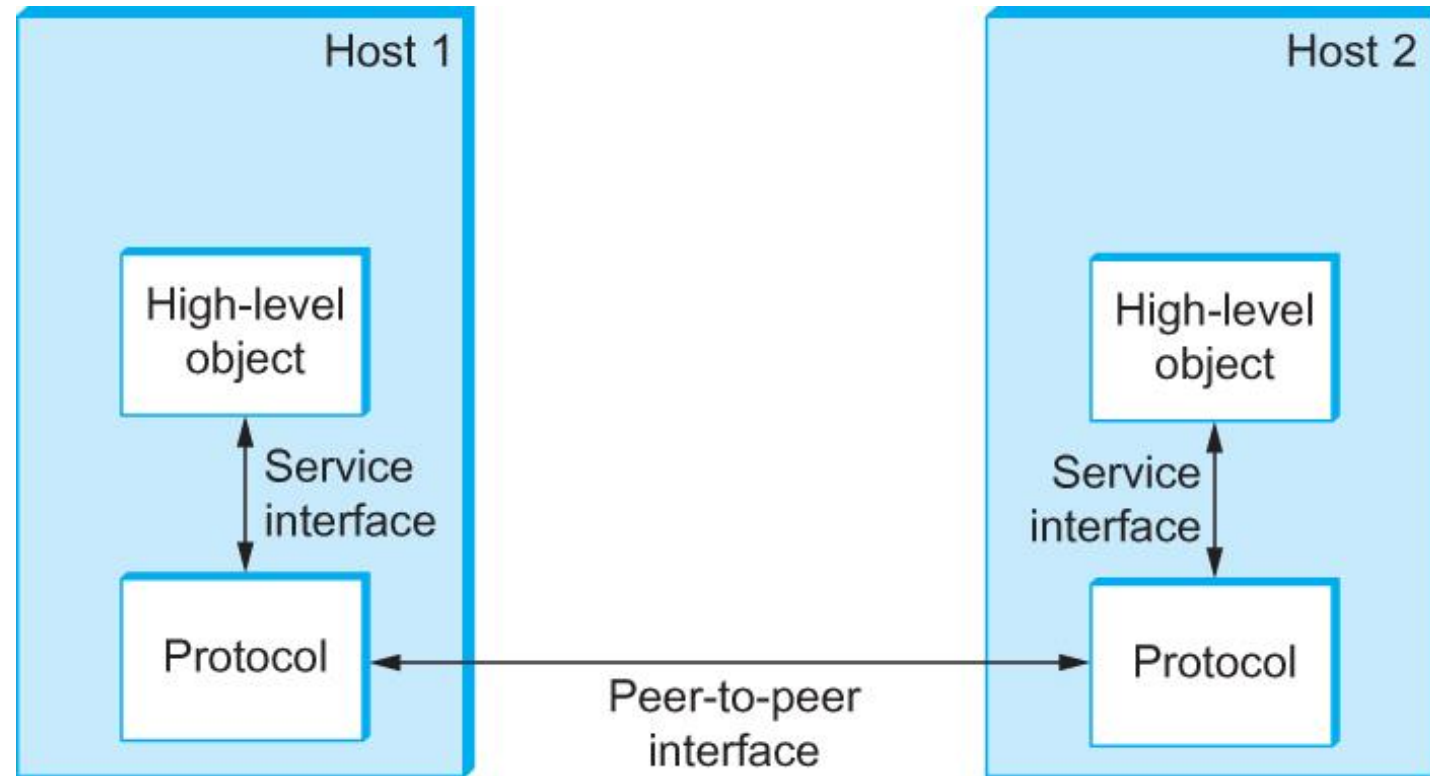
Application programs	
Request/reply channel	Message stream channel
Host-to-host connectivity	
Hardware	

Layered system with alternative abstractions available at a given layer

Protocols

- Protocol defines the interfaces between the layers in the same system and with the (same) layers of peer system
- Building blocks of a network architecture
- Each protocol object has two different interfaces
 - service interface: operations on this protocol called by other layers (e.g., application writes message to protocol)
 - peer-to-peer interface: messages exchanged with peer (e.g., layer 1 on machine A talks with layer 1 on machine B)
- Term “protocol” is overloaded
 - specification of peer-to-peer interface
 - module that implements this interface

