Distributed System Structures

Guihai Chen

Department of Computer Science and Engineering

Shanghai Jiao Tong University

Spring 2020

Distributed System Structures

- Motivation
- Types of Network-Based Operating Systems
- Network Structure
- Network Topology
- Communication Structure
- Communication Protocols
- Robustness
- Design Issues
- An Example: Networking

- Red color title means additionally added stuff.

Chapter Objectives

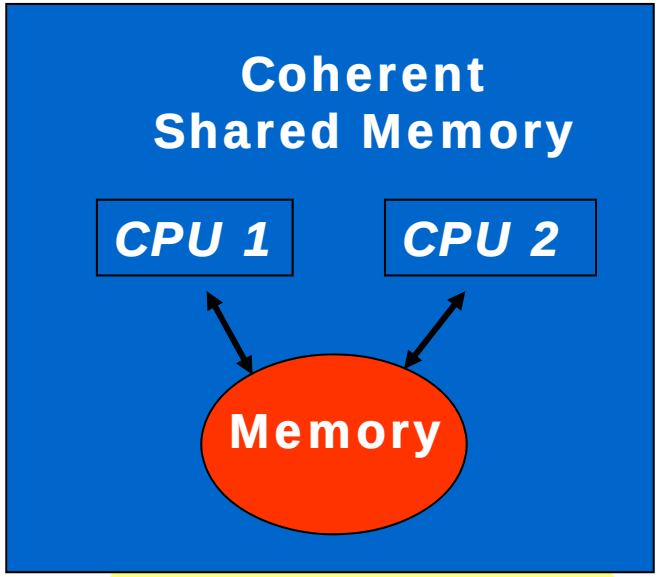
- To provide a high-level overview of distributed systems and the networks that interconnect them
- To discuss the general structure of distributed operating systems

Motivation

- **Distributed system** is collection of loosely coupled processors interconnected by a communications network
- Processors variously called *nodes, computers, machines, hosts*
  - *Site* is location of the processor
- Reasons for distributed systems
  - Resource sharing
    - sharing and printing files at remote sites
    - processing information in a distributed database
    - using remote specialized hardware devices
  - Computation speedup – **load sharing**
  - Reliability – detect and recover from site failure, function transfer, reintegrate failed site
  - Communication – message passing

Shared and Distributed Memory Architectures

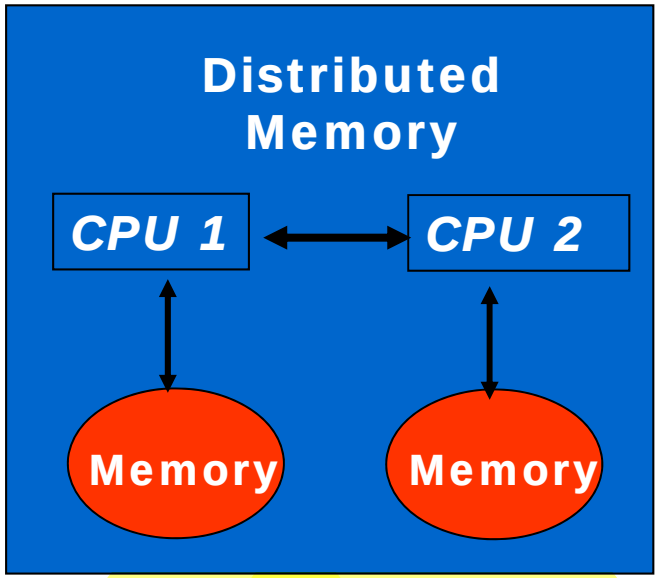
Tightly Coupled Multiprocessor



Easy to Program

Hard to Scale Hardware

Loosely Coupled Multicomputer



Hard to Program

Easy to Scale Hardware

Basic Performance Metrics

- **Workload** (W) = number of ops. required to complete the program
- T_P: execution time using P processors
- T_1: execution time using 1 processor
- **Speed** = W/T_P
- **Speedup (S)** = T_1 / T_P
- **Efficiency** (using P processors) = S / P

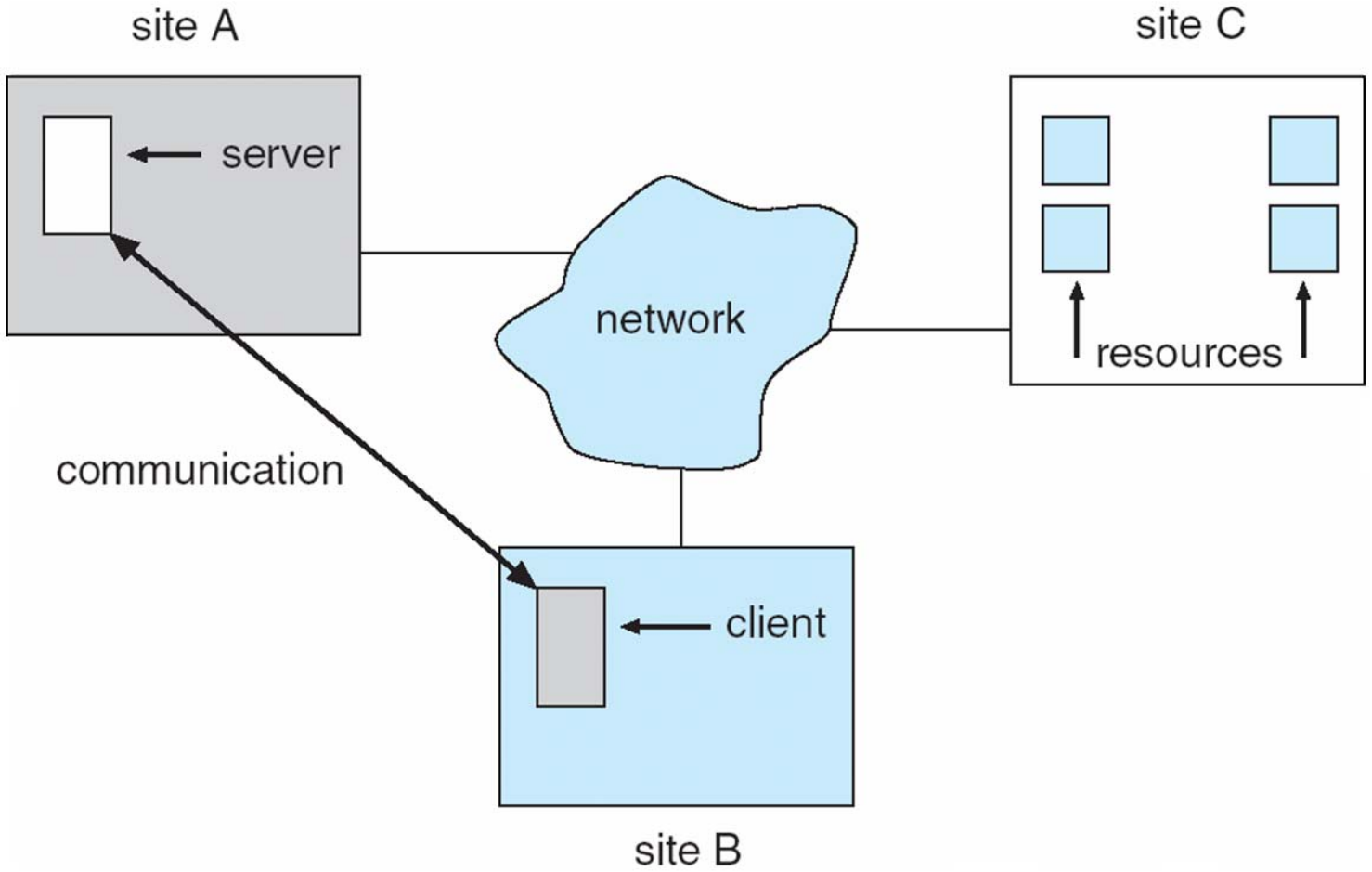
- **Goal:** use metrics which reflect performance delivered to real user programs, real applications.

Speedup

- General concept in parallel computing
  - How much faster an application runs on parallel computer ?
  - What benefits derive from the use of parallelism?

- General agreement :
  - **speedup = serial time/parallel time**

A Distributed System



Types of Distributed Operating Systems

- Network Operating Systems
- Distributed Operating Systems
- Difference?

