

# EEC-682/782 Computer Networks I

## Lecture 1

Wenbing Zhao

w.zhao1@csuohio.edu

(Lecture notes are based on materials supplied by  
Dr. Louise Moser at UCSB and Prentice-Hall)



## Outline

- # Syllabus
- # Introduction to computer networks
  - Definition
  - Uses of computer networks
  - Network hardware
  - Network software

## Course Objectives

- # Become familiar with the basics of computer networks
  - What is a computer network
  - What are the network architectures
  - What are the fundamental protocols
- # Get some hands-on experiences
  - Learn some basic network computing techniques

## Outline of Lectures

- # Introduction to computer networks
- # Physical layer
- # Data link layer
- # Medium access control sublayer
- # Network layer
- # Transport layer
- # Application layer
- # Network security

## Outline of assignments

- # **Assignment 1: *Sockets for inter-process communication***
  - Learn some basic network programming skills
- # **Assignment 2: *Set up your Web homepage***
- # **Optional project**
  - Must be implementation related, potential projects
    - Implementation of a group communication system
    - Implementation of a reliable point-to-point transport service that resembles TCP in the user space using UDP
    - Build a switch using a PC and a few network interface cards
  - Can substitute for a midterm/final exam, or enhance your grade

## Grading Policy

- # **Grade components & relative weights:**
  - Class participants: 5%
  - Assignments: 20%
    - No group is allowed, must do the assignments independently
      - ⊗ Late turn-in accepted, but with penalties
  - 1st midterm exams: 25% (chapters 1-4)
  - 2nd midterm exams: 20% (chapters 5-6)
  - Final exam: 30% (chapters 1-8)
  - Exams are all closed-book closed-notes
  - **Optional project:** can substitute one exam, or 25% extra
    - Individual project
    - A demo, detailed report, and source code required
- # **Do not cheat!**
  - Do not copy other student's homework, exams or project
  - Do not copy someone else's work found on the Internet
    - Including project implementation and report
    - You can quote a sentence or two, but put those in quote and give reference
    - You can build your project or homework on top of open source libraries, but again, you need to explicitly give acknowledgement and state clearly which parts are implemented by you

## Reference Texts

- # **Andrew S. Tanenbaum :**
  - "Computer Networks"
    - 4th Edition, Prentice-Hall, 2003
- # **Richard Stevens:**
  - "Unix Network Programming: Volume 1 Sockets and XTI"
    - 2nd Edition, 1998
- # **Andrew S. Tanenbaum and Marten van Steen:**
  - "Distributed Systems: Principles and Paradigms"
    - Prentice-Hall, 2002

## Misc

- # **Instructor:**
  - Dr. Wenbing Zhao
  - e-mail: w.zhao1@csuohio.edu
  - Lecture hours:
    - ⊗ TTh 6:00-7:50pm
  - Office hours: *TTh 4:00-6:00pm and by appointment*
- # **Networking Lab (SH 306)**
  - Upon request, I can create account on the lab machines for you
  - Can remotely login
- # **Course Web site:**
  - [http://academic.csuohio.edu/zhao\\_w/teaching/eec682.htm](http://academic.csuohio.edu/zhao_w/teaching/eec682.htm)
  - Lecture notes and homework will be posted

## Introduction of Computer Networks

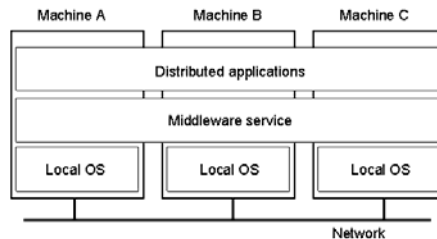
- # Uses of computer networks
- # Network Hardware
- # Network software

## Definition of Computer Network

- # Interconnected collection of autonomous computers
  - Interconnected - able to exchange info via copper wire, fiber, microwaves, satellites, etc.
  - Autonomous - act independently
  - Single network vs. network of networks

