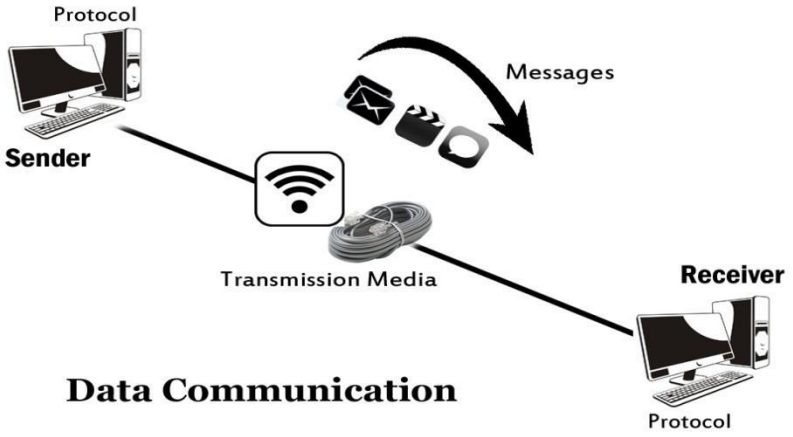
UNIT -I

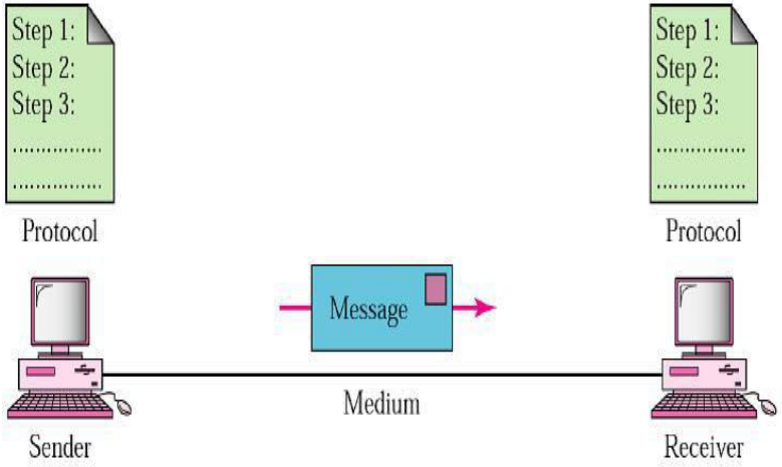
- Data Communications
- Network Topologies
- Reference Models
- Data Link Layer

DATA COMMUNICATIONS

- ***The word data refers to information presented in whatever form is agreed upon by the parties creating and using the data. Data Communications are the exchange of data between two devices via some form of transmission medium such as a wire cable.***



Data Communication

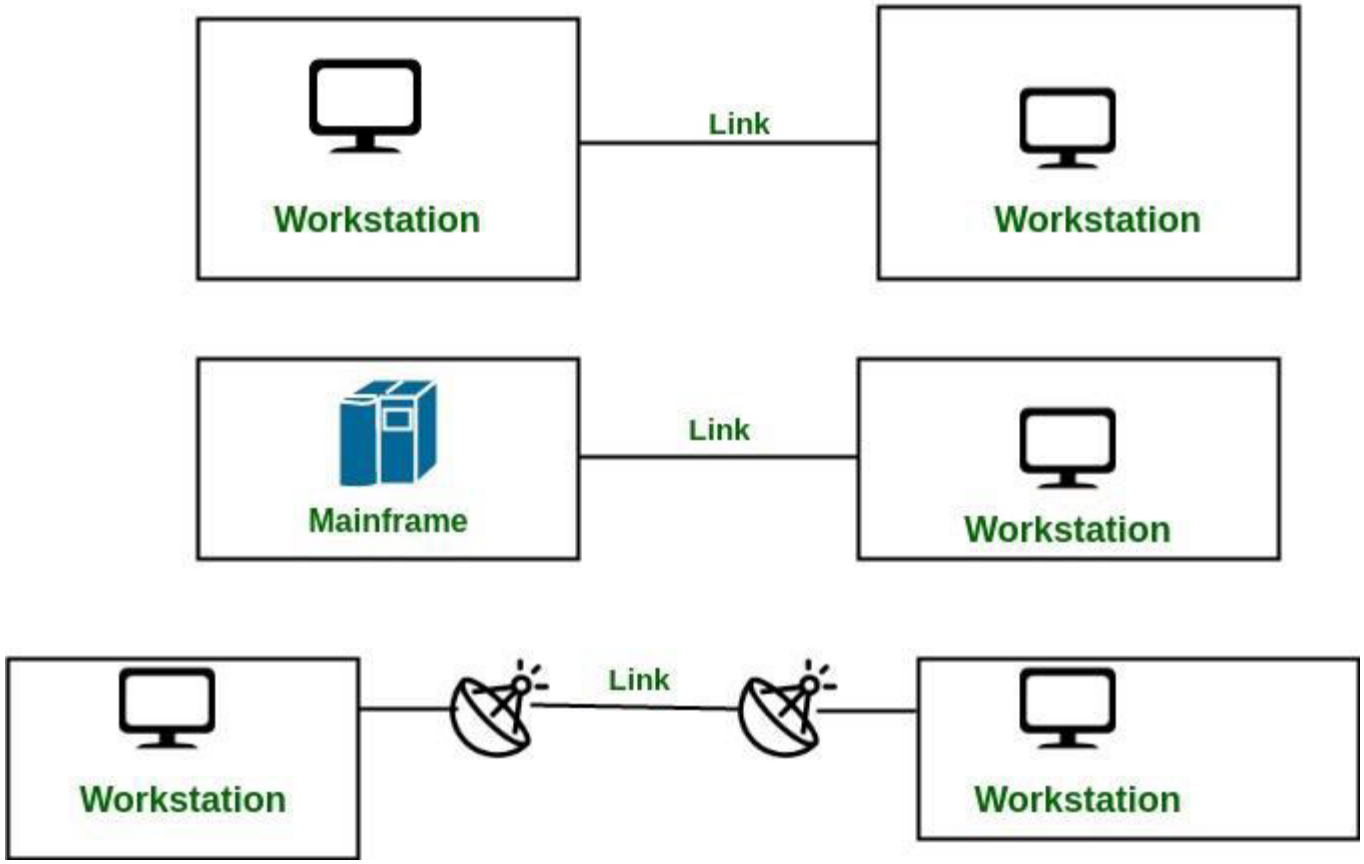


LINE CONFIGURATIONS

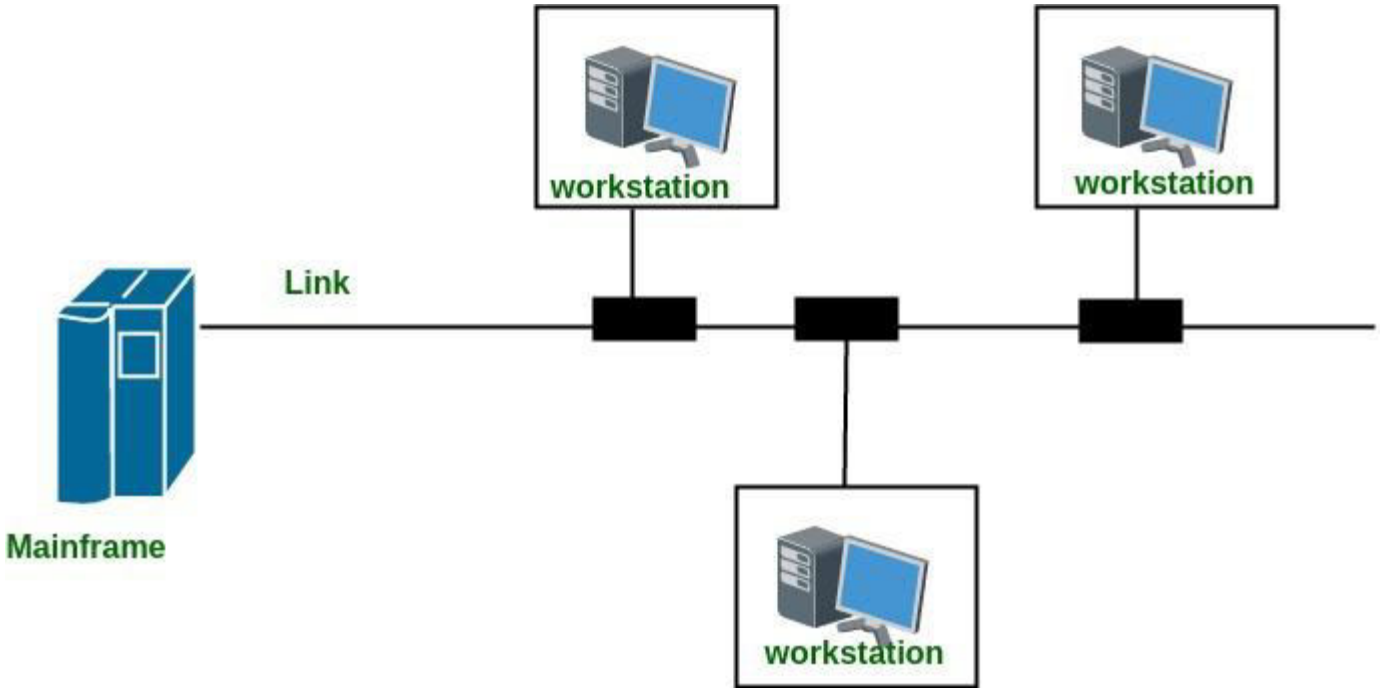
- Line configuration means the way by which two or more communication devices are connected to a link.
- A link is a physical communication pathway that transfers data from one device to another.
- Based on the requirements, there are two possible line configurations.

1. **Point to Point**
2. **Multipoint**

POINT TO POINT



MULTIPOINT

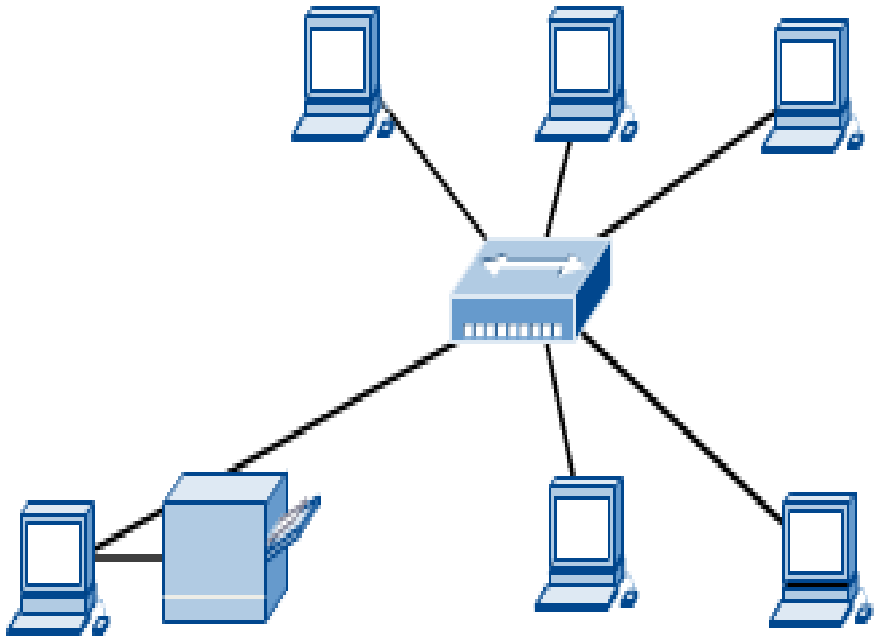


COMPUTER NETWORKS

- A computer network is a group of autonomous computers and other computing hardware devices that are linked together through communication channels to facilitate communication and resource-sharing among a wide range of users.

- Computer network connects two or more autonomous computers.

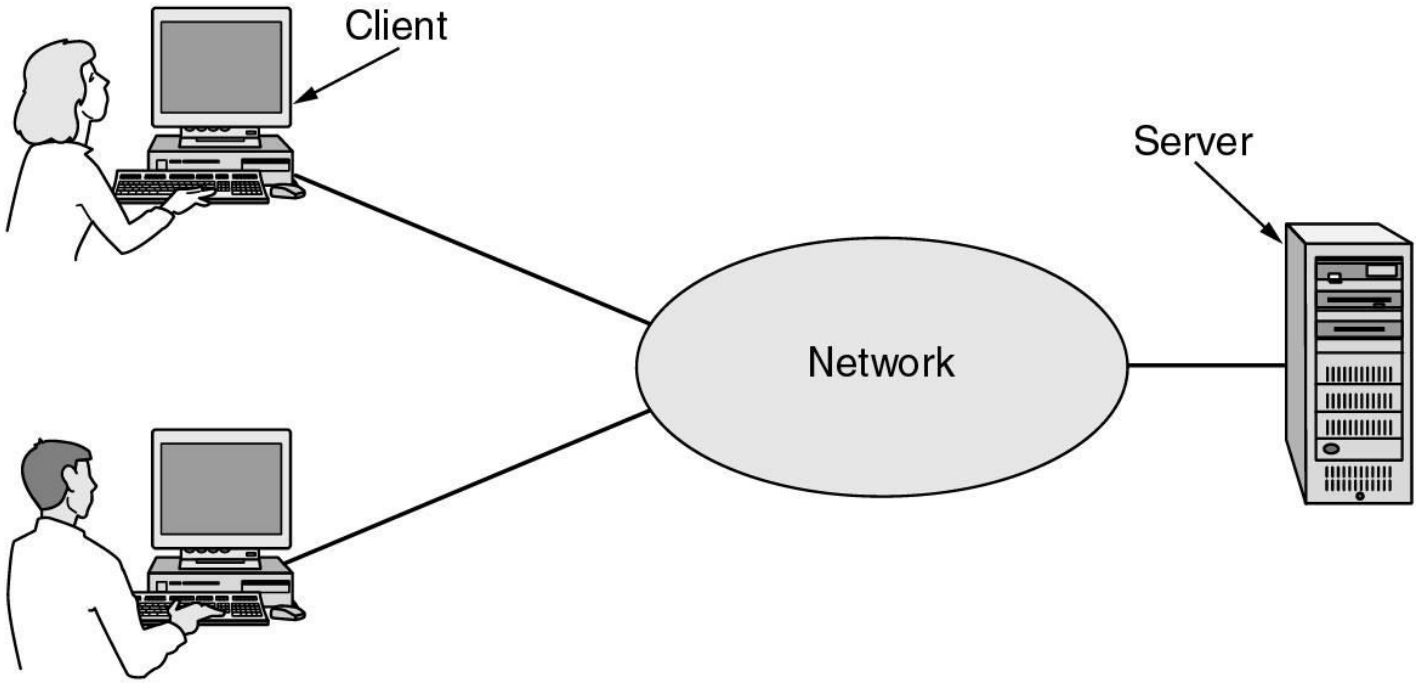
- The computers can be geographically located anywhere.



USES OF COMPUTER NETWORKS

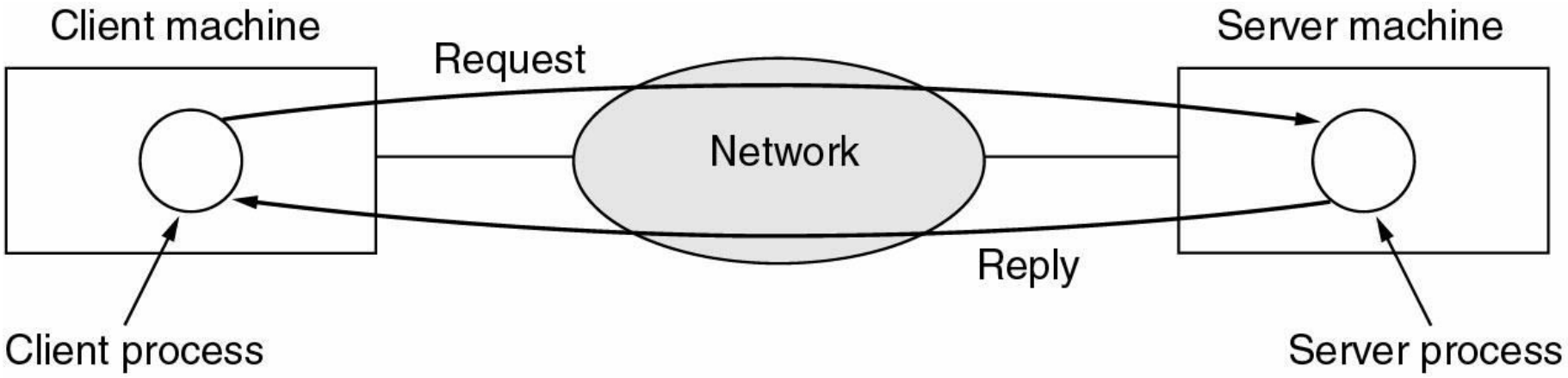
- Business Applications
- Home Applications
- Mobile Users
- Social Issues

BUSINESS APPLICATIONS OF NETWORKS



A network with two clients and one server.

BUSINESS APPLICATIONS OF NETWORKS

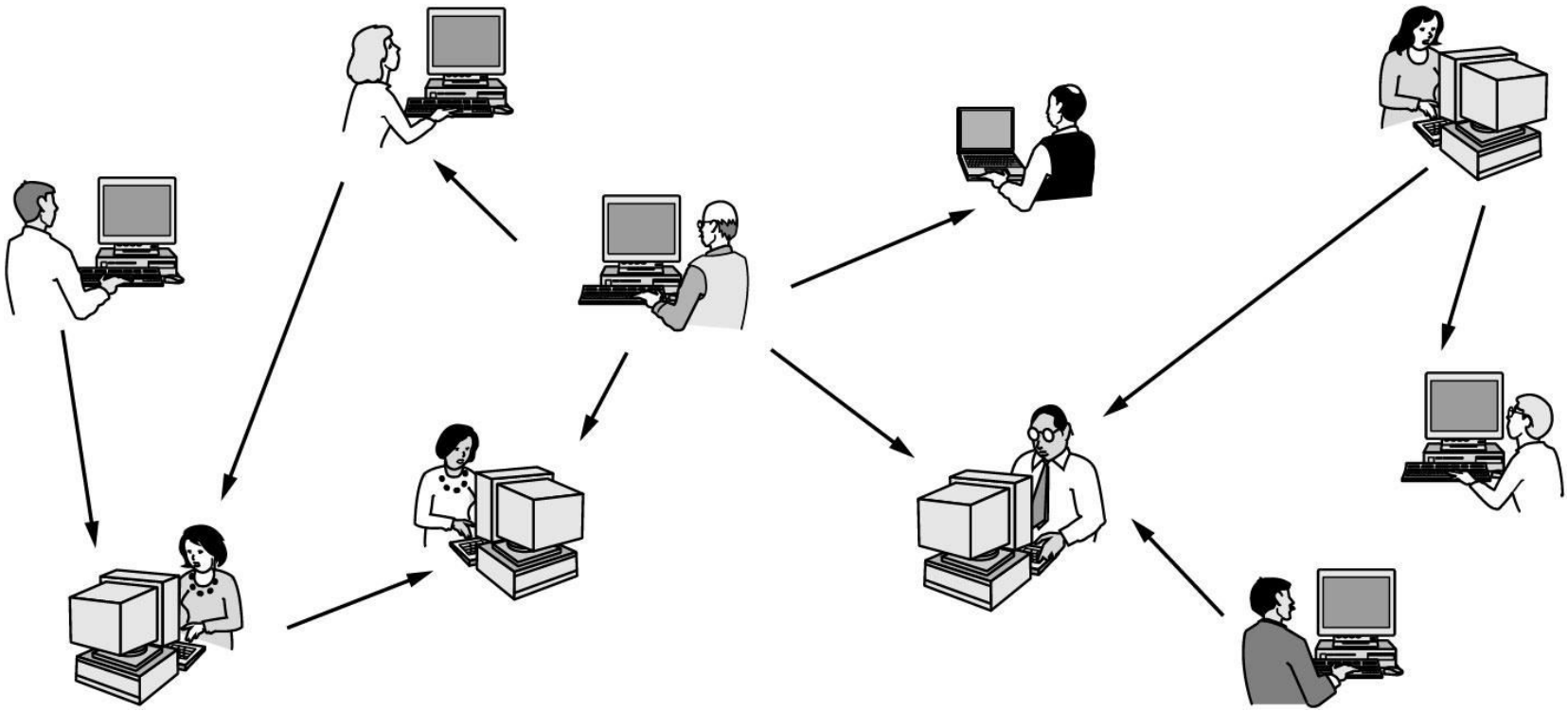


The client-server model involves requests and replies.

Home Network Applications

- Access to remote information
- Person-to-person communication
- Interactive entertainment
- Electronic commerce

Home Network Applications



In peer-to-peer system there are no fixed clients and servers.

Home Network Applications

Tag	Full name	Example
B2C	Business-to-consumer	Ordering books on-line
B2B	Business-to-business	Car manufacturer ordering tires from supplier
G2C	Government-to-consumer	Government distributing tax forms electronically
C2C	Consumer-to-consumer	Auctioning second-hand products on-line
P2P	Peer-to-peer	File sharing

Some forms of e-commerce.

MOBILE NETWORK USERS

Wireless	Mobile	Applications
No	No	Desktop computers in offices
No	Yes	A notebook computer used in a hotel room
Yes	No	Networks in older, unwired buildings
Yes	Yes	Portable office; PDA for store inventory

PDA: Personal Digital Assistants

Combinations of wireless networks and mobile computing.

SOCIAL ISSUES

People to share their views with like minded people

NETWORK HARDWARE

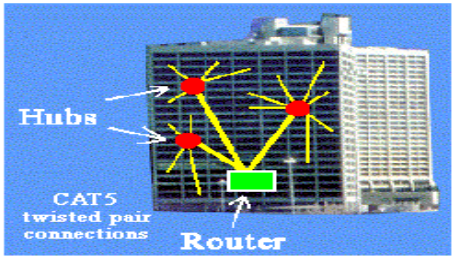
Classifying Networks

1. Types of transmission technology
   - Broadcast links
   - Point-to-point links
2. Scale
   - Local Area Networks
   - Metropolitan Area Networks
   - Wide Area Networks
   - Wireless Networks
   - Home Networks
   - Internetworks

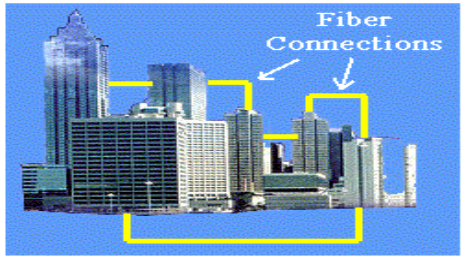
CLASSIFICATION –SCALE

Interprocessor distance	Processors located in same	Example
1 m	Square meter	Personal area network
10 m	Room	Local area network
100 m	Building	
1 km	Campus	
10 km	City	Metropolitan area network
100 km	Country	Wide area network
1000 km	Continent	
10,000 km	Planet	The Internet

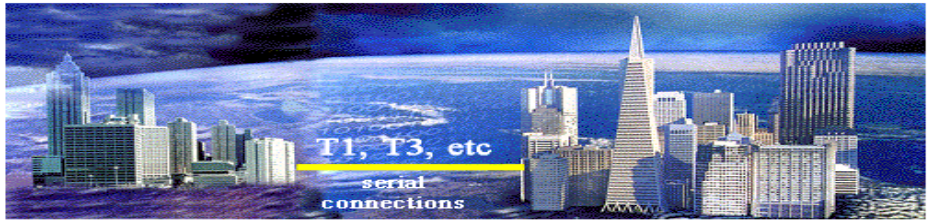
LAN,MAN,WAN



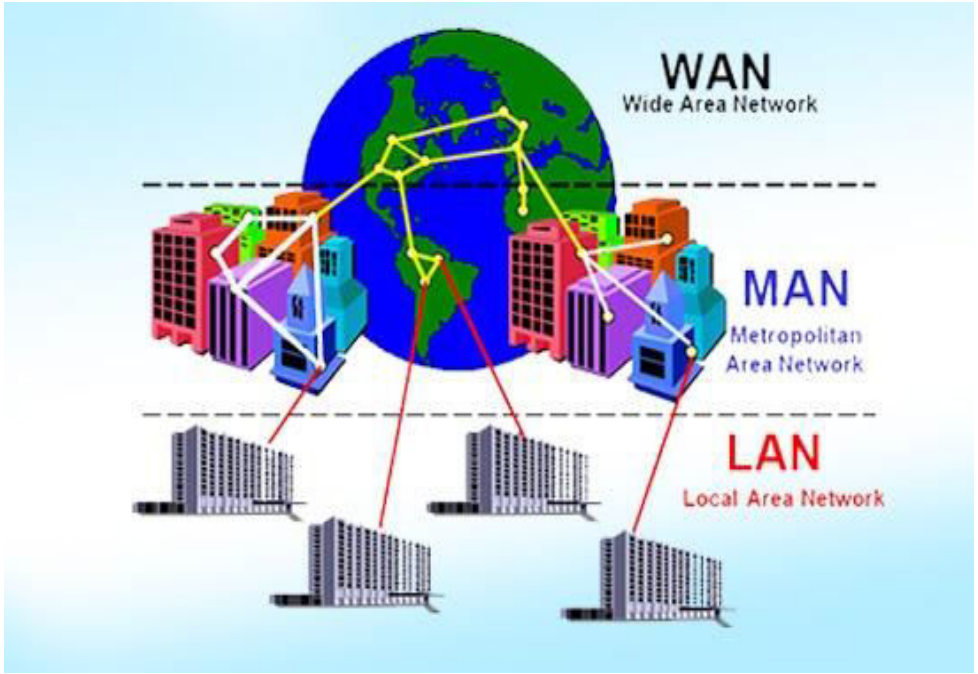
LAN



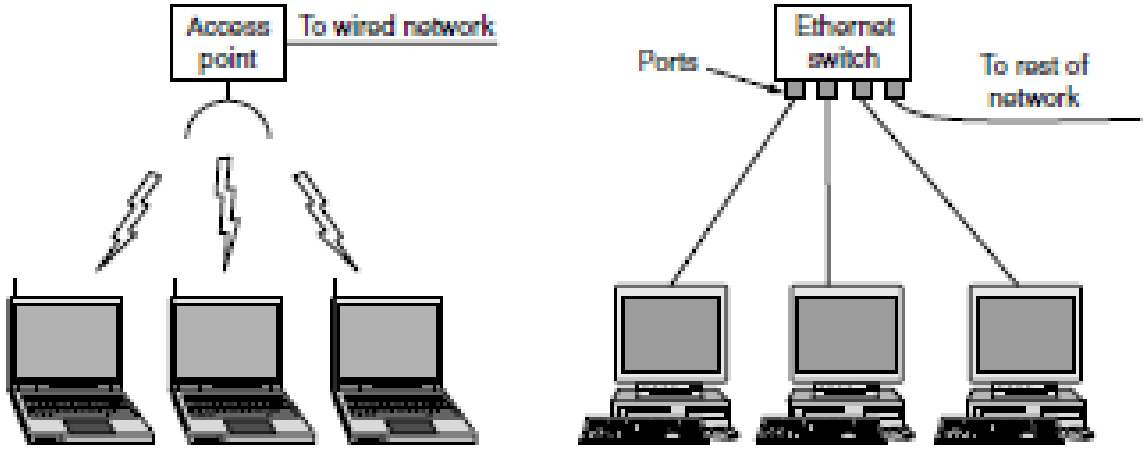
MAN



WAN

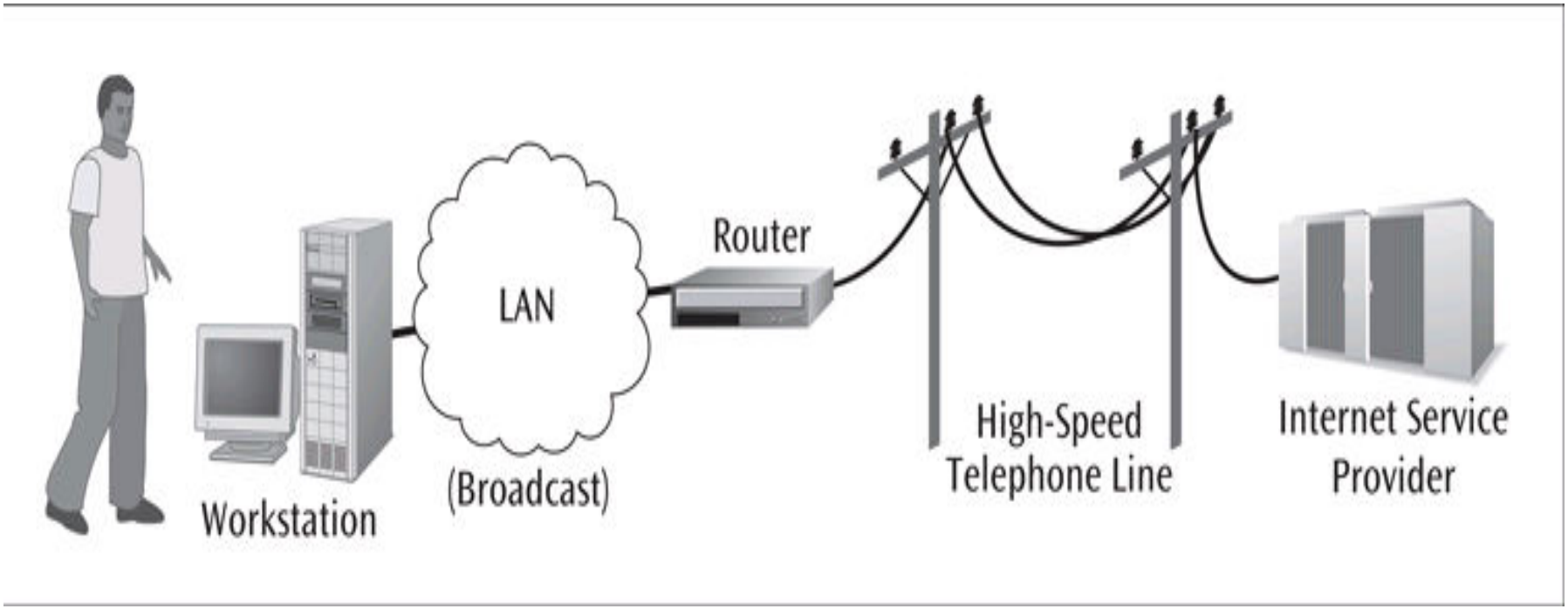


LAN



Wireless-LAN	Wired-LAN
CSMA/CA	CSMA/CD
IEEE 802.11	IEEE 802.3
Lower Speed	High Speed upto Gbps
Wireless Router, Access point	Router, Switch, Hub
Greater Mobility	Greater Security
Air	Copper, Fibre etc.,

USER –ACCESSING INTERNET THROUGH LAN



LAN-ADVANTAGES & DISADVANTAGES

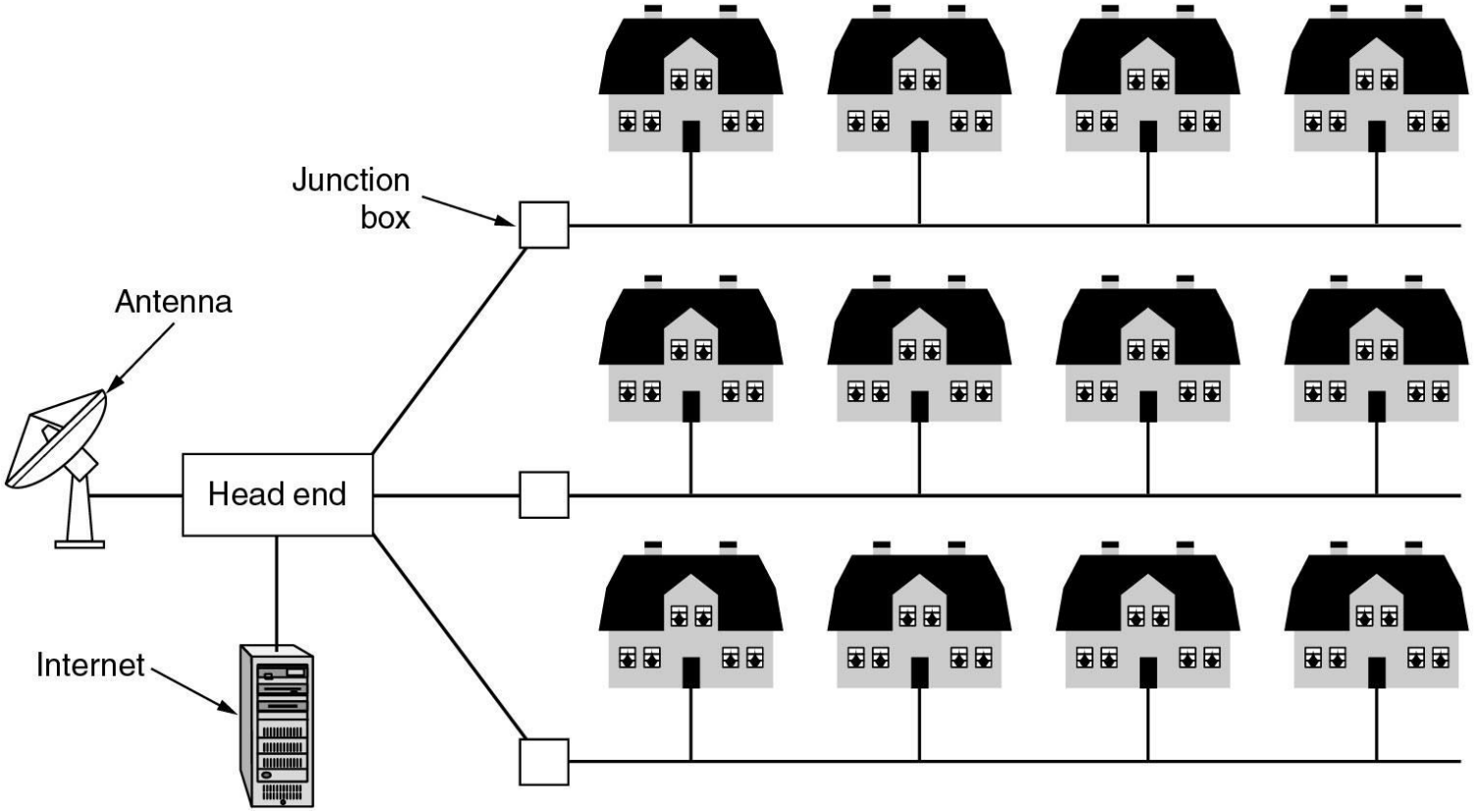
Advantages

- Resource Sharing
- Easy and Cheap Communication
- Centralized Data
- Automate Communication and Manufacturing process

Disadvantages

- High Setup Cost
- LAN maintenance job
- Covers limited area
- All users will be affected if the server problem

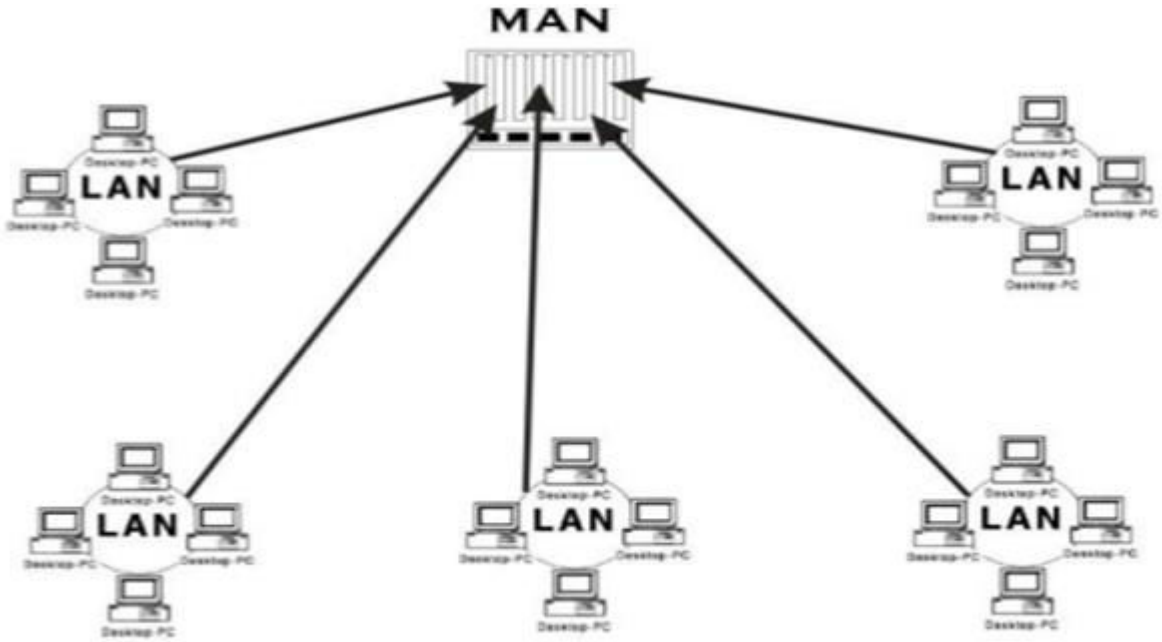
MAN



A metropolitan area network based on cable TV.

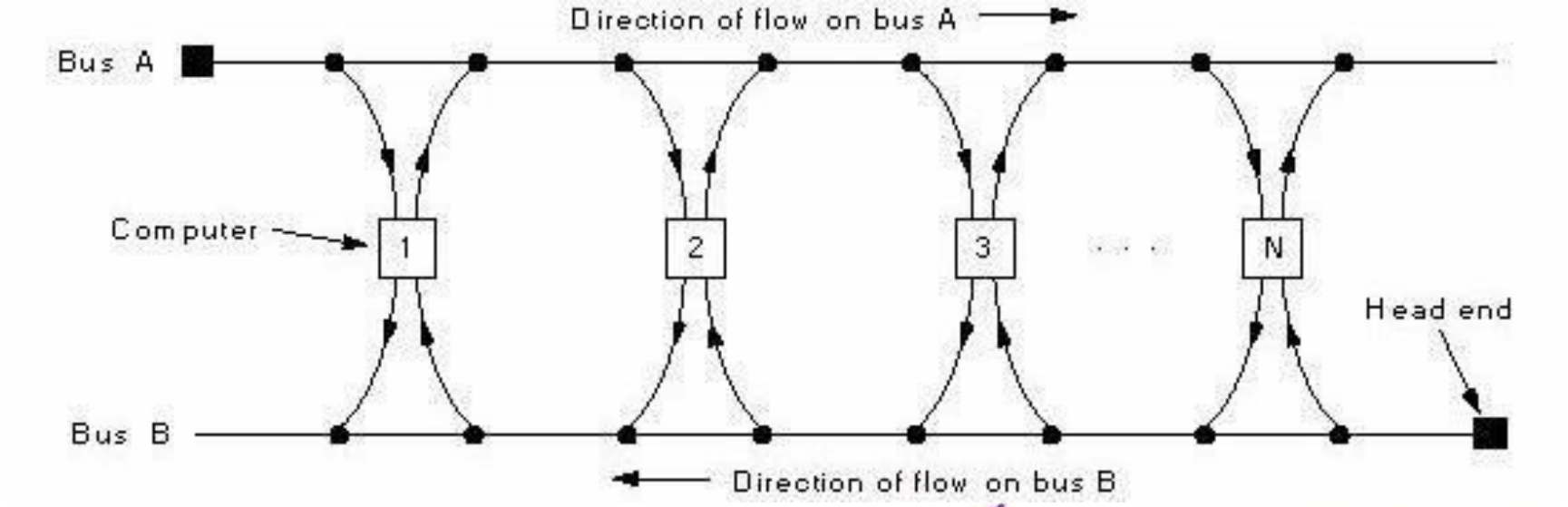
MAN

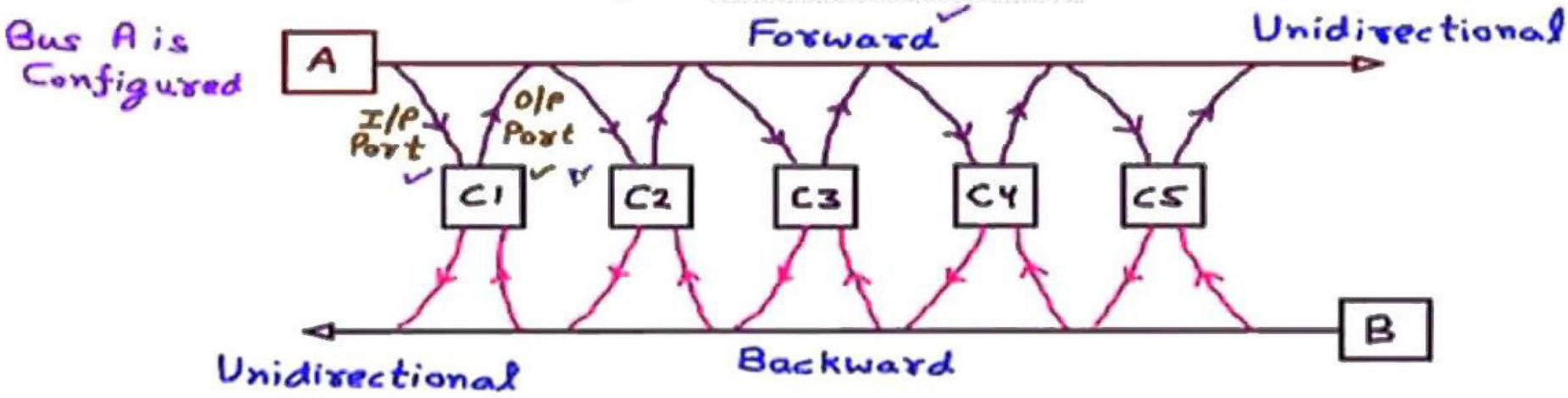
- The metropolitan area network (MAN) is designed to extend over an entire city.
- It may be a single network such as cable television network available in many cities.
- Range: Within 100 km (a city).