Network-Operating Systems

- Users are aware of multiplicity of machines. Access to resources of various machines is done explicitly by:
  - Remote logging into the appropriate remote machine (telnet, ssh)
  - Remote Desktop (Microsoft Windows)
  - Transferring data from remote machines to local machines, via the File Transfer Protocol (FTP) mechanism

Distributed-Operating Systems

- Users not aware of multiplicity of machines
  - Access to remote resources similar to access to local resources
- Data Migration – transfer data by transferring entire file, or transferring only those portions of the file necessary for the immediate task
- Computation Migration – transfer the computation, rather than the data, across the system

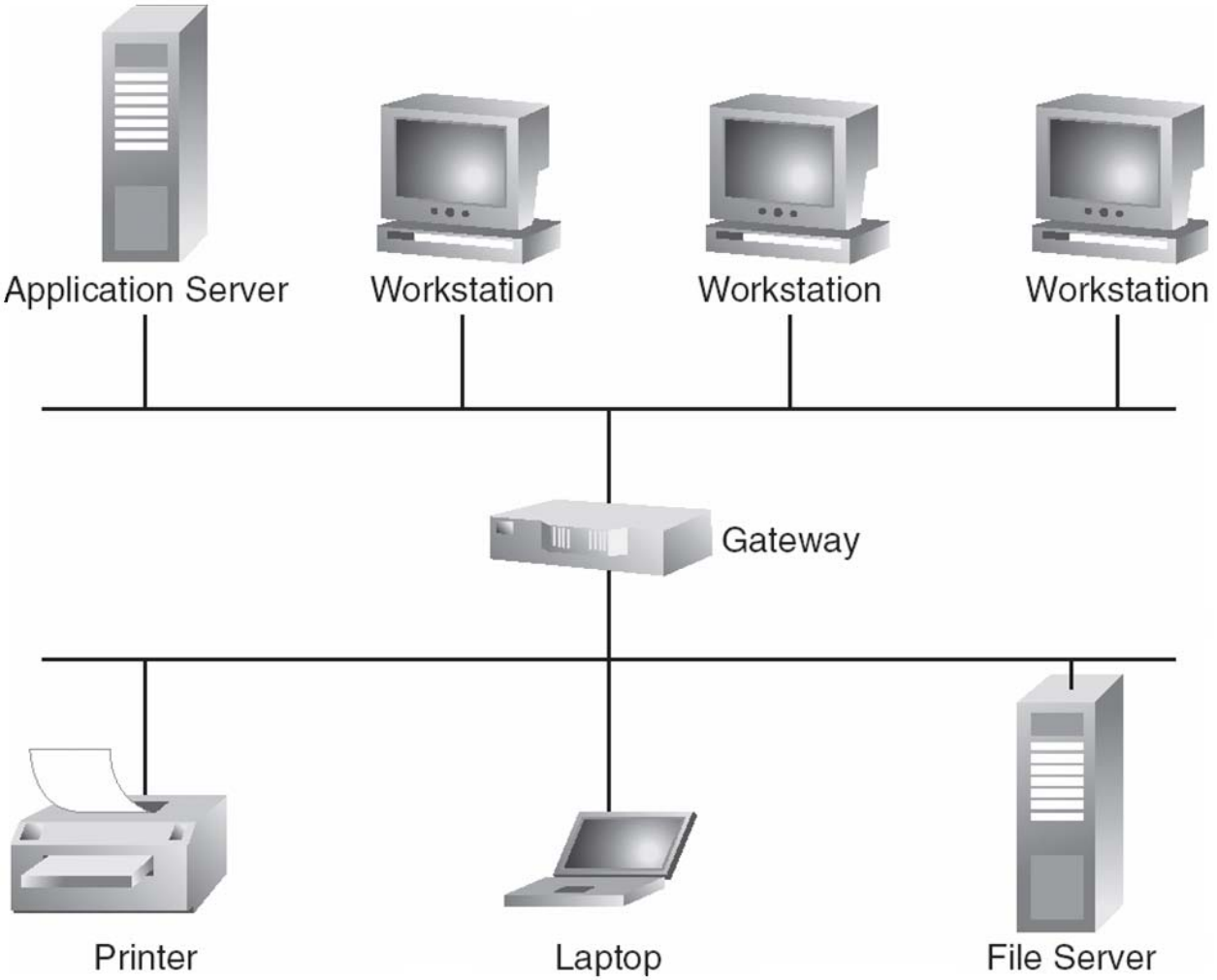
Distributed-Operating Systems (Cont.)

- Process Migration – execute an entire process, or parts of it, at different sites
  - **Load balancing** – distribute processes across network to even the workload
  - **Computation speedup** – subprocesses can run concurrently on different sites
  - **Hardware preference** – process execution may require specialized processor
  - **Software preference** – required software may be available at only a particular site
  - **Data access** – run process remotely, rather than transfer all data locally

Network Structure

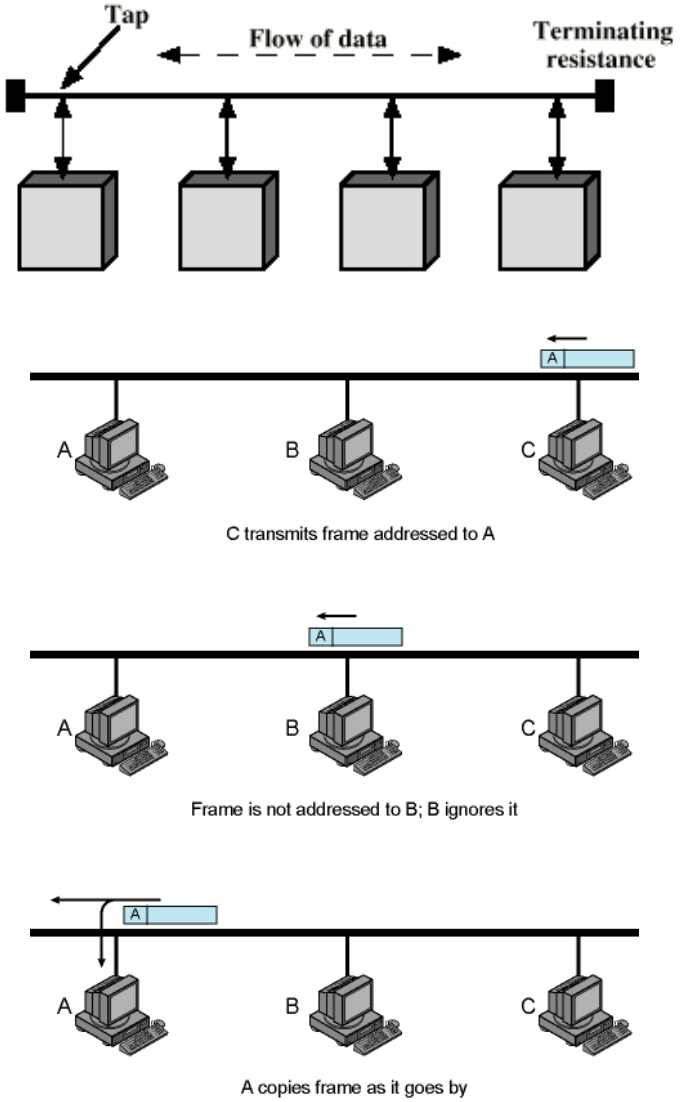
- **Local-Area Network** (**LAN**) – designed to cover small geographical area.
  - Multiaccess bus, ring, or star network
  - Speed ≈ 10 – 100 megabits/second or even Gigabits/second
  - Broadcast is fast and cheap
  - Nodes:
    - usually workstations and/or personal computers
    - a few (usually one or two) mainframes

Depiction of Typical LAN



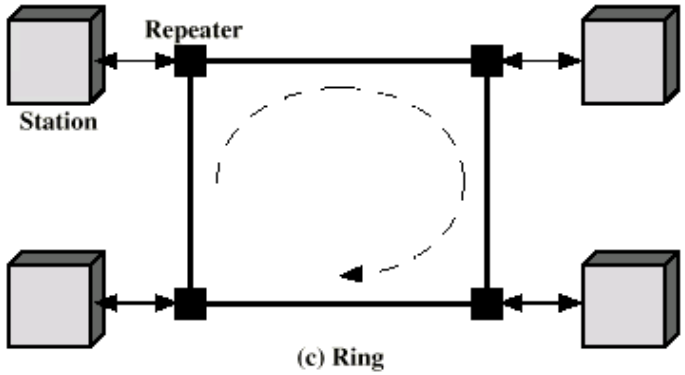
Bus Topology

- Stations attach to linear medium (bus)
  - Via a tap - allows for transmission and reception
- Transmission propagates in medium in both directions
- Received by all other stations
  - Not addressed stations ignore
- Need to identify target station
  - Each station has unique address
  - Destination address included in frame header
- Terminator absorbs frames at the end of medium
- Need to regulate transmission
  - To avoid collisions
    - If two stations attempt to transmit at same time, signals will overlap and become garbage
  - To avoid continuous transmission from a single station.
    - If one station transmits continuously, access is blocked for others
    - Solution: Transmit Data in small blocks–frames