Interfaces

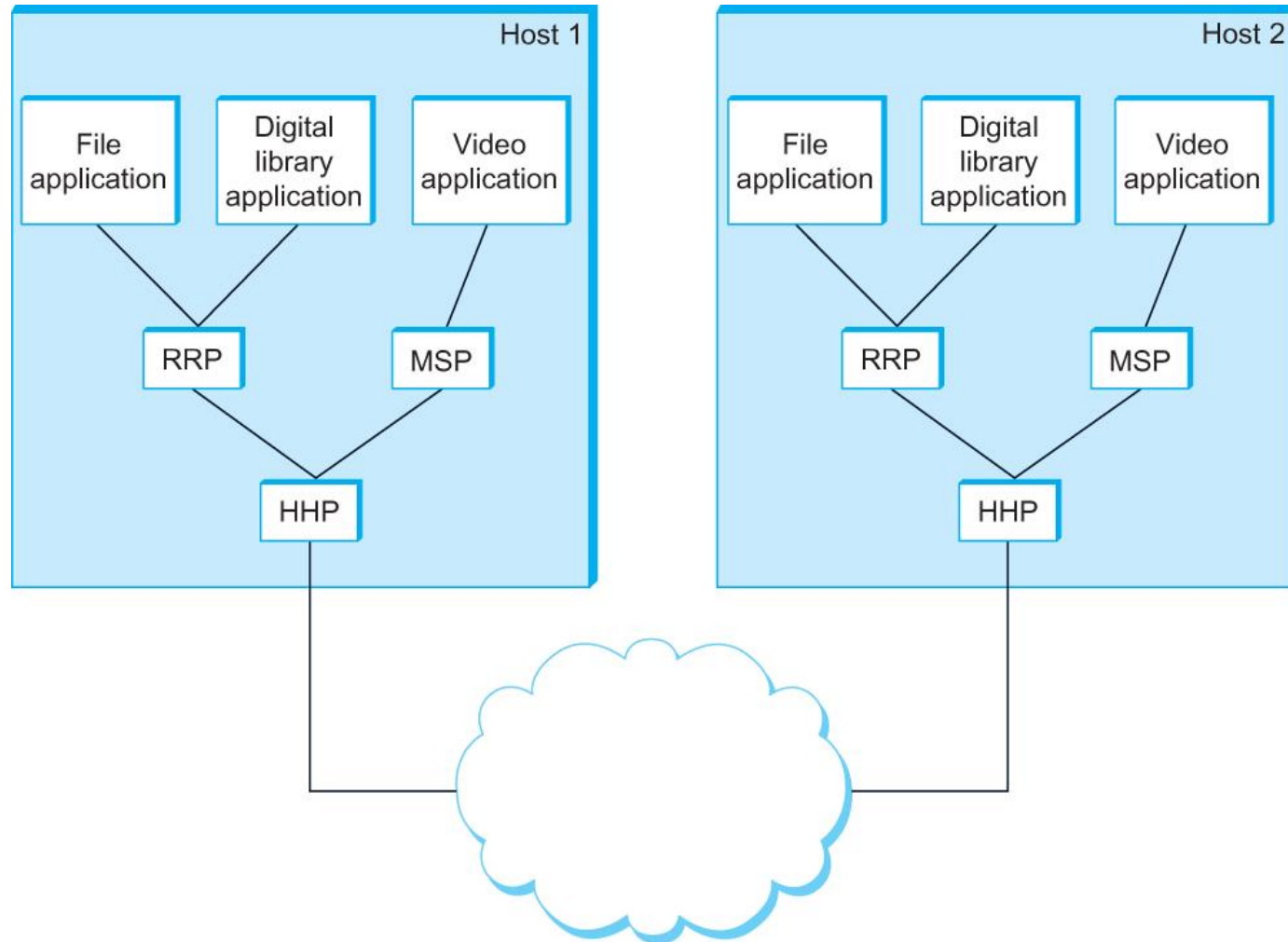


Service and Peer Interfaces

Protocols

- Protocol Specification: prose, pseudo-code, state transition diagram
- Interoperable: when two or more protocols that implement the specification accurately
- IETF: Internet Engineering Task Force