## Computer Network vs Distributed System

- # Distributed system - A collection of independent computers that appear to the users as a single coherent system
  - software specifically designed to provide an integrated computing facility
  - Two aspects (1) independent computers and (2) single system => **middleware**
- # Computer network - User must handle network management explicitly



## Uses of Computer Networks

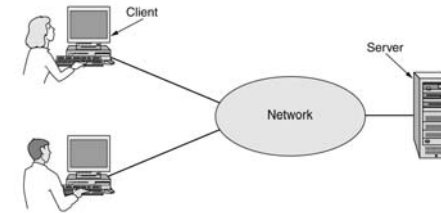
- # Business applications
- # Home applications
- # Mobile users
- # Social issues

## Business Applications of Networks

- ‡ Goals of using computer networks for business
  - Resource sharing
  - Provide powerful communication medium among employees
  - Doing business electronically with other companies
  - Doing business with consumers over the Internet (e-commerce)

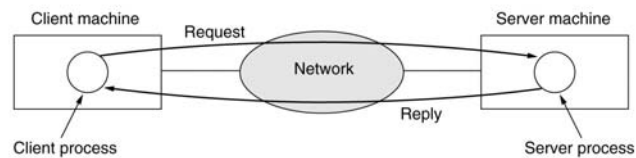
## Resource Sharing

- ‡ Resource sharing
  - equipment, software programs, data
- ‡ Client-server model
  - Server – provides services for clients
  - A network with two clients and one server.



## Client-Server Model

- ‡ The client-server model involves requests and replies.

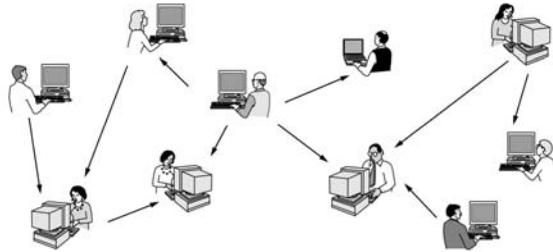


## Home Network Applications

- ‡ Access to remote information
  - WWW, online newspapers, sports news
- ‡ Person-to-person communication
  - Instant messaging, chat room, peer-to-peer communication/file sharing
- ‡ Interactive entertainment
  - Video on demand, game playing
- ‡ Electronic commerce
  - Online banking, online auction, etc.

## Home Network Applications

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



## Home Network Applications

- ✦ Some forms of e-commerce.

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

## Mobile Network Users

- ✦ Wireless network
  - Portable office, military, taxis, trucks
  - Fixed wireless, mobile wireless
- ✦ Combinations of wireless networks and mobile computing

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

- ✦ Future applications
  - Wireless parking meters, vending machine inventory report, utility meter reading, merge of cell phones and PDAs,
  - m-commerce, personal area networks and wearable computers
  - Smart watches, smart dust

## Social Issues

- ✦ Offending content
- ✦ Copyright violations
- ✦ Spams
- ✦ Viruses
- ✦ Who is responsible?
  - Network operator? Software provider? Users?
- ✦ What kind of control is appropriate?

## Network Hardware

- ✦ No generally accepted taxonomy. Two dimensions
  - Transmission technology
  - Scale
- ✦ Transmission technology
  - Broadcast links
  - Point-to-point links (unicasting)
- ✦ Scale
  - Personal area networks
  - Local area networks
  - Metropolitan area networks
  - Wide area networks
  - The Internet

## Classification by Scale

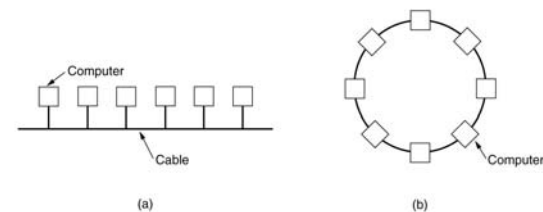
Interprocessor distance	Processors located in same	Example
1 m	Square meter	Personal area network
10 m	Room	
100 m	Building	Local area network
1 km	Campus	
10 km	City	Metropolitan area network
100 km	Country	
1000 km	Continent	Wide area network
10,000 km	Planet	