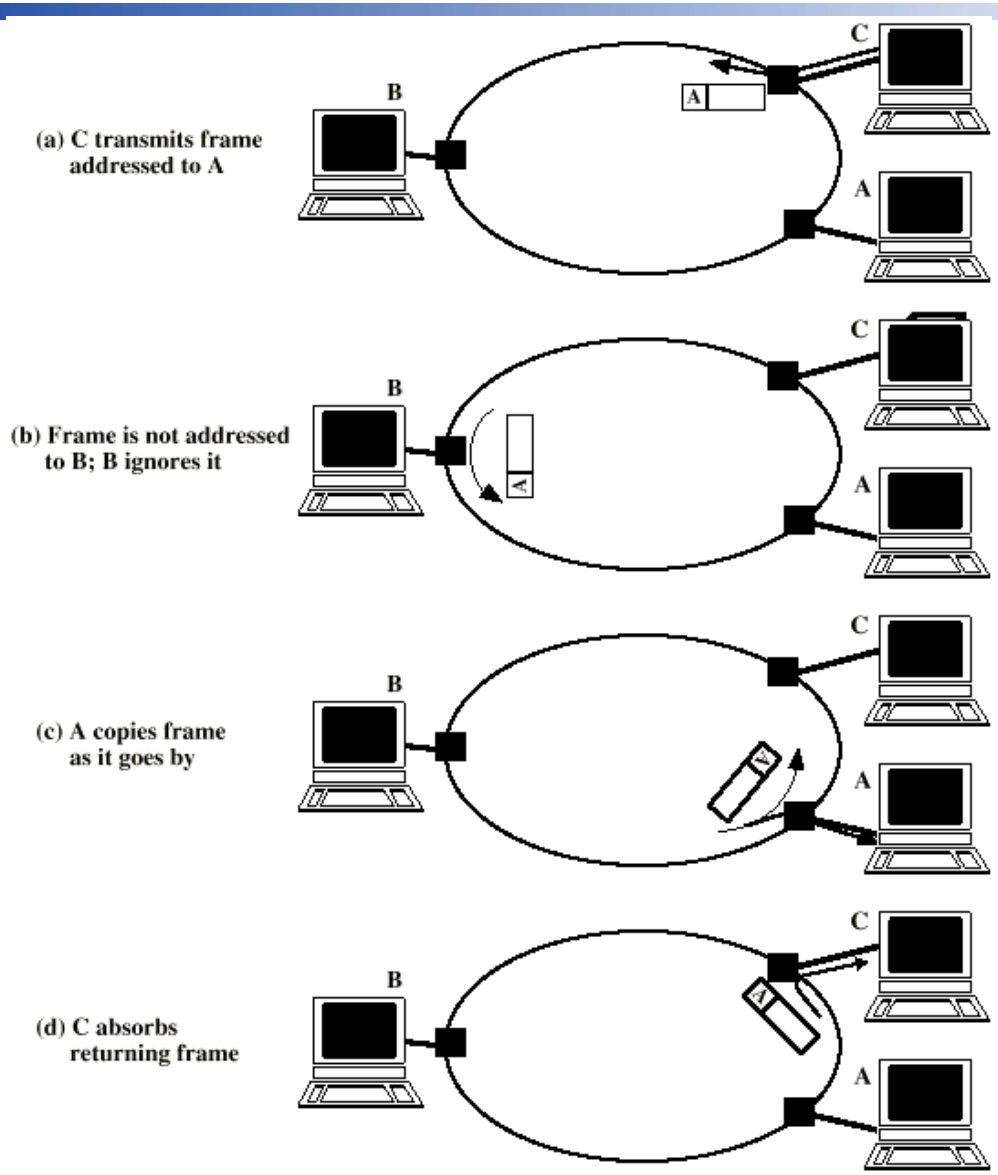


Ring Topology

- Repeaters joined by point-to-point links in closed loop
  - Links are unidirectional
  - Receive data on one link and retransmit on another
  - Stations attach to repeaters
- Data transmitted in frames
  - Frame passes all stations in a circular manner
  - Destination recognizes address and copies frame
  - Frame circulates back to source where it is removed
- Medium access control is needed to determine when station can insert frame, see next page.

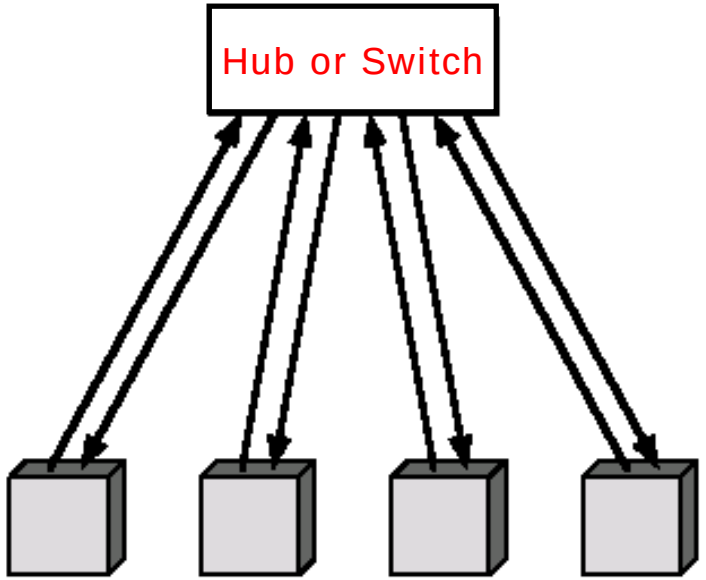


Frame Transmission Ring LAN



Star Topology

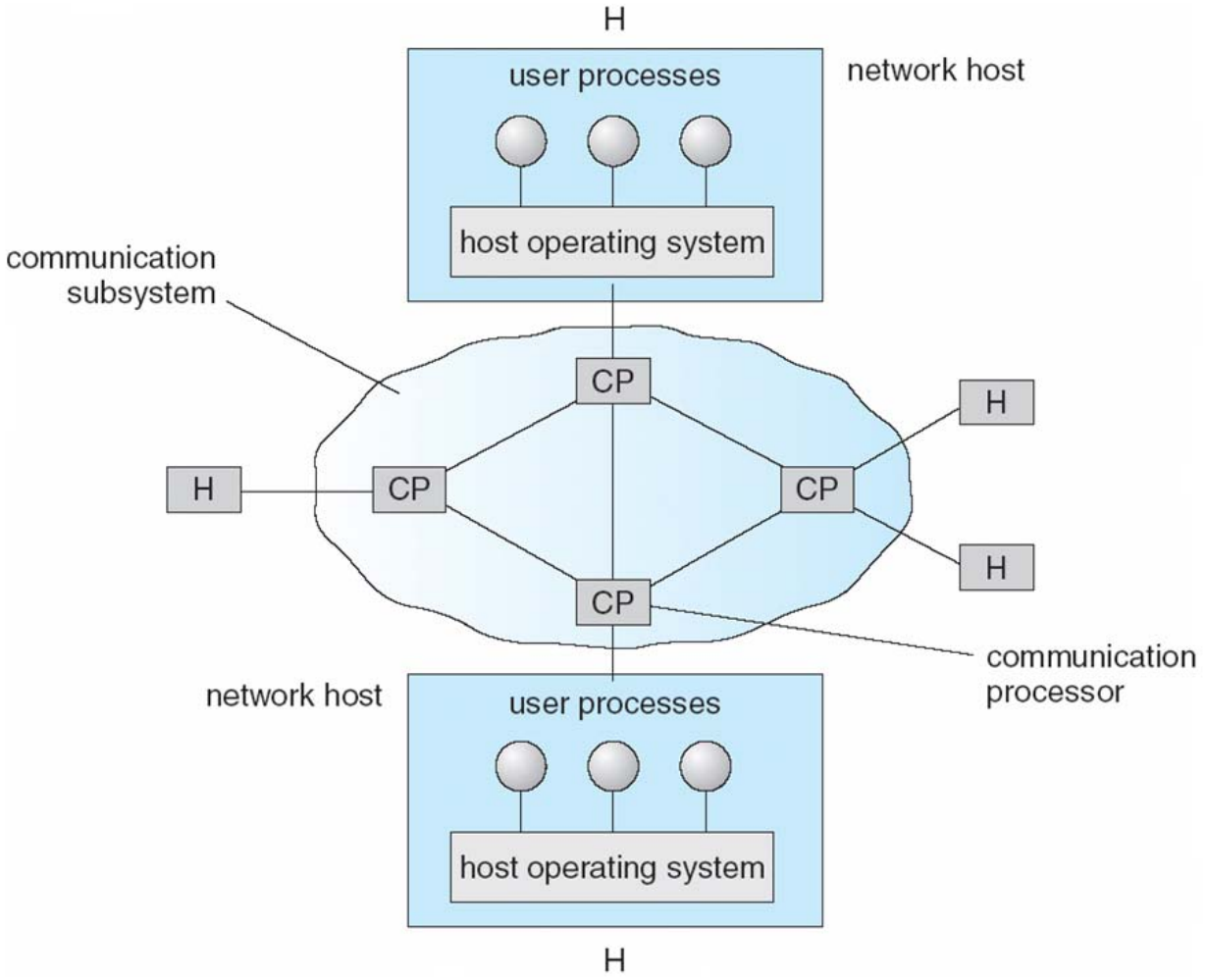
- Each station connected directly to central node
  - using a full-duplex (bi-directional) link
- Central node can broadcast (hub)
  - Physical *star*, but logically like *bus* since broadcast
  - Only one station can transmit at a time; otherwise, collision occurs
- Central node can act as frame switch
  - retransmits only to destination
  - today's technology