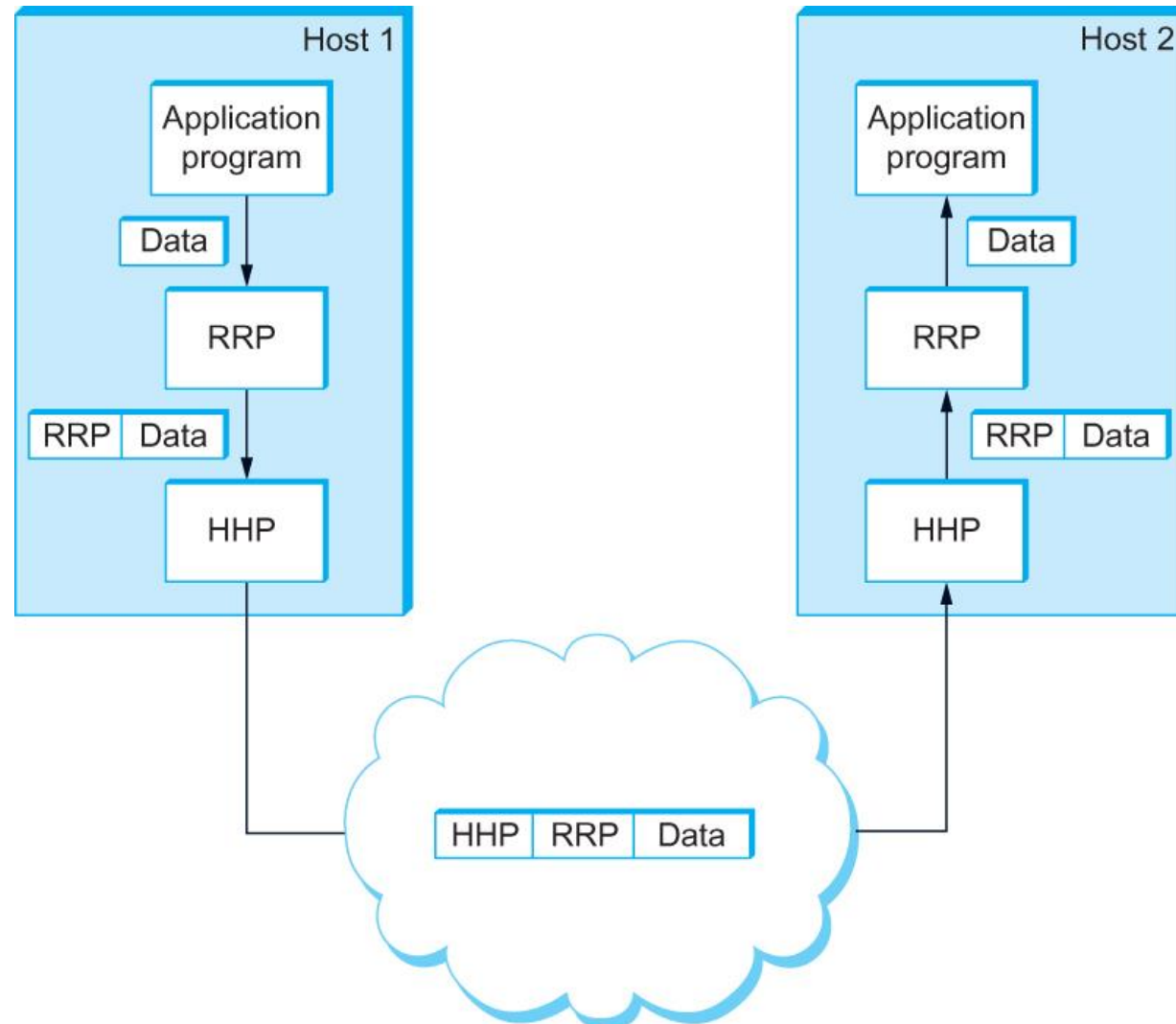
Protocol Graph



Example of a protocol graph

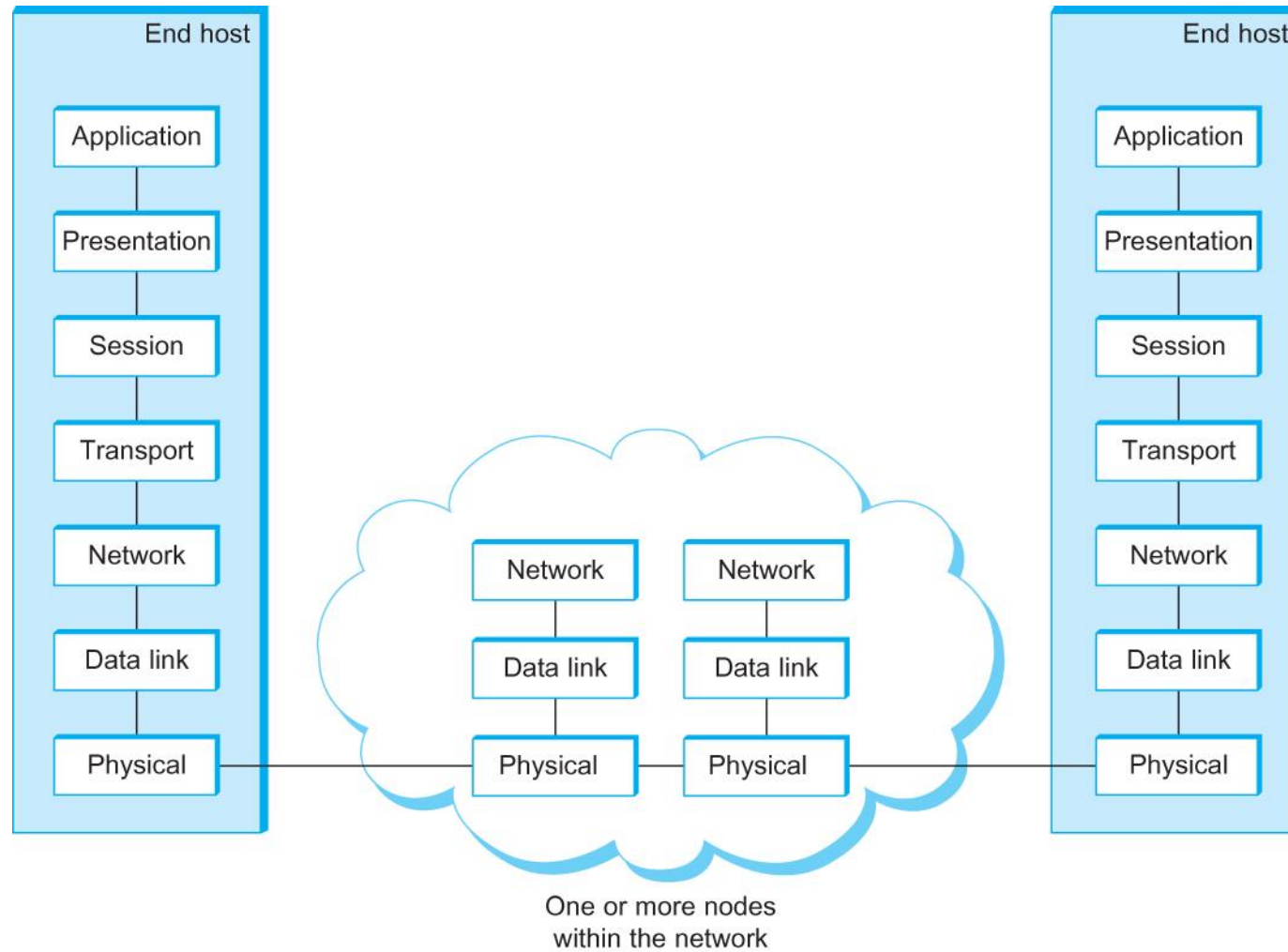
nodes are the protocols and links the “depends-on” relation

Encapsulation



High-level messages are encapsulated inside of low-level messages

OSI Architecture



The OSI 7-layer Model
OSI – Open Systems Interconnection

Description of Layers

Physical Layer

- Handles the transmission of raw bits over a communication link

Data Link Layer

- Collects a stream of bits into a larger aggregate called a *frame*
- Network adaptor along with **device driver** in OS implement the protocol in this layer
- Frames are actually delivered to hosts

Network Layer

- Handles routing among nodes within a packet-switched network
- Unit of data exchanged between nodes in this layer is called a *packet*

The lower three layers are implemented on all network nodes

Description of Layers

Transport Layer

- Implements a process-to-process channel
- Unit of data exchanges in this layer is called a *message*

Session Layer

- Provides a name space that is used to tie together the potentially different transport streams that are part of a single application