MAN TECHNOLOGY-DQDB

IEEE 802.6 standard uses the Distributed Queue Dual Bus(DQDB) network form. This form supports 150 Mbit/s transfer rates. It consists of two unconnected unidirectional buses.





MAN-Advantages & Disadvantages

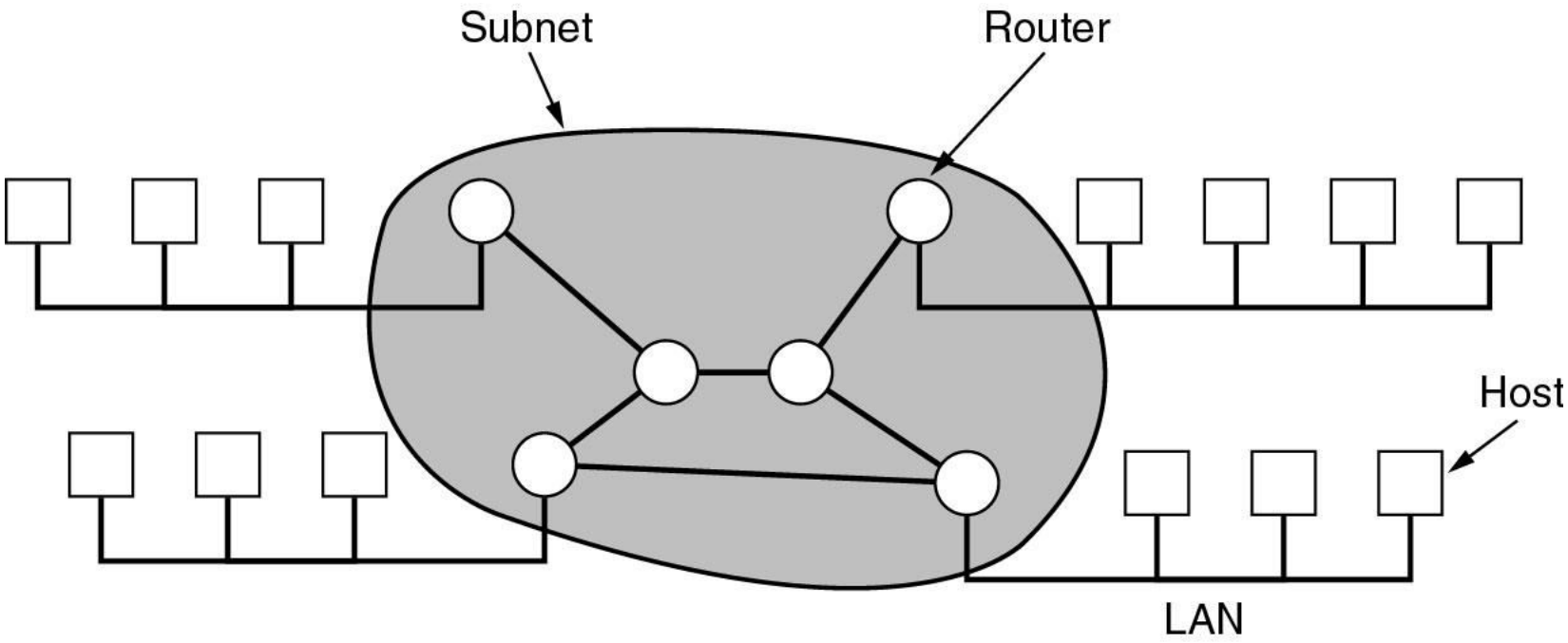
Advantages

- Good Backbone for a larger network and provides greater access to WANs.
- Sharing of the internet.
- Local E-mails.

Disadvantages

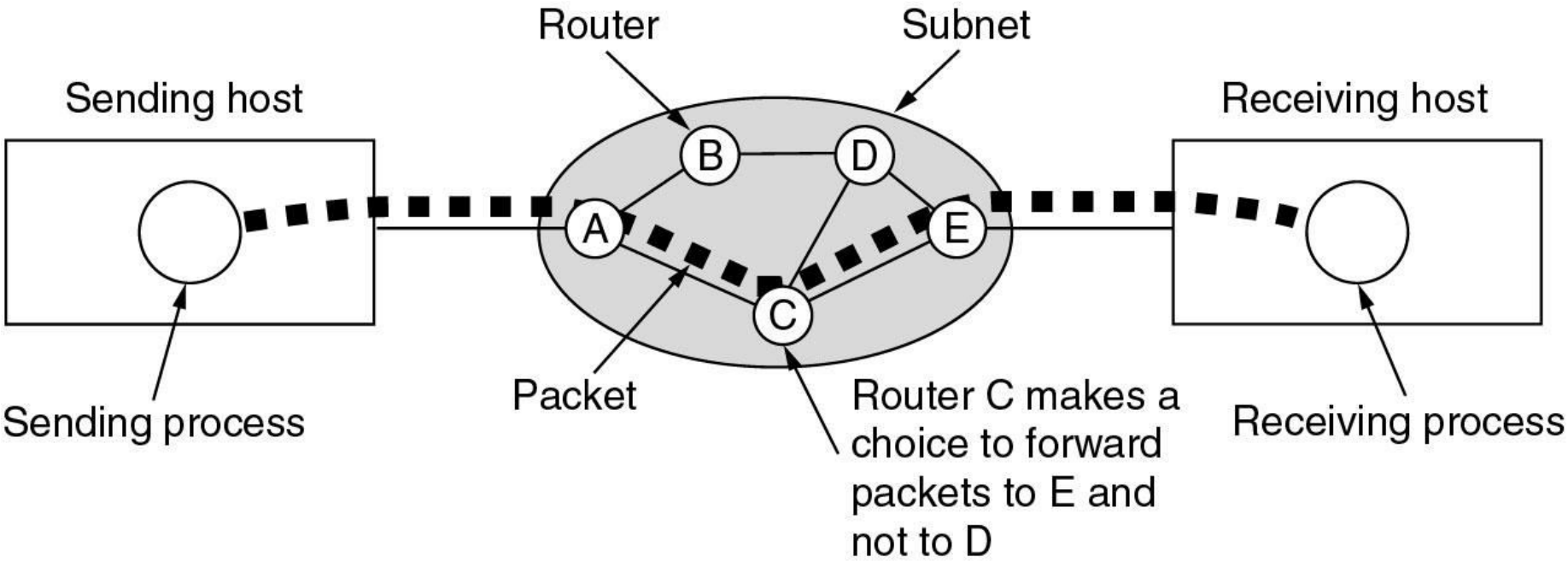
- More Cable is required.
- Difficult to make the system secure from hackers.
- Technical People required to establish.

WAN



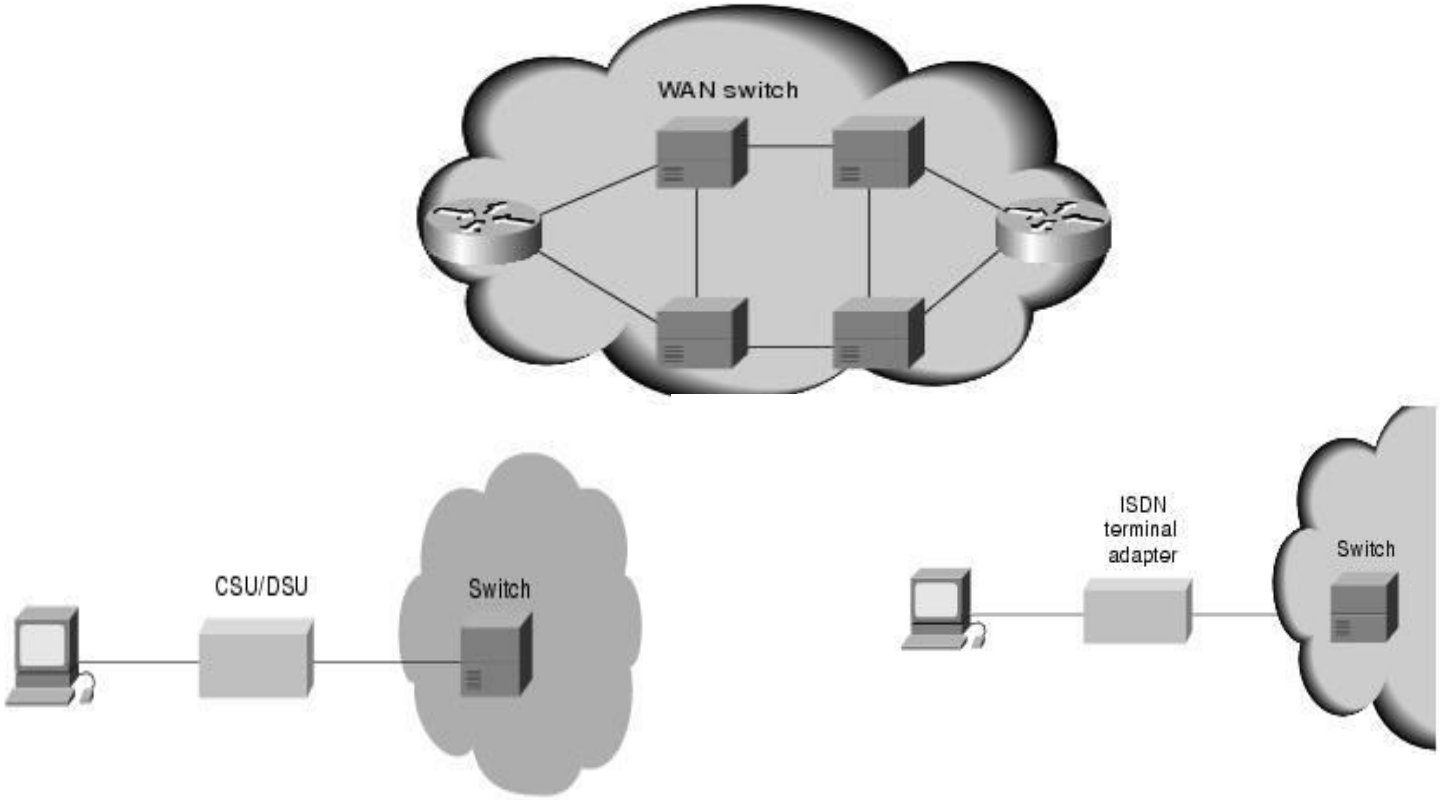
Relation between hosts on LANs and the subnet.

WAN



A stream of packets from sender to receiver.

WAN TECHNOLOGY



*A channel service unit/digital service unit (CSU/DSU) is a digital-interface device used to connect a router to a digital circuit like a T1. The CSU/DSU also provides signal timing for communication between these devices.

*An ISDN terminal adapter is a device used to connect ISDN Basic Rate Interface (BRI) connections to other interfaces, such as EIA/TIA-232 on a router.

*Telecommunication Industry Association /Electronic Industries Association.

WAN-ADVANTAGES & DISADVANTAGES

Advantages

- Covers Large Geographical Area
- Centralizes IT infrastructure
- Increases Bandwidth.

Disadvantages

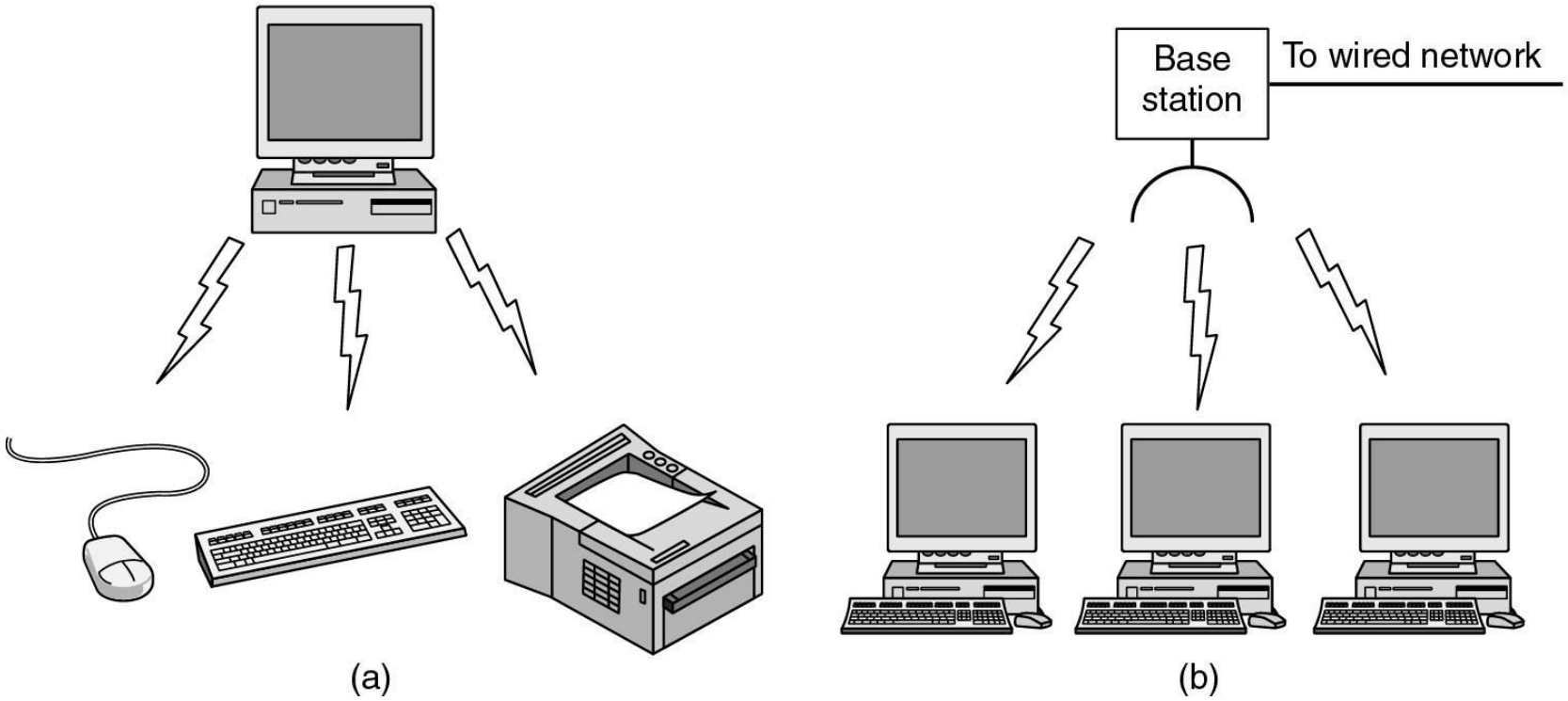
- Need a good Firewall.
- High setup Costs.
- Maintenance issues-Full time job.

WIRELESS NETWORKS

Categories of wireless networks:

- System interconnection
- Wireless LANs
- Wireless WANs

WIRELESS NETWORKS

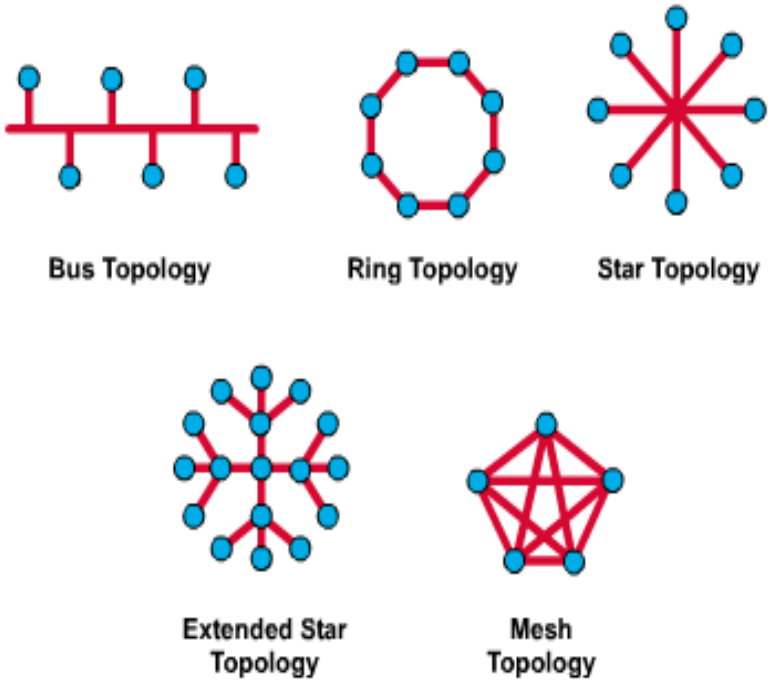


(a) Bluetooth configuration — IEEE 802.15.1

(b) Wireless LAN — IEEE 802.11

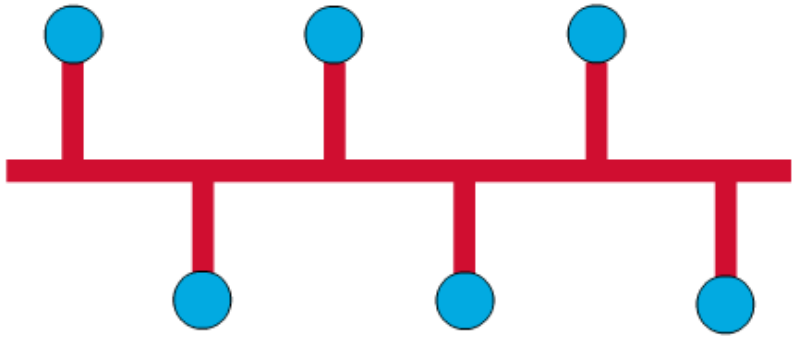
NETWORK TOPOLOGY

- **The network topology defines the way in which computers, printers, and other devices are connected. A network topology describes the layout of the wire and devices as well as the paths used by data transmissions.**



Bus Topology

- Commonly referred to as a linear bus, all the devices on a bus topology are connected by one single cable.



BUS

Advantages

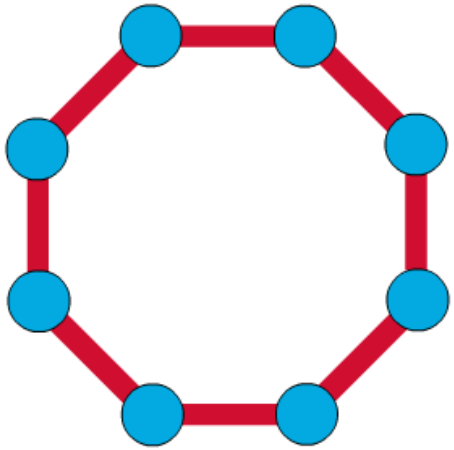
- It is cost effective.
- Cable required is least compared to other network topology.
- Used in small networks.
- It is easy to understand.
- Easy to expand joining two cables together.

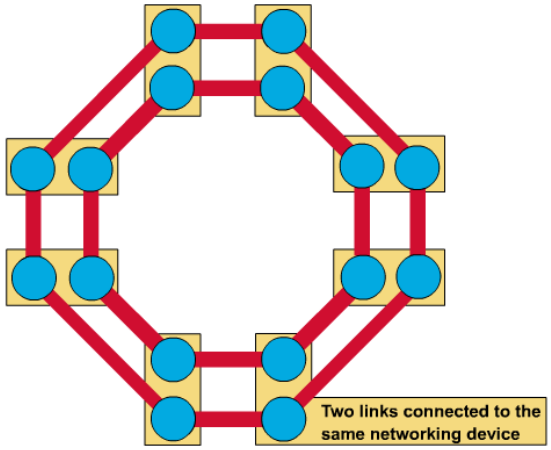
Disadvantages

- Cables fails then whole network fails.
- If network traffic is heavy or nodes are more the performance of the network decreases.
- Cable has a limited length.

Ring Topology

- **A frame travels around the ring, stopping at each node. If a node wants to transmit data, it adds the data as well as the destination address to the frame.**
- **The frame then continues around the ring until it finds the destination node, which takes the data out of the frame.**
  - Single ring – All the devices on the network share a single cable
  - Dual ring – The dual ring topology allows data to be sent in both directions.





RING

Advantages

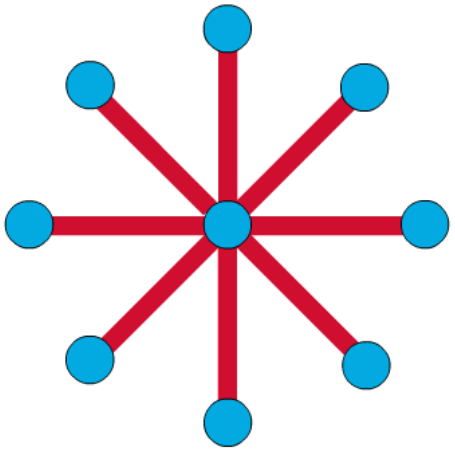
- Transmitting network is not affected by high traffic or by adding more nodes, as only the nodes having tokens can transmit data.
- Cheap to install and expand

Disadvantages

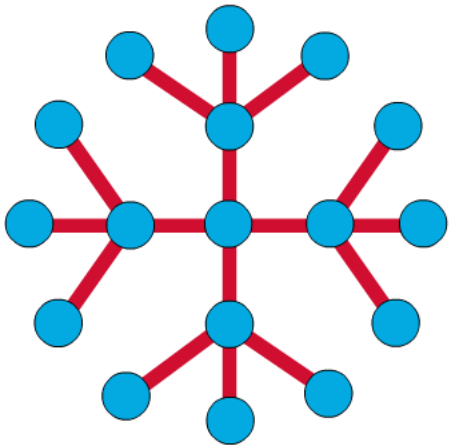
- Troubleshooting is difficult in ring topology.
- Adding or deleting the computers disturbs the network activity.
- Failure of one computer disturbs the whole network.

Star & Tree Topology

- The star topology is the most commonly used architecture in Ethernet LANs.



- When installed, the star topology resembles spokes in a bicycle wheel.
- Larger networks use the extended star topology also called tree topology. When used with network devices that filter frames or packets, like bridges, switches, and routers, this topology significantly reduces the traffic on the wires by sending packets only to the wires of the destination host.



STAR

Advantages

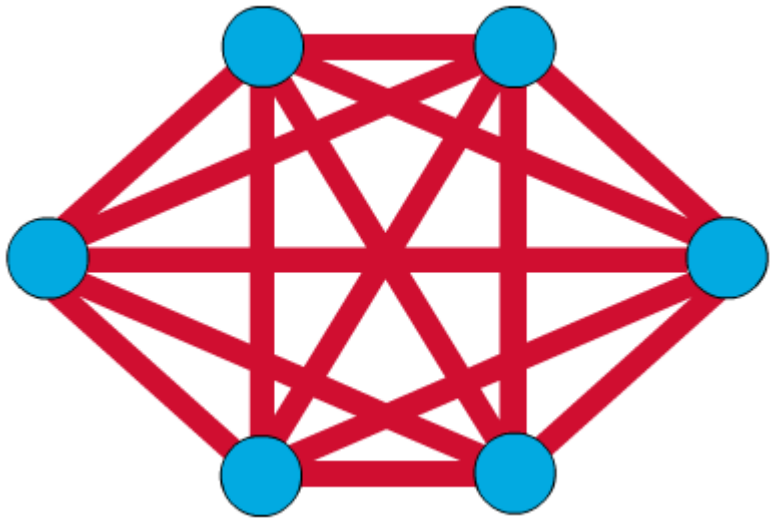
- Fast performance with few nodes and low network traffic.
- Hub can be upgraded easily.
- Easy to troubleshoot.
- Easy to setup and modify.
- Only that node is affected which has failed, rest of the nodes can work smoothly.
- Each device require only 1 port i.e. to connect to the hub.

Disadvantages

- Cost of installation is high.
- Expensive to use.
- If the hub fails then the whole network is stopped because all the nodes depend on the hub.
- Performance is based on the hub that is it depends on its capacity

Mesh Topology

- The mesh topology connects all devices (nodes) to each other for redundancy and fault tolerance.
- It is used in WANs to interconnect LANs and for mission critical networks like those used by banks and financial institutions.
- Implementing the mesh topology is expensive and difficult.



MESH

N number of devices are connected with each other in mesh topology, then total number of dedicated links required to connect them is N(N-1)/2

No. of Ports for each device is N-1

PROBLEMS

For n devices in a network, what is the number of cable links required for a mesh, ring,, and star topology?

n(n-1)/2 cable link are required for mesh, n for ring, and n cable link for star topology.

Assume six devices are arranged in a mesh topology. How many cables are needed? How many ports are needed for each device? How many total ports

Cables needed
(6*5)/2 = 15

Each device needs to be connected to 5 other devices. So, each device needs to have 5 ports.

Six devices times five ports equals 30 total ports.

MESH

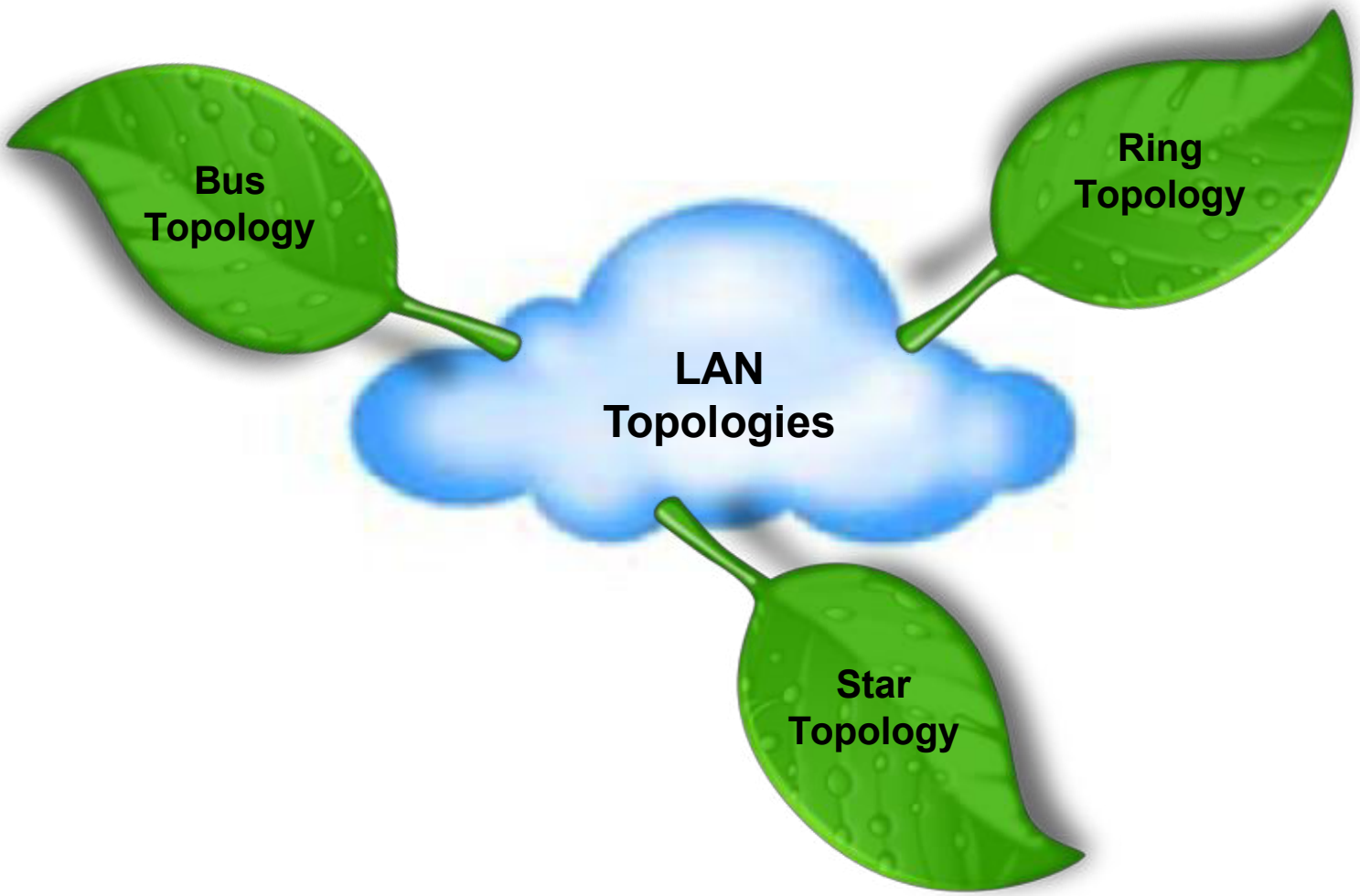
Advantages

- Each connection can carry its own data load.
- It is robust.
- Fault is diagnosed easily.
- Provides security and privacy.

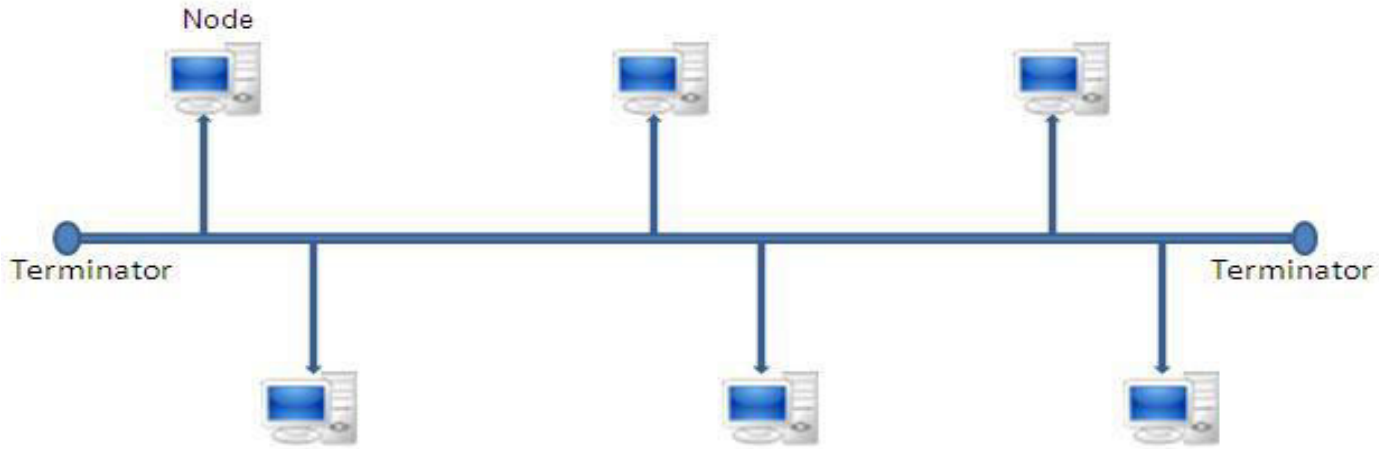
Disadvantages

- Installation and configuration is difficult.
- Cabling cost is more.
- Bulk wiring is required.

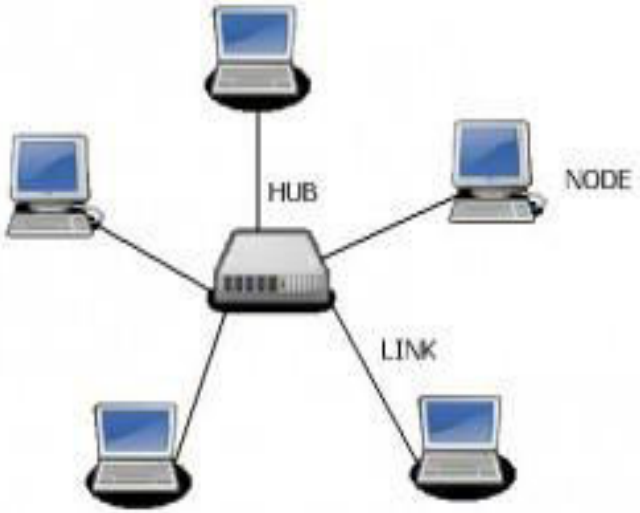
LAN-TOPOLOGIES



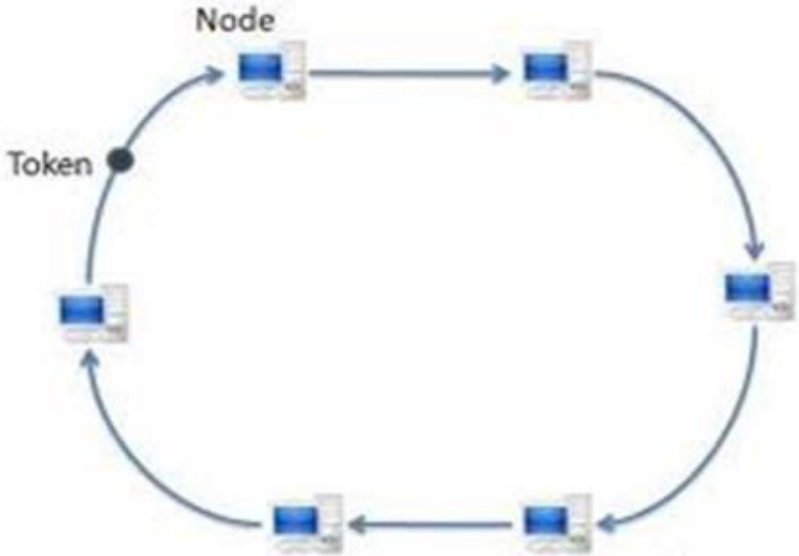
LAN-TOPOLOGIES



Bus Topology



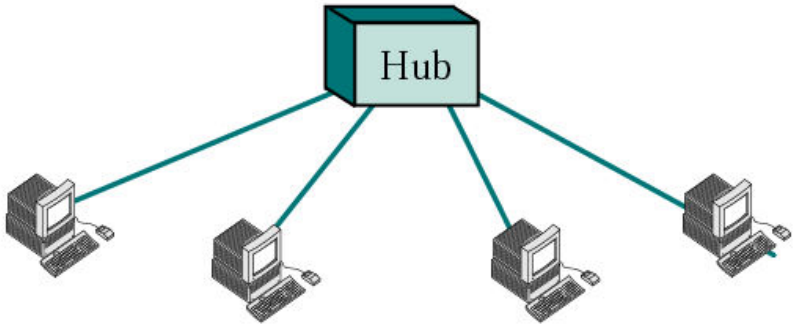
STAR TOPOLOGY



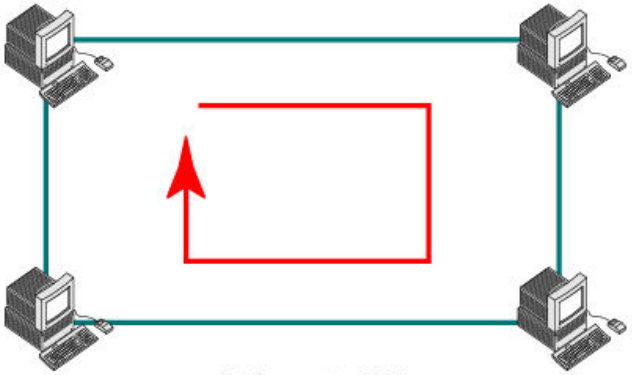
Ring Topology

LAN-TOPOLOGIES

a. Bus LAN



b. Star LAN



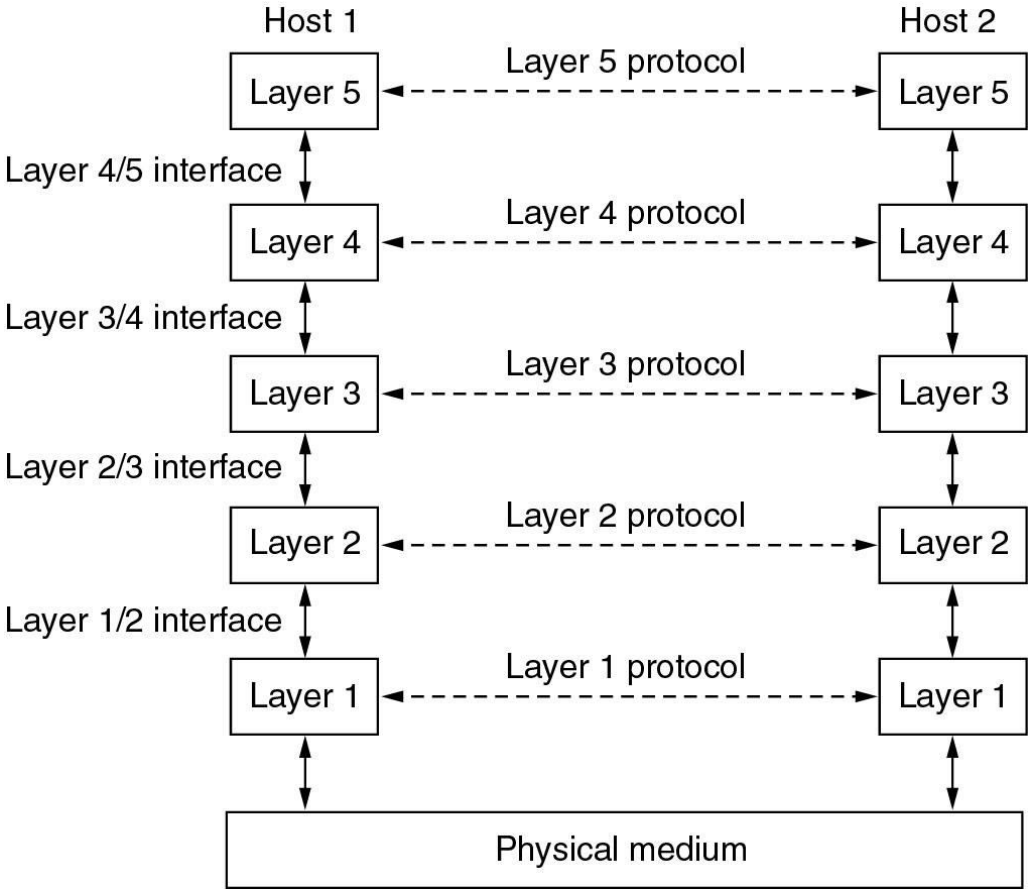
c. Ring LAN

NETWORK SOFTWARE

- Protocol Hierarchies
- Design Issues for the Layers
- Connection-Oriented and Connectionless Services
- Service Primitives
- The Relationship of Services to Protocols

NETWORK SOFTWARE

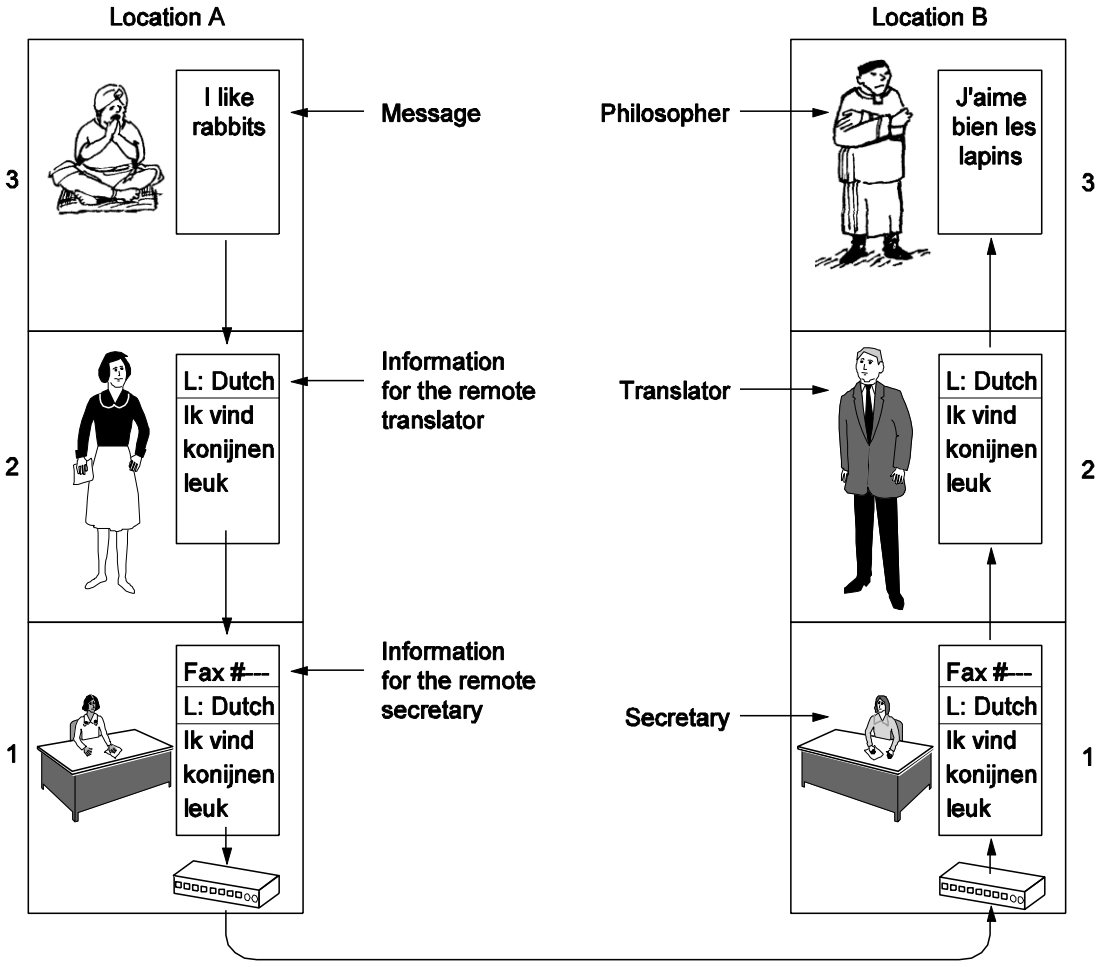
Protocol Hierarchies



Layers, protocols, and interfaces.