Network Types (Cont.)

- **Wide-Area Network** (**WAN**) – links geographically separated sites
  - Point-to-point connections over long-haul lines (often leased from a phone company)
  - Speed ≈ 1.544 – 45 megbits/second
  - Broadcast usually requires multiple messages
  - Nodes:
    - usually a high percentage of mainframes

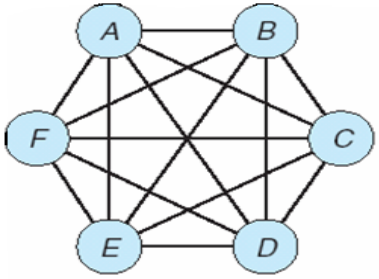
Communication Processors in a Wide-Area Network



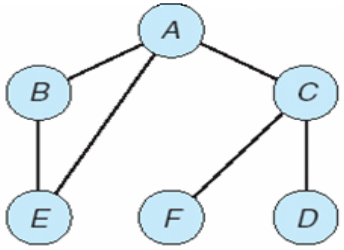
Network Topology

- Sites in the system can be physically connected in a variety of ways; they are compared with respect to the following criteria:
  - **Installation cost** - How expensive is it to link the various sites in the system?
  - **Communication cost** - How long does it take to send a message from site *A* to site *B*?
  - **Reliability** - If a link or a site in the system fails, can the remaining sites still communicate with each other?
- The various topologies are depicted as graphs whose nodes correspond to sites
  - An edge from node *A* to node *B* corresponds to a direct connection between the two sites
- The following six items depict various network topologies

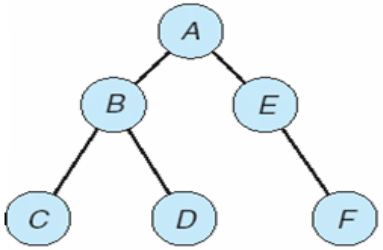
Network Topology



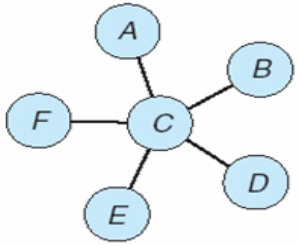
fully connected network



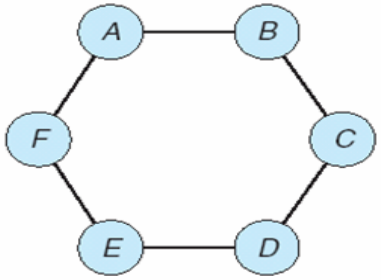
partially connected network



tree-structured network



star network



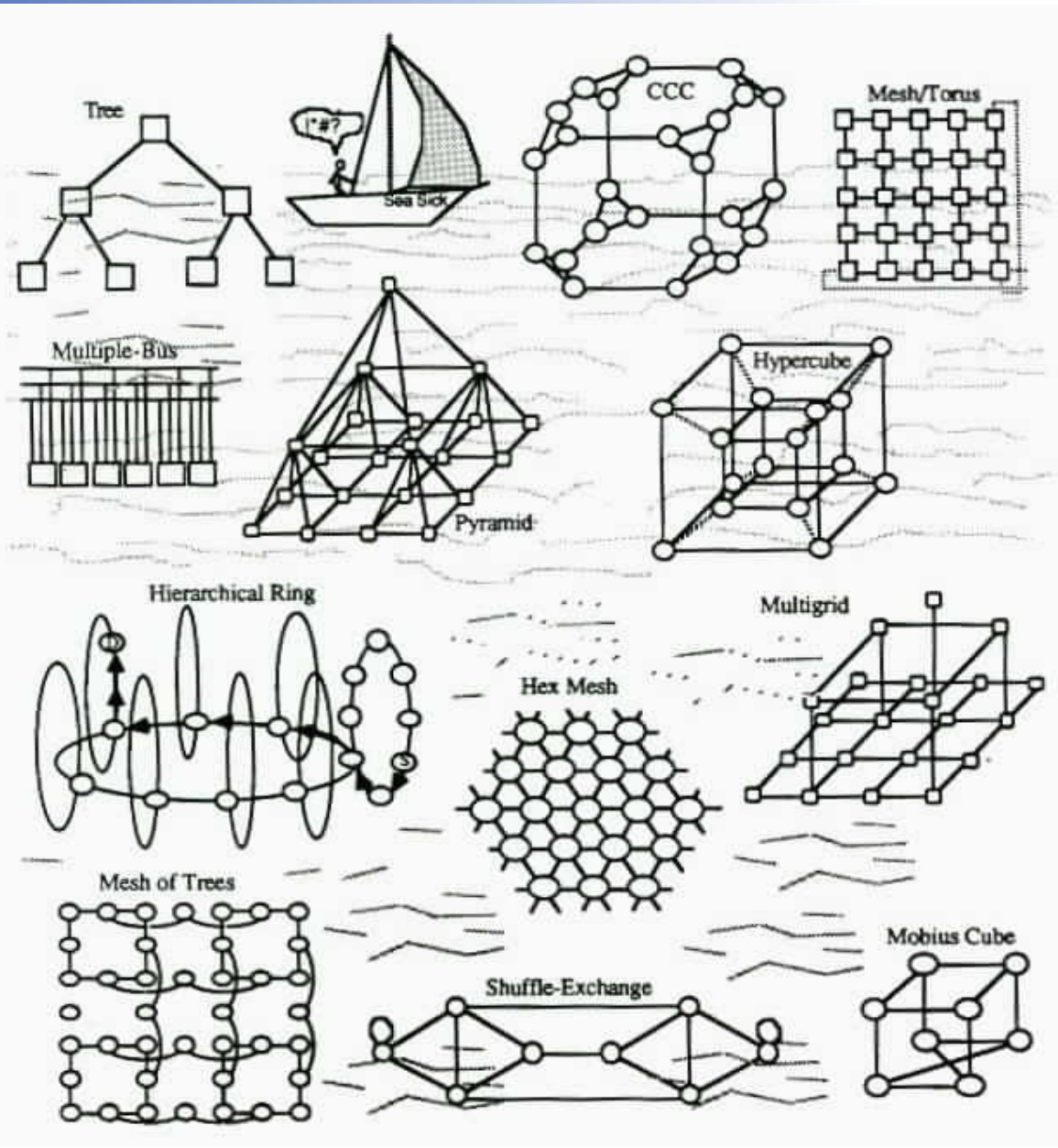
ring network

Network Topology (cont.)

A wide variety of direct interconnection networks have been proposed for, or used in, parallel computers

They differ in topological performance, robustness, and reliability attributes.

Seas of Networks: What is your choice? We will have a separate lecture soon.



Communication Structure

The design of a *communication* network must address four basic issues:

- **Naming and name resolution** - How do two processes locate each other to communicate?
- **Routing strategies** - How are messages sent through the network?
- **Connection strategies** - How do two processes send a sequence of messages?
- **Contention -** The network is a shared resource, so how do we resolve conflicting demands for its use?