## Local Area Networks

- ✦ LAN typically spans a room, a building or a campus
- ✦ LANs are distinguished from other kinds of networks by three characteristics
  - Size – restricted in size => worst-case transmission time is bounded
  - Transmission technology – typically using broadcast
    - Static and dynamic broadcast channel allocation
  - Topology

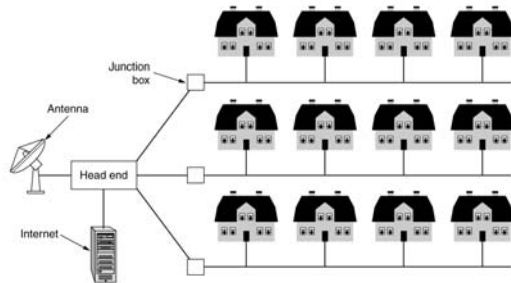
## Local Area Networks

- ✦ Two broadcast networks
  - (a) Bus
  - (b) Ring



## Metropolitan Area Networks

- MAN covers a city
- A metropolitan area network based on cable TV.

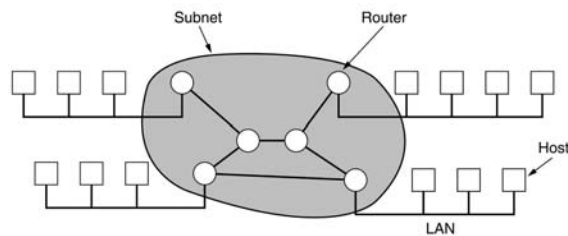


## Wide Area Networks

- Spans a large geographical area, often a country or continent
- Network structure in WAN
  - Host or end system
    - Collection of machines that run user (application) programs
  - Communication Subnet – connects hosts
    - Two distinct components
      - Transmission lines – move bits (circuits, channels, trunks)
      - Routers or switching elements that connect two or more transmission lines
    - Two types of designs
      - Point-to-point channels
      - Broadcast channels

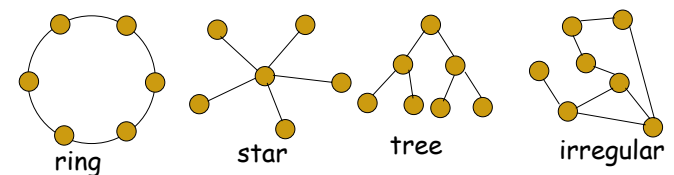
## Wide Area Networks

- Relation between hosts on LANs and the subnet.



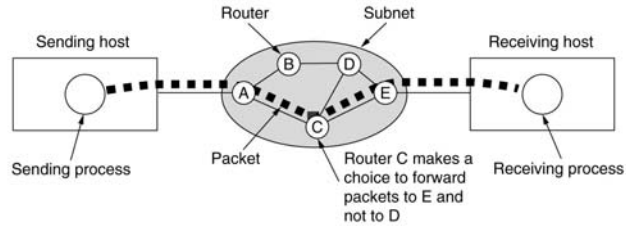
## Wide Area Networks

- Point-to-point channels
  - Store and forward or packet switch
  - If no direct connection, must use intermediate switching elements where store packets until required output line is free and then forward
  - Many different topologies – ring, star, tree, irregular



## Wide Area Networks

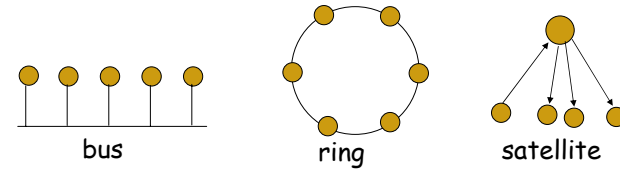
⌘ A stream of packets from sender to receiver.