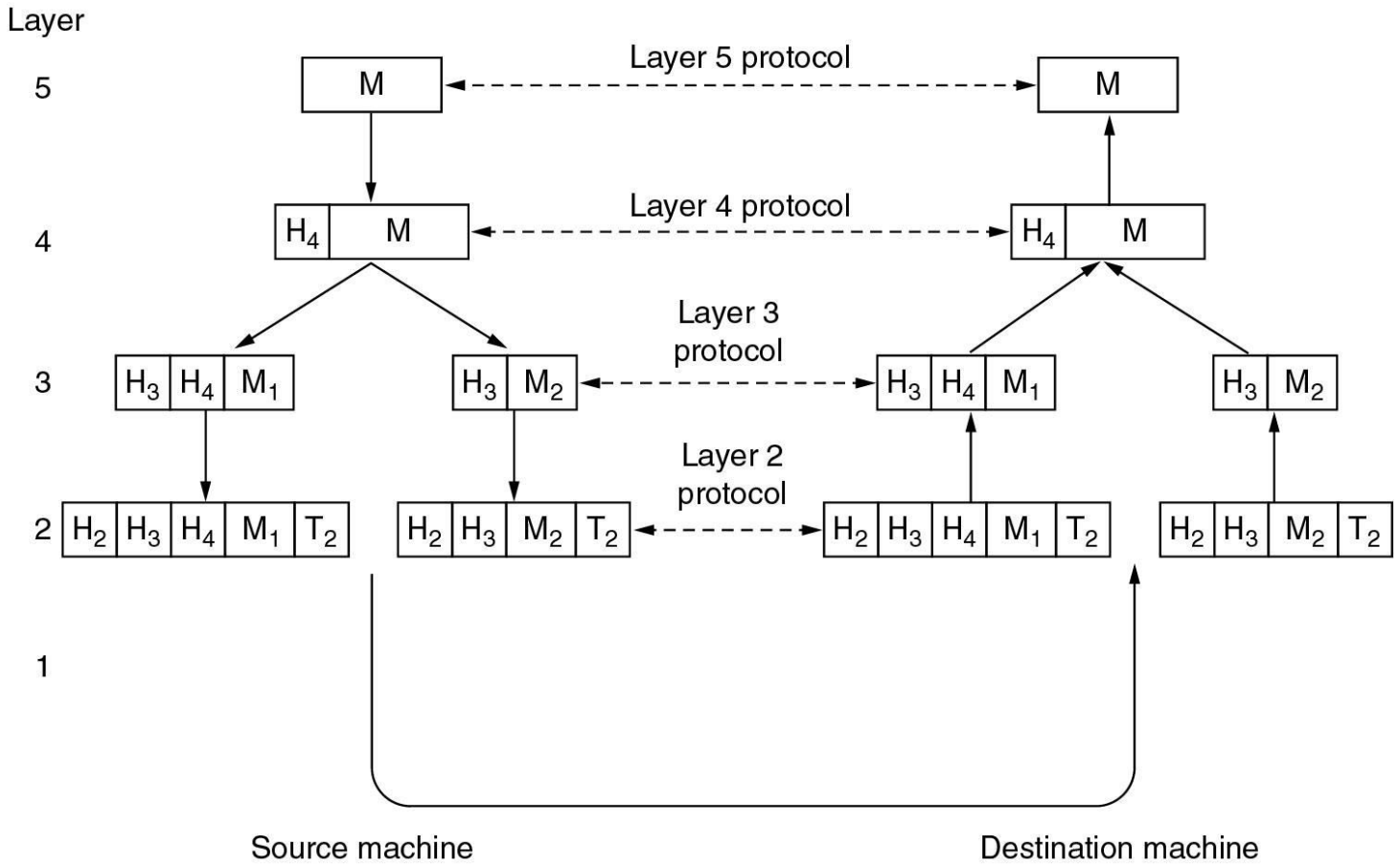
Interface: Interface defines which primitive operations and services the lower layer makes available to the upper layer

Protocol Hierarchies



The philosopher-translator-secretary architecture.

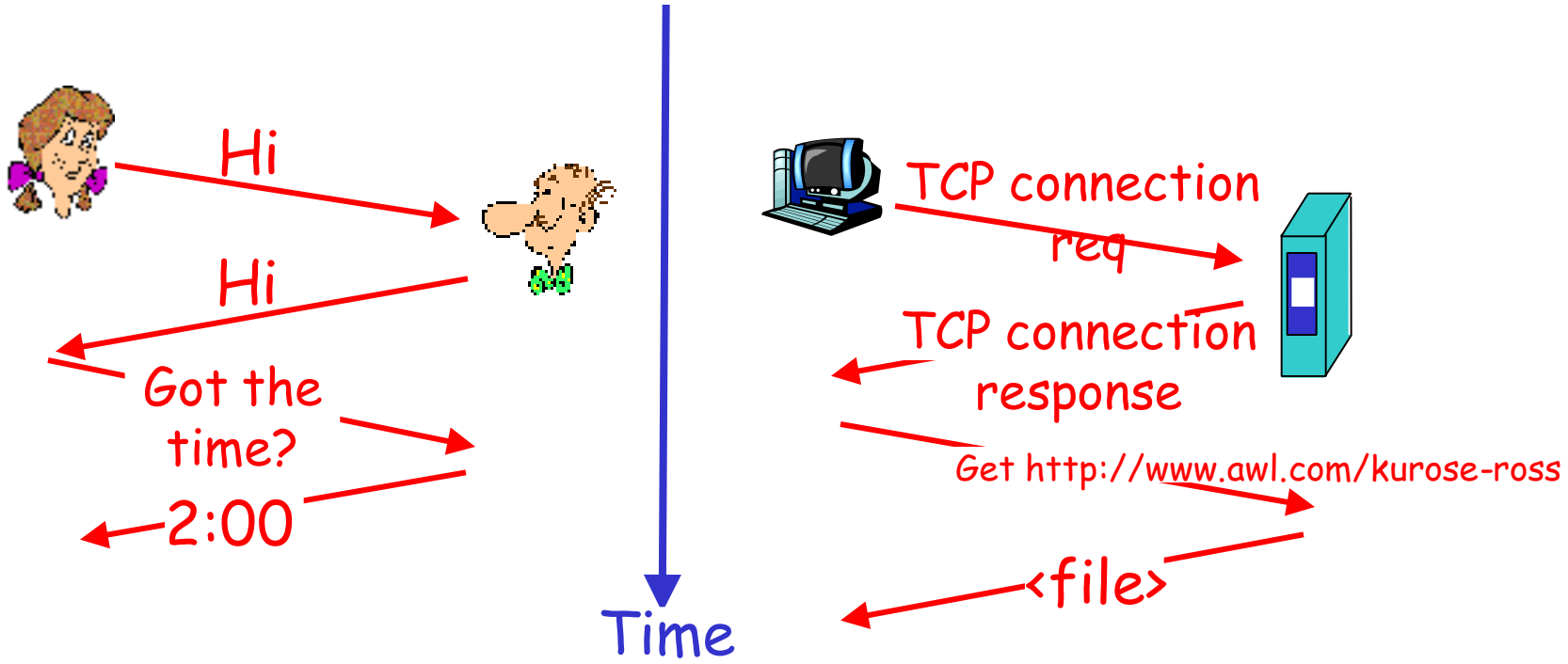
Protocol Hierarchies



Example information flow supporting virtual communication in layer 5.

What's a protocol?

A human protocol and a computer network protocol:



All activity in the Internet that involves two or more communicating remote entities is governed by a protocol. (Routing protocols, Congestion Control protocols, media access protocols, etc.)

DESIGN ISSUES FOR THE LAYERS

- Addressing
- Error Control
- Flow Control
- Multiplexing
- Routing

CONNECTION-ORIENTED & CONNECTIONLESS SERVICES

	Service	Example
Connection-oriented	Reliable message stream	Sequence of pages
	Reliable byte stream	Remote login
	Unreliable connection	Digitized voice
Connection-less	Unreliable datagram	Electronic junk mail
	Acknowledged datagram	Registered mail
	Request-reply	Database query

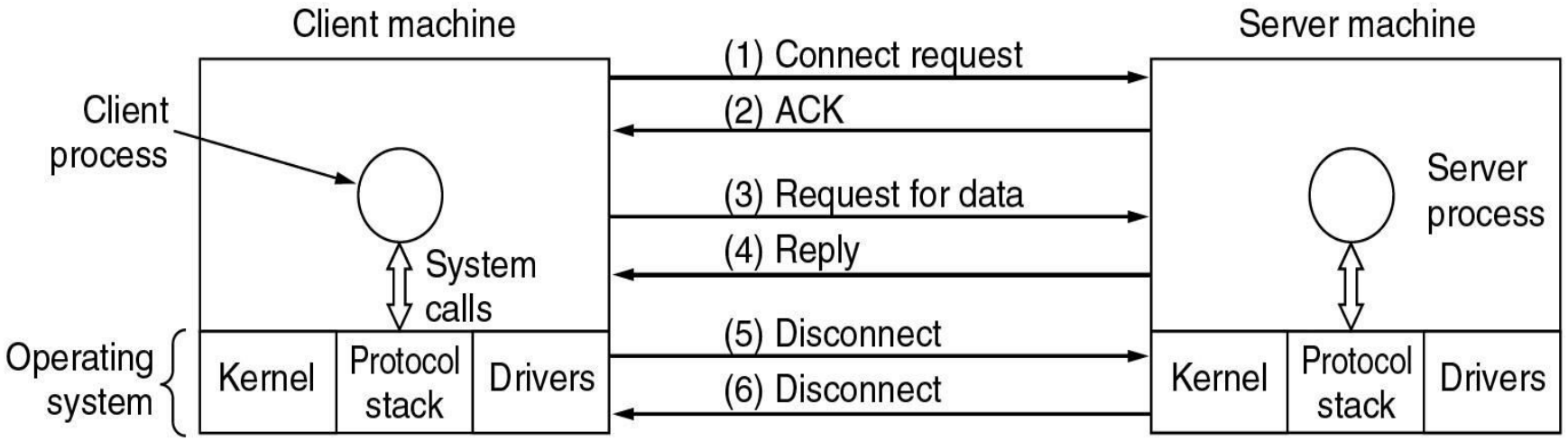
Six different types of service.

SERVICE PRIMITIVES

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

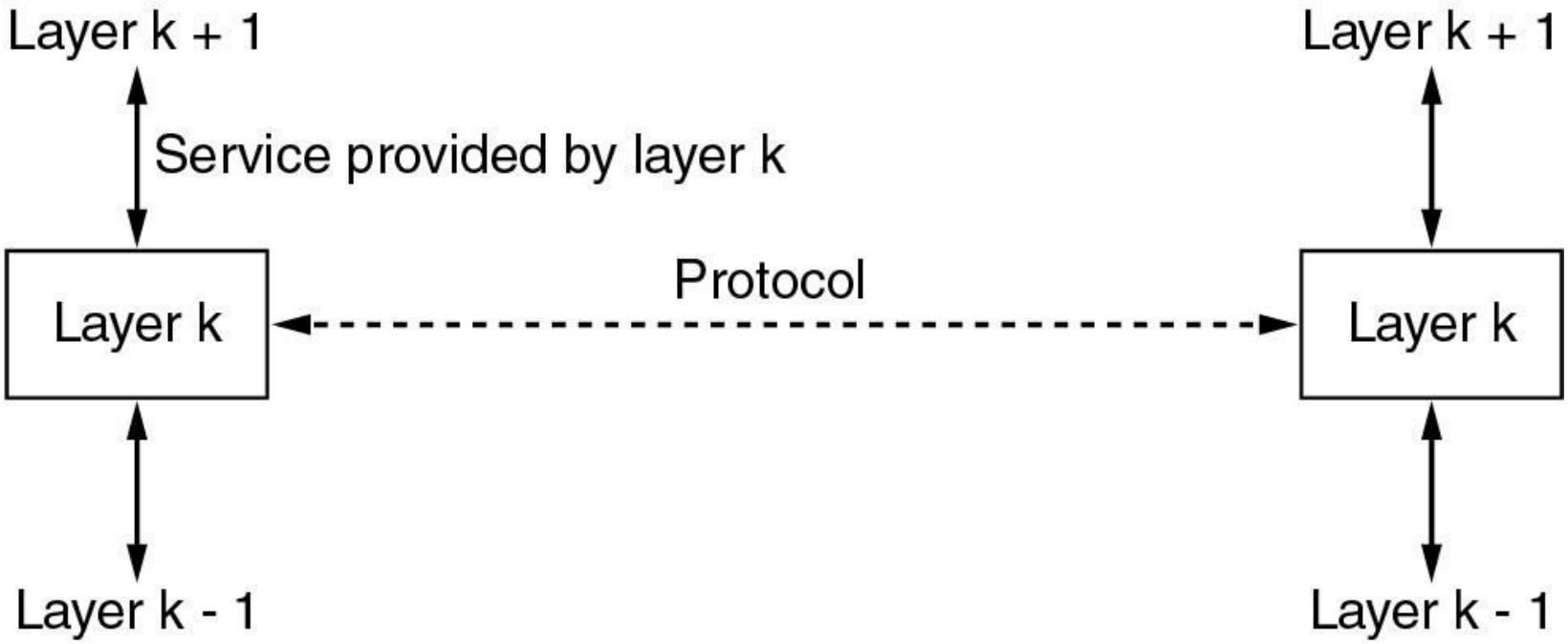
Five service primitives for implementing a simple connection-oriented service.

SERVICE PRIMITIVES



Packets sent in a simple client-server interaction on a connection-oriented network.

SERVICES TO PROTOCOLS RELATIONSHIP



The relationship between a service and a protocol.

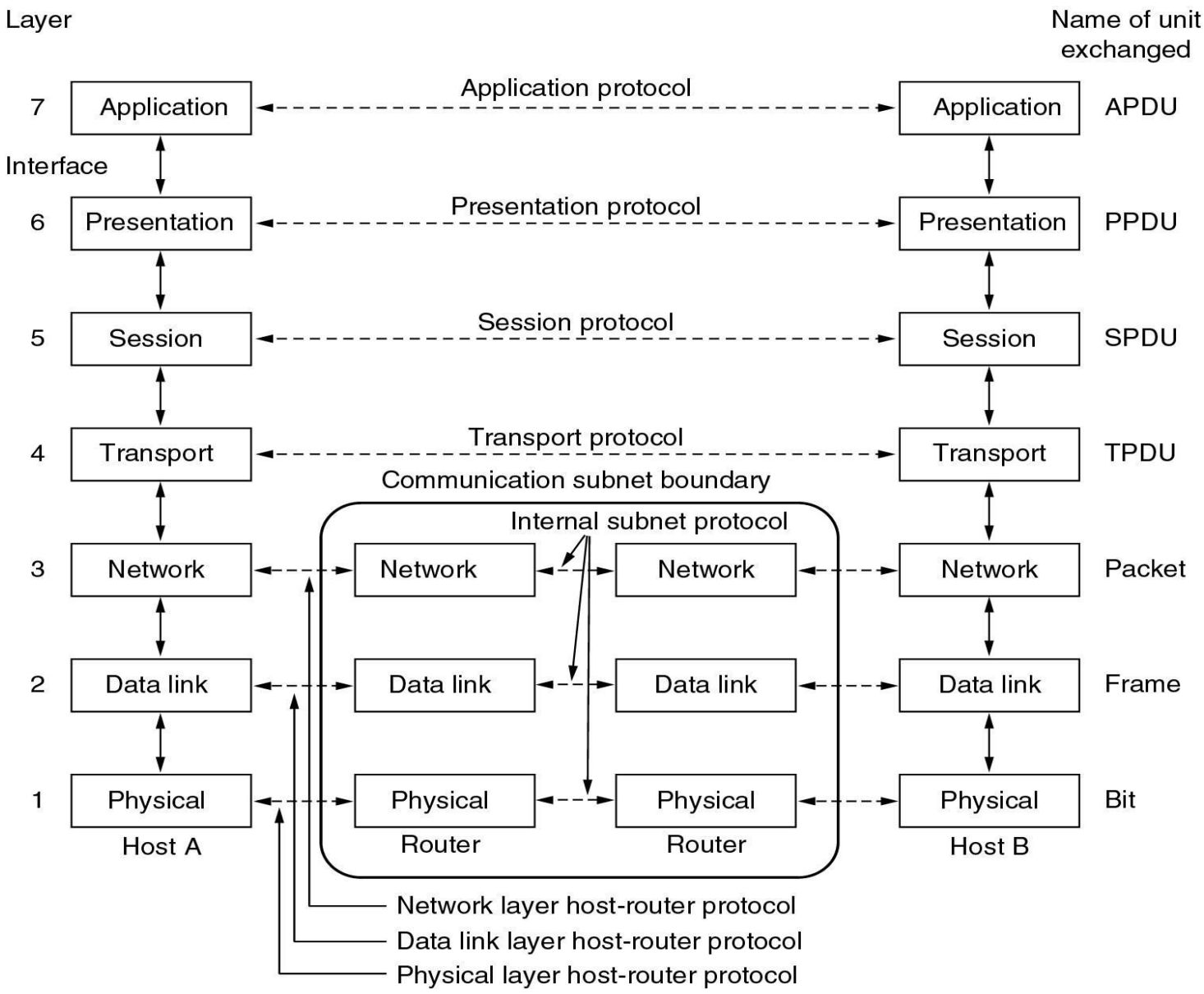
Service: A Service is a set of primitives that a layer provides to the layer above it.

REFERENCE MODELS

- The OSI Reference Model
- The TCP/IP Reference Model
- A Comparison of OSI and TCP/IP
- ATM Reference Model

OSI

The OSI reference model.



OSI

The Physical Layer : The **physical layer is concerned with transmitting raw bits over a communication** channel. The design issues have to do with making sure that when one side sends a 1 bit it is received by the other side as a 1 bit, not as a 0 bit. The design issues largely deal with mechanical, electrical, and timing interfaces, as well as the physical transmission medium, which lies below the physical layer.

The Data Link 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. It does so by masking the real errors so the network layer does not see them. It accomplishes this task by having the sender break up the input data into **data frames (typically** a few hundred or a few thousand bytes) and transmit the frames sequentially. If the service is reliable, the receiver confirms correct receipt of each frame by sending back an **acknowledgement frame.**

Another issue that arises in the data link layer (and most of the higher layers as well) is how to keep a fast transmitter from drowning a slow receiver in data. Some traffic regulation mechanism may be needed to let the transmitter know when the receiver can accept more data.

Broadcast networks have an additional issue in the data link layer: how to control access to the shared channel. A special sub-layer of the data link layer, the **medium access control sub-layer, deals with this problem.**

OSI

The Network Layer: A key design issue is determining how packets are routed from source to destination. Routes can be based on static tables that are "wired into" the network and rarely changed, or more often they can be updated automatically to avoid failed components. Handling congestion is also a responsibility of the network layer. More generally, the quality of service provided (delay, transit time, jitter, etc.) is also a network layer issue.

The 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. Furthermore, all this must be done efficiently and in a way that isolates the upper layers from the inevitable changes in the hardware technology over the course of time. The most popular type of transport connection is an error-free point-to-point channel that delivers messages or bytes in the order in which they were sent. However, other possible kinds of transport service exist, such as the transporting of isolated messages with no guarantee about the order of delivery, and the broadcasting of messages to multiple destinations. The transport layer is a true end-to-end layer; it carries data all the way from the source to the destination.

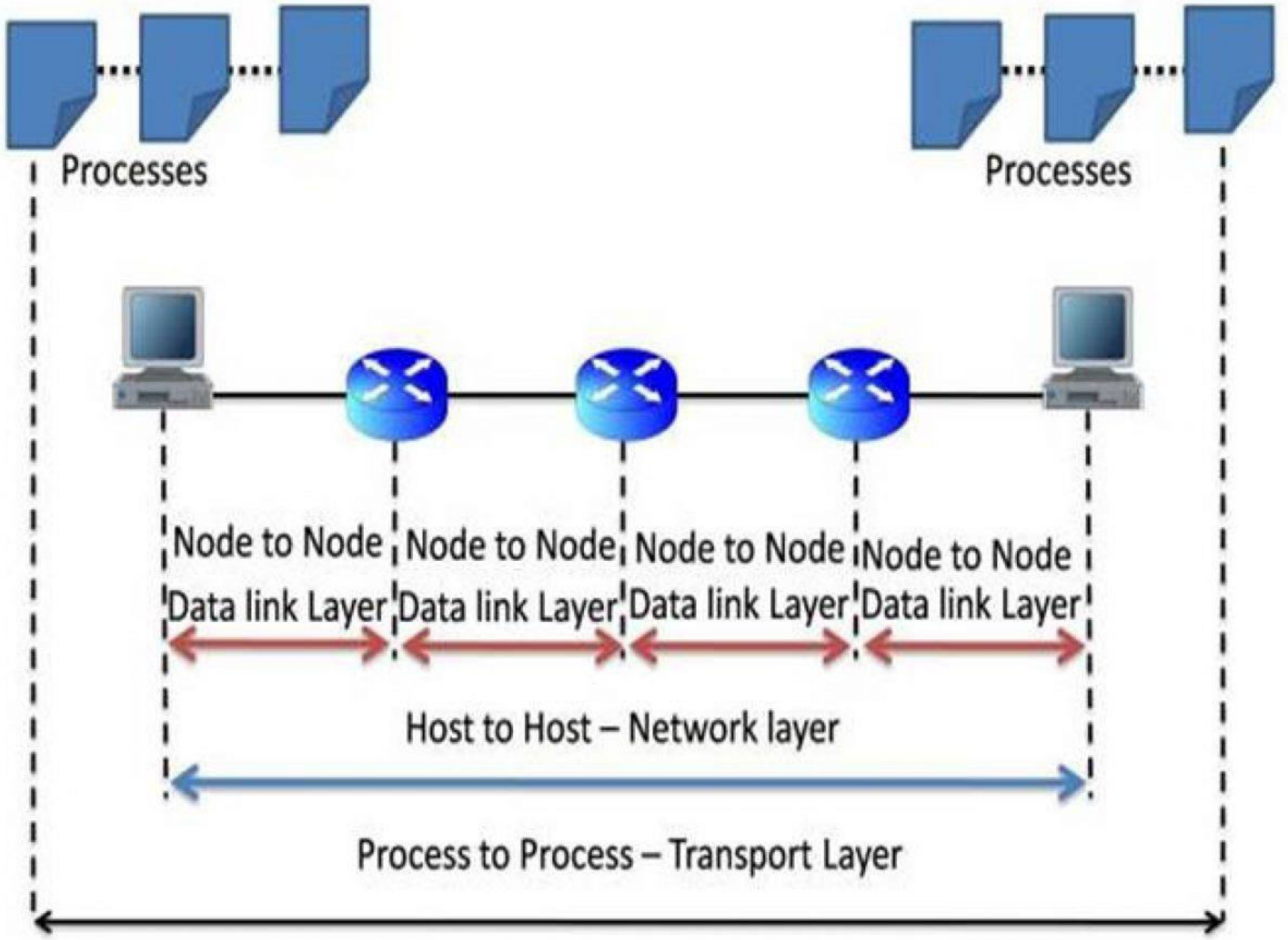
OSI

The 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 simultaneously), and **synchronization** (check pointing long transmissions to allow them to pick up from where they left off in the event of a crash and subsequent recovery).

.**The Presentation Layer:** Unlike the lower layers, which are mostly concerned with moving bits around, the **presentation layer is concerned with the syntax and semantics of the information** transmitted. In order to make it possible for computers with different internal data representations to communicate, the data structures to be exchanged can b 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.

The 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. When a** browser wants a Web page, it sends the name of the page it wants to the server hosting the page using HTTP. The server then sends the page back. Other application protocols are used for file transfer, electronic mail, and network news.

TL Vs DL


Processes
Processes
Node to Node
Data link Layer
Node to Node
Data link Layer
Node to Node
Data link Layer
Node to Node
Data link Layer
Host to Host – Network layer
Process to Process – Transport Layer

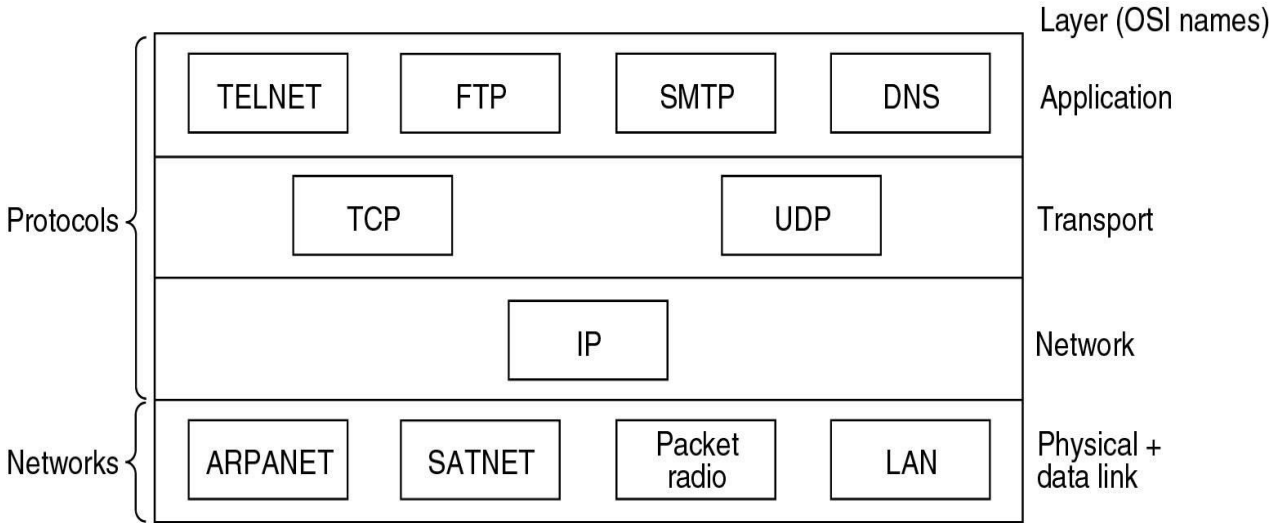
TCP/IP

Application

Transport

Internet

Link



Protocols and networks in the TCP/IP model initially.

The ARPANET was a research network sponsored by the DoD (U.S. Department of Defence). 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. Thus, from nearly the beginning, the ability to connect multiple networks in a seamless way was one of the major design goals. This architecture later became known as the **TCP/IP Reference Model, after its** two primary protocols. It was first described by Cerf and Kahn (1974), and later refined and defined as a standard in the Internet community (Braden, 1989). The design philosophy behind the model is discussed by Clark (1988).

TCP/IP

The Link Layer: The lowest layer in the model, the **link layer describes what links such as serial lines and classic Ethernet** must do to meet the needs of this connectionless internet layer. It is not really a layer at all, in the normal sense of the term, but rather an interface between hosts and transmission links.

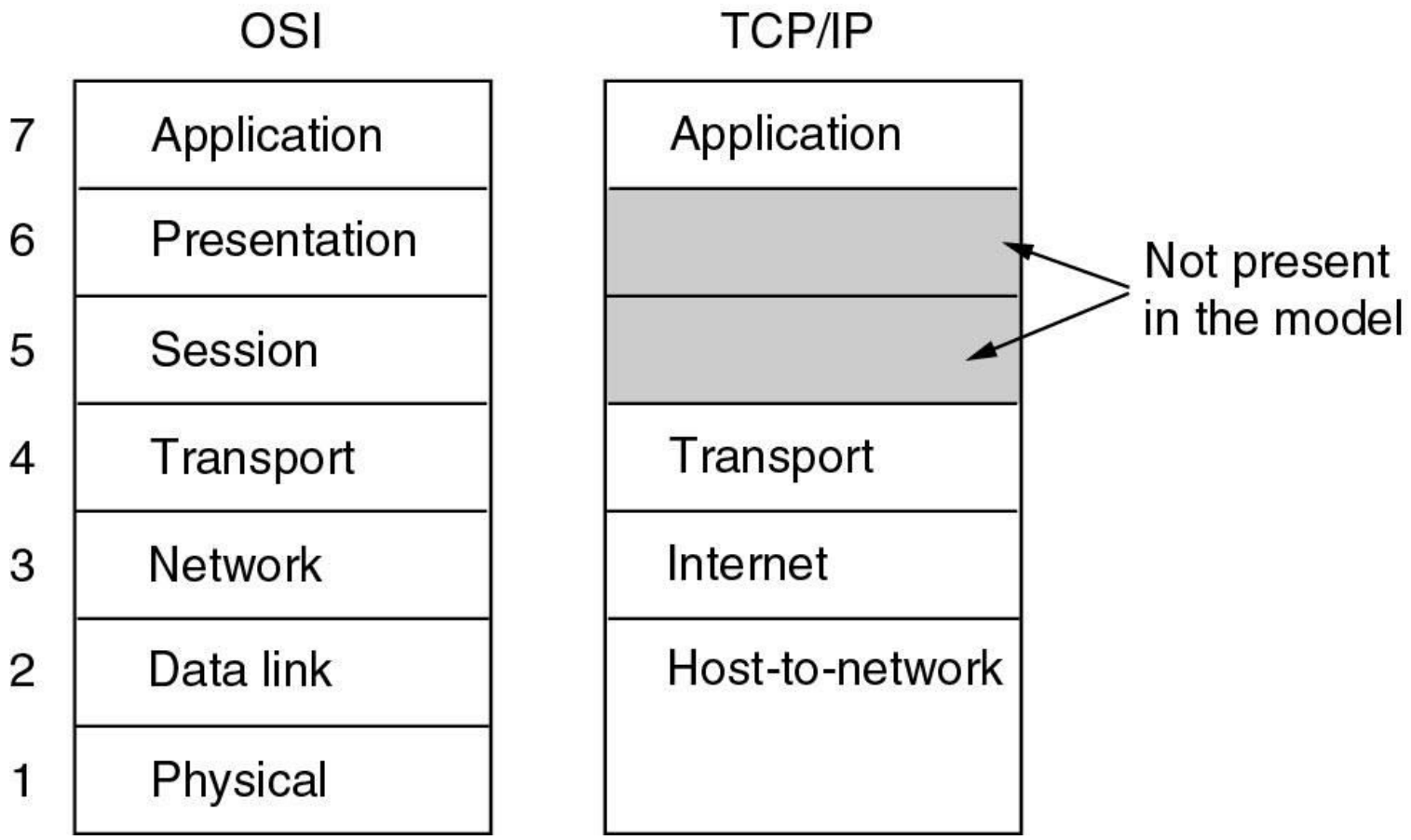
The Internet Layer: Its job is to permit hosts to inject packets into any network and have them travel independently to the destination (potentially on a different network). They may even arrive in a completely different order than they were sent, in which case it is the job of higher layers to rearrange them, if in-order delivery is desired. Note that "internet" is used here in a generic sense, even though this layer is present in the Internet. The internet layer defines an official packet format and protocol called **IP** (**Internet Protocol), plus a companion protocol called ICMP (Internet Control Message Protocol) that helps it function. The job of the internet layer is to** deliver IP packets where they are supposed to go. Packet routing is clearly a major issue here, as is congestion (though IP has not proven effective at avoiding congestion).

TCP/IP

The Transport Layer: Two end-to-end transport protocols have been defined here. The first one, **TCP** (**Transmission Control Protocol), is a reliable connection-oriented protocol that** allows a byte stream originating on one machine to be delivered without error on any other machine in the internet. It segments the incoming byte stream into discrete messages and passes each one on to the internet layer. At the destination, the receiving TCP process reassembles the received messages into the output stream. TCP also handles flow control to make sure a fast sender cannot swamp a slow receiver with more messages than it can handle. The second protocol in this layer, **UDP (User Datagram Protocol), is an** unreliable, connectionless protocol for applications that do not want TCP's sequencing or flow control and wish to provide their own. It is also widely used for one-shot, client-server-type request-reply queries and applications in which prompt delivery is more important than accurate delivery, such as transmitting speech or video.

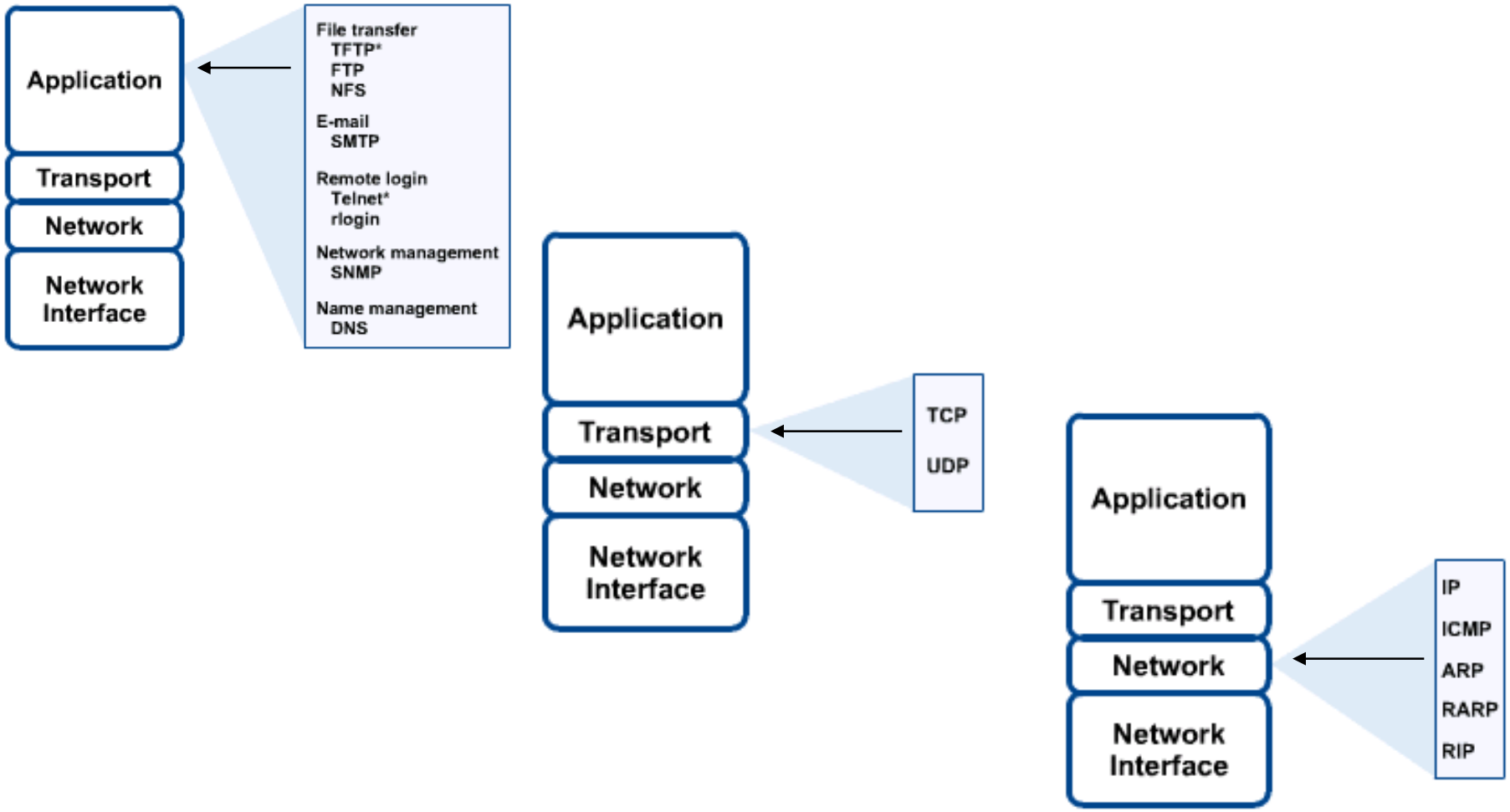
The Application Layer: On top of the transport layer is the **application layer. It contains all the higher-** level protocols. The early ones included virtual terminal (TELNET), file transfer (FTP), and electronic mail (SMTP).. Domain Name System (DNS), for mapping host names onto their network addresses, HTTP, the protocol for fetching pages on the World Wide Web, and **Real-time Transport Protocol(**RTP), the protocol for delivering real-time media such as voice or movies.

OSI vs TCP/IP

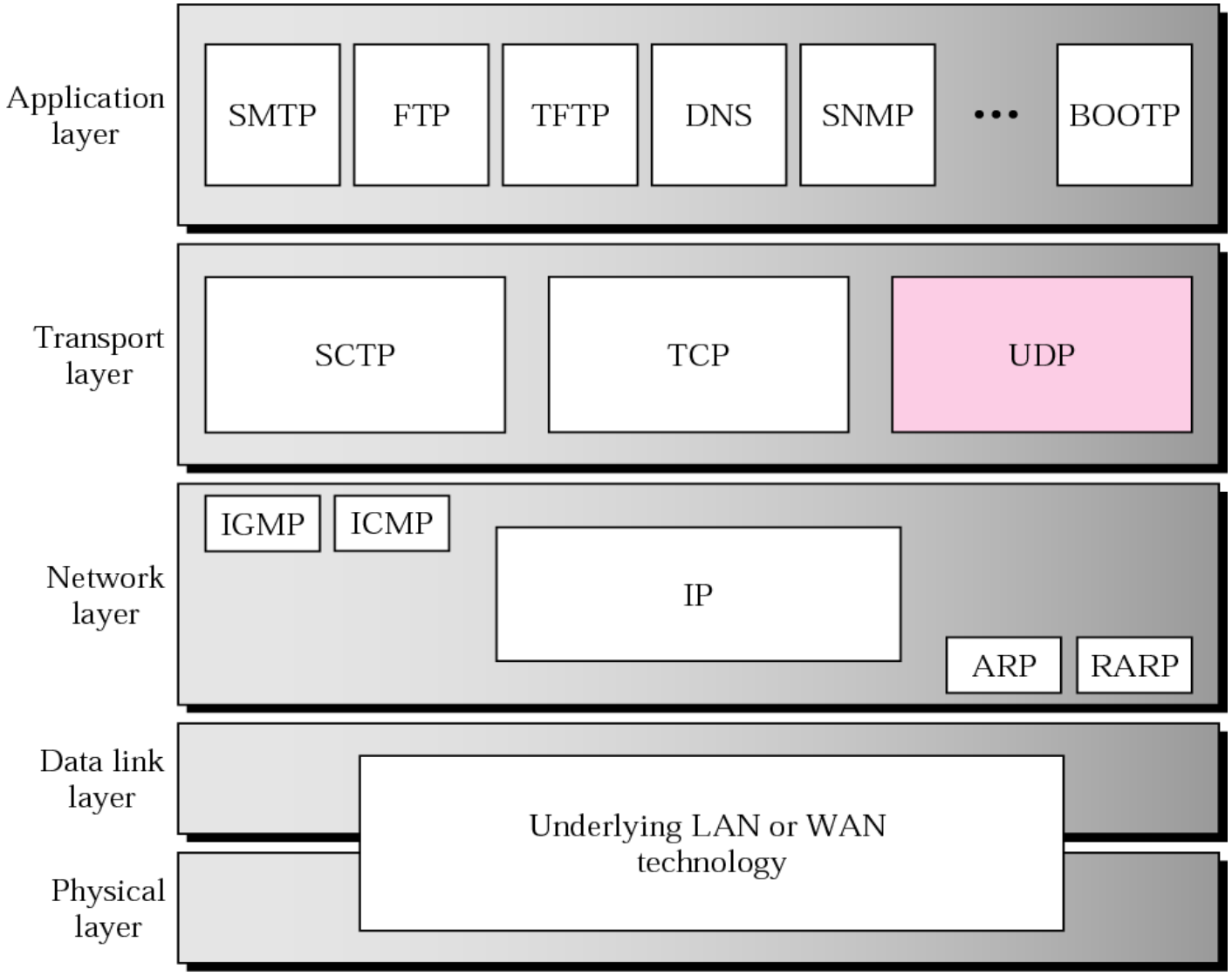


The TCP/IP reference model.