Naming and Name Resolution

- Name systems in the network
- Address messages with the process-id
- Identify processes on remote systems by

  <host-name, identifier> pair
- **Domain name service** (**DNS**) – specifies the naming structure of the hosts, as well as name to address resolution (Internet)

Naming and Name Resolution (Cont.)

- Different layer, different naming in TCP/IP
- Application layer: www.google.com.hk; www.sjtu.edu.cn
- Transport layer: <IP, port number> pair to locate a process
- Network layer: IP to locate a host
- Data layer: Mac address

Routing Strategies

- **Fixed routing** - A path from *A* to *B* is specified in advance; path changes only if a hardware failure disables it
  - Since the shortest path is usually chosen, communication costs are minimized
  - Fixed routing cannot adapt to load changes
  - Ensures that messages will be delivered in the order in which they were sent

- **Virtual circuit** - A path from *A* to *B* is fixed for the duration of one session. Different sessions involving messages from *A* to *B* may have different paths
  - Partial remedy to adapting to load changes
  - Ensures that messages will be delivered in the order in which they were sent

Routing Strategies (Cont.)

- **Dynamic routing** - The path used to send a message form site *A* to site *B* is chosen only when a message is sent
  - Usually a site sends a message to another site on the link least used at that particular time
  - Adapts to load changes by avoiding routing messages on heavily used path
  - Messages may arrive out of order
    - This problem can be remedied by appending a sequence number to each message

Connection Strategies

- **Circuit switching** - A permanent physical link is established for the duration of the communication (i.e., telephone system)
- **Message switching** - A temporary link is established for the duration of one message transfer (i.e., post-office mailing system)
- **Packet switching** - Messages of variable length are divided into fixed-length packets which are sent to the destination
  - Each packet may take a different path through the network
  - The packets must be reassembled into messages as they arrive
- Circuit switching requires setup time, but incurs less overhead for shipping each message, and may waste network bandwidth
  - Message and packet switching require less setup time, but incur more overhead per message

Contention

Several sites may want to transmit information over a link simultaneously. Techniques to avoid repeated collisions include:

- **CSMA/CD** - Carrier sense with multiple access (CSMA); collision detection (CD)
  - A site determines whether another message is currently being transmitted over that link. If two or more sites begin transmitting at exactly the same time, then they will register a CD and will stop transmitting
  - When the system is very busy, many collisions may occur, and thus performance may be degraded

- CSMA/CD is used successfully in the Ethernet system, the most common network system

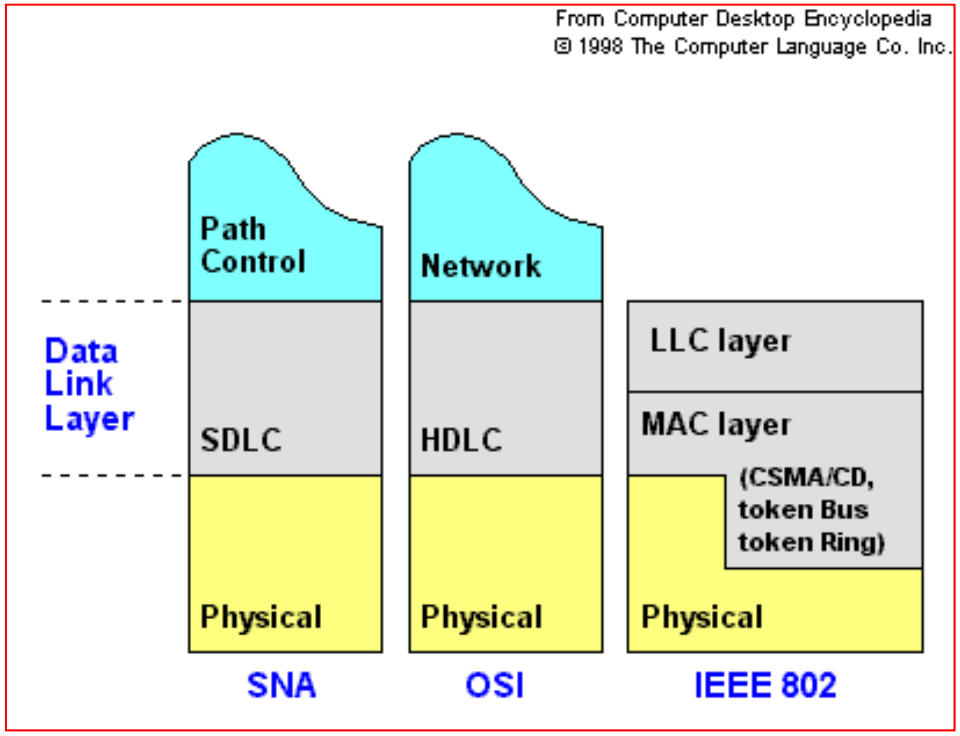
Contention (Cont.)

- **Token passing** - A unique message type, known as a token, continuously circulates in the system (usually a ring structure)
  - A site that wants to transmit information must wait until the token arrives
  - When the site completes its round of message passing, it retransmits the token
  - A token-passing scheme is used by some IBM and HP/Apollo systems

- **Message slots** - A number of fixed-length message slots continuously circulate in the system (usually a ring structure)
  - Since a slot can contain only fixed-sized messages, a single logical message may have to be broken down into a number of smaller packets, each of which is sent in a separate slot
  - This scheme has been adopted in the experimental Cambridge Digital Communication Ring

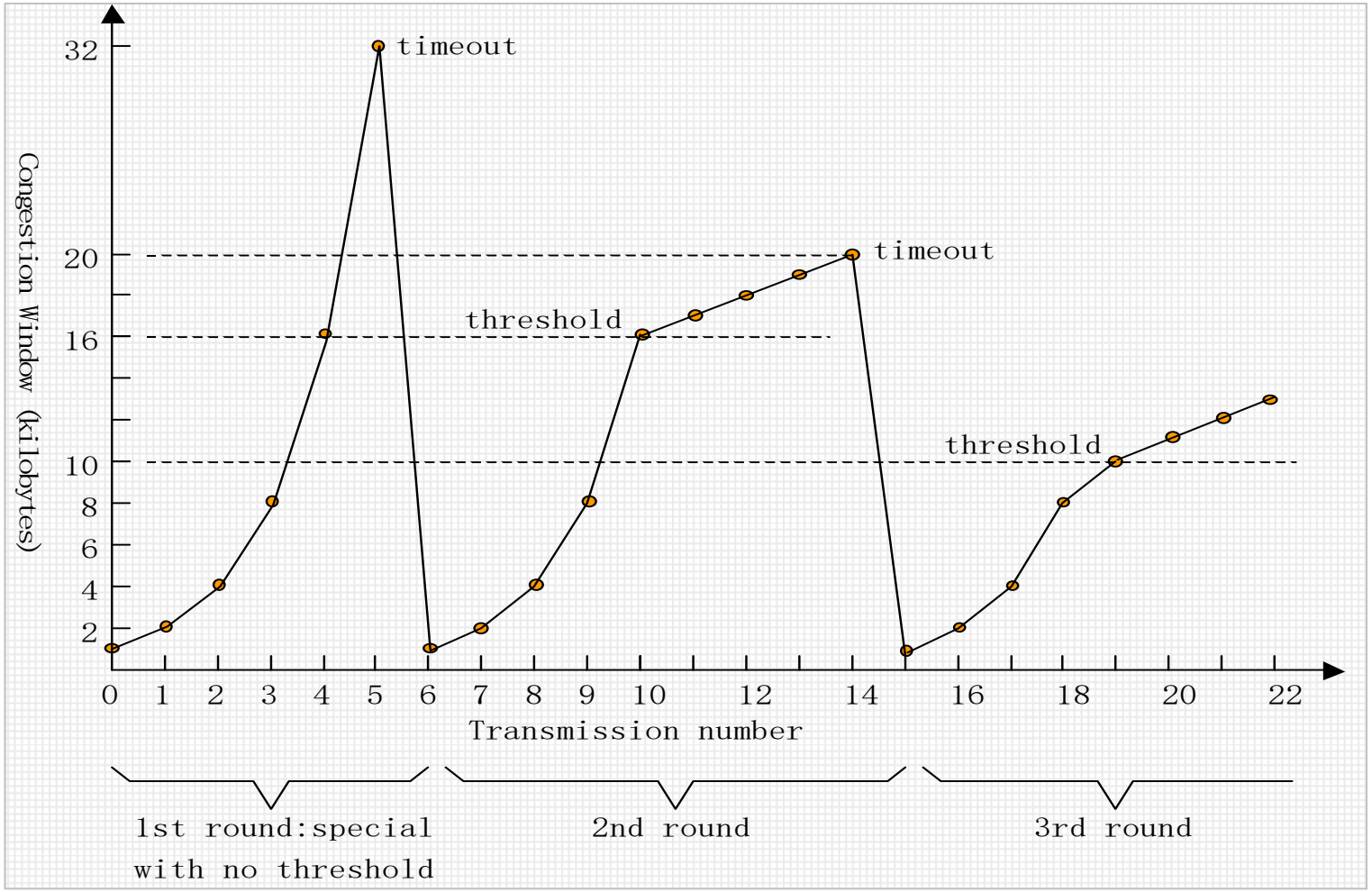
Contention (Cont.)