## Wide Area Networks

⌘ Broadcast channels

- Sent by one, received by all
- Address field specifies intended destinations
  - Multicast if more than one destination
- When machine receives packet, checks address ignores if not for it
- Topologies



## Wide Area Networks

⌘ Broadcast channels

- Need to arbitrate simultaneous access to medium

■ Two approaches

■ Static

- ⌘ Divide time into slots, use round robin strategy
- ⌘ Waste channel capacity when nothing to send

■ Dynamic

- ⌘ Centralized - bus arbitration unit decides who goes next
- ⌘ Decentralized - each machine decides for itself



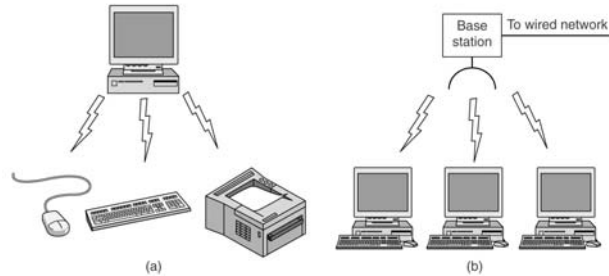
## Wireless Networks

⌘ Categories of wireless networks

- System interconnection
- Wireless LANs
- Wireless WANs

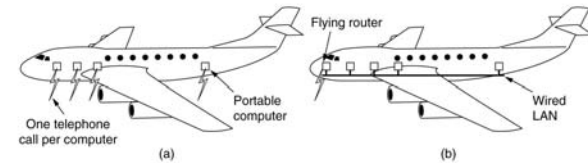
## Wireless Networks

- (a) Bluetooth configuration
- (b) Wireless LAN



## Wireless Networks

- (a) Individual mobile computers
- (b) A flying LAN



## Home Network Categories

- # Computers
  - Desktop PC, PDA, shared peripherals
- # Entertainment
  - TV, DVD, VCR, camera, stereo, MP3
- # Telecomm
  - Telephone, cell phone, intercom, fax
- # Appliances
  - Microwave, fridge, clock, furnace, air conditioner
- # Telemetry
  - Utility meter, burglar alarm, babycam

## Network Software

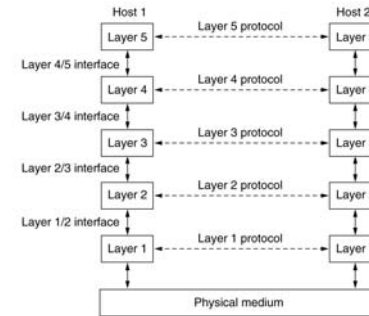
- # Protocol hierarchies / Network architectures
- # Design issues for the layers
- # Connection-oriented and connectionless services
- # Service primitives
- # The relationship of services to protocols

# Protocol Hierarchies

- ✦ **Protocol hierarchies** are organized into layers or levels with different protocols at each layer
- ✦ Each layer offers certain services to higher layers, hiding the details of implementation of those services
- ✦ Layer n on one machine communicates with layer n on another machine
- ✦ Interface between adjacent layers defines operations and services offered by lower layer to upper layer

# Protocol Hierarchies

- ✦ Layers, protocols, and interfaces.

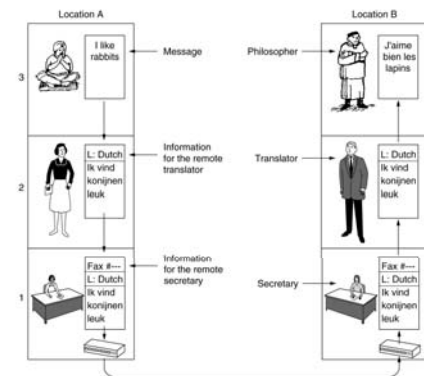


# Network Architectures

- ✦ **Network Architecture** - A set of layers and protocols
  - Must contain enough information for implementation
  - Does not contain details of the implementation and specification of the interfaces
  - It is not even necessary that the interfaces on all machines in a network be the same
- ✦ **Protocol Stack** - A list of protocols used by a certain system, one protocol per layer
- ✦ The subjects of network architectures, protocols stacks, and the protocols themselves are the principal topics of this course