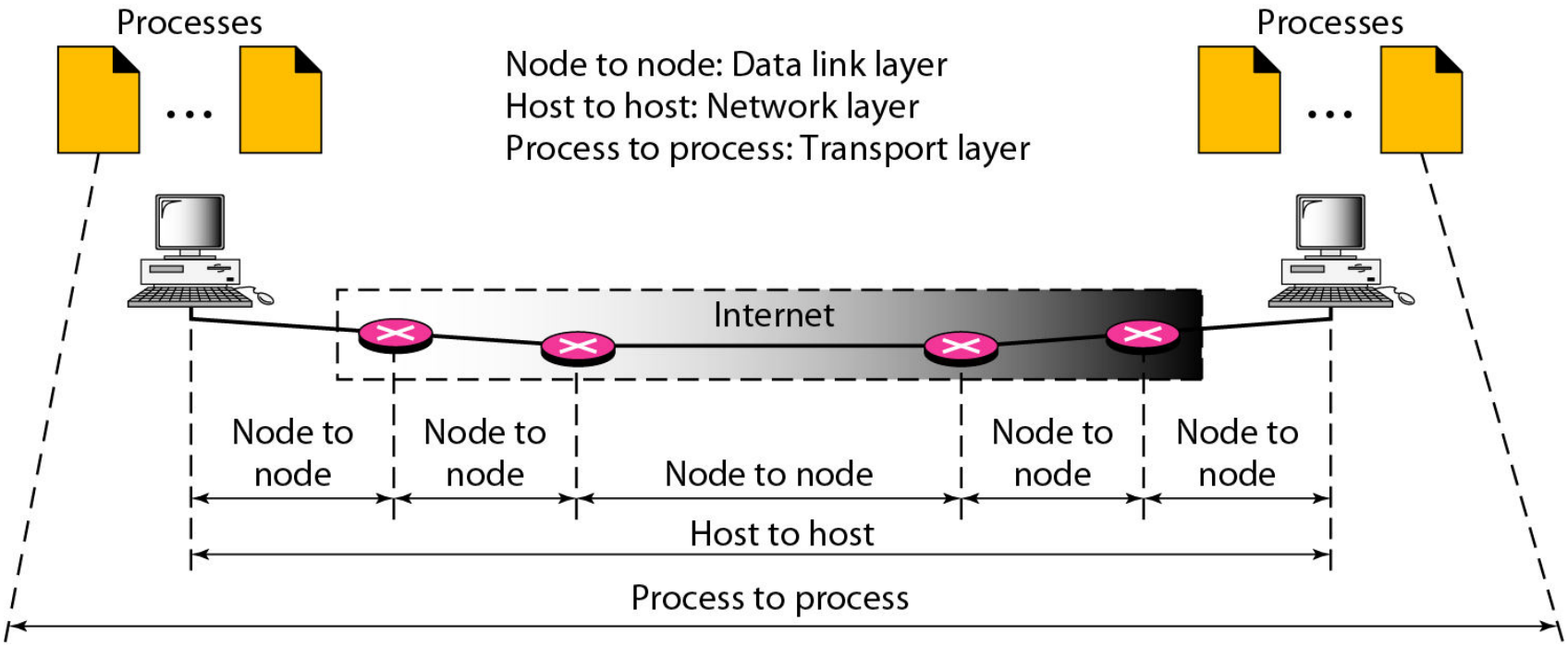
Networking Protocol: TCP/IP



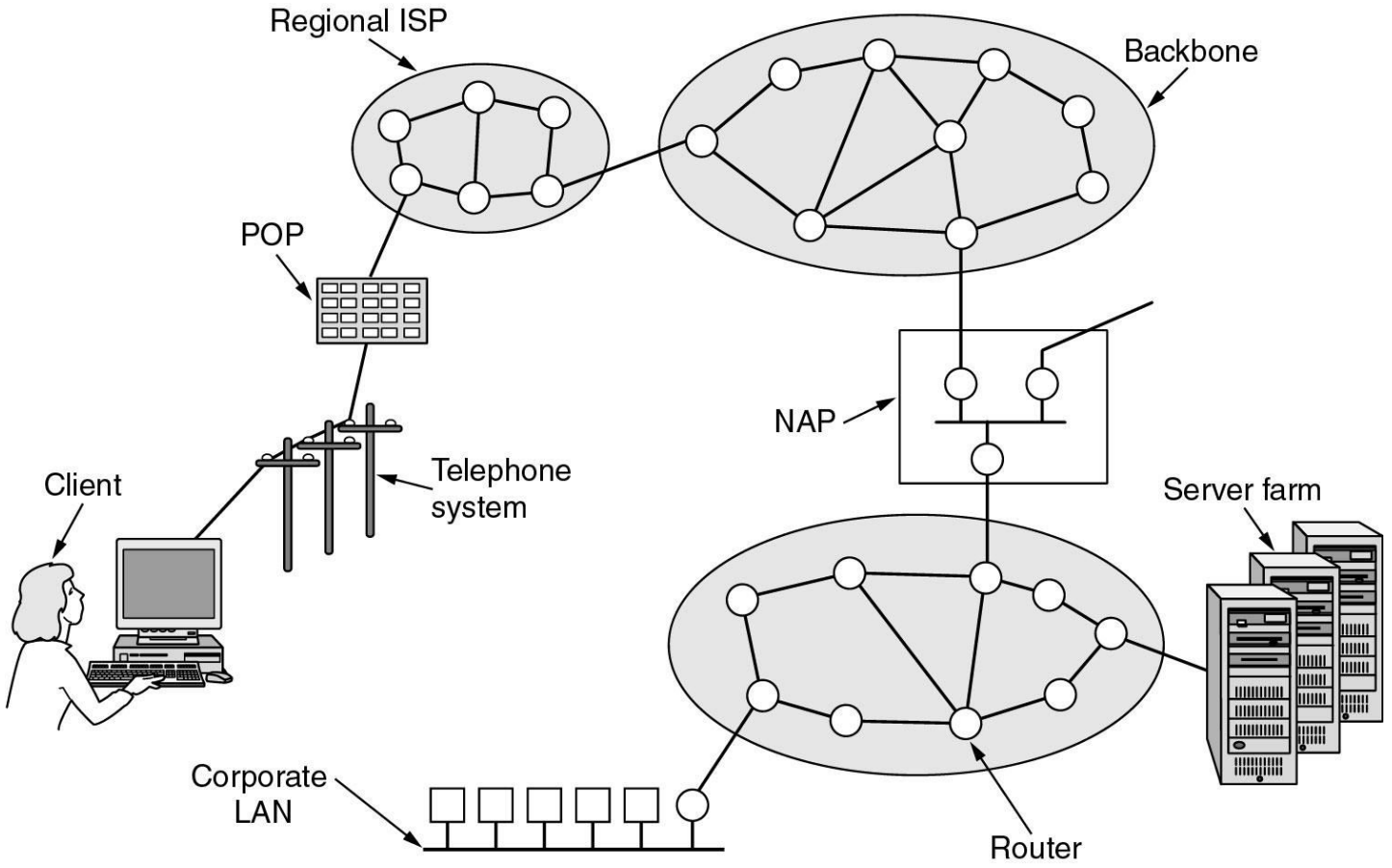
TCP/IP


Application layer
SMTP
FTP
TFTP
DNS
SNMP
• • •
BOOTP
Transport layer
SCTP
TCP
UDP
Network layer
IGMP
ICMP
IP
ARP
RARP
Data link layer
Underlying LAN or WAN technology
Physical layer

TYPES OF DATA DELIVERIES



ARCHITECTURE OF THE INTERNET



What's the Internet: "nuts and bolts" view

millions of connected computing devices: *hosts, end-systems*

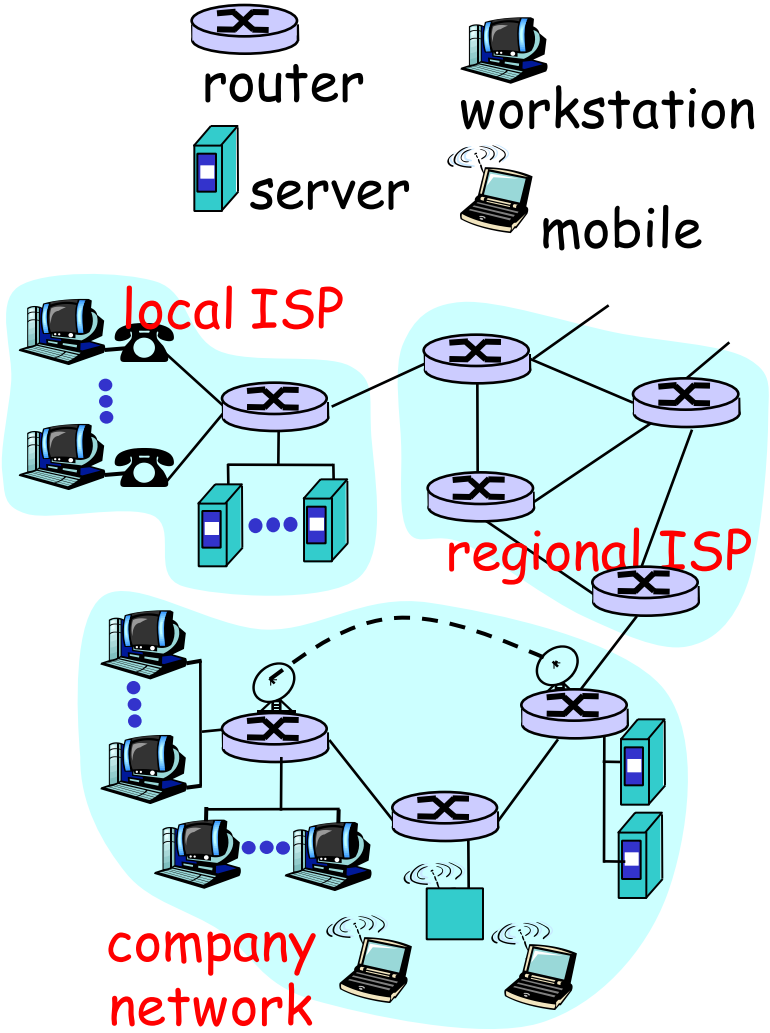
- PCs workstations, servers
- PDAs phones, toasters

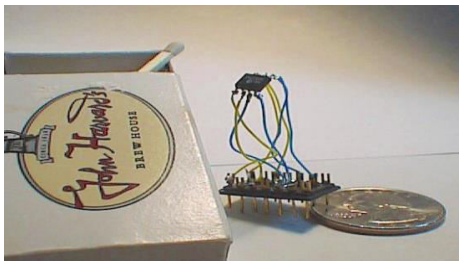
running *network apps*

communication links

- fiber, copper, radio, satellite
- transmission rate = ***bandwidth***

routers: forward packets (chunks of data)



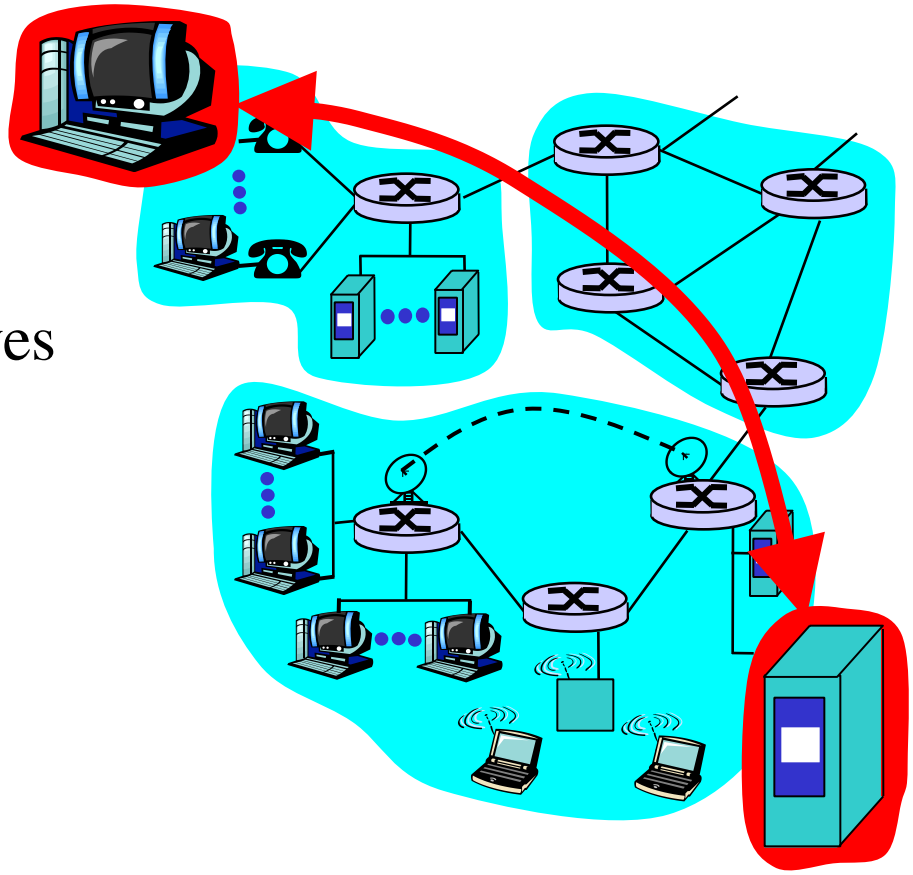


World's smallest web server

The network edge:

end systems (hosts):

- run application programs
- e.g. Web, email
- at "edge of network"



client/server model

- client host requests, receives service from always-on server
- e.g. Web browser/server; email client/server

peer-peer model:

- minimal (or no) use of dedicated servers
- e.g. Gnutella, KaZaA

Internet protocol stack

application: supporting network applications

- FTP, SMTP, STTP

transport: host-host data transfer

- TCP, UDP

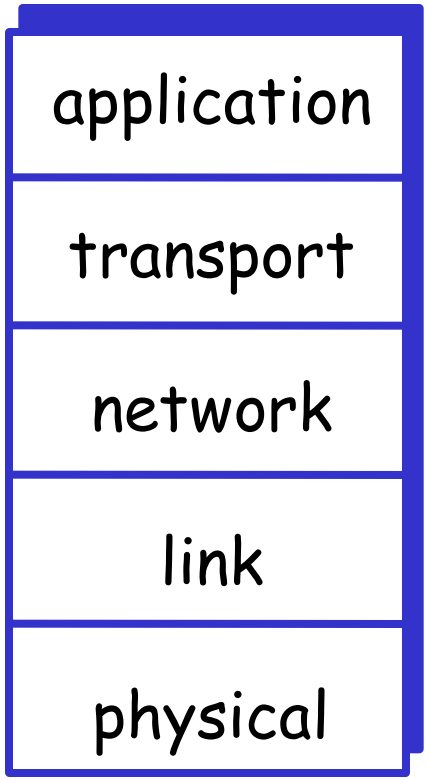
network: routing of datagrams from source to destination

- IP, routing protocols

link: data transfer between neighboring network elements

- PPP, Ethernet

physical: bits "on the wire"



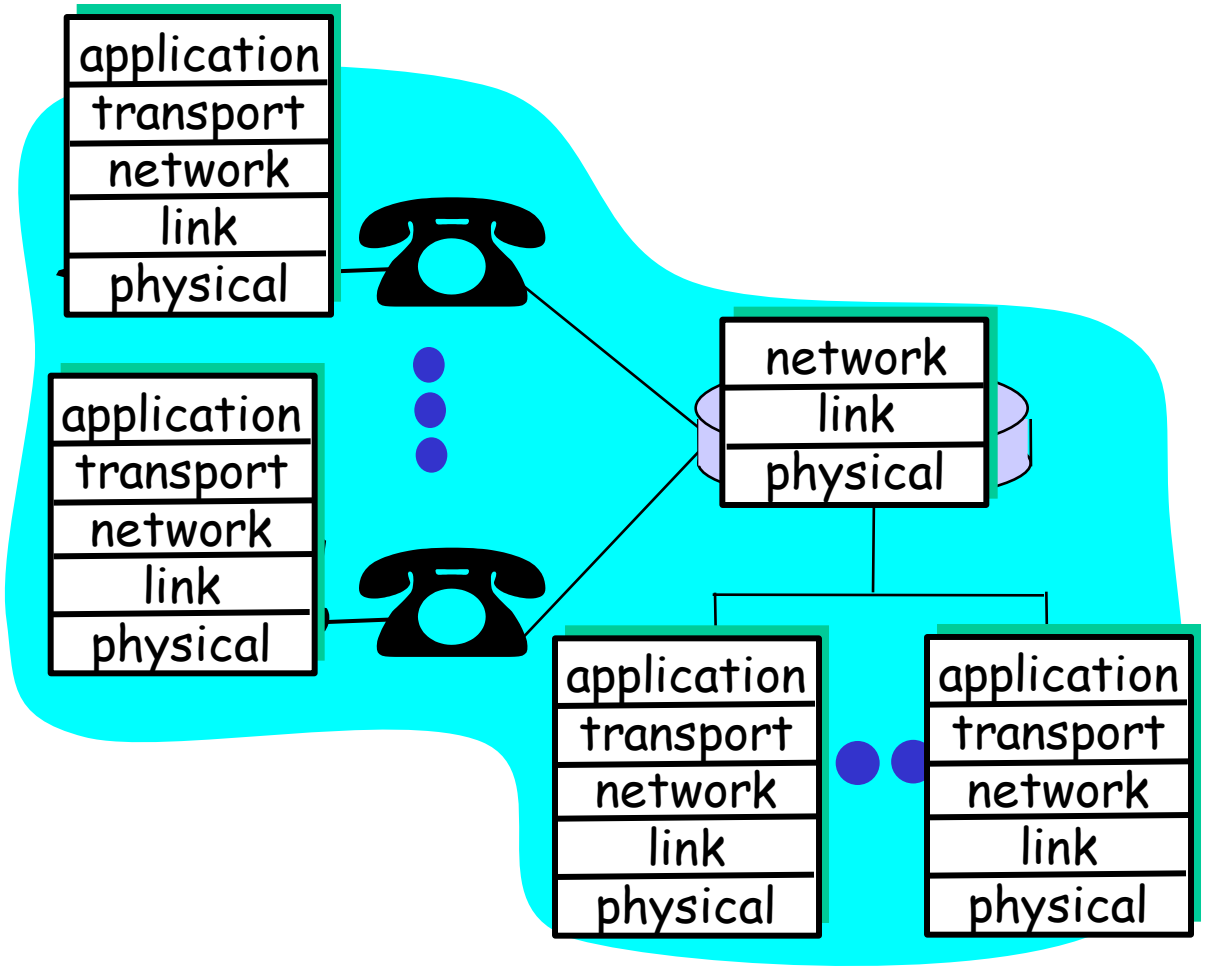
Layering: logical communication

Each layer:

distributed "entities"

implement layer functions at each node

entities perform actions, exchange messages with peers



Layering: *logical* communication

E.g.: transport

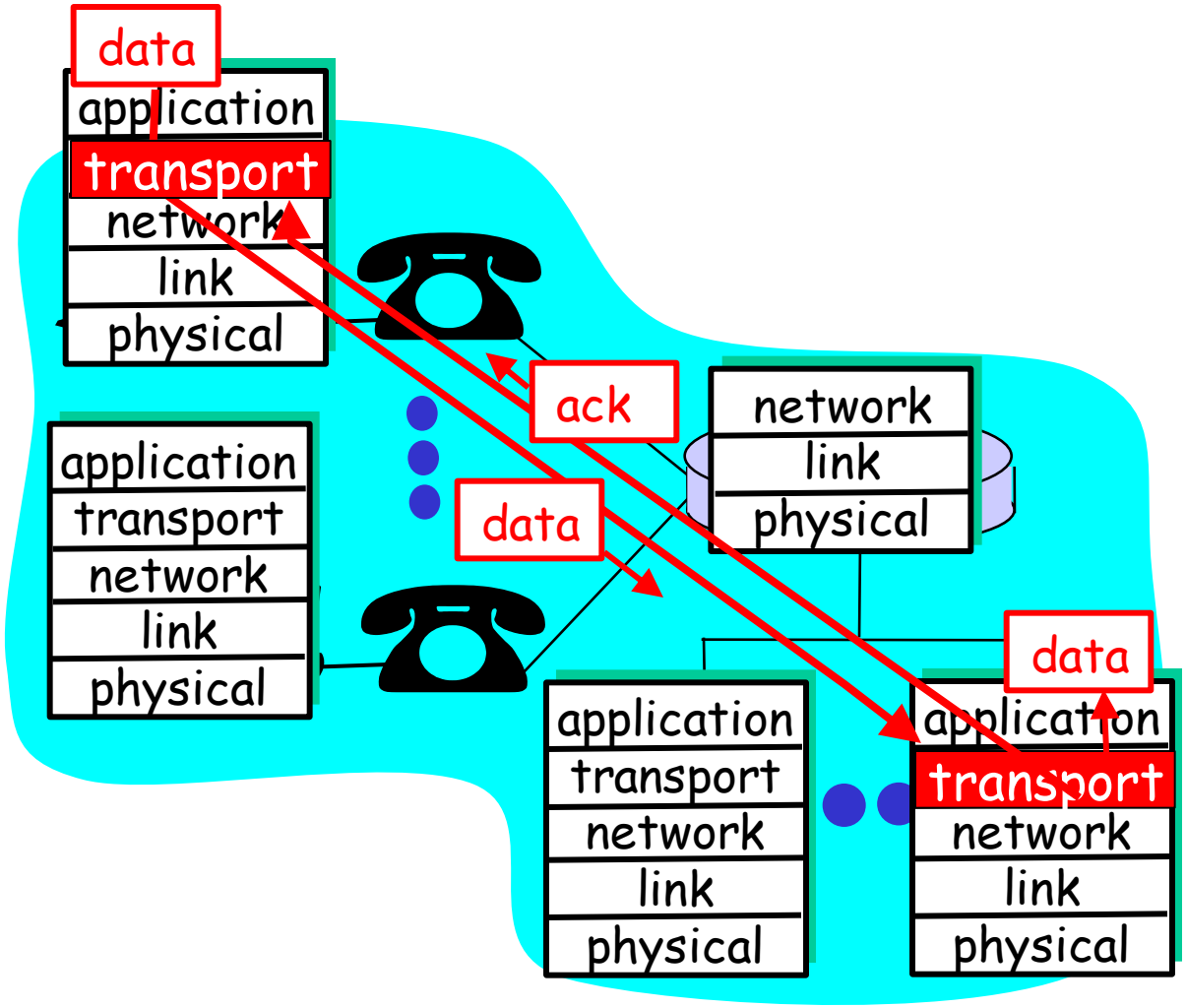
take data from app

add addressing, reliability check info to form "datagram"

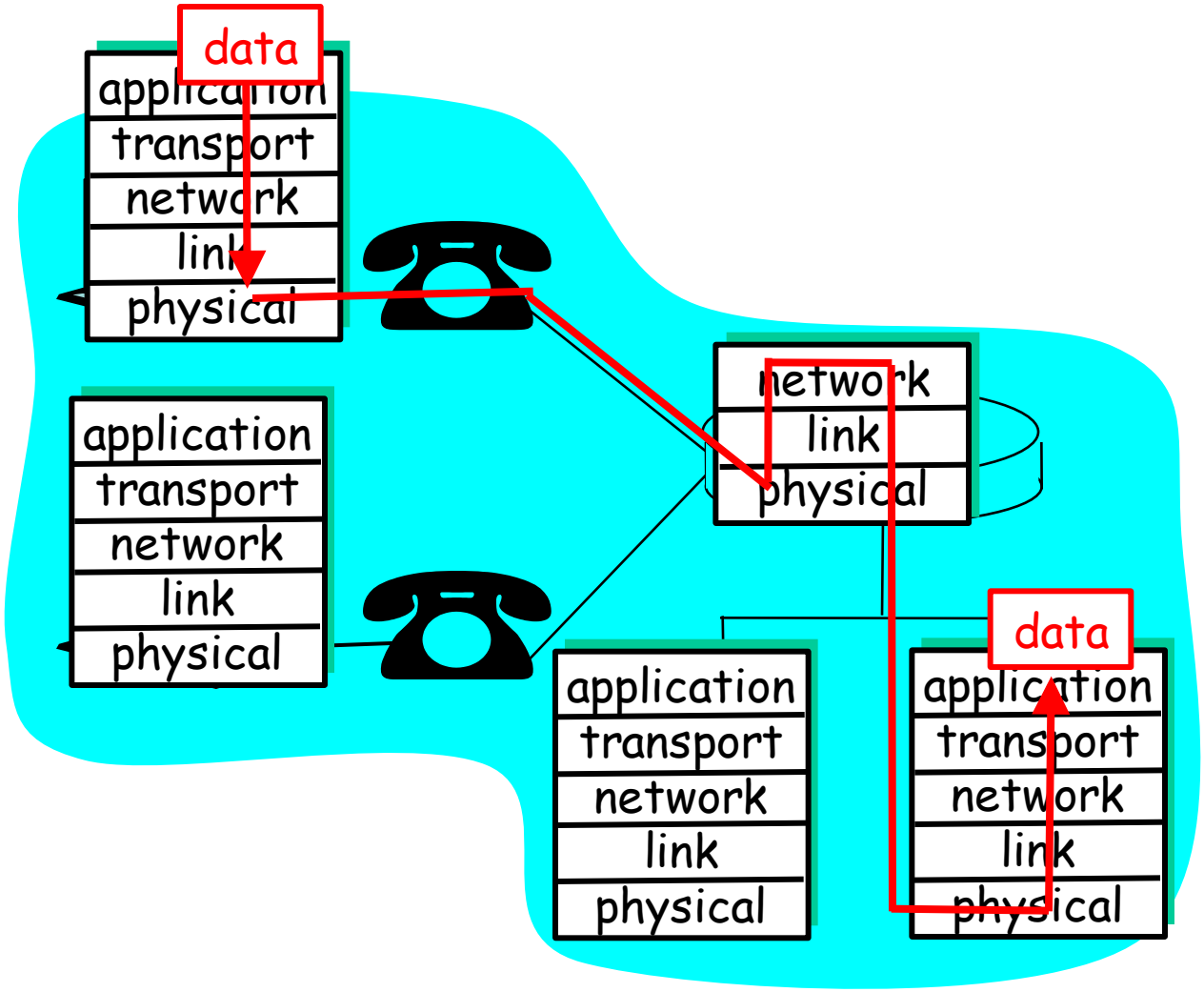
send datagram to peer

wait for peer to ack receipt

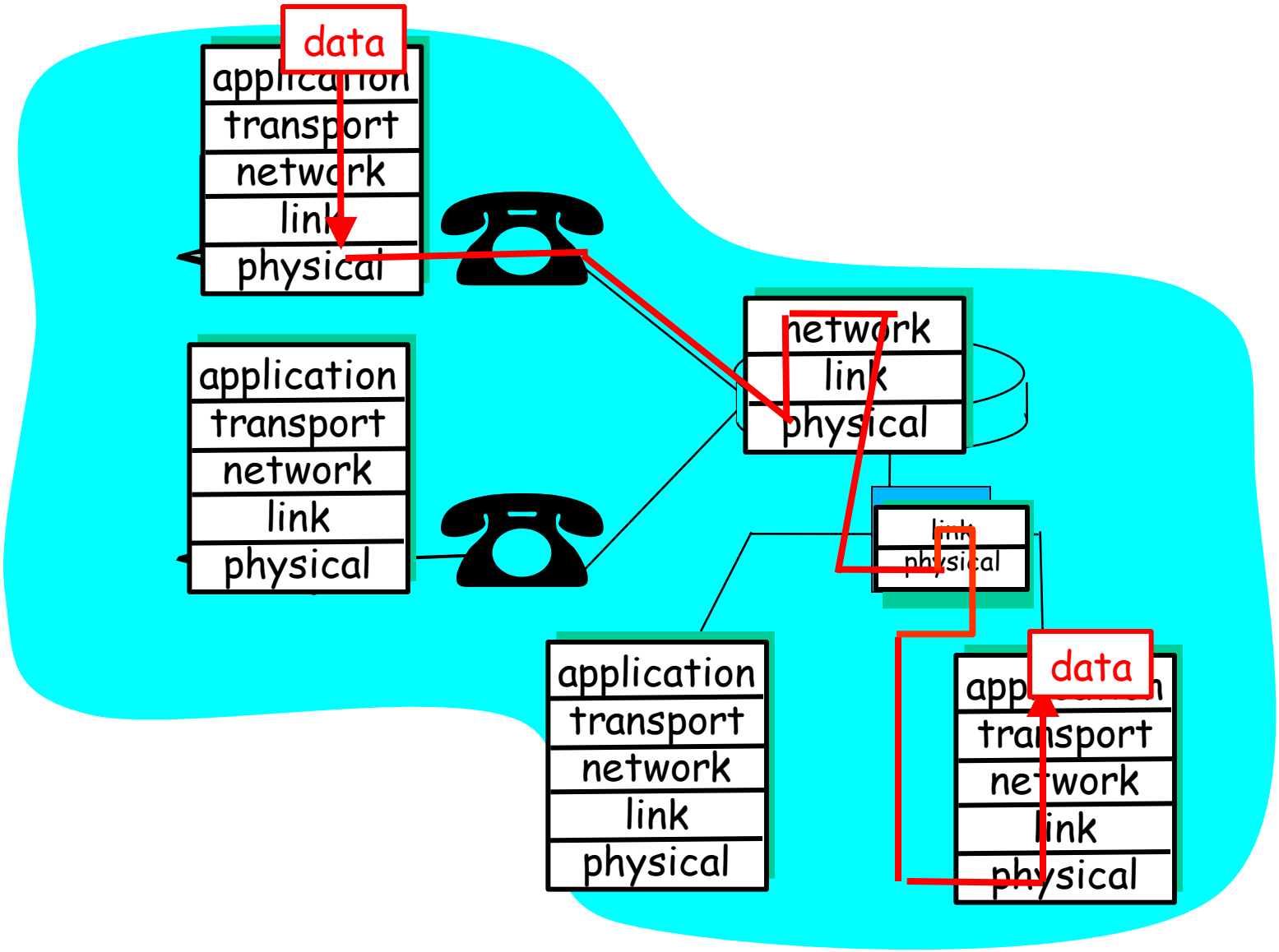
analogy: post office



Layering: physical communication



Layering: physical communication



COMPARING OSI AND TCP/IP MODELS

Concepts central to the OSI model

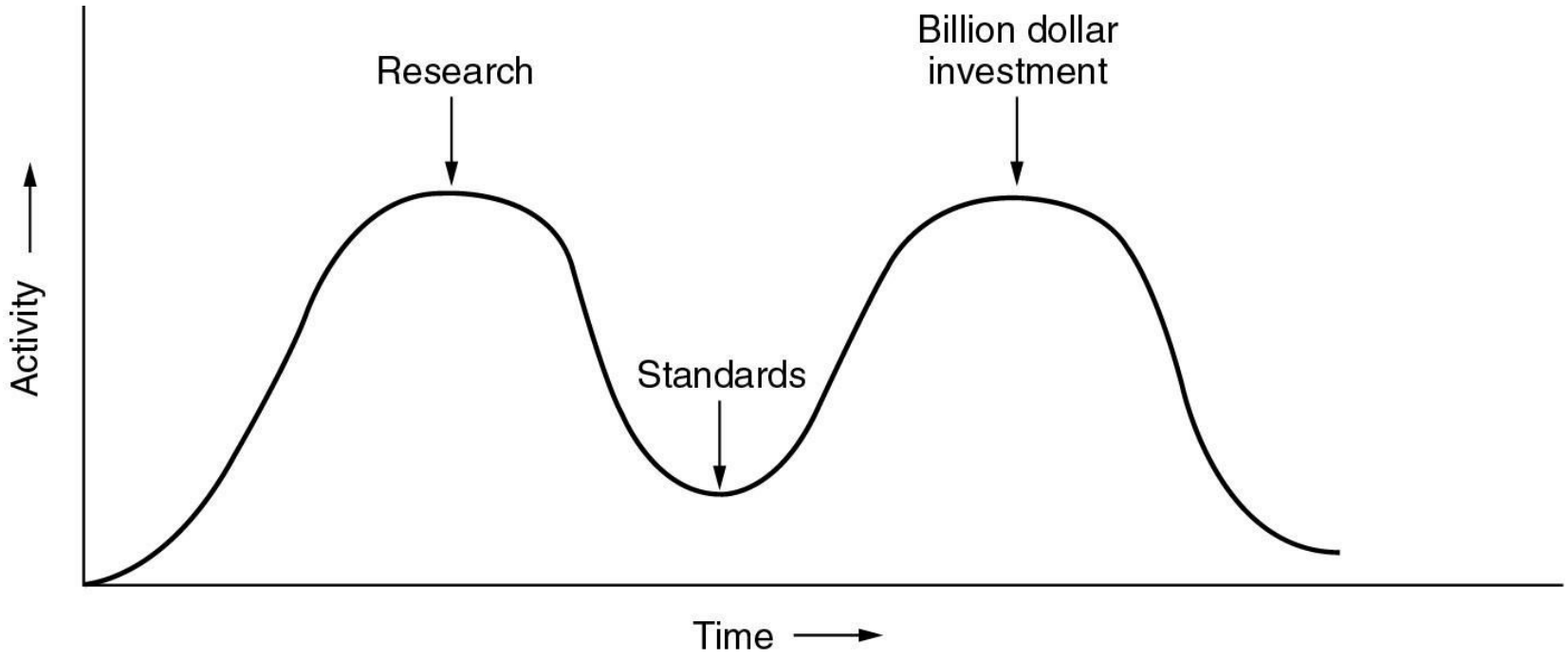
- Services
- Interfaces
- Protocols

A CRITIQUE OF THE OSI MODEL AND PROTOCOLS

Why OSI did not take over the world

- Bad timing
- Bad technology
- Bad implementations
- Bad politics

BAD TIMING



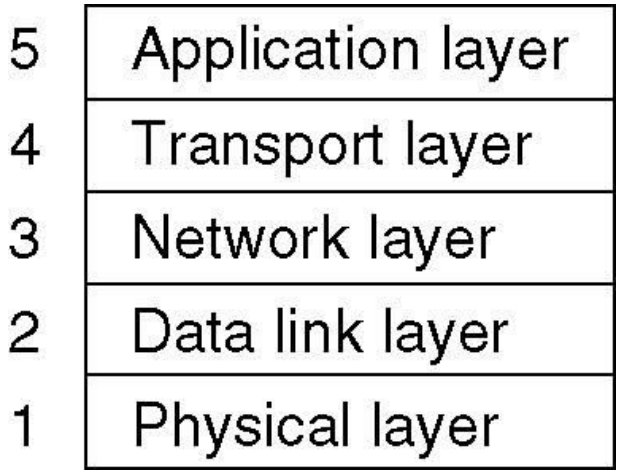
The apocalypse of the two elephants.

A CRITIQUE OF THE TCP/IP REFERENCE MODEL

Problems:

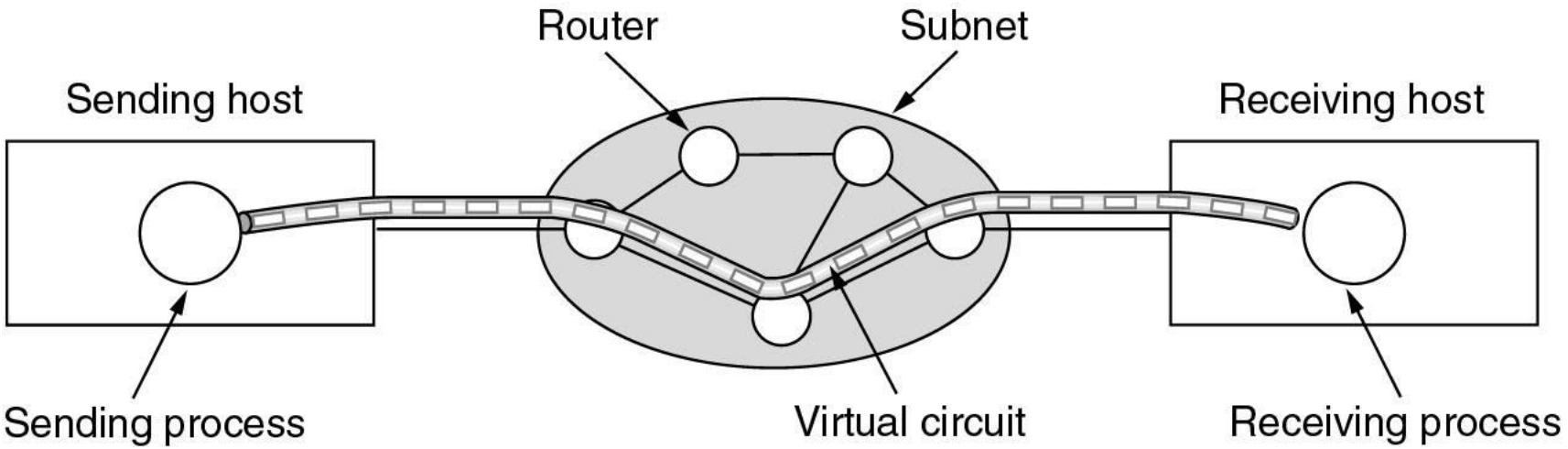
- Service, interface, and protocol not distinguished
- Not a general model
- Host-to-network “layer” not really a layer
- No mention of physical and data link layers
- Minor protocols deeply entrenched, hard to replace

HYBRID MODEL



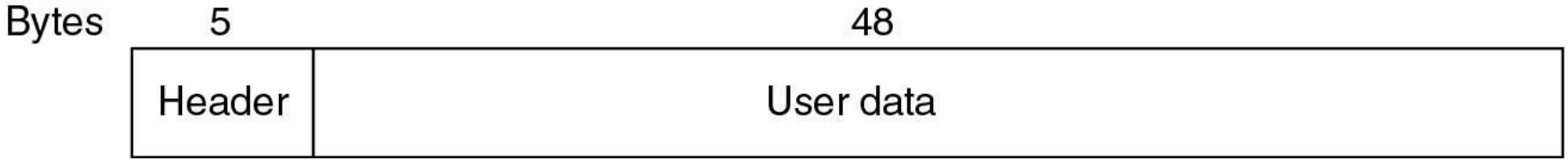
The hybrid reference model to be used in this book.

ATM VIRTUAL CIRCUITS



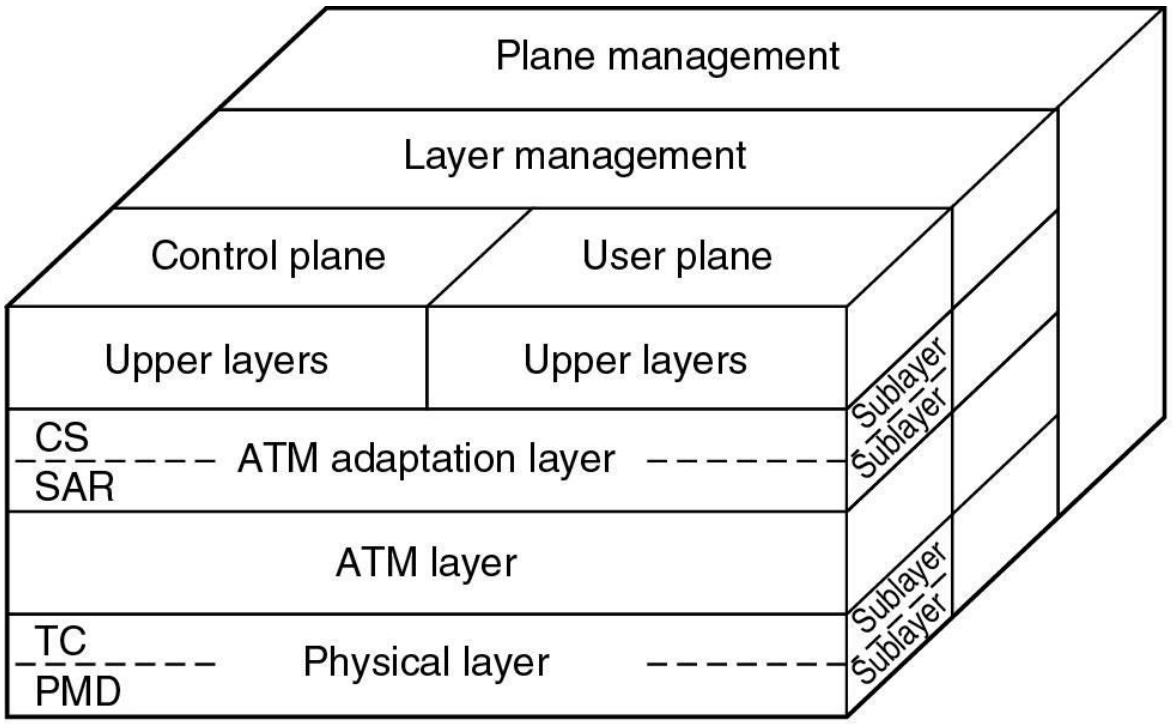
A virtual circuit.