- Contention occurs at every layer.
- We have only discussed contention at Mac Layer or data link layer. That is Contention on Media Access. That is why we need MAC!
- Other Contentions Example: TCP Congestion Control[Jacobson88]



Contention (Cont.)

- TCP Congestion Control[Jacobson88]



Communication Protocol

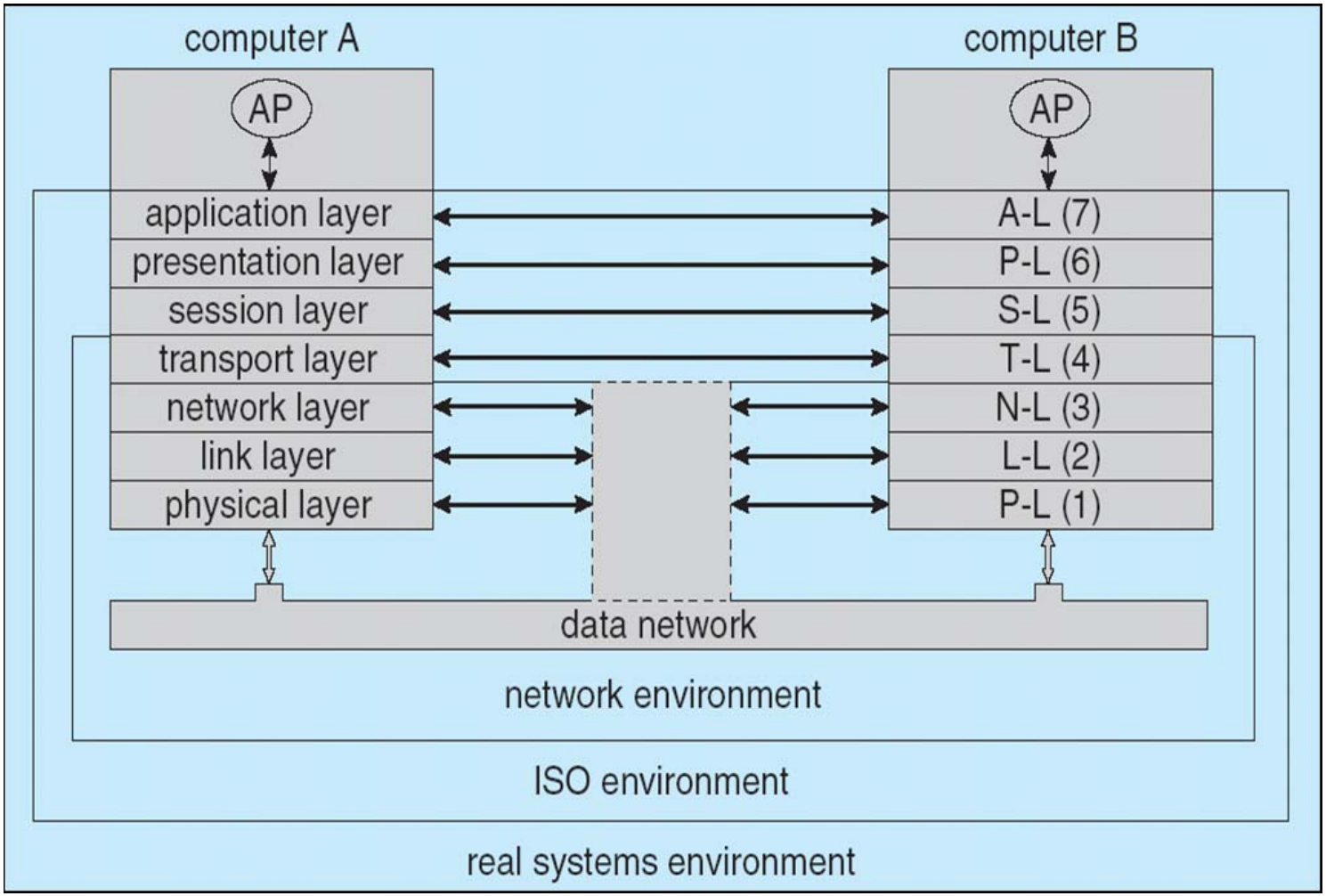
The communication network is partitioned into the following multiple layers:

- **Physical layer** – handles the mechanical and electrical details of the physical transmission of a bit stream
- **Data-link layer** – handles the *frames*, or fixed-length parts of packets, including any error detection and recovery that occurred in the physical layer
- **Network layer** – provides connections and routes packets in the communication network, including handling the address of outgoing packets, decoding the address of incoming packets, and maintaining routing information for proper response to changing load levels

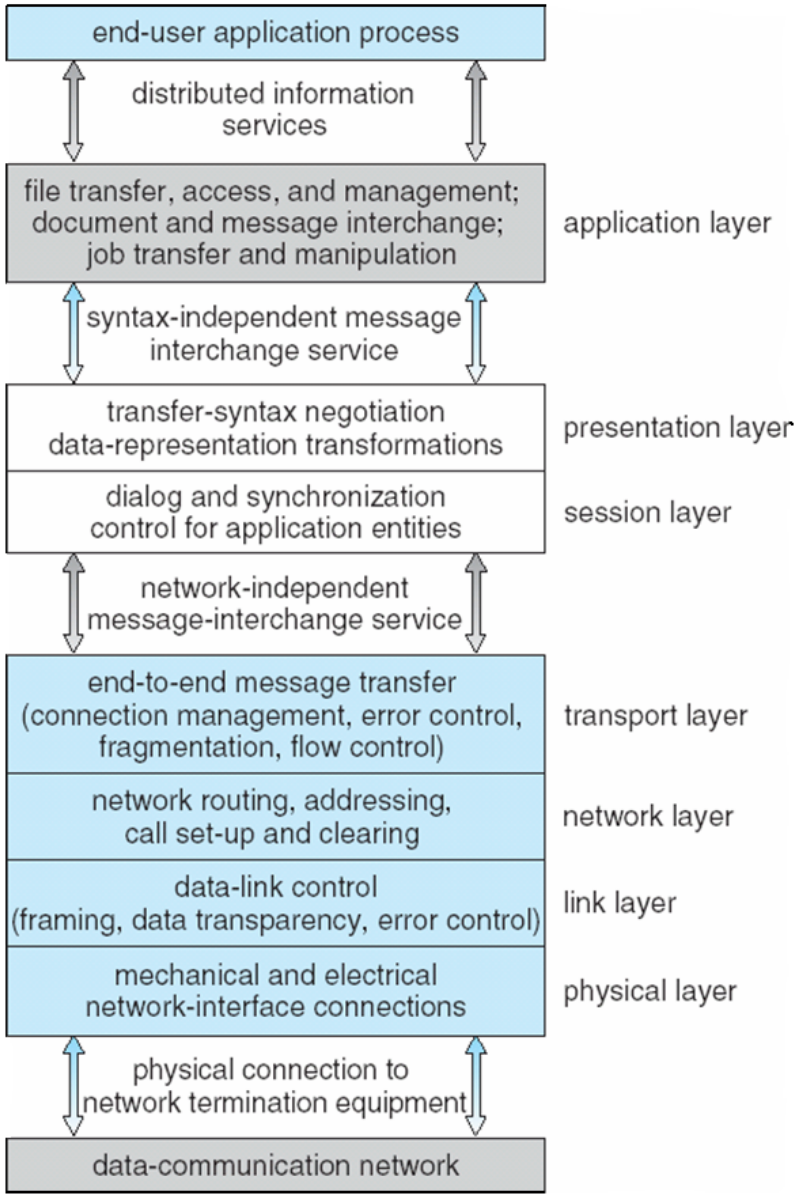
Communication Protocol (Cont.)