Presentation Layer

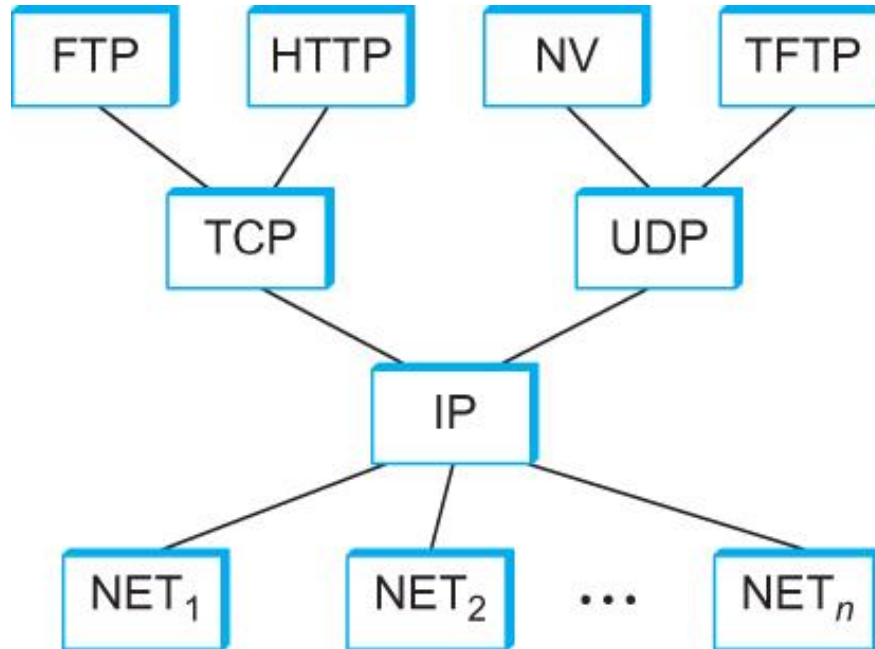
- Concerned about the format of data exchanged between peers

Application Layer

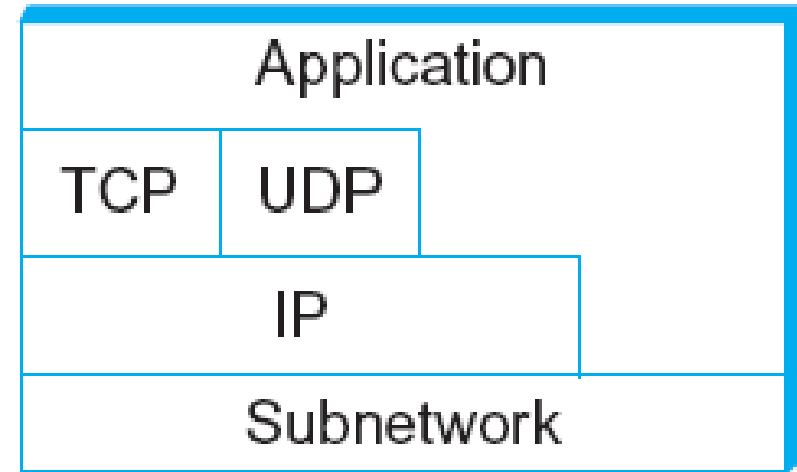
- Standardize common type of exchanges

The transport layer and the higher layers typically run only on end-hosts and not on the intermediate switches and routers

Internet Architecture



Internet Protocol Graph



Alternative view of the Internet architecture. The “Network” layer shown here is sometimes referred to as the “sub-network” or “link” layer.

Internet Architecture

Defined by IETF

Three main features

- Does not imply strict layering. The application is free to bypass the defined transport layers and to directly use IP or other underlying networks
- An hour-glass shape – wide at the top, narrow in the middle and wide at the bottom. IP serves as the focal point for the architecture
- In order for a new protocol to be officially included in the architecture, there needs to be both a protocol specification and at least one (and preferably two) representative implementations of the specification

Applications

Most people know about the Internet (a computer network) through applications

- World Wide Web
- Email
- Online Social Network
- Streaming Audio Video
- File Sharing
- Instant Messaging
- ...

Application Level Protocols

Application-level protocol is a shared language between distributed processes of a particular application

- WWW (world wide web)
HTTP (hypertext transfer protocol)
Used for viewing and sending data to web sites
- Email
SMTP (simple mail transfer protocol)
Used for transmitting email
- DNS (domain name service)
DNS protocol
Used for looking up the server address of a domain name
- SSH (Secure Shell)
SSH protocol
Used to securely control a shell on a remote machine

May be different programs participating in an application

- For example: Putty on Windows connecting to OpenSSH server on Linux

Application Protocol Example

Applications may require multiple protocols, for example, a browser connecting to a web server shown below

URL

Uniform resource locator

<http://www.cs.princeton.edu/~llp/index.html>

HTTP

Hyper Text Transfer Protocol

TCP

Transmission Control Protocol

17 messages for one URL request

6 to find the IP (Internet Protocol) address

3 for connection establishment of TCP

4 for HTTP request and acknowledgement

Request: I got your request and I will send the data

Reply: Here is the data you requested; I got the data

4 messages for tearing down TCP connection