ATM VIRTUAL CIRCUITS



An ATM cell.

THE ATM REFERENCE MODEL



User plane deals with Data Transport, Flow Control, Error correction etc.,

Control Plane is concerned with connection management

CS: Convergence sublayer
SAR: Segmentation and reassembly sublayer
TC: Transmission convergence sublayer
PMD: Physical medium dependent sublayer

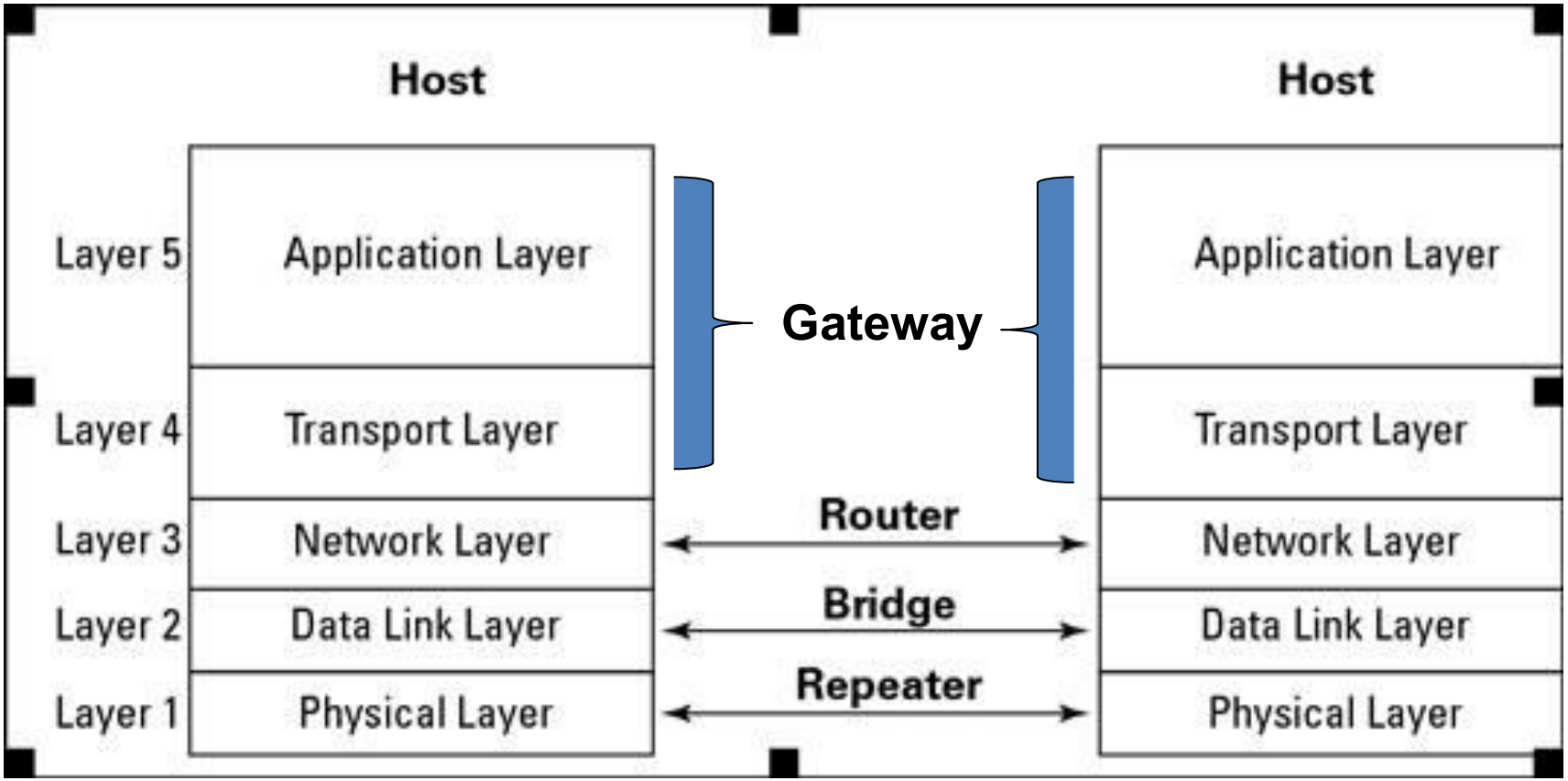
The ATM reference model.

THE ATM REFERENCE MODEL

OSI layer	ATM layer	ATM sublayer	Functionality
3/4	AAL	CS	Providing the standard interface (convergence)
		SAR	Segmentation and reassembly
2/3	ATM		Flow control Cell header generation/extraction Virtual circuit/path management Cell multiplexing/demultiplexing
2	Physical	TC	Cell rate decoupling Header checksum generation and verification Cell generation Packing/unpacking cells from the enclosing envelope Frame generation
1		PMD	Bit timing Physical network access

The ATM layers and sublayers and their functions.

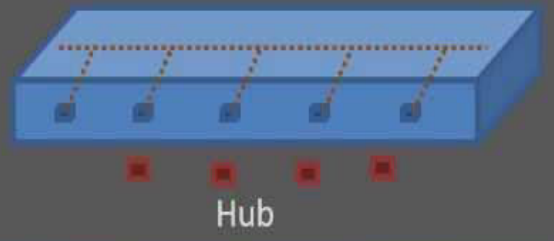
CONNECTING DEVICES



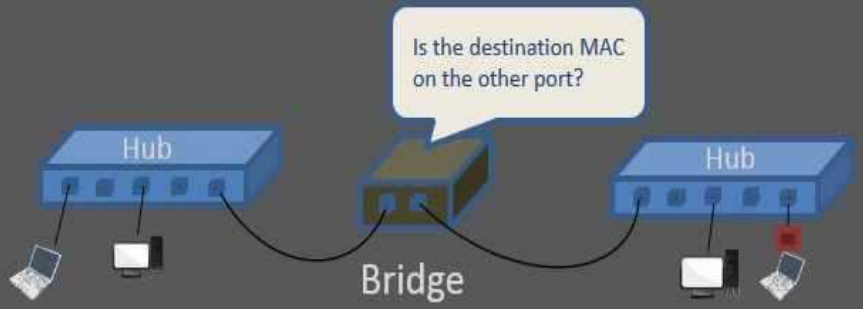
HUB/BRIDGE/SWITCH

Hub vs. Bridge vs. Switch

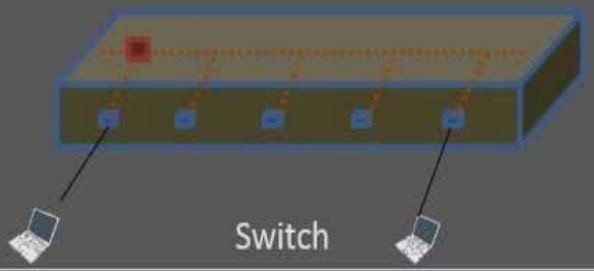
- Hub is really a *repeater*
- A message sent by one host is sent to all other hosts.
- One of the simplest ways to create a network.



- Bridge is a more intelligent form of Hub
- Packets are processed based on MAC address (Hardware Address) inside the incoming packet.



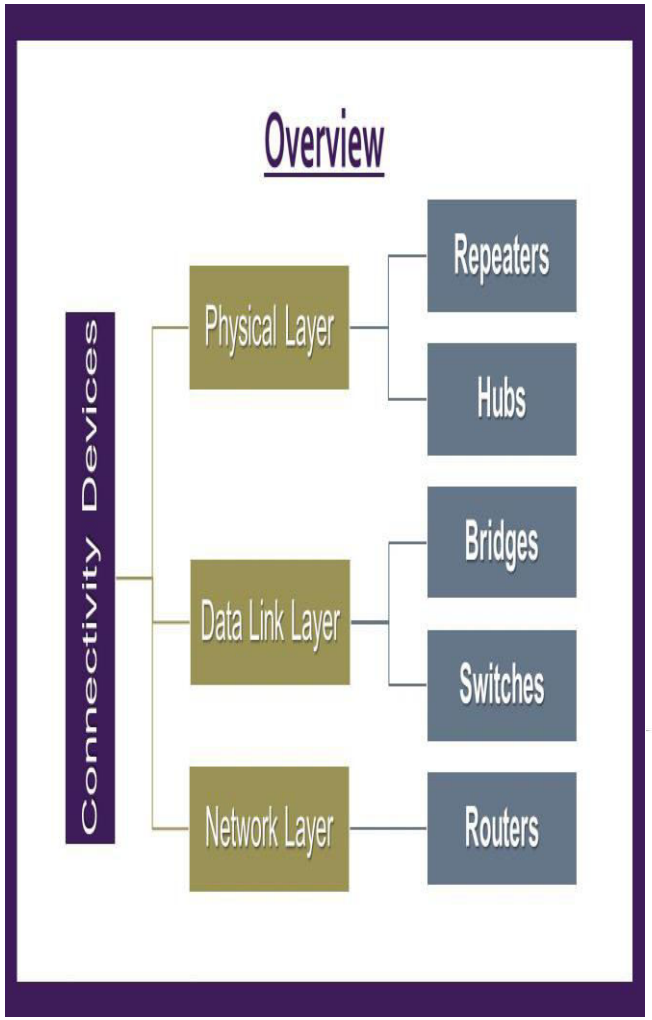
- Switch = Bridge with more than 2 Ports
- More scalable and practical
  - Bridge is not very useful for end-computing devices
  - Hubs cannot handle large data traffic



COMPARISON

BASIS FOR COMPARISON	BRIDGE	SWITCH
Basic	A bridge can connect fewer LAN.	A switch can connect more networks compared to the bridge.
Buffer	Bridges do not have buffers.	Switch has a buffer for each link connected to it.
Types	Simple bridge, multiport bridge and transparent bridge.	Store-and-forward switch and cut-through switch.
Error	Bridges do not perform error checking.	Switches perform error checking.

NETWORKING DEVICES



Hub

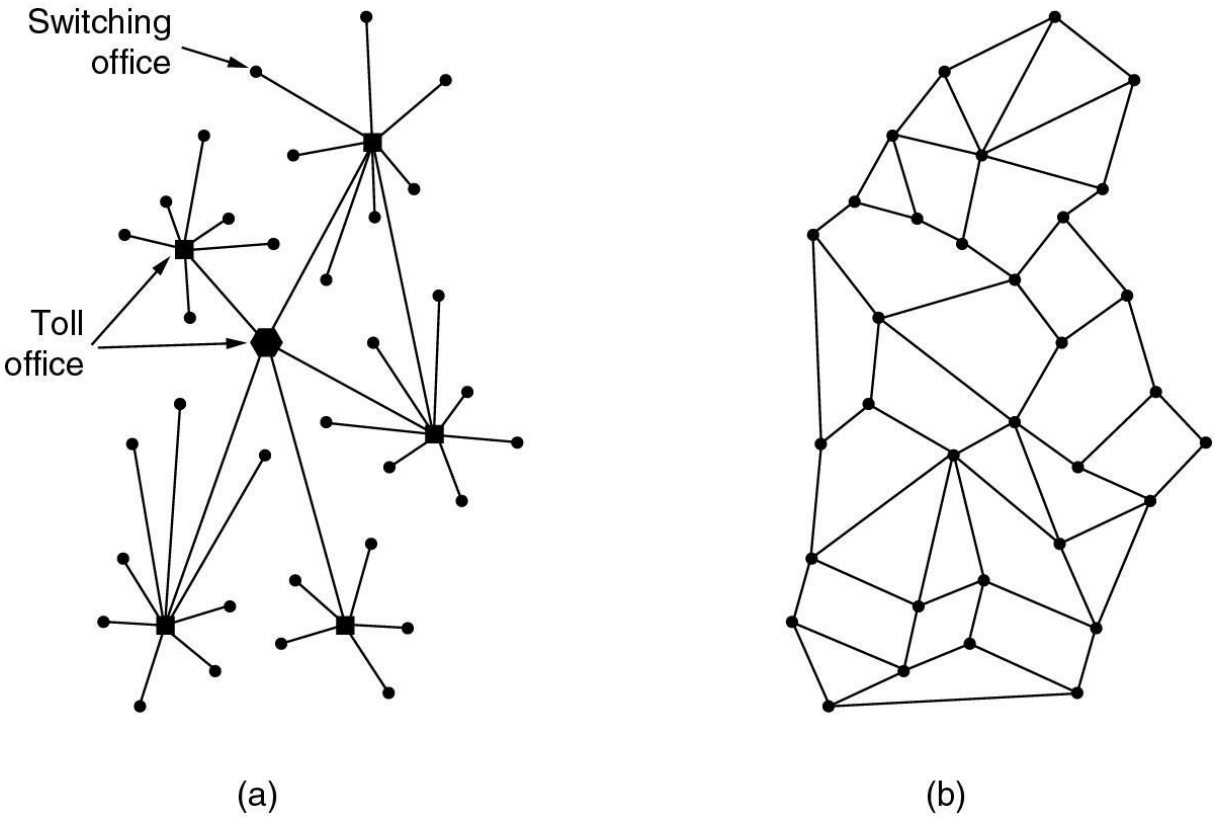
NETGEAR 5 Port Network Switch

ComputerHope.com

EXAMPLE NETWORKS

- The Internet
- Connection-Oriented Networks:
  X.25, Frame Relay, and ATM
- Ethernet
- Wireless LANs: 802:11

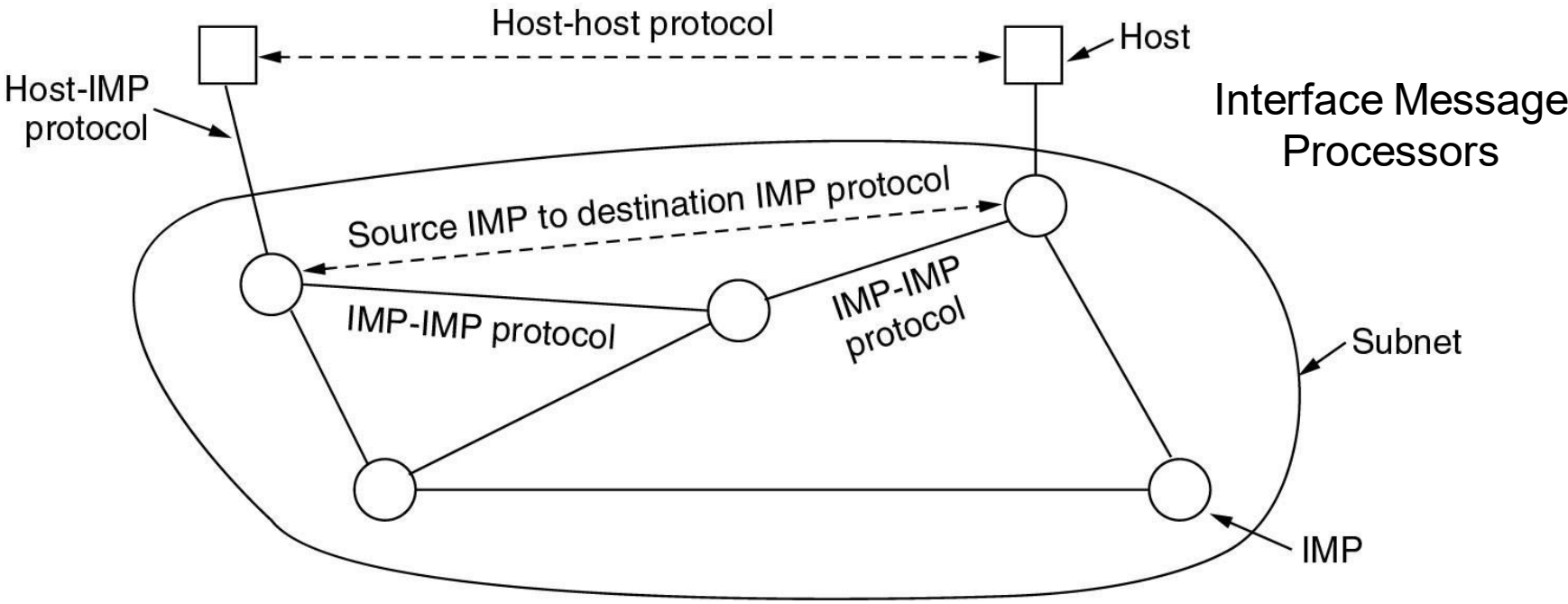
ARPANET



(a) Structure of the telephone system.

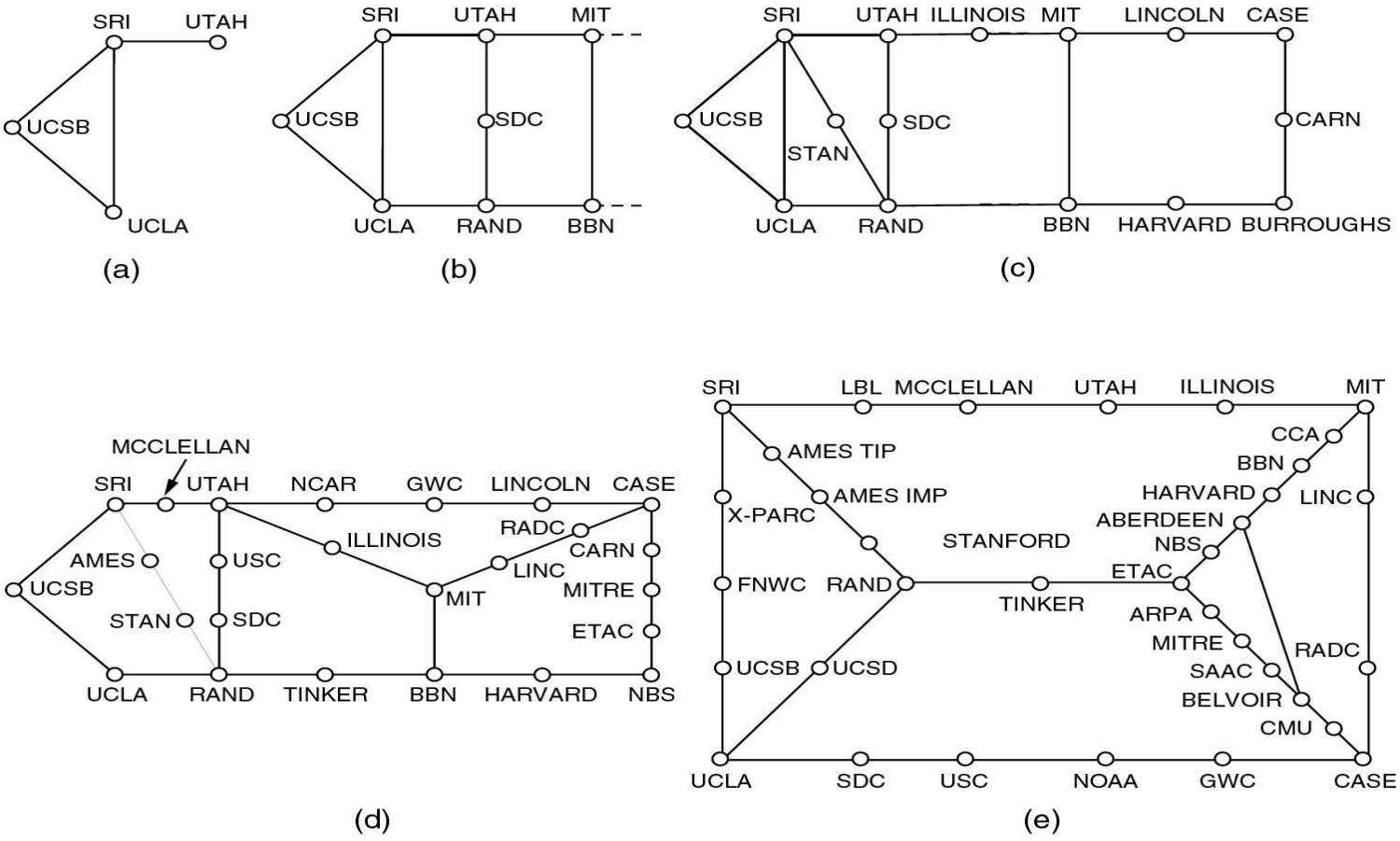
(b) Baran's proposed distributed switching system.

ARPANET



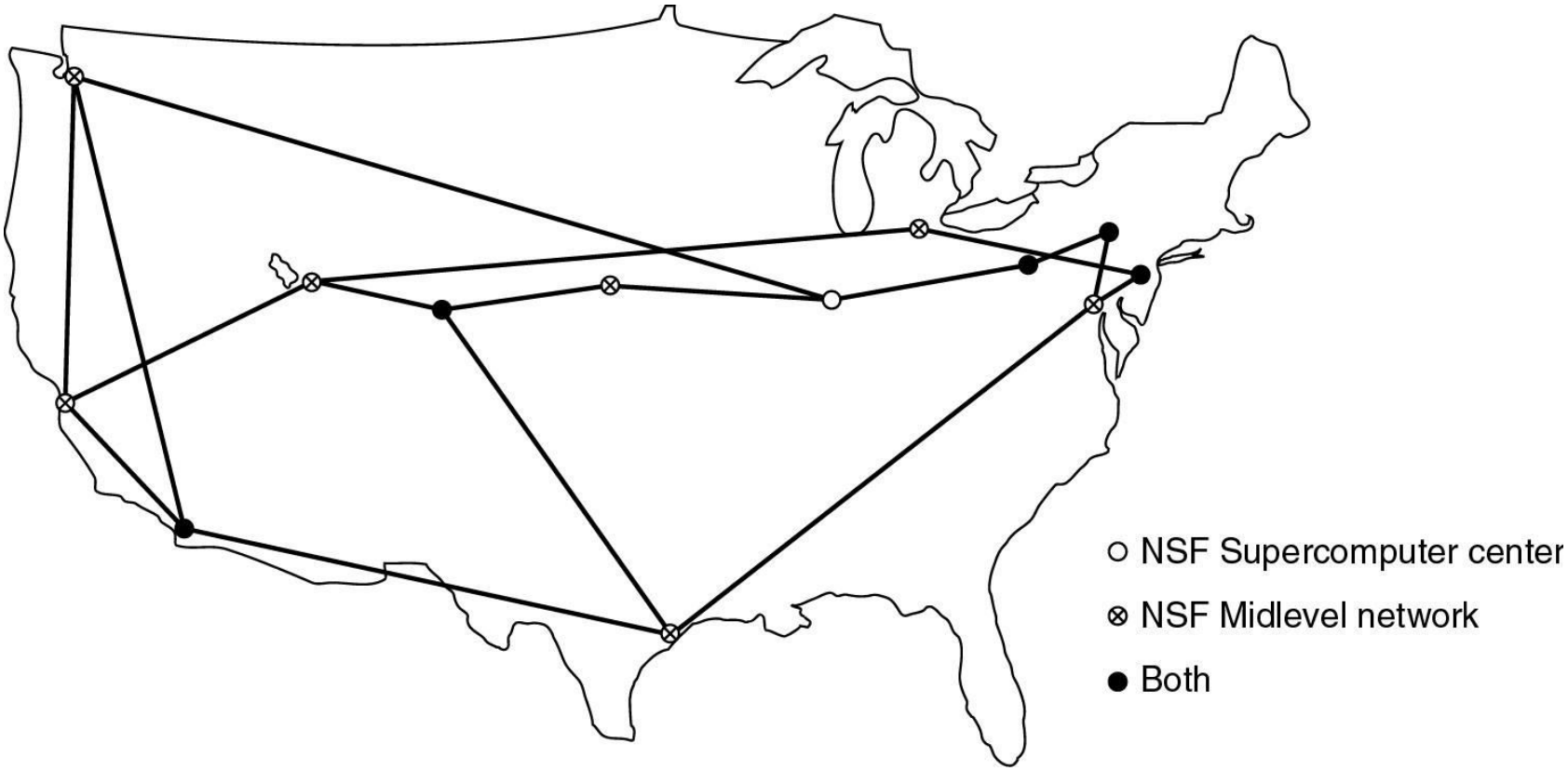
The original ARPANET design.

ARPANET



Growth of the ARPANET (a) December 1969. (b) July 1970. (c) March 1971. (d) April 1972. (e) September 1972.

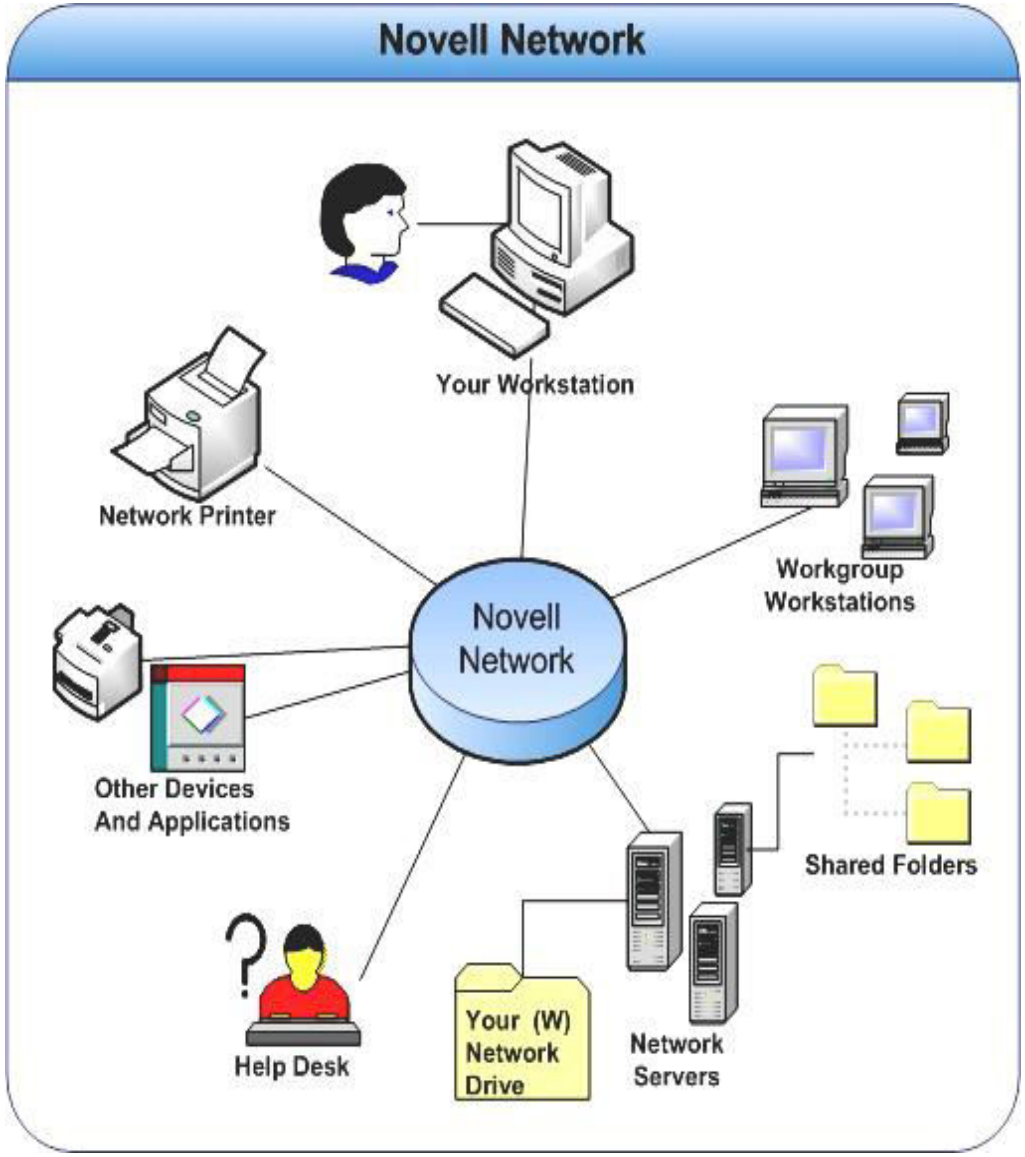
NSFNET-(NATIONAL SCIENCE FOUNDATION)



The NSFNET backbone in 1988.

Each super computer was given a little brother called a fuzzball. To ease the transition and make every regional network could communicate with every other regional network through Network Operators-Network Access Point

NOVELL NETWORK

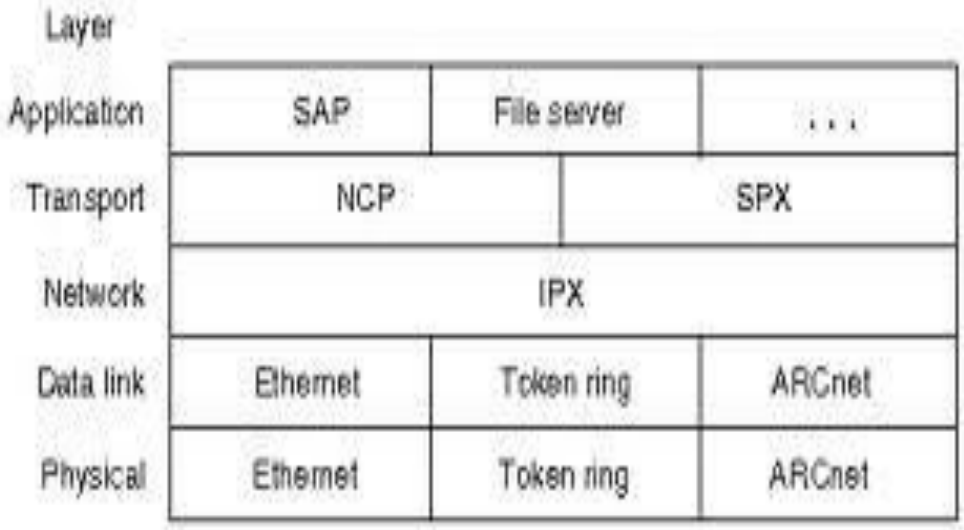


The Novell Netware network connects your workstation to file servers, printers, and other shared computing and information resources

Novell NetWare is the most popular network system in the PC world. It was designed to be used by companies downsizing from a mainframe to a network of PCs. Novell NetWare is based on the client-server model.

NOVELL NETWORK

Internetwork Packet Exchange (**IPX**)



The Sequenced Packet Exchange (**SPX**), is **Novell's** legacy transport layer protocol providing a packet delivery service for **Novell** NetWare network.

SAP- Service Advertising Protocol

NCP, or NetWare Core Protocol is the file-sharing protocol between server and client(s) on a Novell NetWare **network**. **NCP** controls many requests to the file **...**

SAP services provide information on all the known servers throughout the entire internetwork. These servers can include file servers, print servers, NetWare access servers, remote console servers and so on.

IEEE 802 STANDARDS

Number	Topic
802.1	Overview and architecture of LANs
802.2 ↓	Logical link control
802.3 *	Ethernet
802.4 ↓	Token bus (was briefly used in manufacturing plants)
802.5	Token ring (IBM's entry into the LAN world)
802.6 ↓	Dual queue dual bus (early metropolitan area network)
802.7 ↓	Technical advisory group on broadband technologies
802.8 †	Technical advisory group on fiber optic technologies
802.9 ↓	Isochronous LANs (for real-time applications)
802.10 ↓	Virtual LANs and security
802.11 *	Wireless LANs
802.12 ↓	Demand priority (Hewlett-Packard's AnyLAN)
802.13	Unlucky number. Nobody wanted it
802.14 ↓	Cable modems (defunct: an industry consortium got there first)
802.15 *	Personal area networks (Bluetooth)
802.16 *	Broadband wireless
802.17	Resilient packet ring

The 802 working groups. The important ones are marked with *. The ones marked with ↓ are hibernating. The one marked with † gave up.

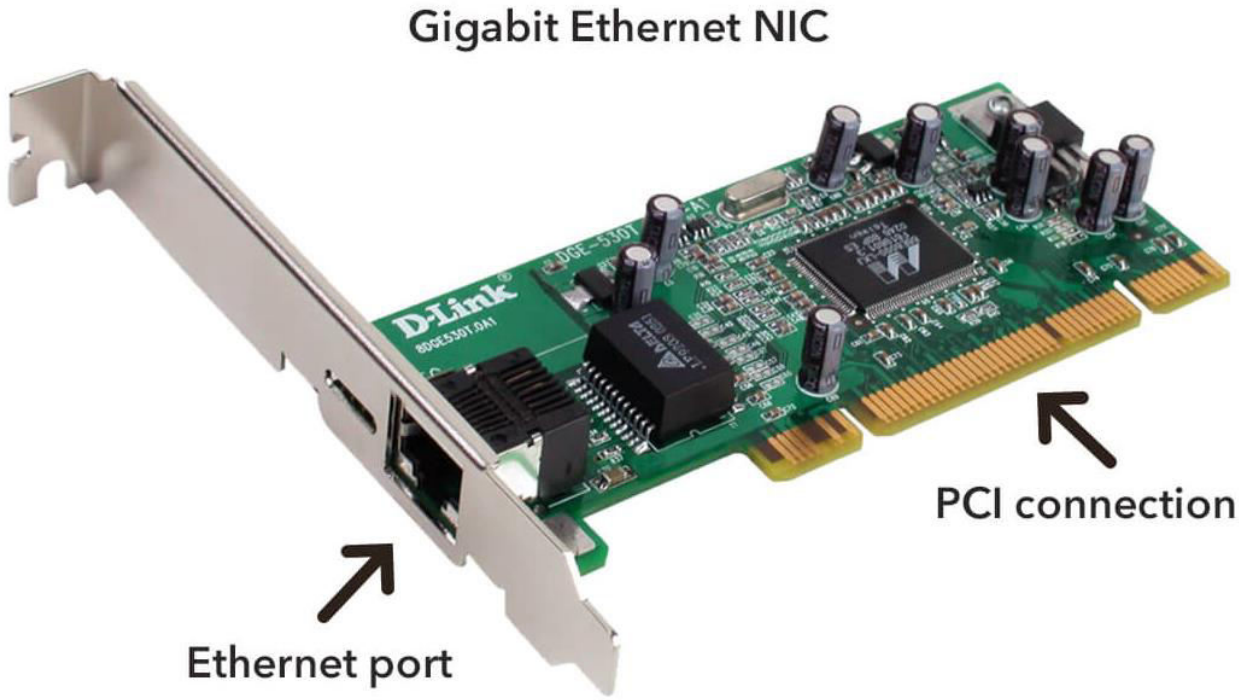
REFERENCES

1. Andrew S Tanenbaum “Computer Networks” 5/ed. Pearson Education.
2. Behrouz A. Forouzan “Data Communication and Networking” 3/e, TMH.
3. William Stallings “Data and Computer Communications”, 8/e, PHI, 2004.
4. S.Keshav “An Engineering Approach to Computer Networks” 2/e, PE
5. Behrouz A. Forouzan “TCP/IP protocol suite” 4/e, TMH, 2010.
6.https://www.slideshare.net/arushigarg714/lan-man-and-wan-ppt-final

UNIT-I; PART-II

The Data Link Layer

The Data Link Layer



The network interface card operates on layer two. Every single NIC has a unique number on it. This number is burned into the card at the factory in which it is made, and it can't be changed.

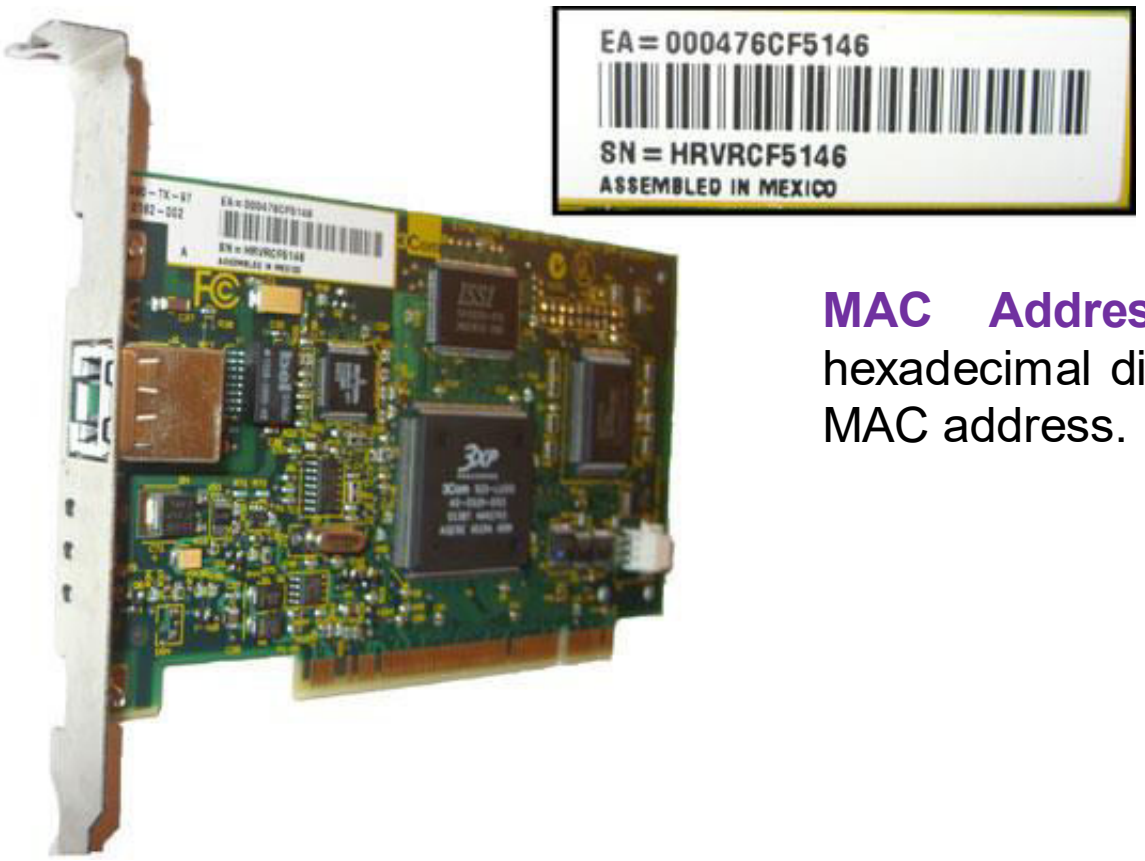
WHAT TO LOOK FOR IN A NIC: There are three primary things to look for in a network interface card: the **bandwidth** it supports, the **type of media** is supports, and the type of **network architecture** it supports.

The Data Link Layer

Working of Network Interface Card :

- The foremost step is to establish a connection and gather the data.
- The purpose of NIC is to collect the data present on the motherboard. Then it is transferred to the buffer present on the card.
- While gathering the information and in between transferring the destination address where the data should be sent is inserted.
- Along with destination, NIC adds its own address.
- At the buffer, a checksum is calculated by NIC.
- Further, the data is transmitted to the network.
- After the data received NIC performs checksum an compares it with original checksum.
- After the verification, if no errors are detected then the acknowledgement is sent.

The Data Link Layer



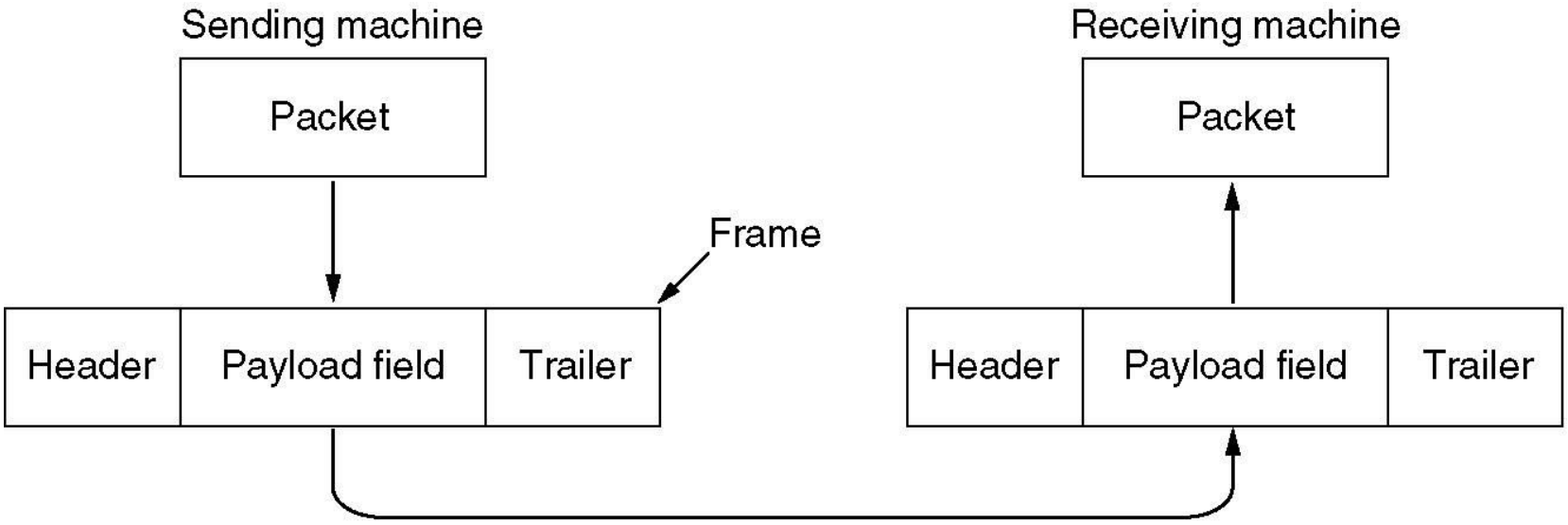
MAC Address: We have to write 12 hexadecimal digits. You need 48 bits to store a MAC address.