- **Transport layer** – responsible for low-level network access and for message transfer between clients, including partitioning messages into packets, maintaining packet order, controlling flow, and generating physical addresses
- **Session layer** – implements sessions, or process-to-process communications protocols
- **Presentation layer** – resolves the differences in formats among the various sites in the network, including character conversions, and half duplex/full duplex (echoing)
- **Application layer** – interacts directly with the users' deals with file transfer, remote-login protocols and electronic mail, as well as schemas for distributed databases

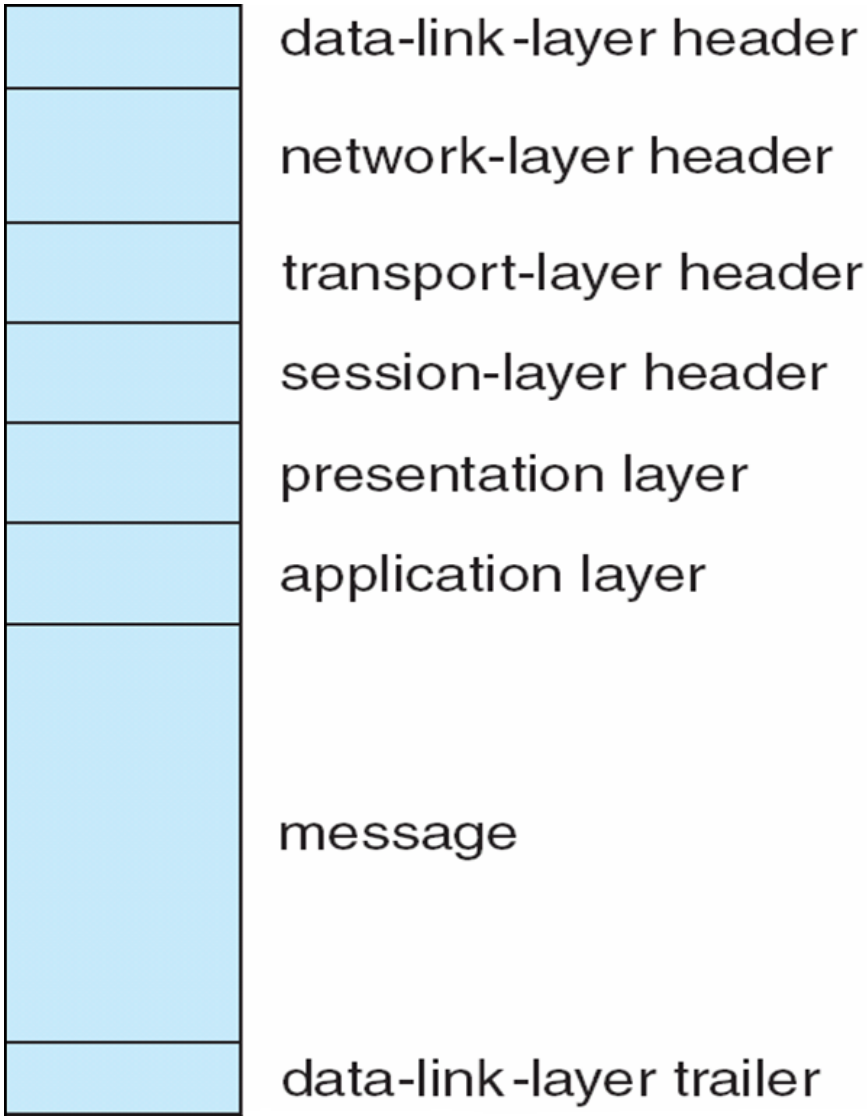
Communication Via ISO Network Model



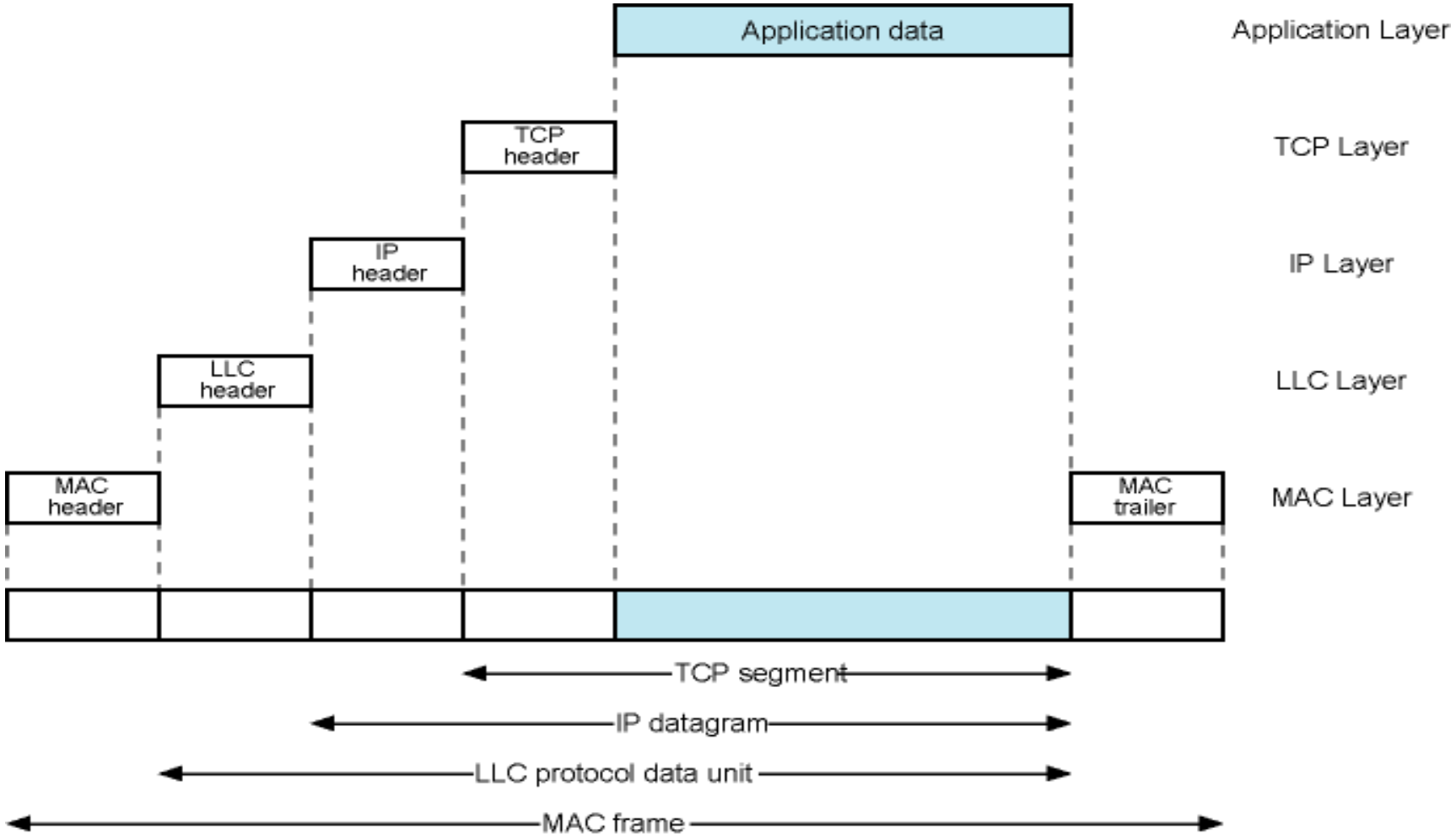
The ISO Protocol Layer



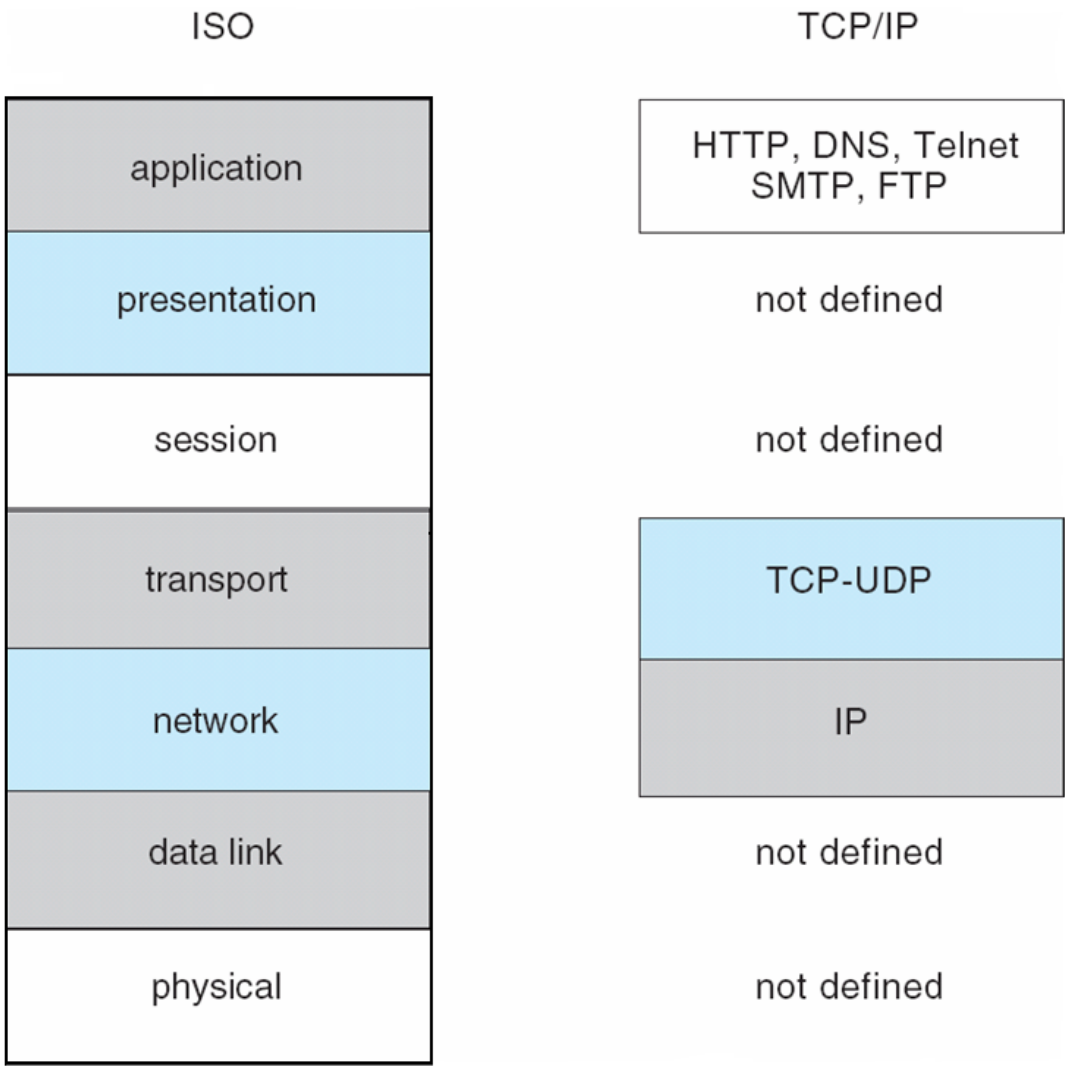
The ISO Network Message



Header, Header and Header



The TCP/IP Protocol Layers



Robustness

- Failure detection
- Reconfiguration

Failure Detection

- Detecting hardware failure is difficult
- To detect a link failure, a handshaking protocol can be used
- Assume Site A and Site B have established a link
  - At fixed intervals, each site will exchange an *I-am-up* message indicating that they are up and running
- If Site A does not receive a message within the fixed interval, it assumes either (a) the other site is not up or (b) the message was lost
- Site A can now send an *Are-you-up?* message to Site B
- If Site A does not receive a reply, it can repeat the message or try an alternate route to Site B

Failure Detection (Cont.)

- If Site A does not ultimately receive a reply from Site B, it concludes some type of failure has occurred

- Types of failures:
  - Site B is down