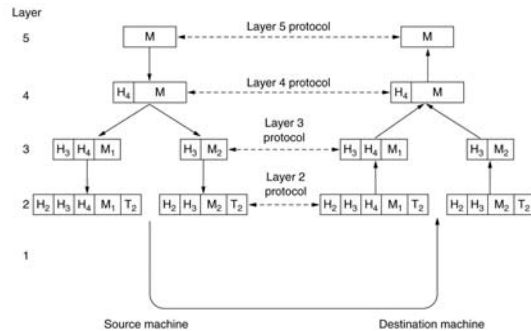
# Protocol Hierarchies

- ✦ The philosopher-translator-secretary architecture.



## Protocol Hierarchies

- ✦ Example information flow supporting virtual communication in layer 5.



## Design Issues for the Layers

- ✦ Each layer needs
  - Mechanisms for
    - connection establishment and addressing
    - Connection termination or release
  - Rules for data transfer
    - Simplex – one direction
    - Half duplex – either direction, but not simultaneously
    - Full duplex – both directions simultaneously
    - Number of logical channels and their priority
  - Error control
    - Error detecting and correcting codes
    - Positive and negative acks
    - Re-sequencing packets that arrives out of order

## Design Issues for the Layers

- ✦ Each layer needs
  - Flow control
    - Need to keep a fast sender from swamping a slow receiver
    - Use feedback from receiver to sender
  - Finite buffers
    - Inability of receiver to accept arbitrarily long messages
    - Use mechanisms for disassembling, transmitting, reassembling
  - Multiplexing
    - Several unrelated conversations between pair of communicating processes
  - Routing
    - Which route to use if there are several

## Connection-Oriented and Connectionless Services

- ✦ Connection-oriented service
  - Modeled after telephone system – establish connection before communication
  - Some service allow a **negotiation** among sender, receiver and subnet regarding the parameters to be used, such as max message size, etc.
- ✦ Connectionless service
  - Modeled after postal system – a message carries full destination address, and each one is routed through the system independent of all the others
    - Ordering of messages are not guaranteed

## Connection-Oriented and Connectionless Services

- # **Quality of service** - characterize each service
- # **Reliable service** - it does not lose data
  - Implemented by having the receiver acknowledge the receipt of each message
  - The acknowledgement process introduces overhead and delays
  - Two variations
    - Message sequences - message boundaries are preserved
    - Byte streams - message boundaries are **not** preserved
  - Example: file transfer
- # **Unreliable connectionless service - datagram service**
  - No acknowledgement is used
  - In analogy with telegram service
- # **Acknowledged datagram service**
  - Like sending a registered letter and requesting a return receipt
- # **Request-reply service** - used in client/server model

## Connection-Oriented and Connectionless Services

- # Six different types of service.

	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

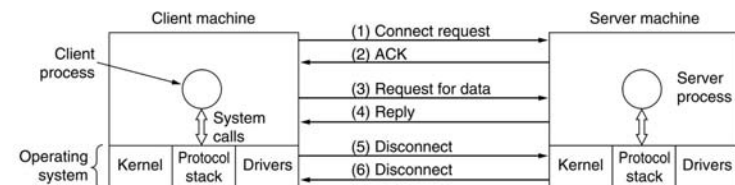
## Service Primitives

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

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

## Service Primitives

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



## Relationship of Services to Protocols

- ⌘ **Services** and **protocols** are **distinct** concepts
- ⌘ A **service** is a set of primitives (operations) that a layer provides to the layer above it
  - Defines what operations the layer is prepared to perform on behalf of its users
  - How service is implemented is not revealed (hidden)
  - Relates to an interface between two layers, with lower layer being the service provider and the upper layer being service user
- ⌘ A **protocol** is a set of rules governing the format and meaning of the packets, or messages that are exchanged by the peer entities within a layer
  - Entities use protocols to implement their service definitions
  - Different protocol could be used to provide the same service
  - Protocol and service are completely decoupled

## Relationship of Services to Protocols

- ⌘ The relationship between a service and a protocol.