NIC is both a physical layer and a data link layer device, i.e. it provides the necessary hardware circuitry so that the physical layer processes and some data link layer processes can run on it.

DATA LINK LAYER DESIGN ISSUES

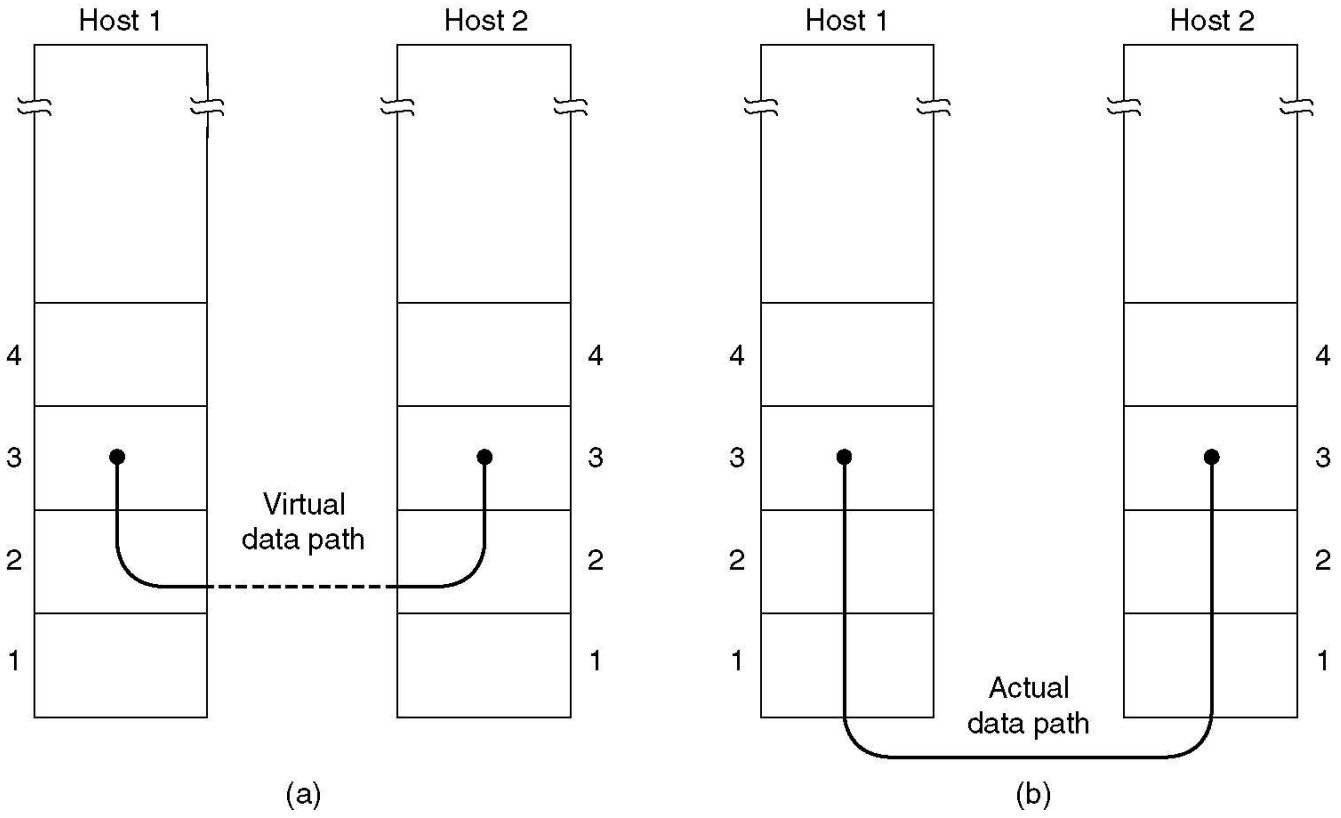
- **Services Provided to the Network Layer**
  - **Framing**
  - **Error Control**
  - **Flow Control**

FUNCTIONS OF THE DL LAYER



Relationship between packets and frames.

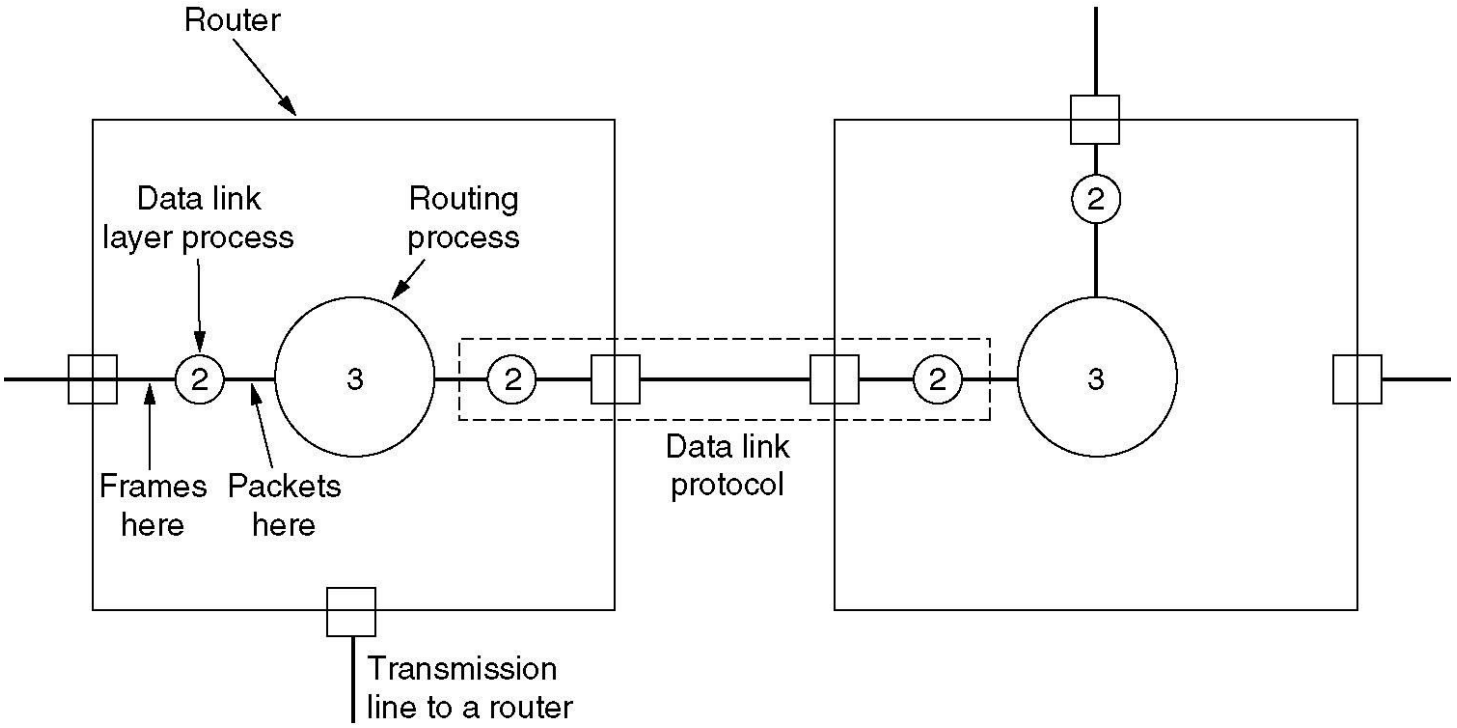
SERVICES PROVIDED TO NETWORK LAYER



(a) Virtual communication.

(b) Actual communication.

SERVICES PROVIDED TO NETWORK LAYER

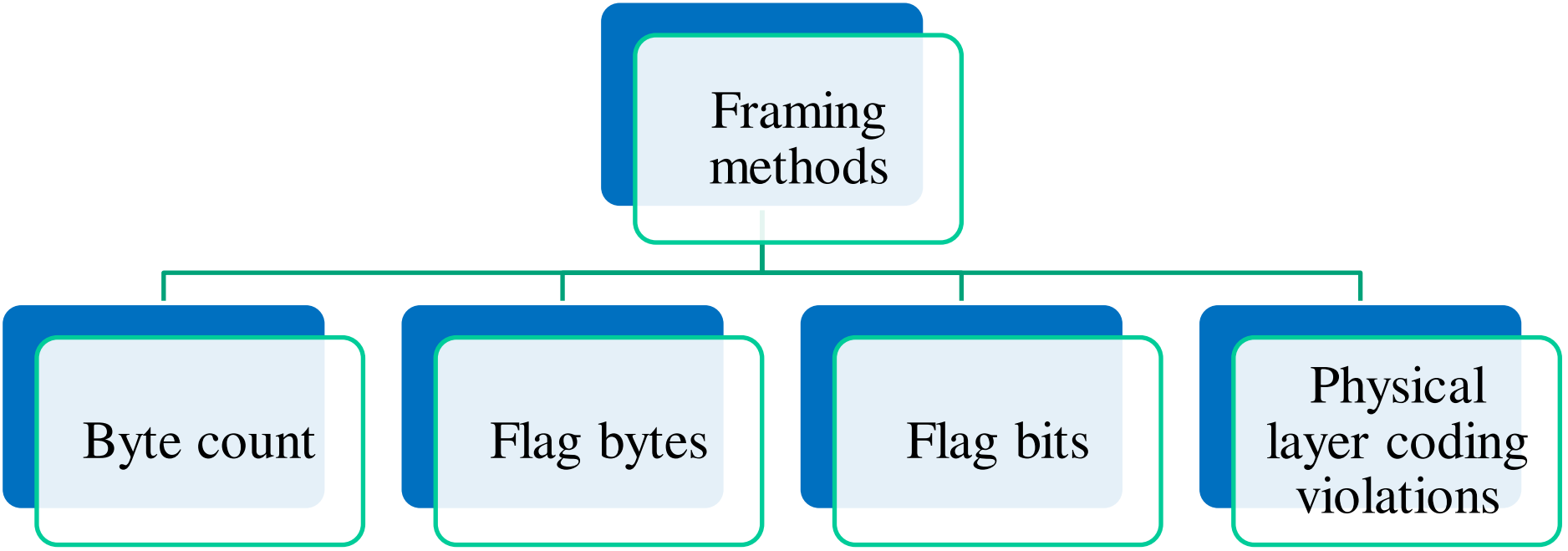


Placement of the data link protocol.

SERVICES PROVIDED TO NETWORK LAYER

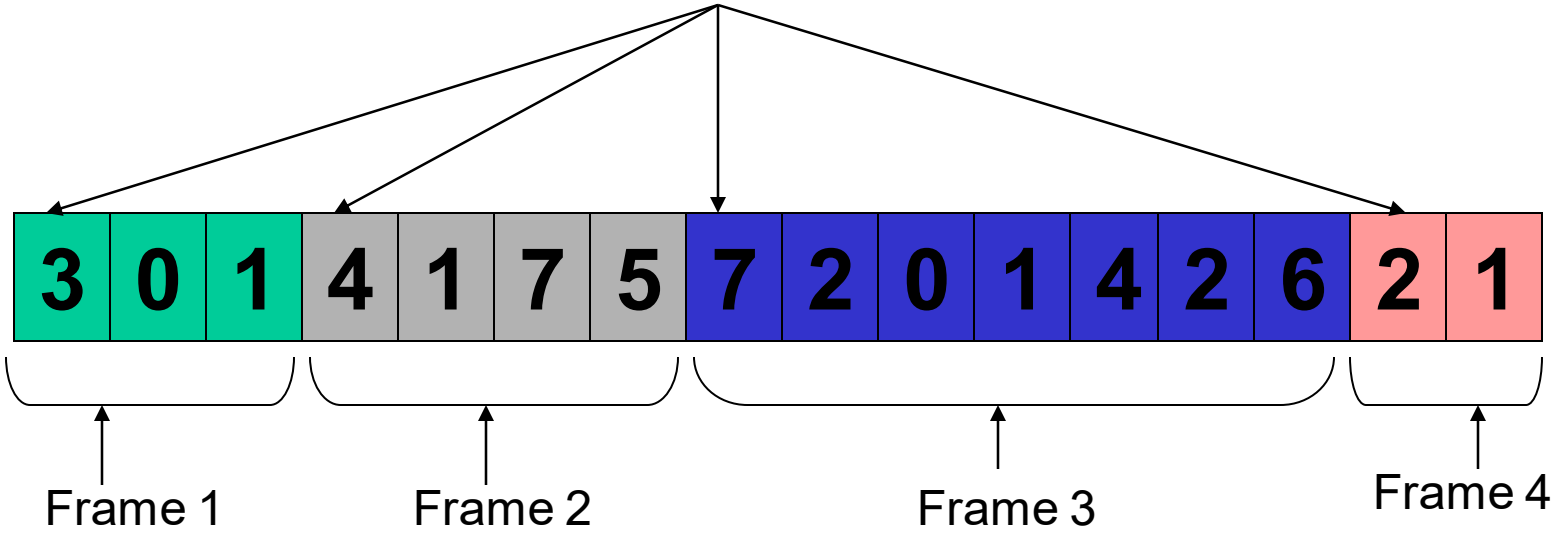
Unacknowledged connectionless service	• No logical connection is established before and or released afterward. • Ex: Ethernet
Acknowledged connectionless service	• There are still no logical connections used, but each frame sent is individually acknowledged. • Ex:802.11 (WiFi)
Acknowledged connection-oriented service.	• Establish a connection before any data are transferred. Each frame sent over the connection is numbered, and the data link layer guarantees that each frame sent is indeed received. • Ex: Satellite channel or a long-distance telephone circuit

FRAMING


Framing methods
Byte count
Flag bytes
Flag bits
Physical layer coding violations

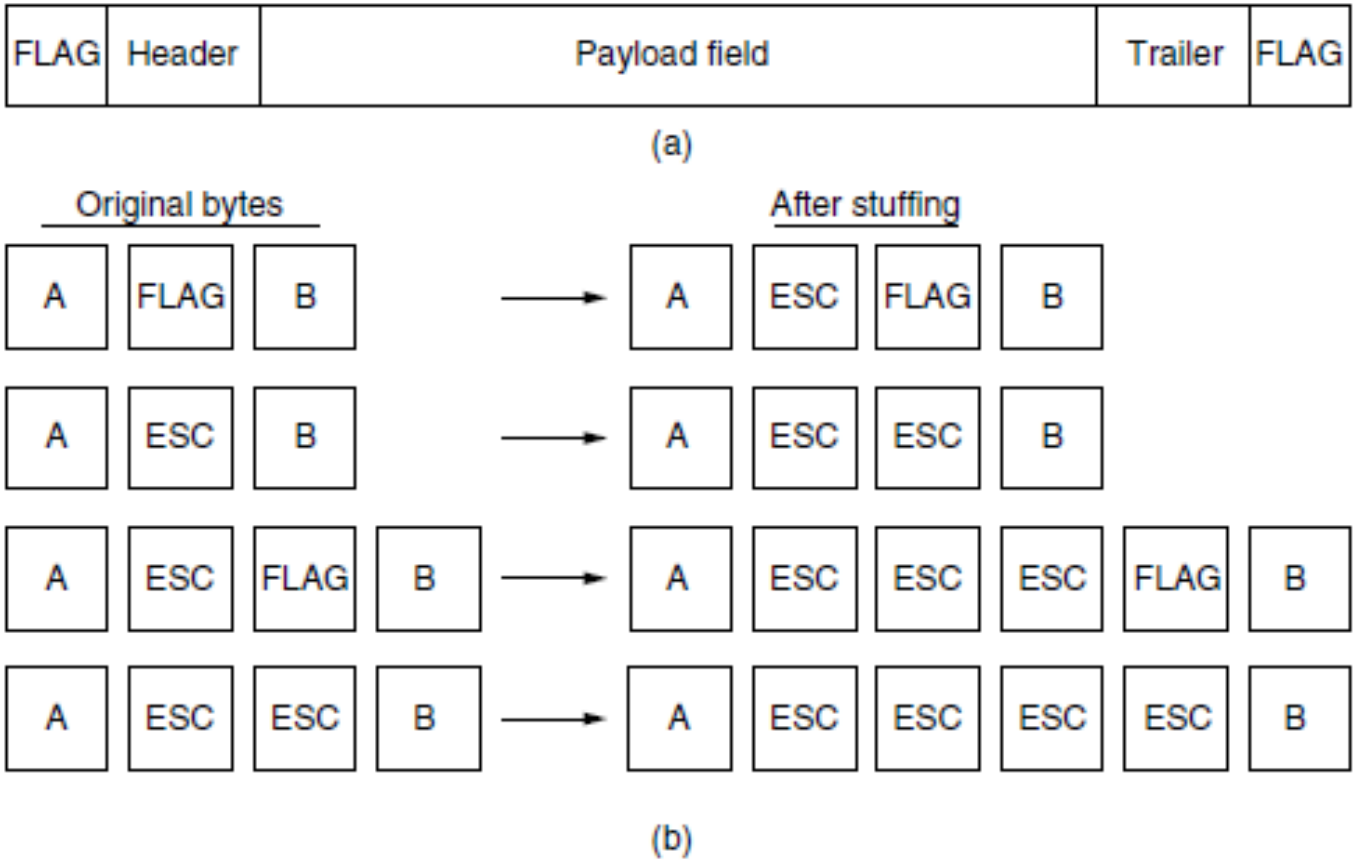
FRAMING

1. Byte count



FRAMING

2. Flag Bytes

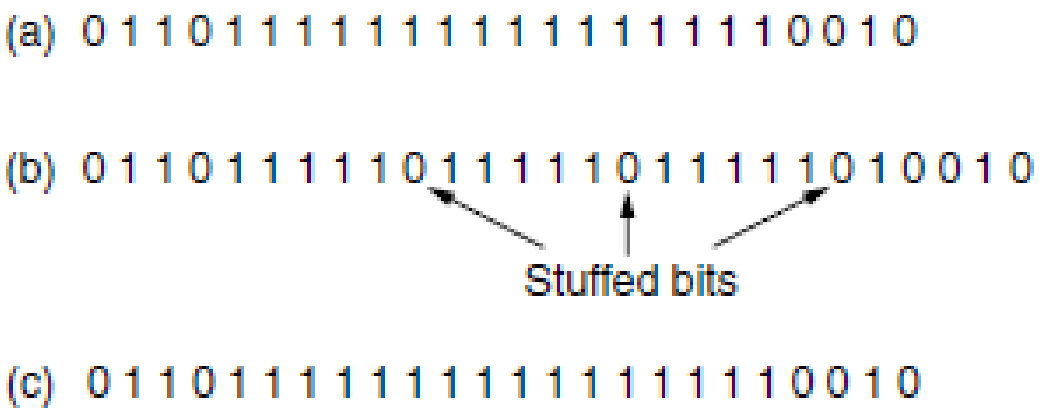


(a) A frame delimited by flag bytes. (b) Four examples of byte sequences before and after byte stuffing.

FRAMING

3. Flag Bits

Each frame begins and ends with a special bit pattern, 01111110



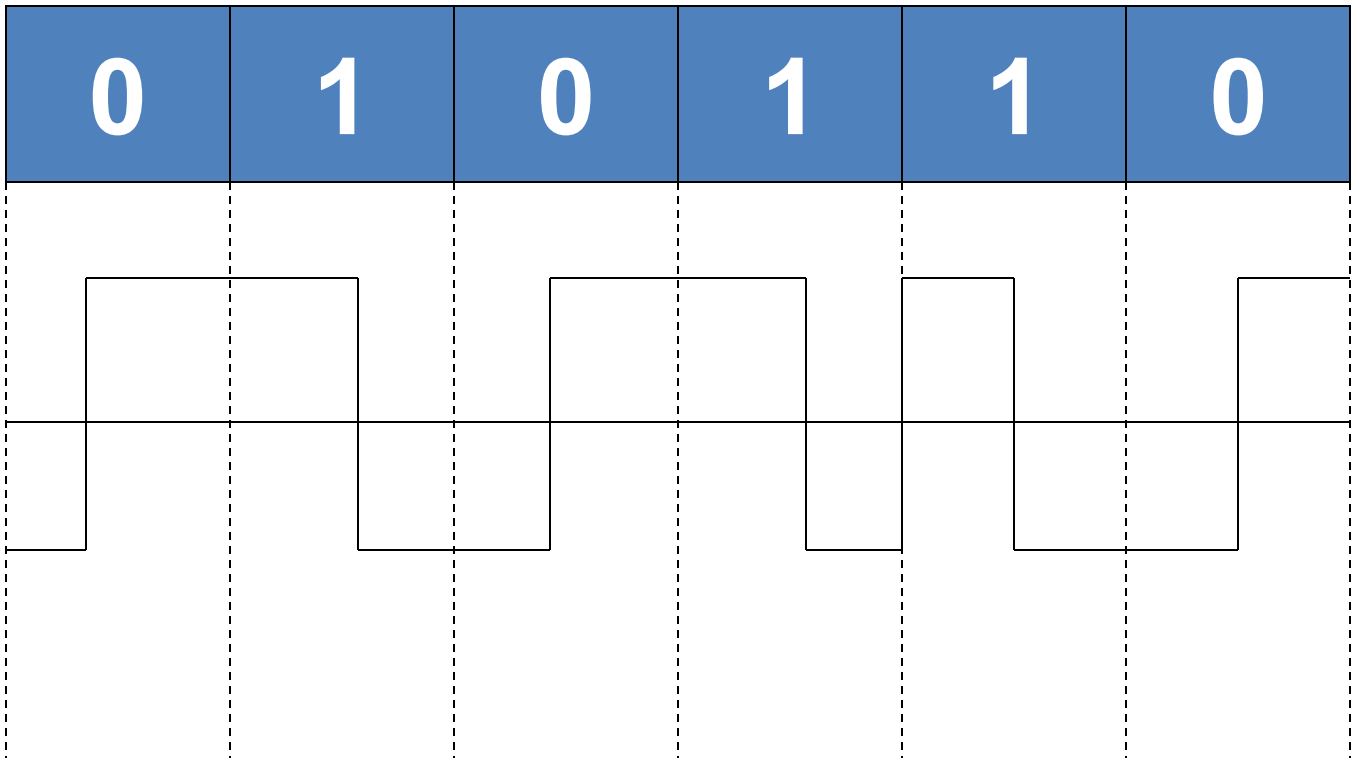
Bit stuffing. (a) The original data. (b) The data as they appear on the line. (c) The data as they are stored in the receiver's memory after destuffing.

FRAMING

4. Physical layer coding violations

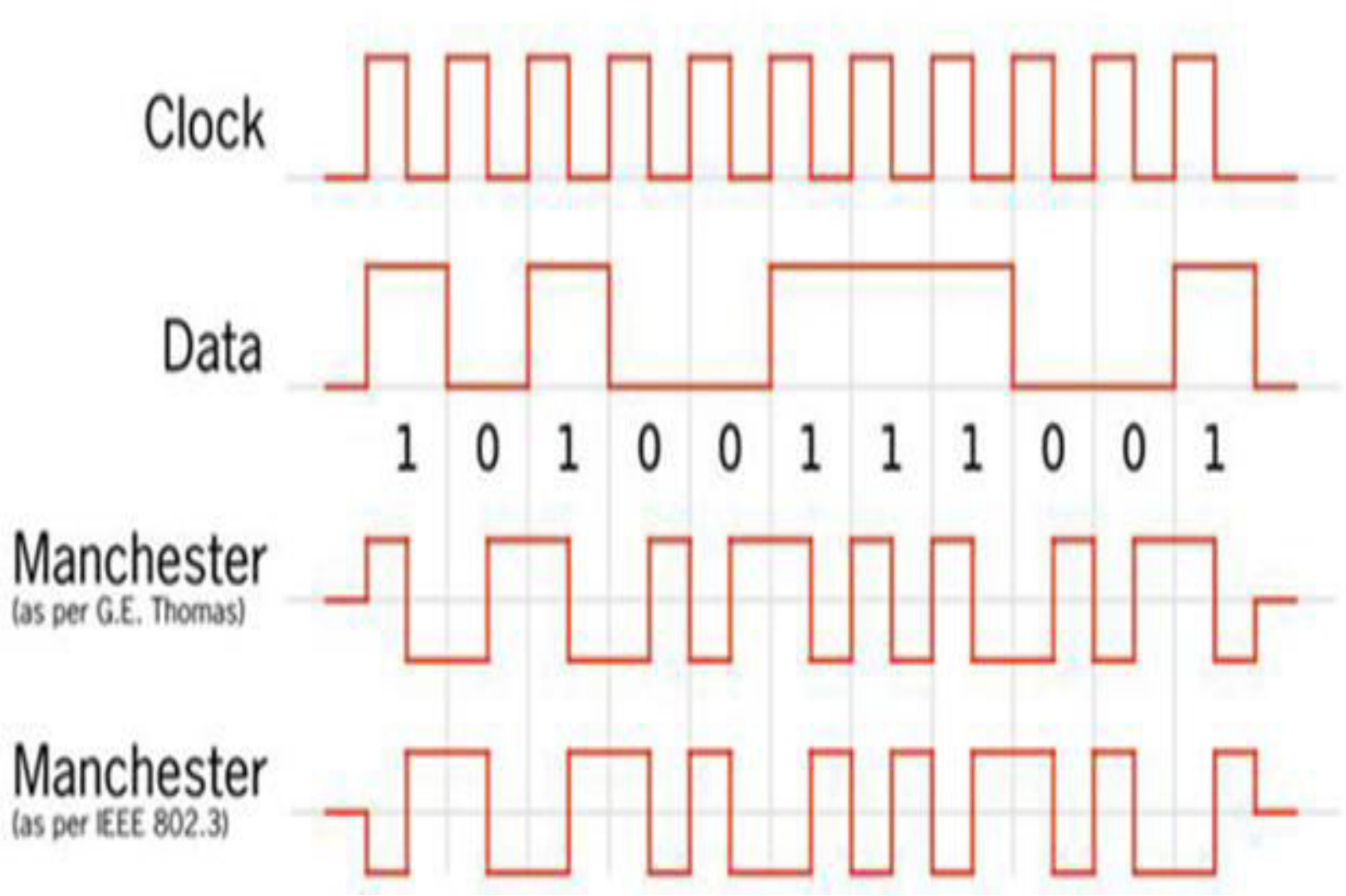
We can use some reserved signals to indicate the start and end of frames. In effect, we are using "coding violations" to delimit frames. The beauty of this scheme is that, because they are reserved signals, it is easy to find the start and end of frames and there is no need to stuff the data.

Physical Layer Coding Violation.

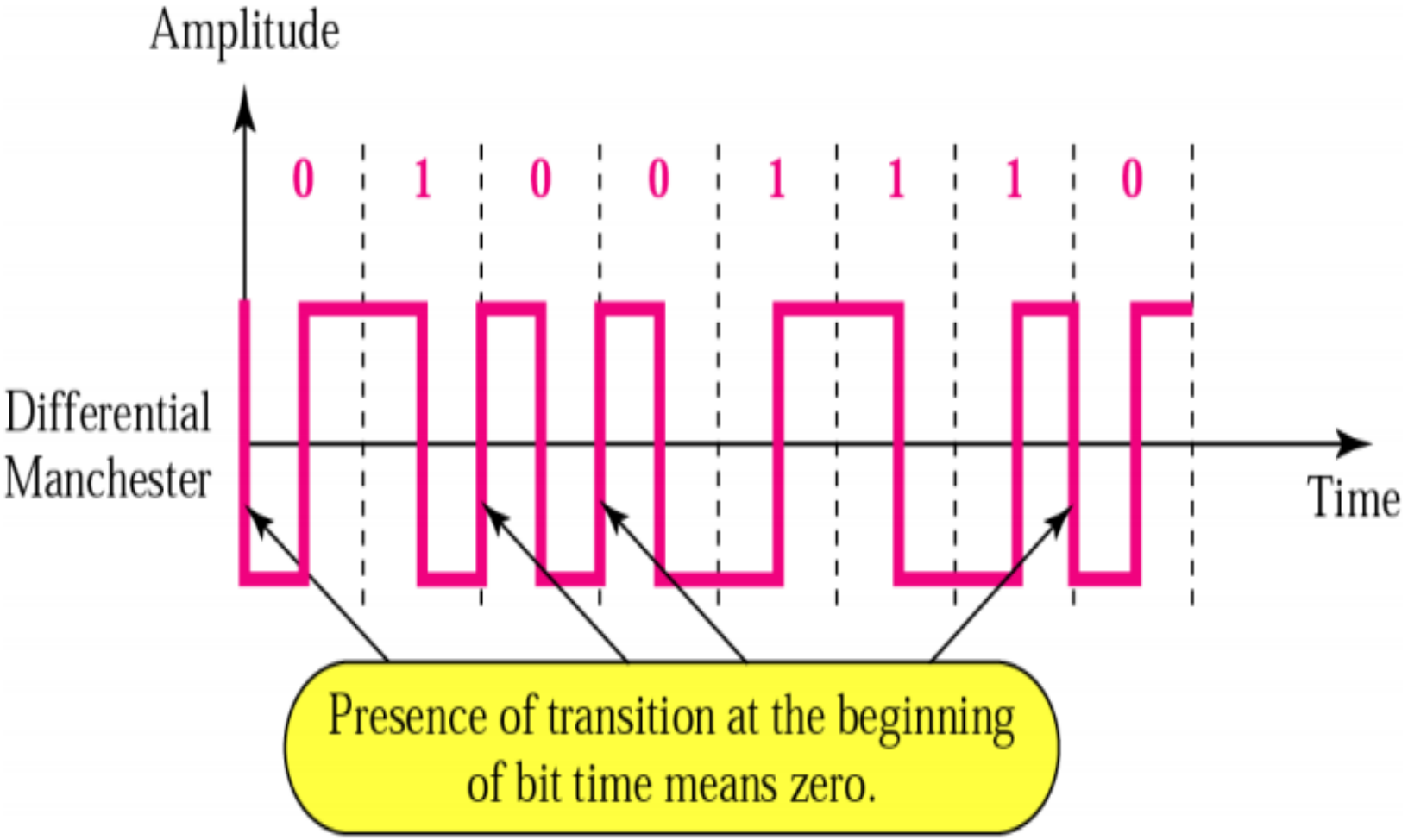


Manchester Encoding

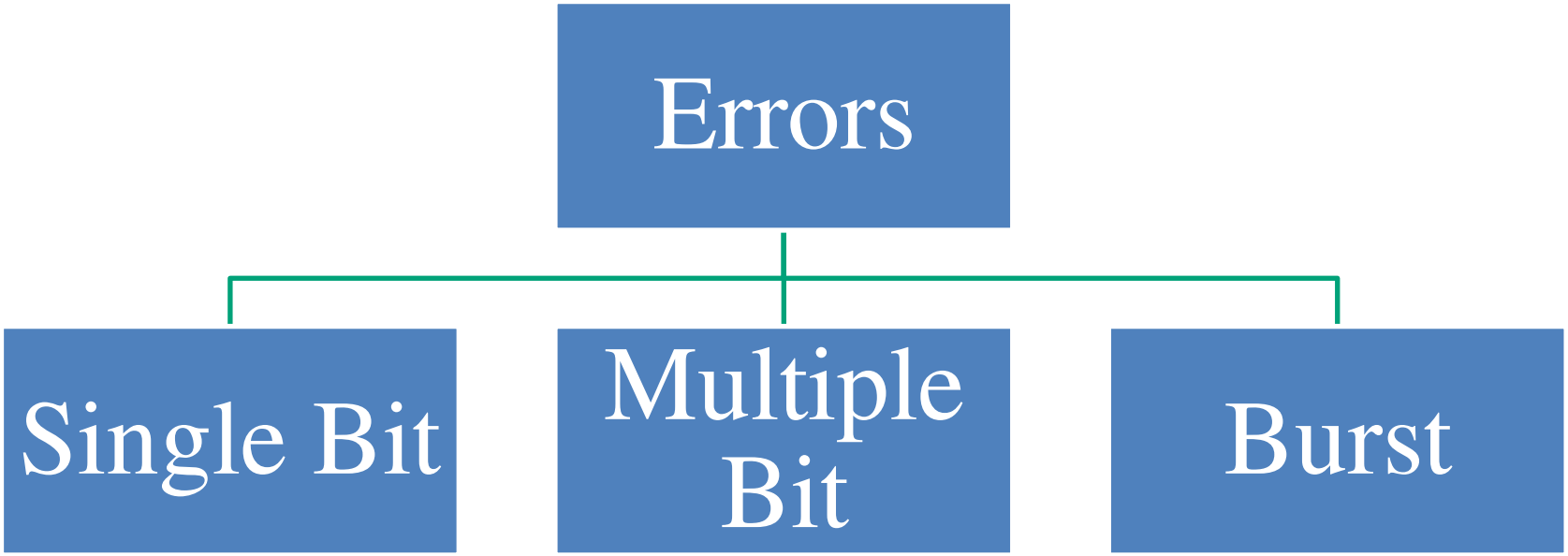
MANCHESTER ENCODING



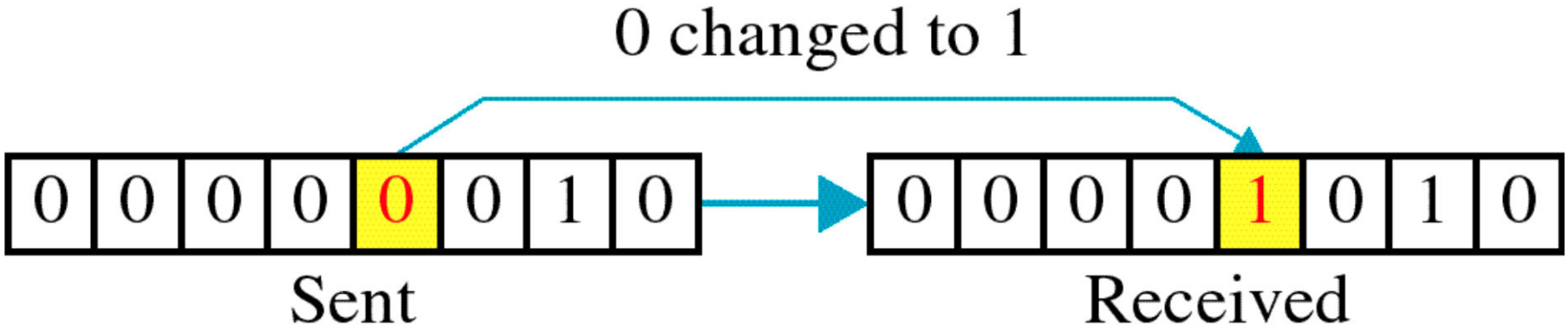
DIFFERENTIAL MANCHESTER ENCODING



TYPES OF ERRORS



Single-bit error

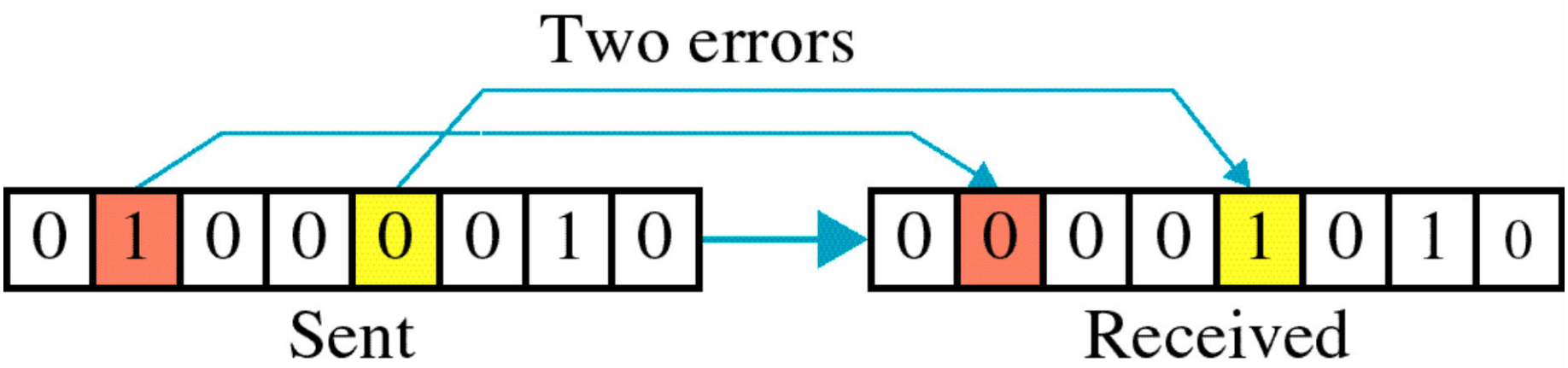


Single bit errors are the **least likely** type of errors in serial data transmission because the noise must have a very short duration which is very rare. However this kind of errors can happen in parallel transmission.

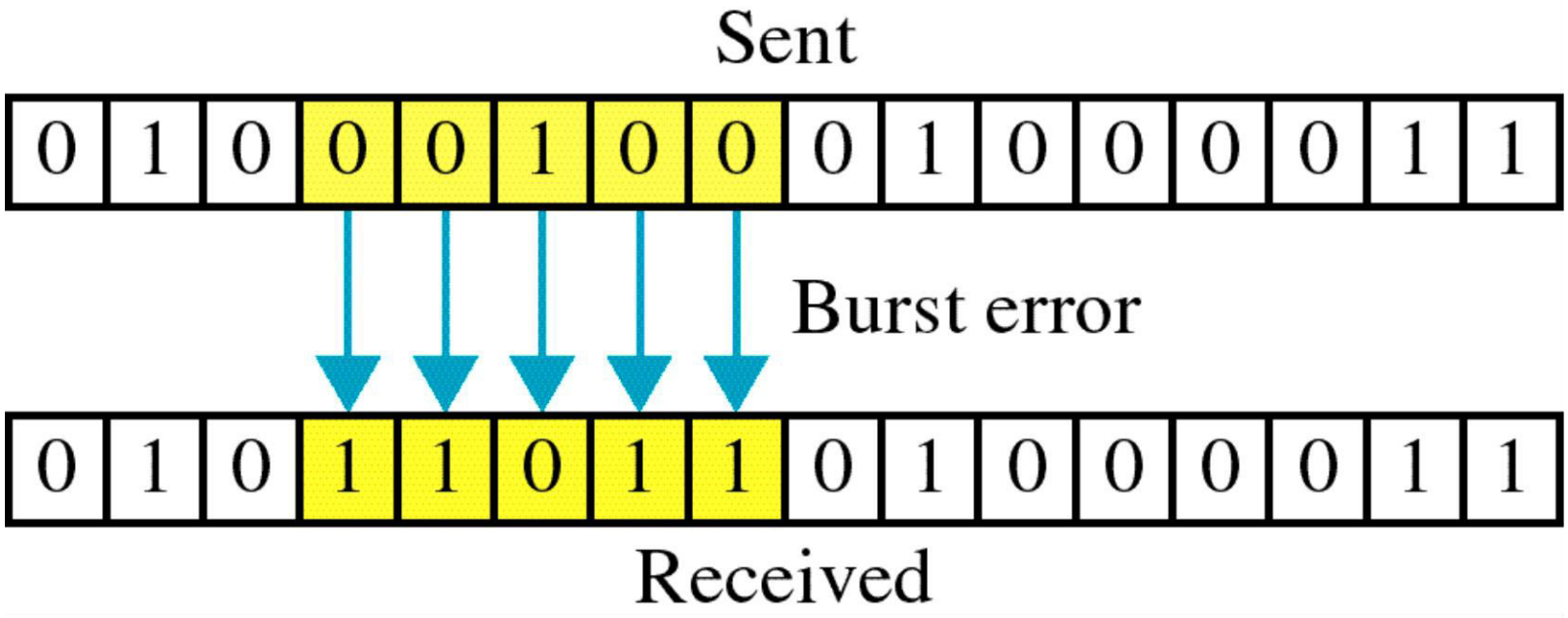
Example:

- ★ If data is sent at 1Mbps then each bit lasts only 1/1,000,000 sec. or 1 μs.
- ★ For a single-bit error to occur, the noise must have a duration of only 1 μs, which is very rare.