  - The direct link between A and B is down
  - The alternate link from A to B is down
  - The message has been lost

- However, Site A cannot determine exactly **why** the failure has occurred

Reconfiguration

- When Site A determines a failure has occurred, it must reconfigure the system:

  1. If the link from A to B has failed, this must be broadcast to every site in the system

  2. If a site has failed, every other site must also be notified indicating that the services offered by the failed site are no longer available

- When the link or the site becomes available again, this information must again be broadcast to all other sites

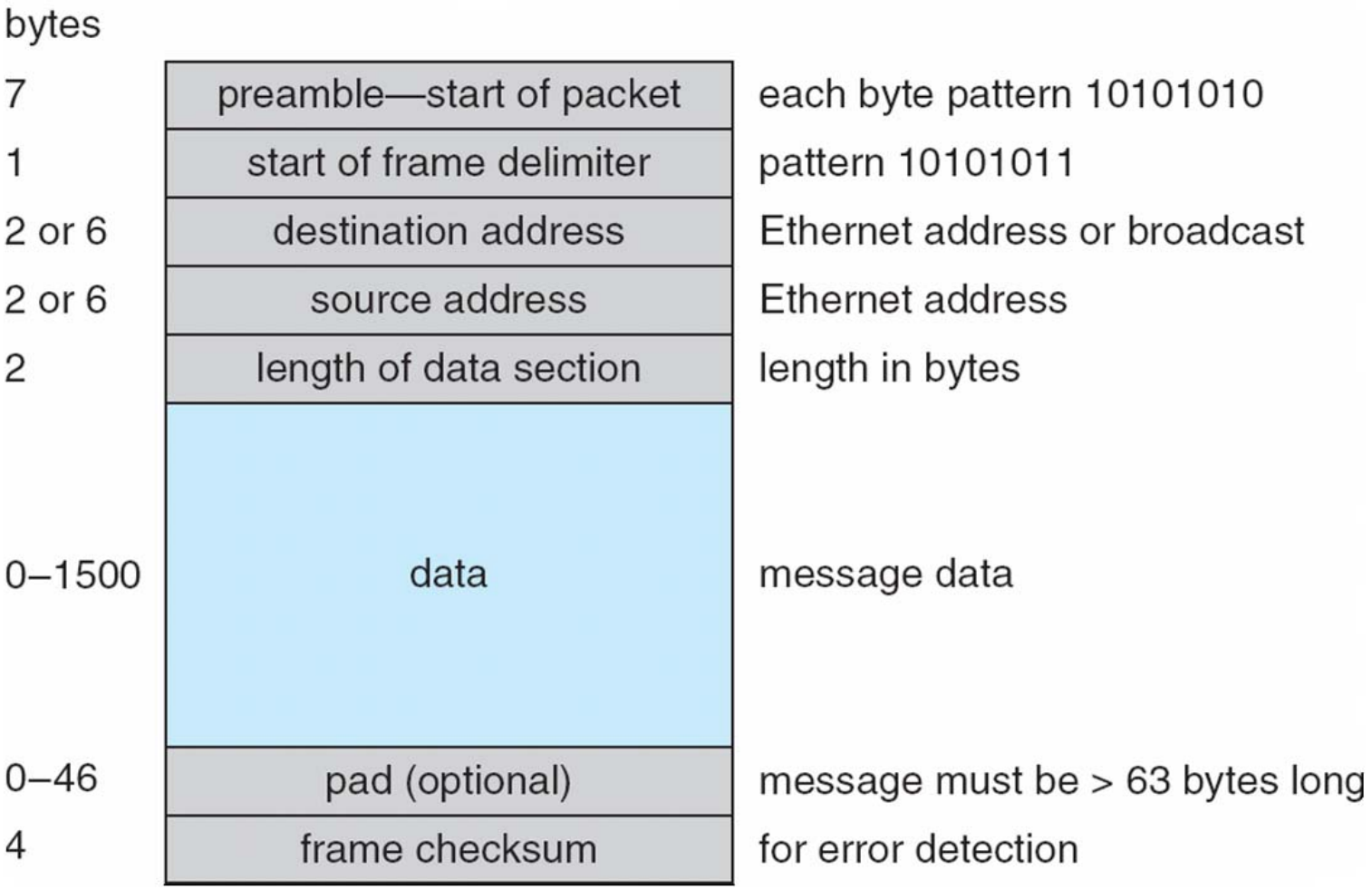
Designing Issues

- **Transparency** – the distributed system should appear as a conventional, centralized system to the user
- **Fault tolerance** – the distributed system should continue to function in the face of failure
- **Scalability** – as demands increase, the system should easily accept the addition of new resources to accommodate the increased demand
- **Clusters** – a collection of semi-autonomous machines that acts as a single system

Example: Networking

- The transmission of a network packet between hosts on an Ethernet network
- Every host has a unique IP address and a corresponding Ethernet (MAC) address
- Communication requires both addresses
- Domain Name Service (DNS) can be used to acquire IP addresses
- Address Resolution Protocol (ARP) is used to map MAC addresses to IP addresses
- If the hosts are on the same network, ARP can be used
  - If the hosts are on different networks, the sending host will send the packet to a *router* which routes the packet to the destination network

An Ethernet Packet



Homework

- Reading
  - Chapter 19