Two-bit error



Burst error


Sent
0 1 0 0 0 1 0 0 0 1 0 0 0 0 1 1
Burst error
0 1 0 1 1 0 1 1 0 1 0 0 0 0 1 1
Received

★ **Burst error is most likely to happen in serial transmission** since the duration of noise is normally longer than the duration of a bit.

★ The number of bits affected depends on the data rate and duration of noise.

Example:

➔If data is sent at rate = 1Kbps then a noise of 1/100 sec can affect 10 bits.(1/100)*1000)

➔If same data is sent at rate = 1Mbps then a noise of 1/100 sec can affect 10,000 bits.(1/100)*10^6)

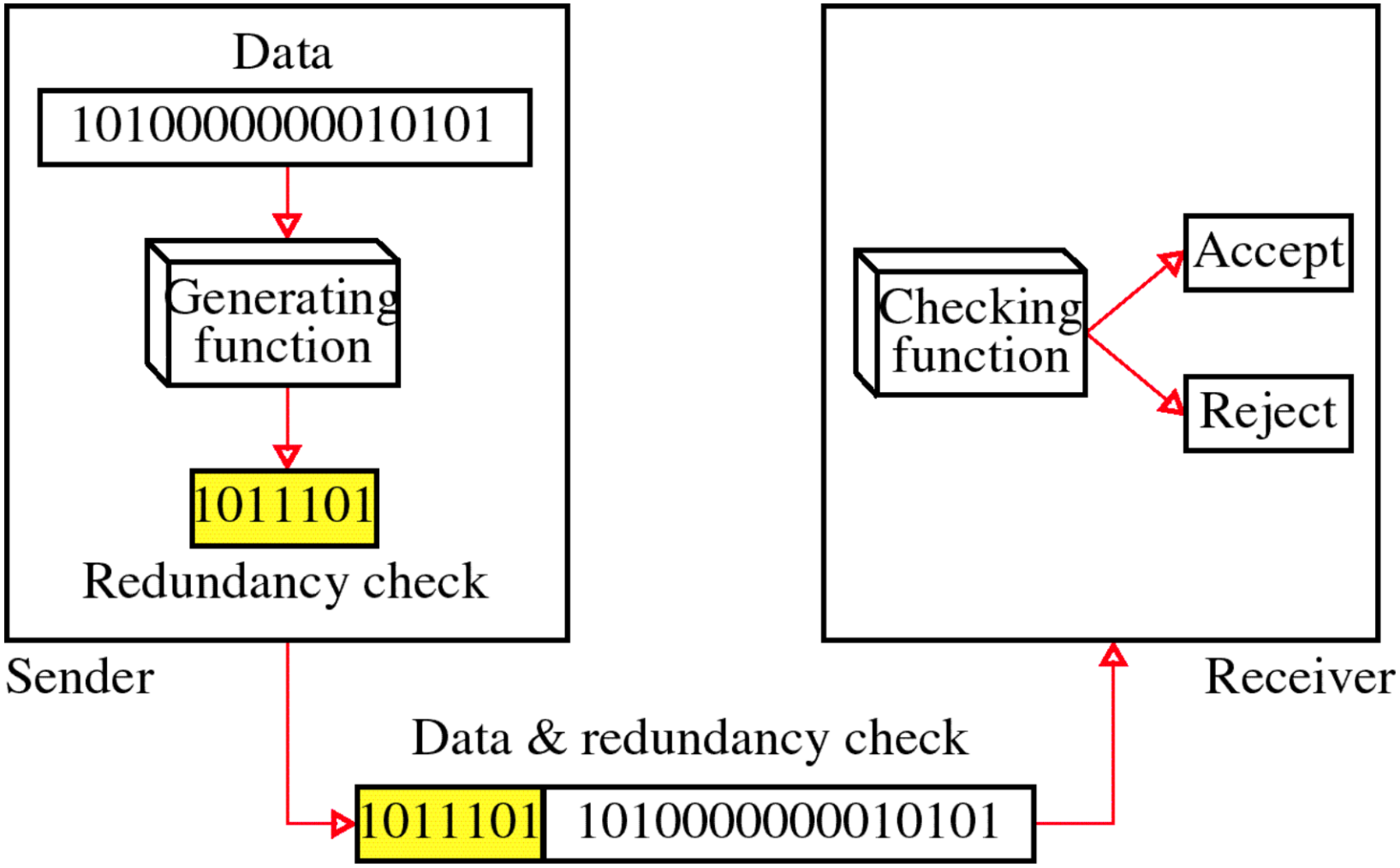
ERROR DETECTION AND CORRECTION

- **Error-Detecting Codes**
- **Error-Correcting Codes**

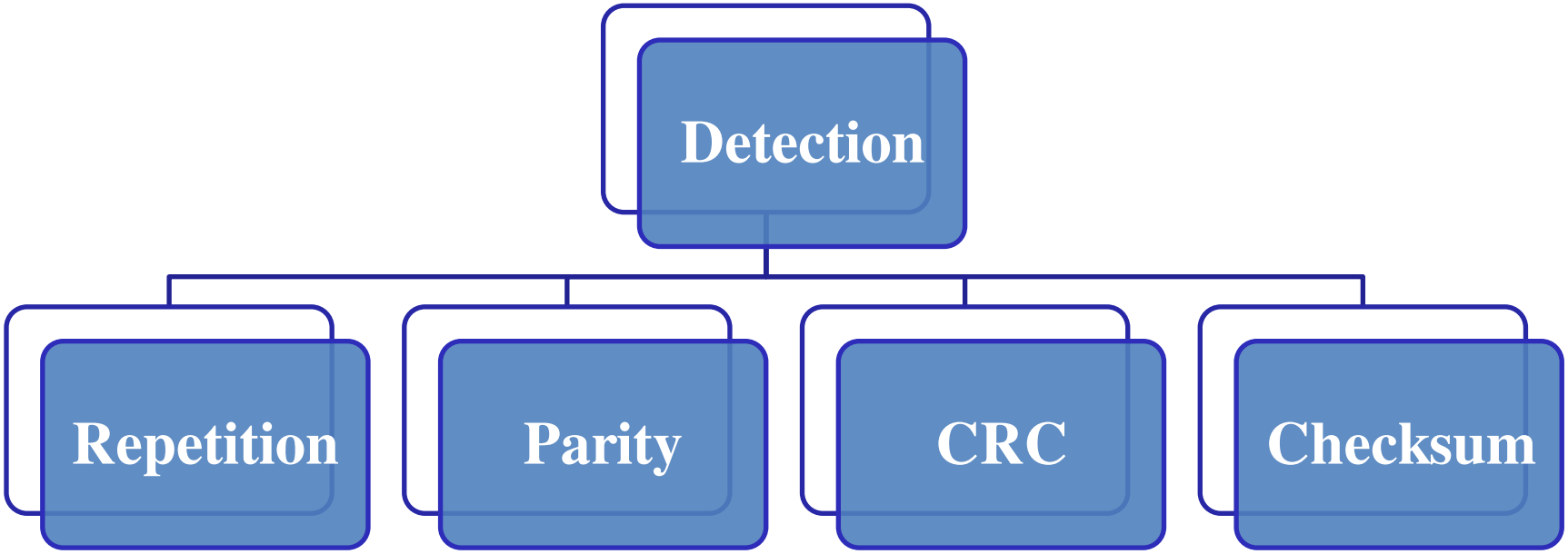
ERROR DETECTION

Error detection means to decide whether the received data is correct or not without having a copy of the original message. Error detection uses the concept of redundancy, which means adding extra bits for detecting errors at the destination.

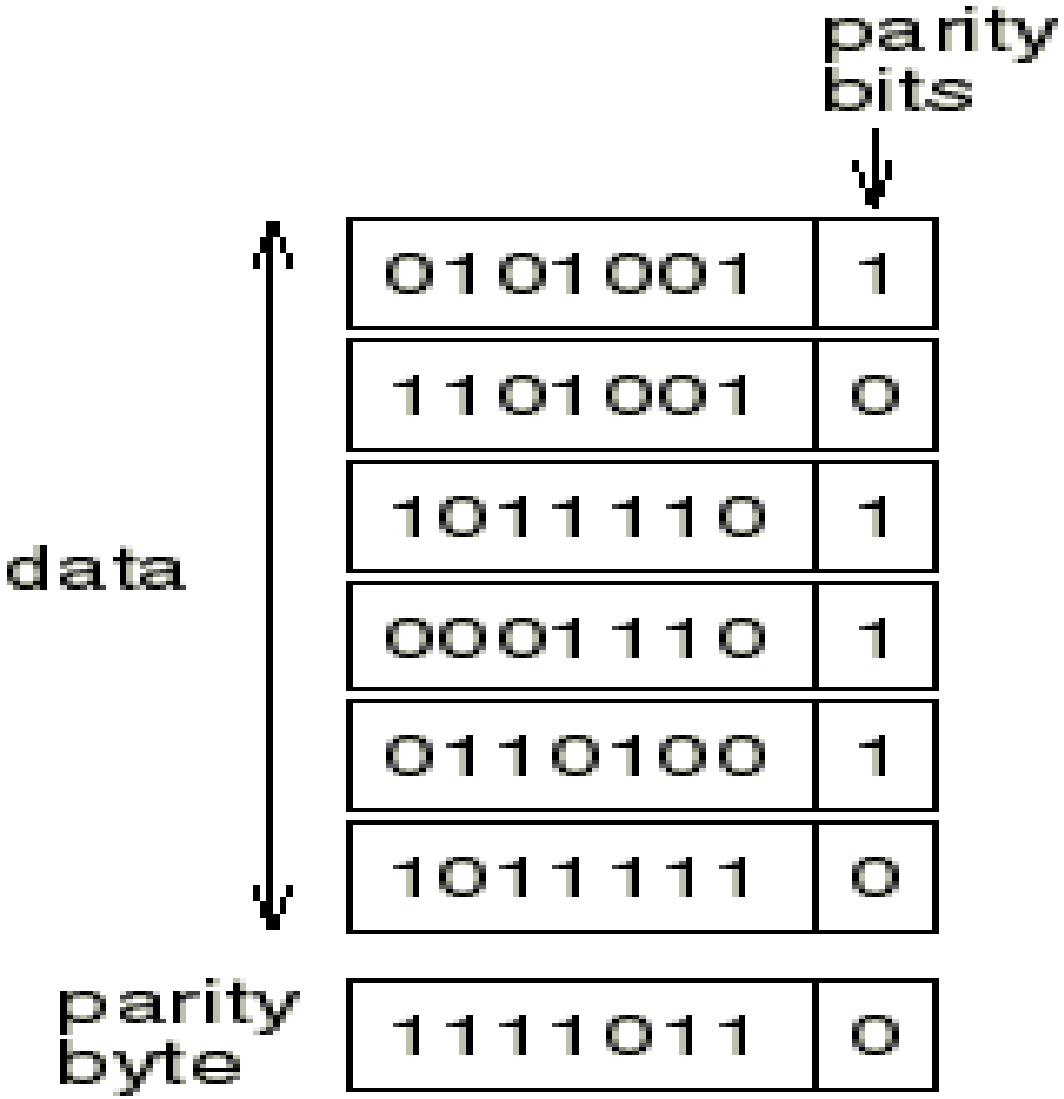
REDUNDANCY


Data
1010000000010101
Generating function
1011101
Redundancy check
Sender
Checking function
Accept
Reject
Receiver
Data & redundancy check
1011101
1010000000010101

ERROR DETECTION

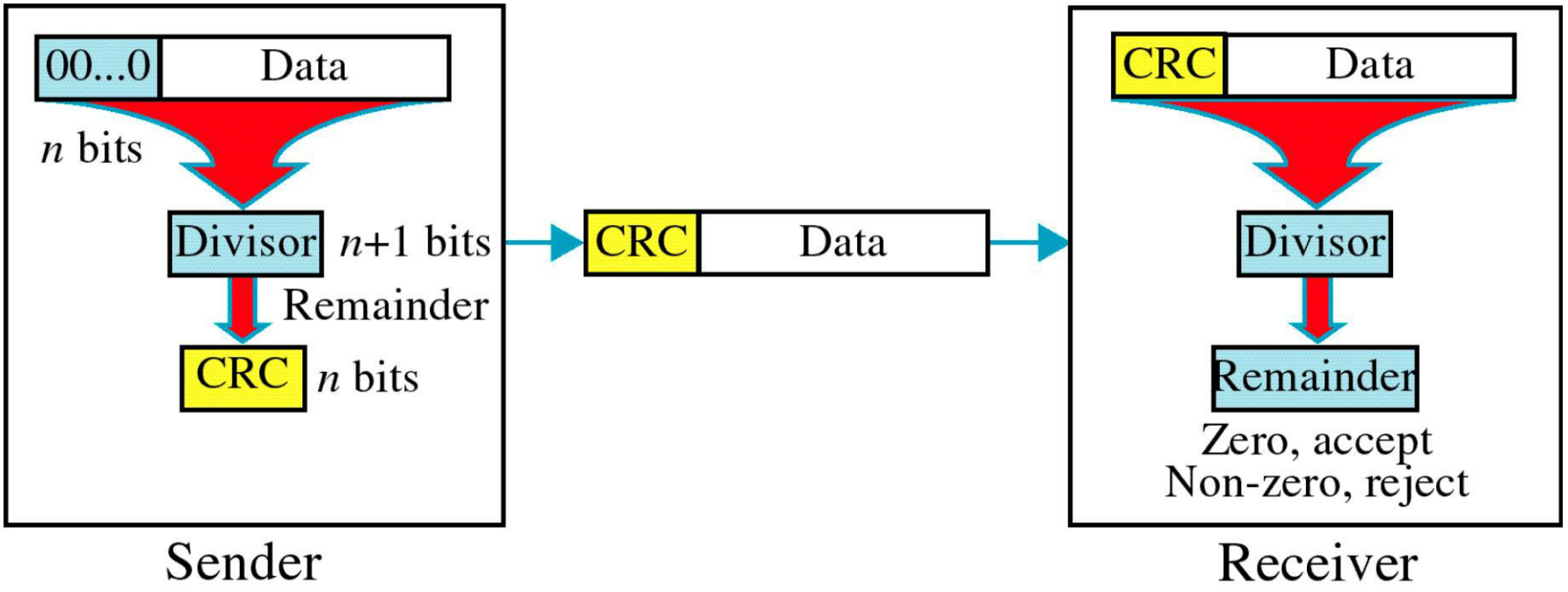


TWO-DIMENSIONAL PARITY

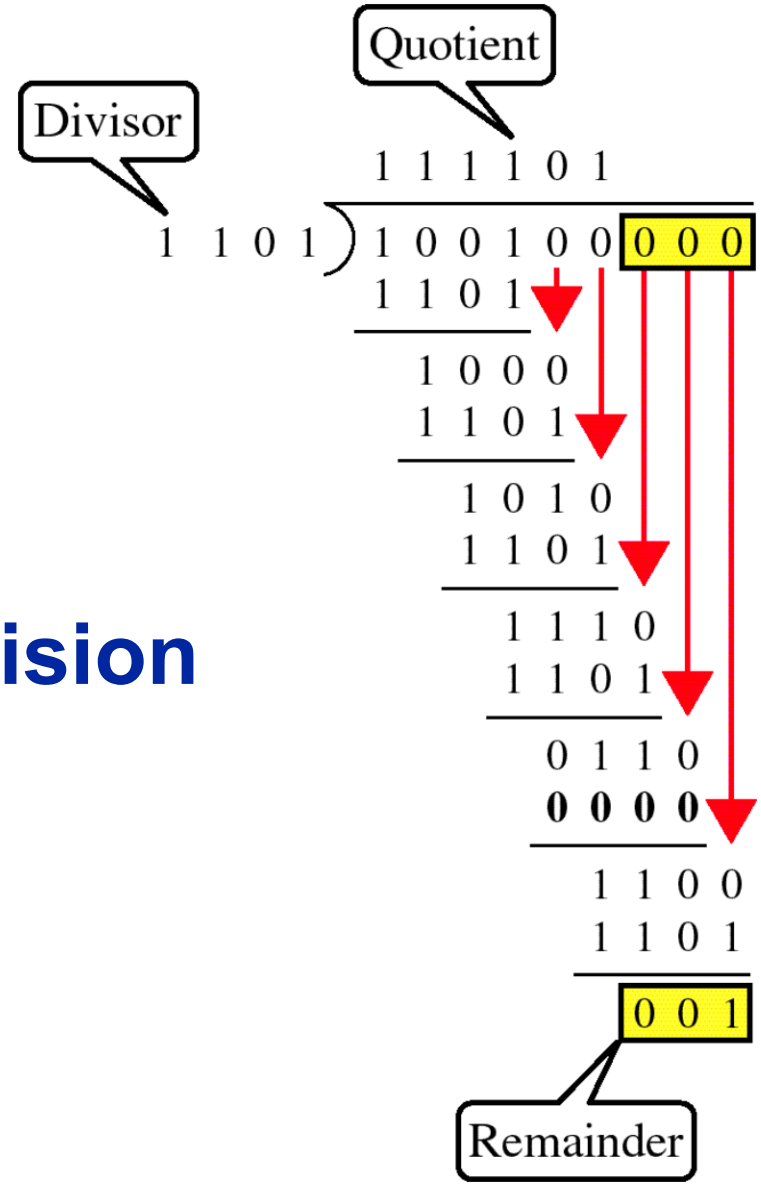


CYCLIC REDUNDANCY CHECK

CRC



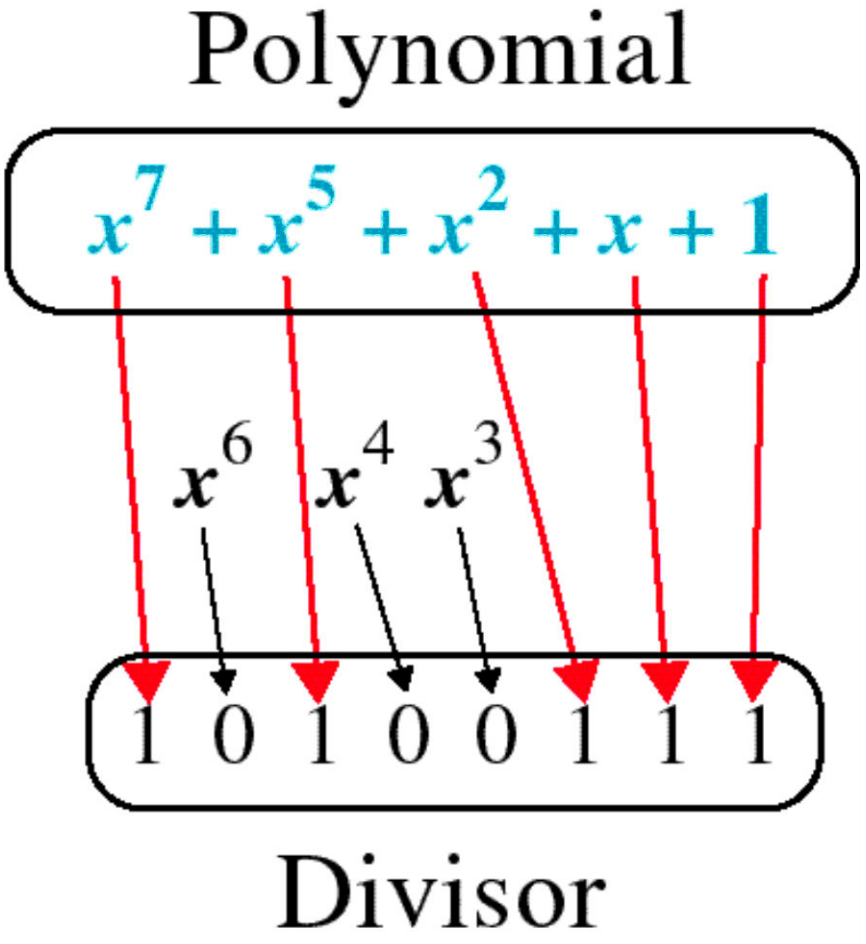
Binary Division



POLYNOMIAL

$$x^7 + x^5 + x^2 + x + 1$$

POLYNOMIAL AND DIVISOR



STANDARD POLYNOMIALS

CRC-12

$$x^{12}+x^{11}+x^{3}+x+1$$

CRC-16

$$x^{16}+x^{15}+x^{2}+1$$

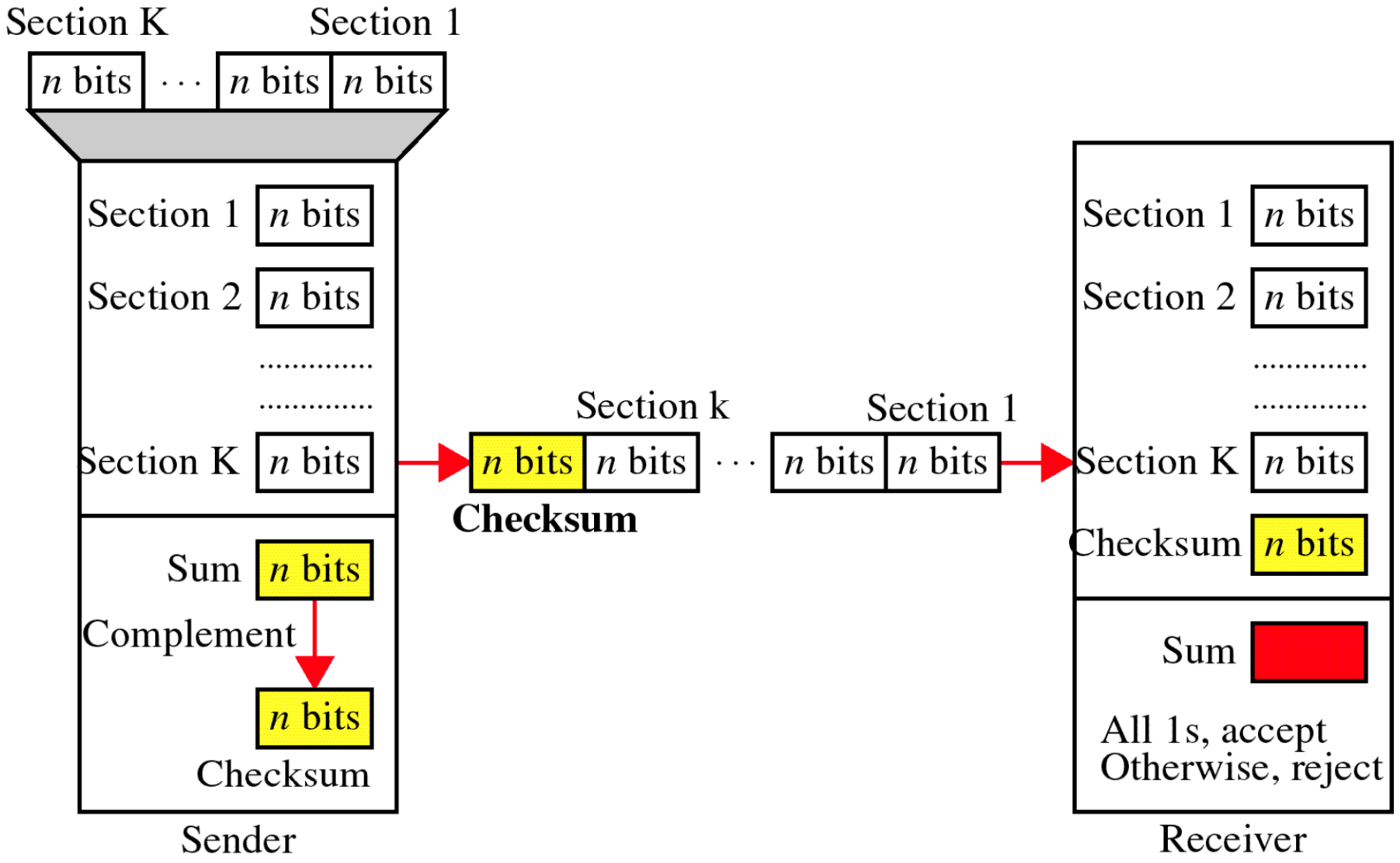
CRC-ITU

$$x^{16}+x^{12}+x^{5}+1$$

CRC-32

$$x^{32}+x^{26}+x^{23}+x^{22}+x^{16}+x^{12}+x^{11}+x^{10}+x^{8}+x^{7}+x^{5}+x^{4}+x^{2}+x+1$$

CHECKSUM


Section K
Section 1
n bits
n bits
n bits
Section 1
n bits
Section 2
n bits
Section K
n bits
Sum
n bits
Complement
n bits
Checksum
Sender
Section k
Section 1
n bits
n bits
n bits
n bits
Checksum
Section 1
n bits
Section 2
n bits
Section K
n bits
Checksum
n bits
Sum
All 1s, accept
Otherwise, reject
Receiver

AT THE SENDER

- The unit is divided into *k* sections, each of *n* bits.
- All sections are added together using one's complement to get the sum.
- The sum is complemented and becomes the checksum.
- The checksum is sent with the data

AT THE RECEIVER

- The unit is divided into *k* sections, each of *n* bits.
- All sections are added together using one's complement to get the sum.
- The sum is complemented.
- If the result is zero, the data are accepted: otherwise, they are rejected.

ERROR CORRECTION

It can be handled in two ways:

1) **receiver can have the sender retransmit the entire data unit.**
2) **The receiver can use an error-correcting code, which automatically corrects certain errors.**

SINGLE-BIT ERROR CORRECTION

To correct an error, the receiver reverses the value of the altered bit. To do so, it must know which bit is in error.

Number of redundancy bits needed

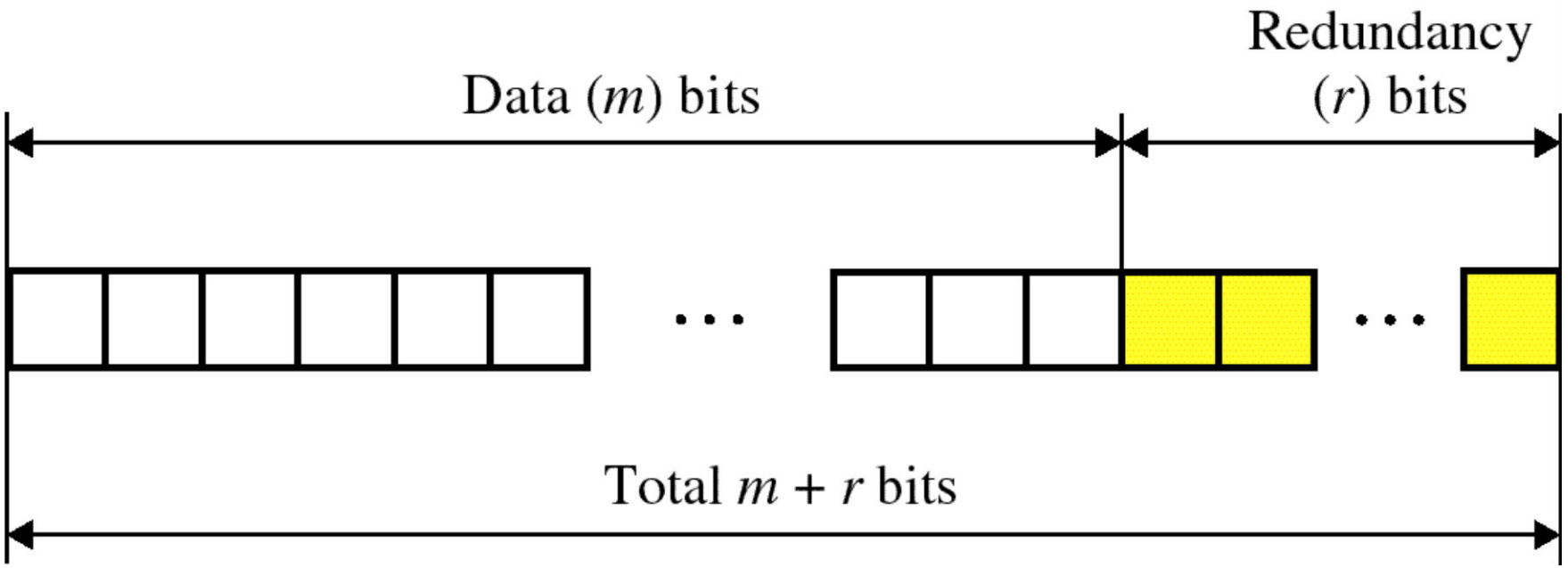
- **Let data bits = *m***
- **Redundancy bits = *r***

∴ Total message sent = *m*+*r*

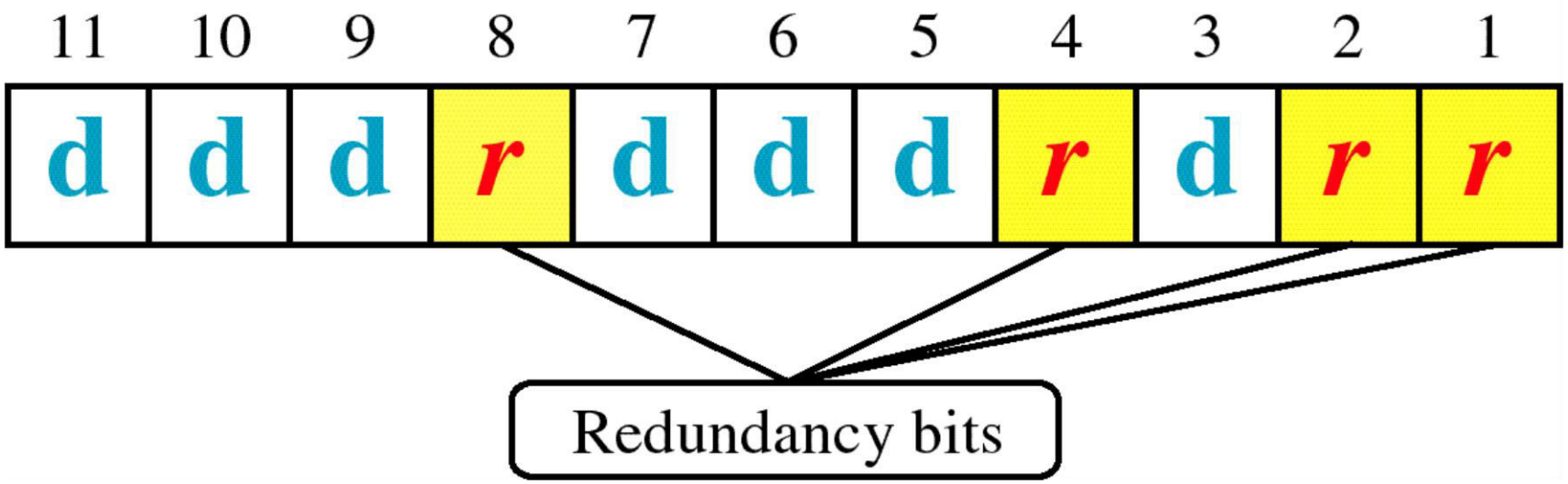
The value of r must satisfy the following relation:

$$2^r \geq m+r+1$$

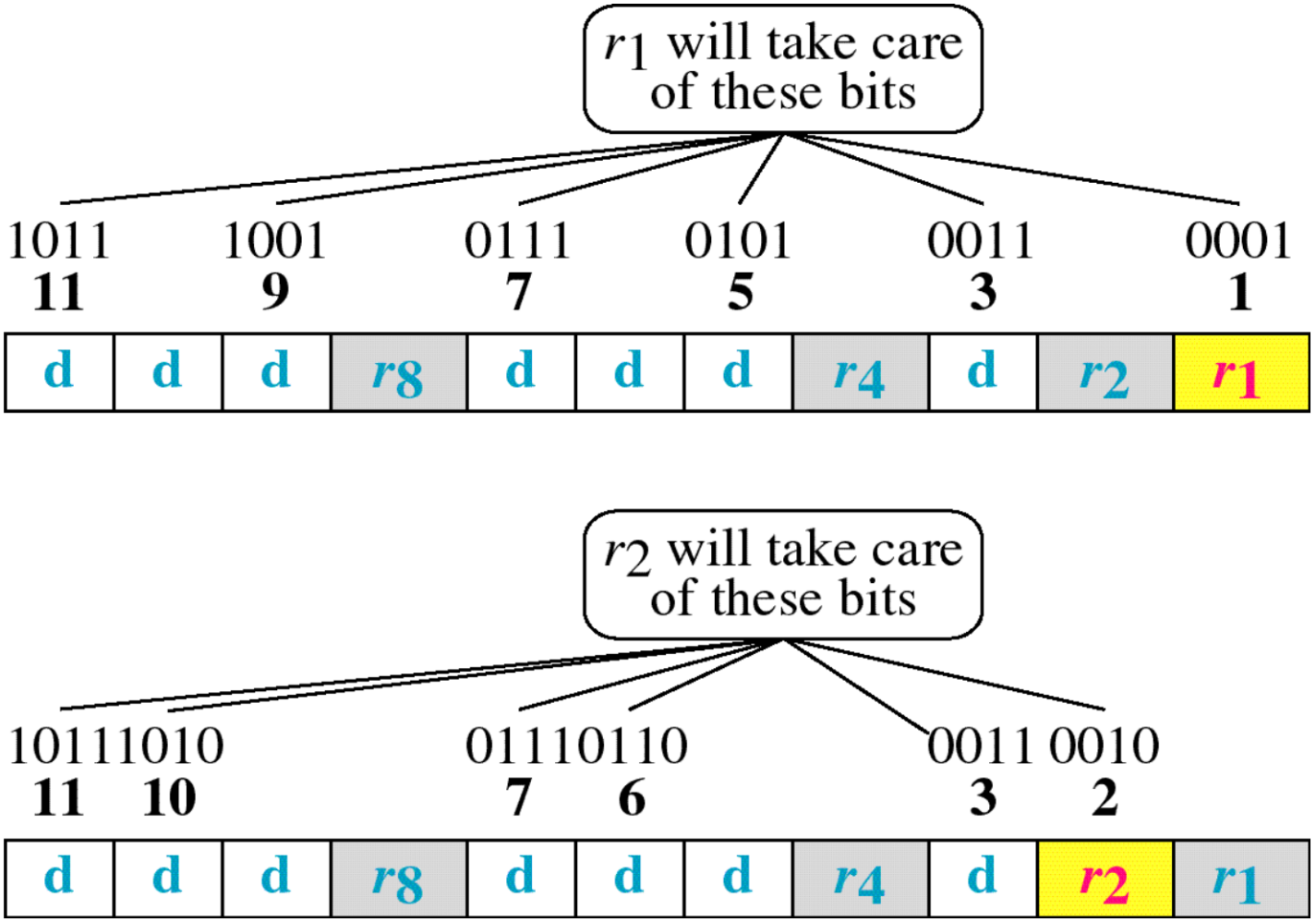
Error Correction



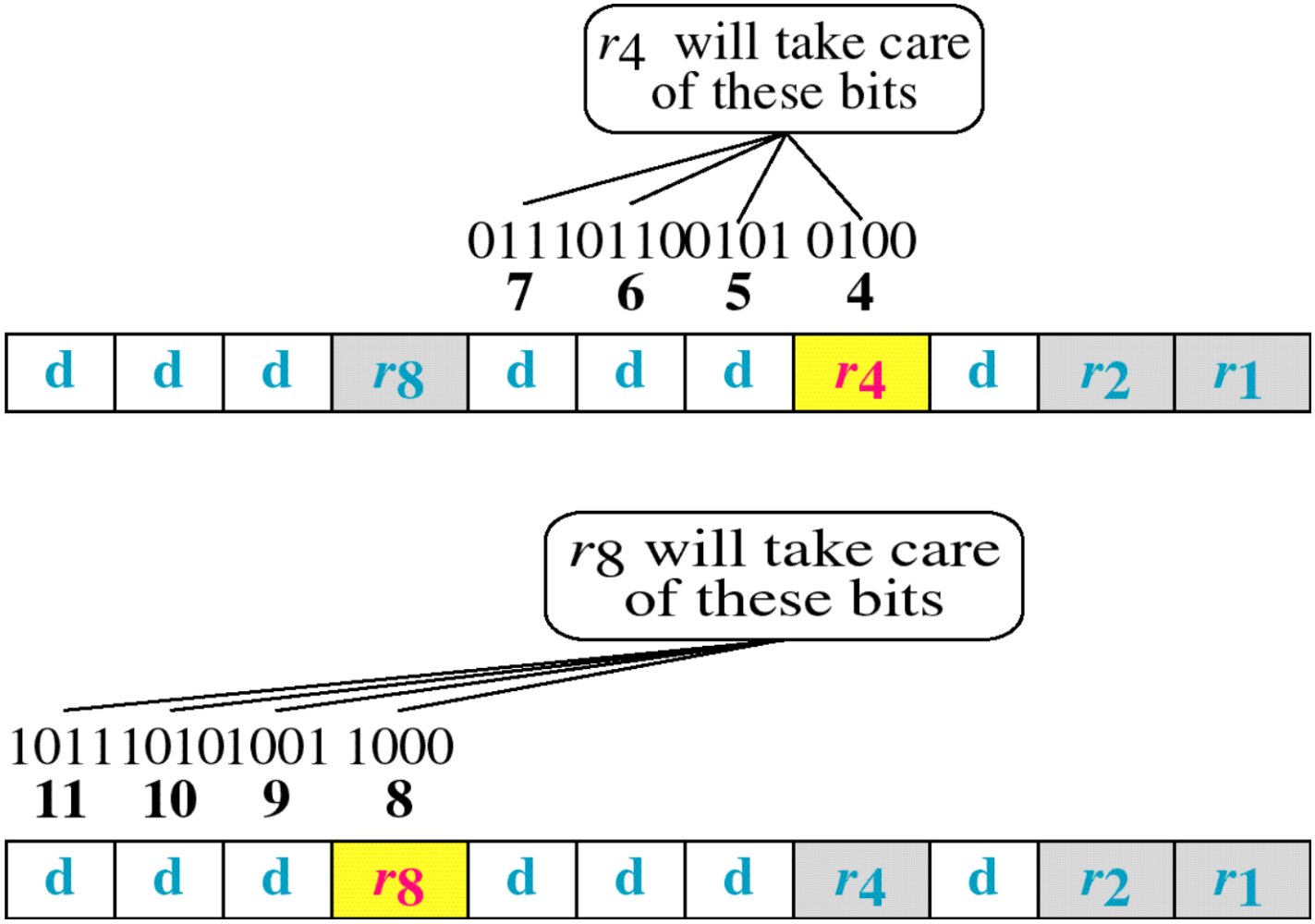
Hamming Code



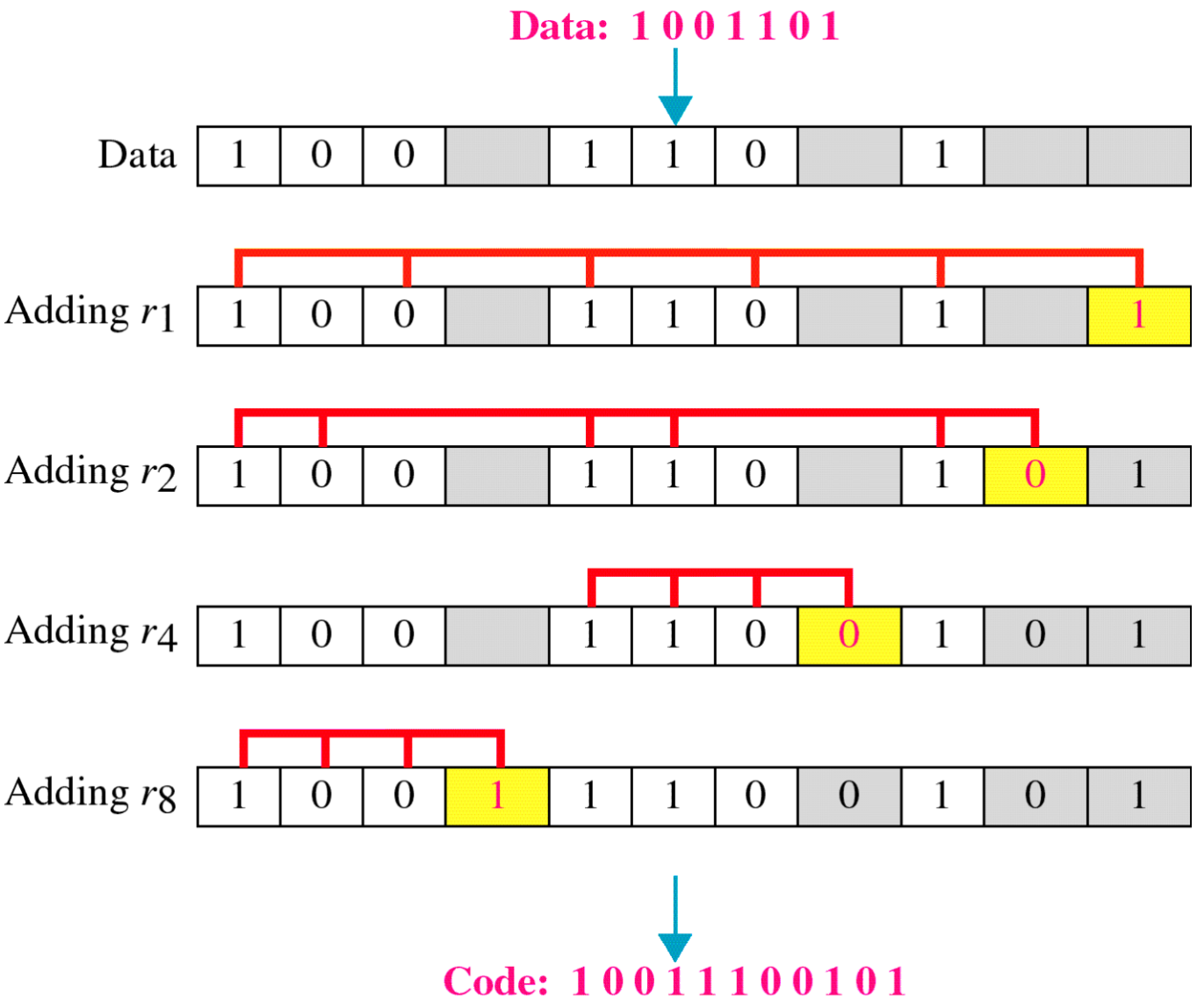
Hamming Code



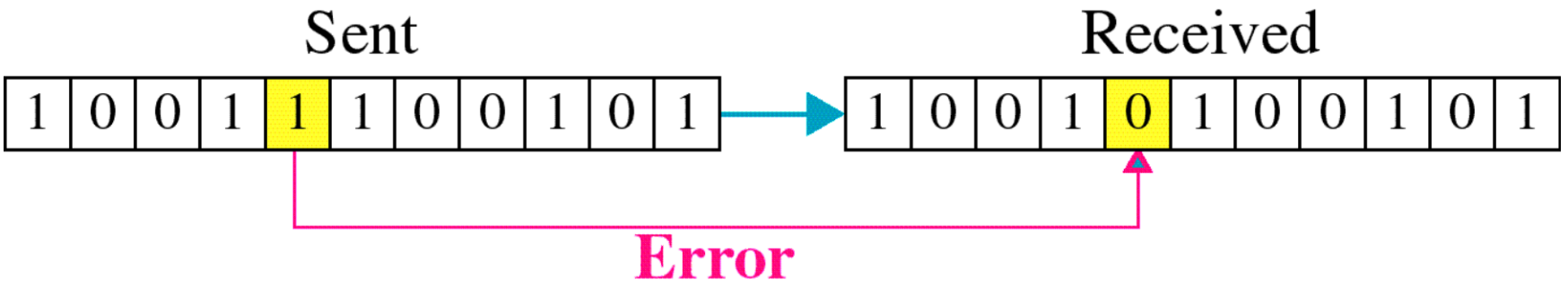
Hamming Code



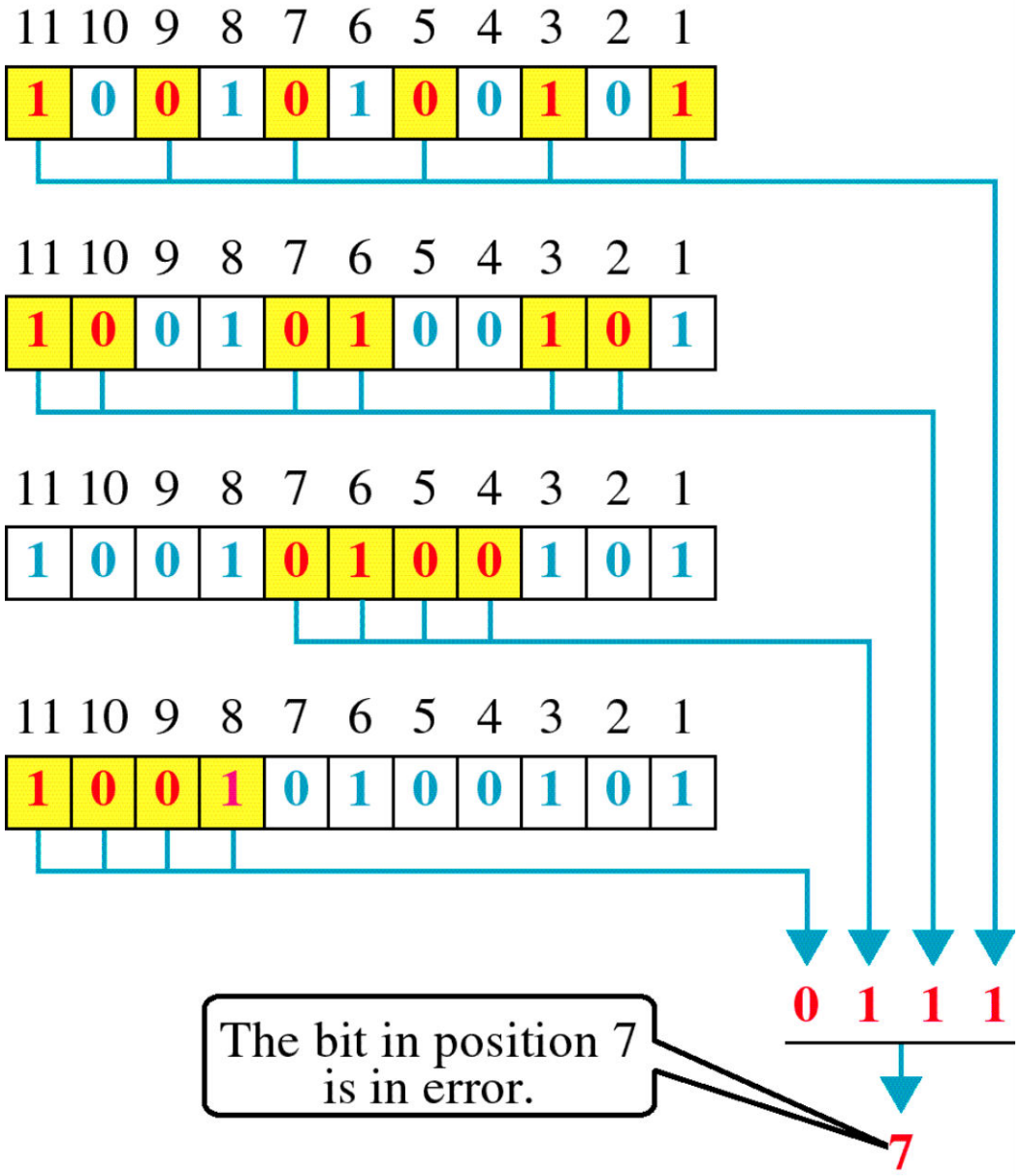
Example of Hamming Code



Single-bit error



Error Detection



ELEMENTARY DATA LINK PROTOCOLS

- An Unrestricted Simplex Protocol
- A Simplex Stop-and-Wait Protocol
- A Simplex Protocol for a Noisy Channel

UNRESTRICTED SIMPLEX PROTOCOL

- Data is transmitted in one direction only
- Transmitting and receiving network layers are always ready
- Processing time can be ignored
- Infinite buffer space is available
- Communication channels never damage/loses frames

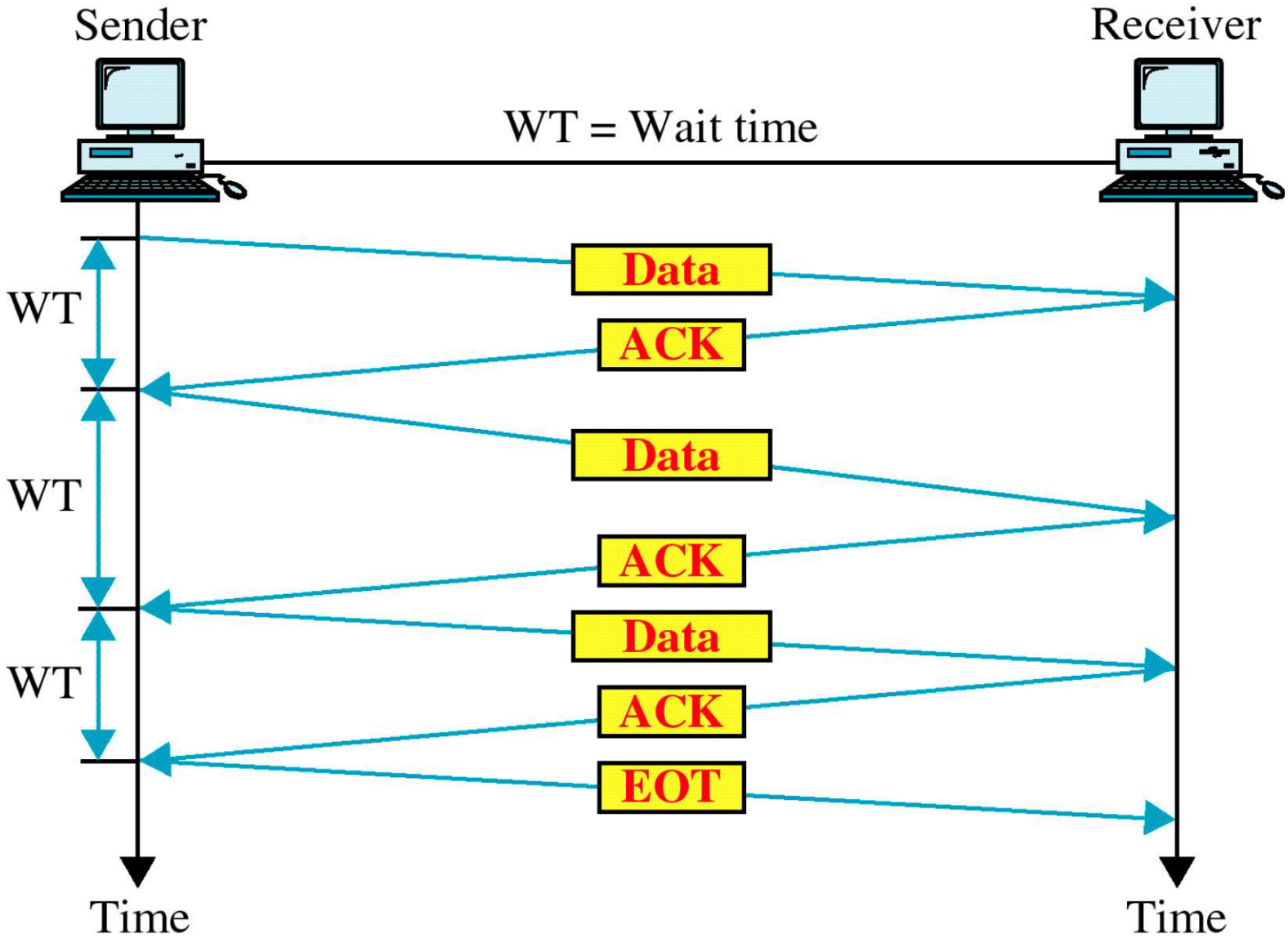
Unrestricted Simplex Protocol

```
/* Protocol 1 (utopia) provides for data transmission in one direction only, from
   sender to receiver.  The communication channel is assumed to be error free,
   and the receiver is assumed to be able to process all the input infinitely quickly.
   Consequently, the sender just sits in a loop pumping data out onto the line as
   fast as it can. */

typedef enum {frame_arrival} event_type;
#include "protocol.h"

void sender1(void)
{
  frame s;                              /* buffer for an outbound frame */
  packet buffer;                        /* buffer for an outbound packet */

  while (true) {
      from_network_layer(&buffer);      /* go get something to send */
      s.info = buffer;                  /* copy it into s for transmission */
      to_physical_layer(&s);            /* send it on its way */
  }                                      /*  * Tomorrow, and tomorrow, and tomorrow,
                                            Creeps in this petty pace from day to day
                                            To the last syllable of recorded time
                                                - Macbeth, V, v */
}

void receiver1(void)
{
  frame r;
  event_type event;                     /* filled in by wait, but not used here */

  while (true) {
      wait_for_event(&event);           /* only possibility is frame_arrival */
      from_physical_layer(&r);          /* go get the inbound frame */
      to_network_layer(&r.info);        /* pass the data to the network layer */
  }
}
```

SIMPLEX STOP-AND-WAIT PROTOCOL

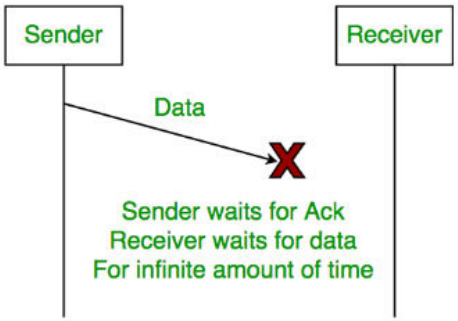
- How to prevent the sender from flooding the receiver with data
- Receiver provide feedback to the sender
- Protocols in which the sender sends one frame and then waits for an acknowledgement before proceeding are called stop-and-wait

Stop and Wait

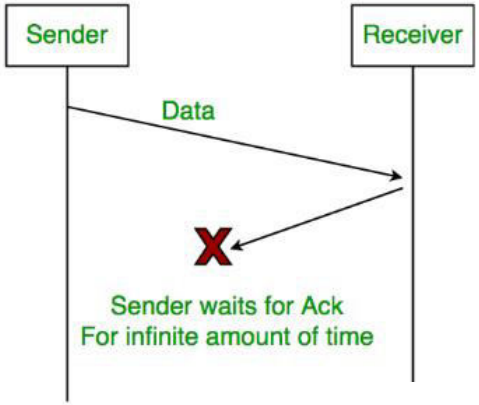


Stop and Wait-ARQ

Lost Data

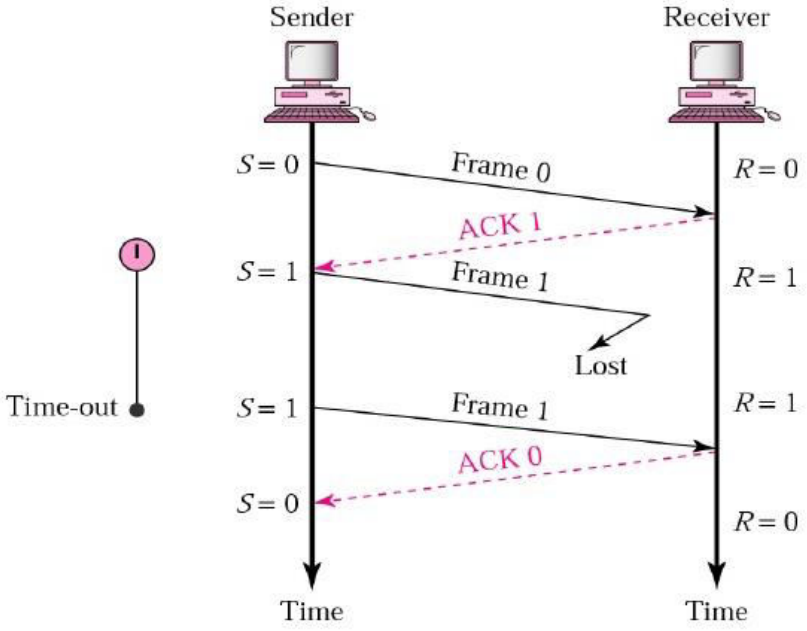


Lost Acknowledgement



Delayed Acknowledgement/Data

Above 3 problems are resolved by Stop and Wait ARQ (Automatic Repeat Request) that does both error control and flow control.



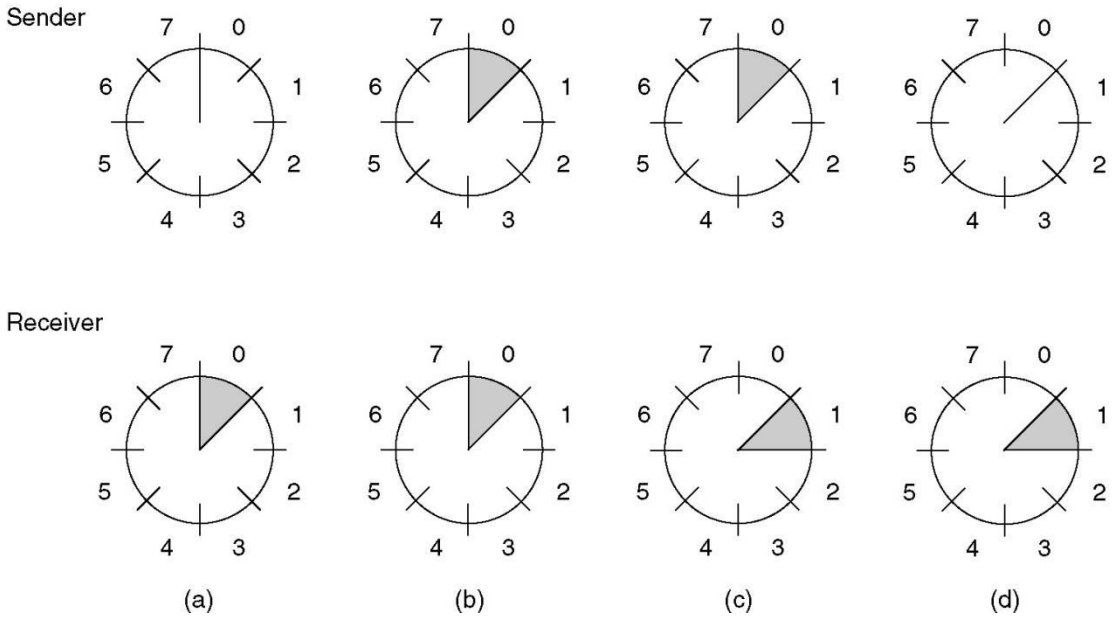
Simplex Stop-and-Wait Protocol

```
/* Protocol 2 (stop-and-wait) also provides for a one-directional flow of data from
   sender to receiver. The communication channel is once again assumed to be error
   free, as in protocol 1. However, this time, the receiver has only a finite buffer
   capacity and a finite processing speed, so the protocol must explicitly prevent
   the sender from flooding the receiver with data faster than it can be handled. */

typedef enum {frame_arrival} event_type;
#include "protocol.h"

void sender2(void)
{
  frame s;                              /* buffer for an outbound frame */
  packet buffer;                        /* buffer for an outbound packet */
  event_type event;                     /* frame_arrival is the only possibility */

  while (true) {
      from_network_layer(&buffer);      /* go get something to send */
      s.info = buffer;                  /* copy it into s for transmission */
      to_physical_layer(&s);            /* bye bye little frame */
      wait_for_event(&event);           /* do not proceed until given the go ahead */
  }
}

void receiver2(void)
{
  frame r, s;                           /* buffers for frames */
  event_type event;                     /* frame_arrival is the only possibility */
  while (true) {
      wait_for_event(&event);           /* only possibility is frame_arrival */
      from_physical_layer(&r);          /* go get the inbound frame */
      to_network_layer(&r.info);        /* pass the data to the network layer */
      to_physical_layer(&s);            /* send a dummy frame to awaken sender */
  }
}
```

SLIDING WINDOW PROTOCOLS

- **Need for transmitting data in both the directions.**
- **The technique of temporarily delaying outgoing acknowledgements so that they can be hooked onto the next outgoing data frame is known as piggybacking**

SLIDING WINDOW PROTOCOLS



A sliding window of size 1, with a 3-bit sequence number.

(a) Initially.

(b) After the first frame has been sent.

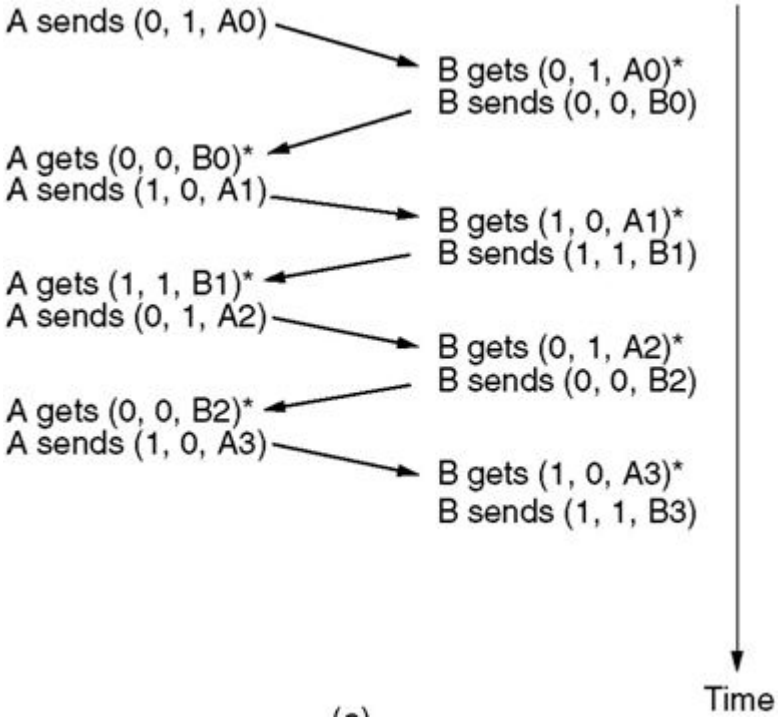
(c) After the first frame has been received.

(d) After the first acknowledgement has been received.

SLIDING WINDOW PROTOCOLS

- A One-Bit Sliding Window Protocol
- A Protocol Using Go Back N
- A Protocol Using Selective Repeat

A ONE-BIT SLIDING WINDOW PROTOCOL



The notation is (seq, ack, packet number). An asterisk indicates where a network layer accepts a packet.

A ONE-BIT SLIDING WINDOW PROTOCOL

```
/* Protocol 4 (sliding window) is bidirectional. */
#define MAX_SEQ 1                                   /* must be 1 for protocol 4 */
typedef enum {frame_arrival, cksum_err, timeout} event_type;
#include "protocol.h"
void protocol4 (void)
{
  seq_nr next_frame_to_send;                        /* 0 or 1 only */
  seq_nr frame_expected;                            /* 0 or 1 only */
  frame r, s;                                       /* scratch variables */
  packet buffer;                                    /* current packet being sent */
  event_type event;
  next_frame_to_send = 0;                           /* next frame on the outbound stream */
  frame_expected = 0;                               /* frame expected next */
  from_network_layer(&buffer);                      /* fetch a packet from the network layer */
  s.info = buffer;                                  /* prepare to send the initial frame */
  s.seq = next_frame_to_send;                       /* insert sequence number into frame */
  s.ack = 1 – frame_expected;                       /* piggybacked ack */
  to_physical_layer(&s);                            /* transmit the frame */
  start_timer(s.seq);                               /* start the timer running */
```

Continued →

A One-Bit Sliding Window Protocol (ctd.)

```
while (true) {
    wait_for_event(&event);                 /* frame_arrival, cksum_err, or timeout */
    if (event == frame_arrival) {           /* a frame has arrived undamaged. */
        from_physical_layer(&r);            /* go get it */

        if (r.seq == frame_expected) {      /* handle inbound frame stream. */
            to_network_layer(&r.info);      /* pass packet to network layer */
            inc(frame_expected);            /* invert seq number expected next */
        }

        if (r.ack == next_frame_to_send) {  /* handle outbound frame stream. */
            stop_timer(r.ack);              /* turn the timer off */
            from_network_layer(&buffer);    /* fetch new pkt from network layer */
            inc(next_frame_to_send);        /* invert senderís sequence number */
        }
    }
    s.info = buffer;                        /* construct outbound frame */
    s.seq = next_frame_to_send;             /* insert sequence number into it */
    s.ack = 1 – frame_expected;             /* seq number of last received frame */
    to_physical_layer(&s);                  /* transmit a frame */
    start_timer(s.seq);                     /* start the timer running */
  }
}
```

PROBLEM

Consider a 50 kbps satellite channel with a 500 msec roundtrip propagation delay. Use one bit SWP to send 1000 bit frame via satellite. What is blocking percentage?

At t=0 the sender starts sending the first frame

Time for 1000 bits?
1 sec = 50000
1 bit =1/50000
1000 bit=(1/50000)1000=20 msec
Roundtrip=500msec
We will get acknowledgement after 520msec

Sender blocking %= 500/520=96%

PROBLEM

Consider a 50 kbps satellite channel with a 500 msec roundtrip propagation delay. Use one bit SWP to send 1000 bit frame via satellite.

The problem described here can be viewed as a consequence of the rule requiring a sender to wait for an acknowledgement before sending another frame. If we relax that restriction, much better efficiency can be achieved. Basically, the solution lies in allowing the sender to transmit up to ***w frames before blocking, instead* of just 1**. With a large enough choice of *w the sender will be able to continuously* transmit frames since the acknowledgements will arrive for previous frames before the window becomes full, preventing the sender from blocking.

PROBLEM

Consider a 50 kbps satellite channel with a 500 msec roundtrip propagation delay. Use one bit SWP to send 1000 bit frame via satellite.

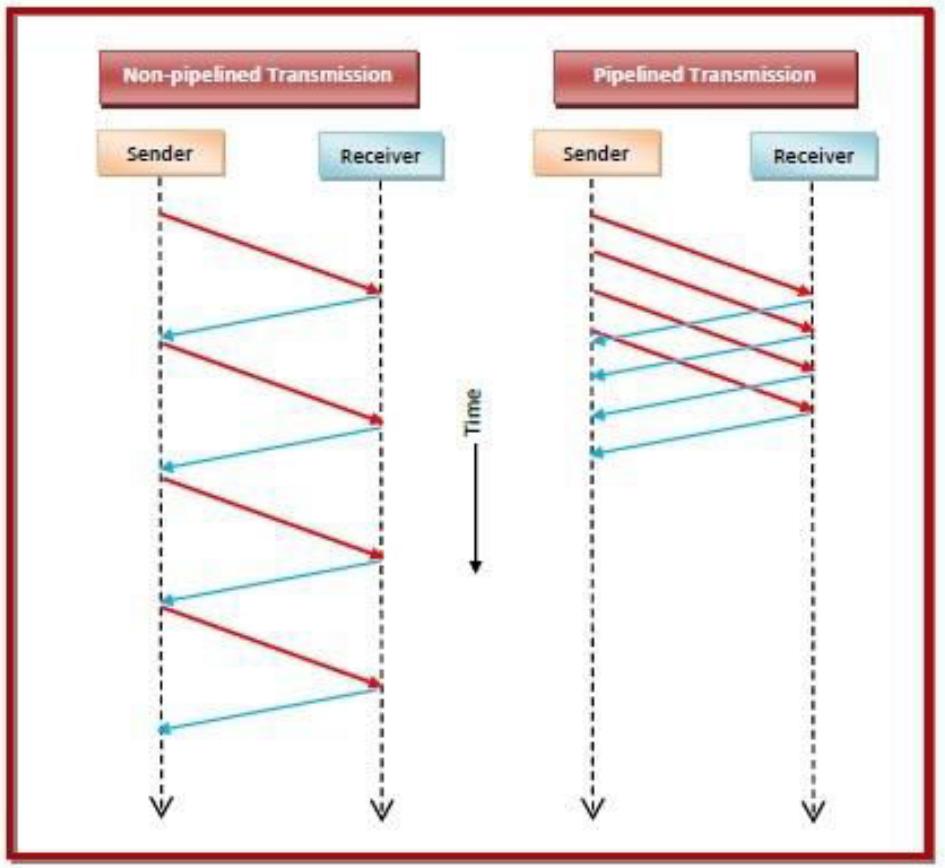
This capacity is determined by the bandwidth in bits/sec multiplied by the one-way transit time, or the **bandwidth-delay product of the link. We can divide this quantity by the** number of bits in a frame to express it as a number of frames. Call this quantity *BD. Then* ***w should be set to 2BD + 1****. Twice the bandwidth-delay is the number* of frames that can be outstanding if the sender continuously sends frames when the round-trip time to receive an acknowledgement is considered. The "+1" is because an acknowledgement frame will not be sent until after a complete frame is received.

2BD + 1 is then 26 frames. Assume the sender begins sending frame 0 as before and sends a new frame every 20 msec. By the time it has finished sending 26 frames, at *t = 520 msec, the acknowledgement for frame 0 will have just arrived.* Thereafter, acknowledgements will arrive every 20 msec, so the sender will always get permission to continue just when it needs it. From then onwards, 25 or 26 unacknowledged frames will always be outstanding. Put in other terms, the **sender's maximum window size is 26**.

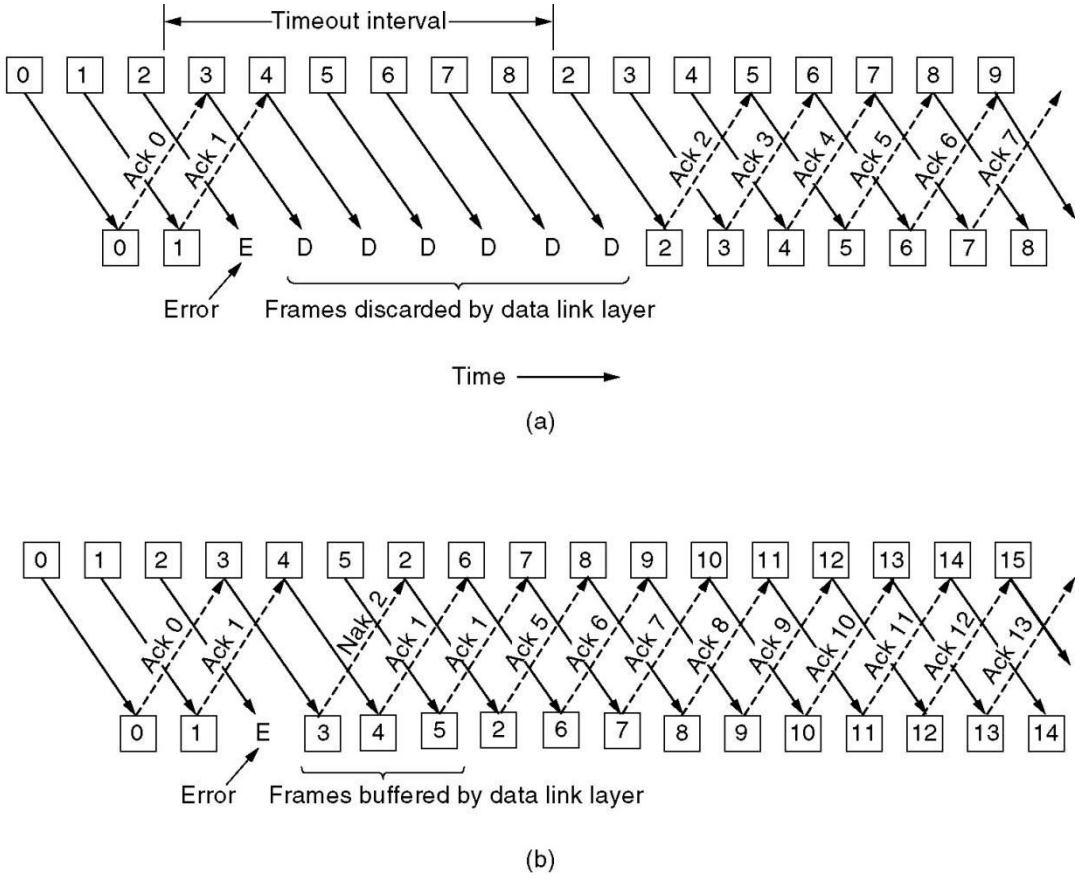
Pipelining

The technique of keeping multiple frames in flight is called pipelining.

In computer networking, pipelining is the method of **sending multiple data units without waiting for an acknowledgment for the first frame sent**. Pipelining ensures better utilization of network resources and also increases the speed of delivery, particularly in situations where a large number of data units make up a message to be sent.



A PROTOCOL -GO BACK N & SELECTIVE REPEAT



Selective repeat is often combined with having the receiver send a negative acknowledgement (NAK) when it detects an error, for example, when it receives a checksum error or a frame out of sequence. NAKs stimulate retransmission before the corresponding timer expires and thus improve performance.

Pipelining and error recovery. Effect on an error when

(a) Receiver's window size is 1.

(b) Receiver's window size is large.

A PROTOCOL -GO BACK N

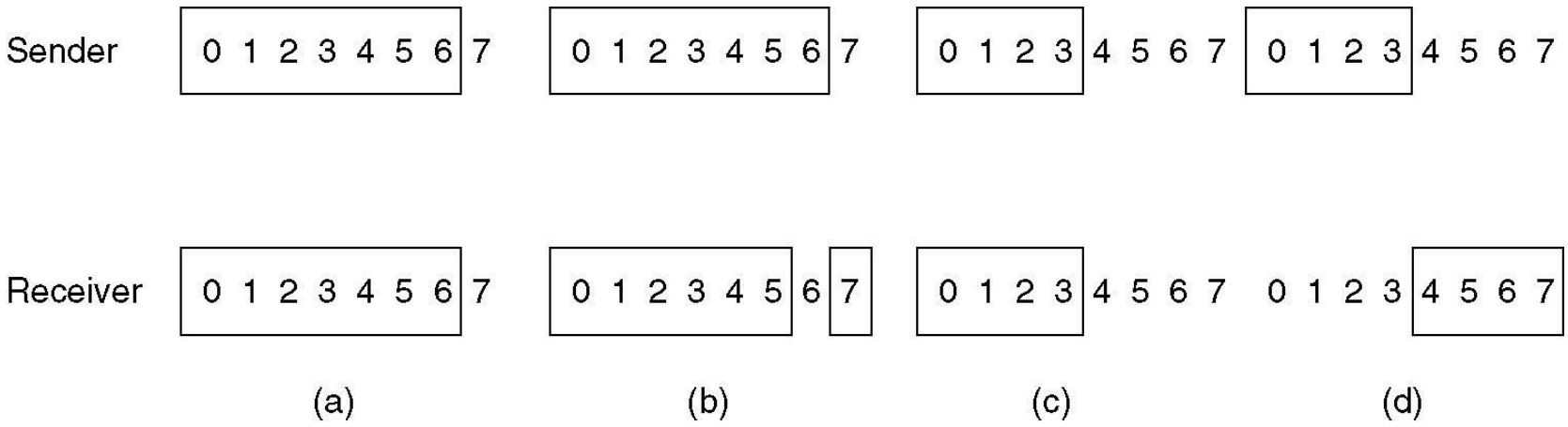
The maximum number of frames that may be outstanding at any instant is not the same as the size of the sequence number space. For go-back-n, *MAX_SEQ* frames may be outstanding at any instant, even though there are *MAX_SEQ* + 1 distinct sequence numbers (which are 0, 1, . . . , *MAX_SEQ*). We will see an even tighter restriction for the next protocol, selective repeat. To see why this restriction is required, consider the following scenario with *MAX_SEQ* = 7:

1. The sender sends frames 0 through 7.
2. A piggybacked acknowledgement for 7 comes back to the sender.
3. The sender sends another eight frames, again with sequence numbers 0 through 7.
4. Now another piggybacked acknowledgement for frame 7 comes in.

A PROTOCOL –Selective Repeat

The way out of this dilemma lies in making sure that after the receiver has advanced its window there is no overlap with the original window. To ensure that there is no overlap, the maximum window size should be at most half the range of the sequence numbers. With 3 bits, the sequence numbers range from 0 to 7. Only four unacknowledged frames should be outstanding at any instant. That way, if the receiver has just accepted frames 0 through 3 and advanced its window to permit acceptance of frames 4 through 7, it can unambiguously tell if subsequent frames are retransmissions (0 through 3) or new ones (4 through 7). In general, the window size for protocol SR will be *(MAX SEQ + 1)/2.*

A Protocol Using Selective Repeat



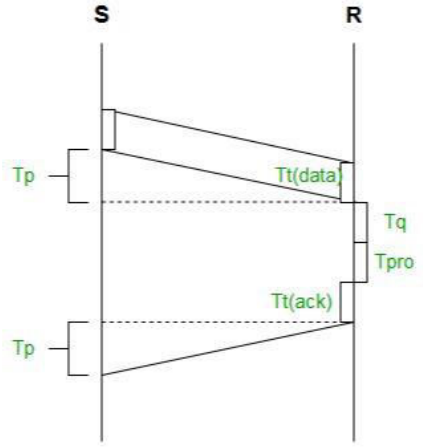
(a) Initial situation with a window size seven.

(b) After seven frames sent and received, but not acknowledged.

(c) Initial situation with a window size of four.

(d) After four frames sent and received, but not acknowledged.

Properties	Stop and Wait	Go Back N	Selective Repeat
Sender window size	1	N	N
Receiver Window size	1	1	N
Minimum Sequence number	2	N+1	2N
Efficiency	1/(1+2*a)	N/(1+2*a)	N/(1+2*a)
Number of retransmissions in case of packet drop	1	N	1

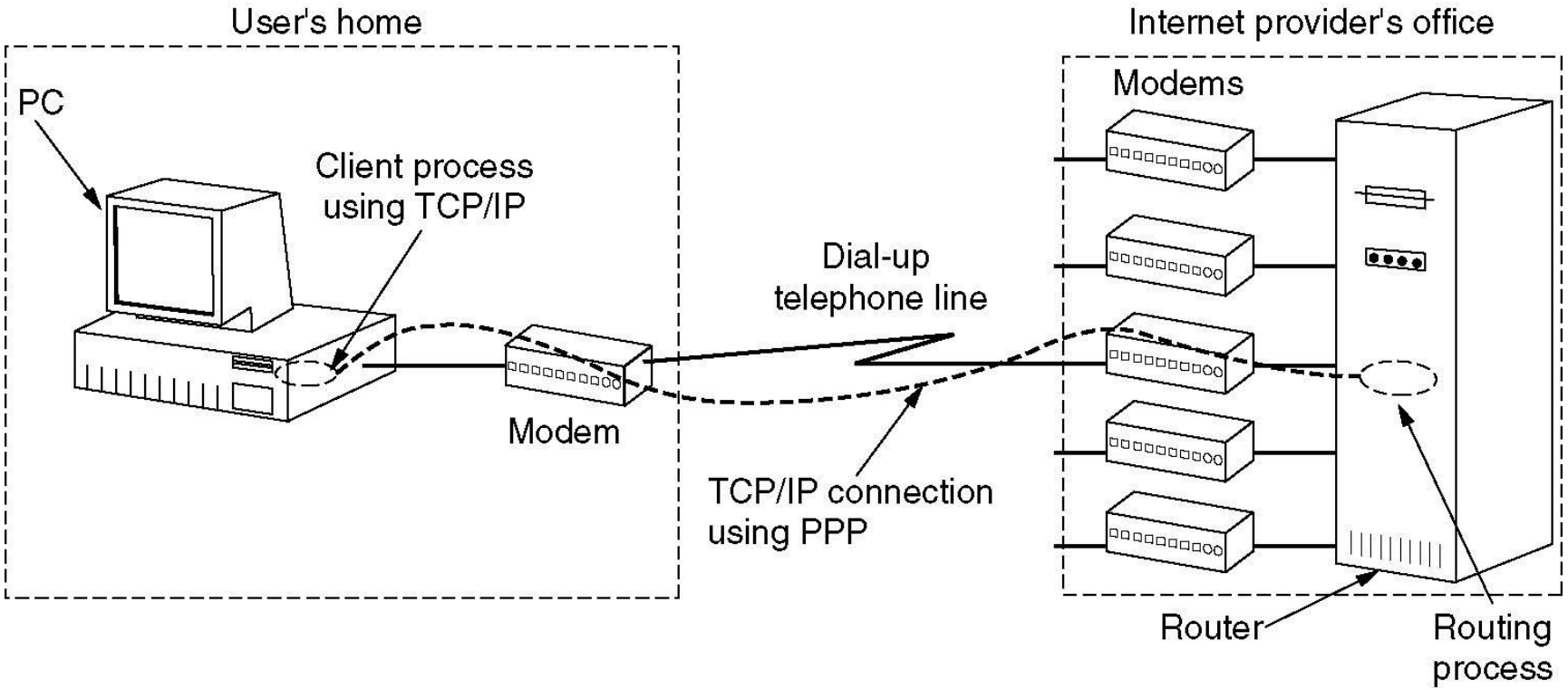


Efficiency= Useful time / Total cycle time.
= T_t / (T_t + 2*T_p) = 1 / (1+2*(T_p/ T_t))
= 1 / (1+2*a) where, a = T_p / T_t

a = Ratio of Propagation delay and Transmission delay

At N=1, Go Back N is effectively reduced to Stop and Wait

The Data Link Layer in the Internet



A home personal computer acting as an internet host.

REFERENCES

1. Andrew S Tanenbaum “Computer Networks” 5/ed. Pearson Education.
2. Behrouz A. Forouzan “Data Communication and Networking” 3/e, TMH.
3. William Stallings “Data and Computer Communications”, 8/e, PHI, 2004.
4. S.Keshav “An Engineering Approach to Computer Networks” 2/e, PE
5. Behrouz A. Forouzan “TCP/IP protocol suite” 4/e, TMH, 2010.
6. Arzoo Kataria “Data Link Layer in Internet”.
7.https://nptel.ac.in/courses/106105080/pdf/M3L4.pdf