COMP 484 - Web Engineering I

Instructor:
Tania Babakhanlou

Reference:
Computer Networks, Tanenbaum

Computer Networks, Reference Models, Internet

Outline:

- Uses of Computer Networks
- Network Hardware
- Network Software
- Reference Models
- Example Networks
- Network Standardization

Uses of Computer Networks

- **Computer networks** are collections of autonomous computers, interconnected by a single technology. Example: internet, intranet

- Two computers are said to be interconnected if they are able to exchange information.

Internet: network of networks

Networks usage:

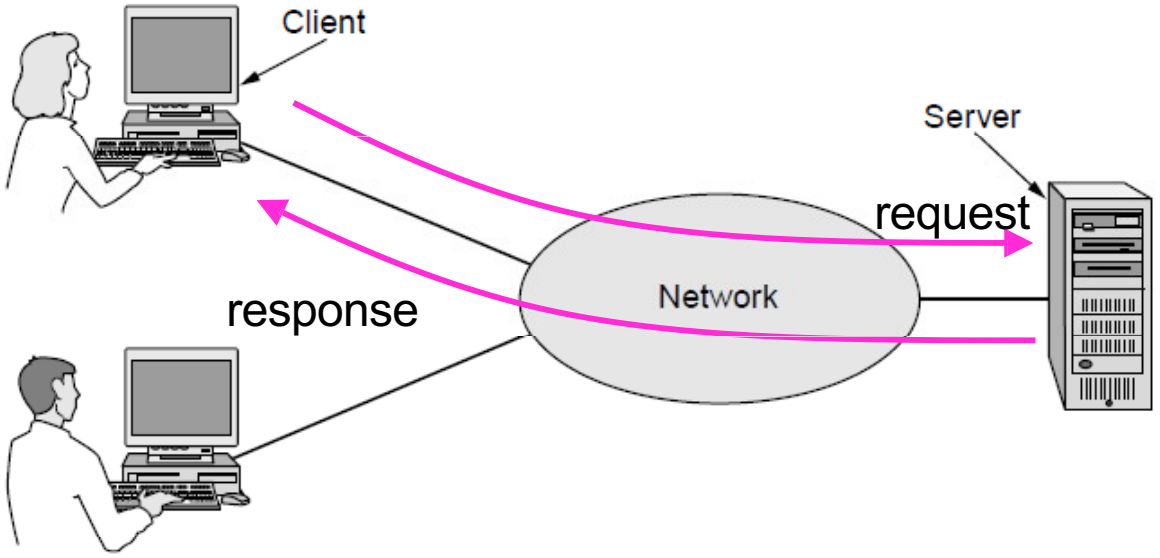
- Business Applications
- Home Applications
- Mobile Users

These uses raise:

- Social Issues

1- Business Applications:
Client Server Model(1)

- A distinct computer(s) called "Server" is designated to provide a specific service to others called "Clients"
- Example: Web server, mail server, file server
- The server can be a distributed application system
- Companies use networks and computers for **resource sharing** (data, program, equipment), with the client-server model
- The server itself can be a client in turn for a different type of service



Client Server Model(2): Web application

Another example of Client-Server model is Web Applications.

- The server generates Web page content based on its database and information from the client in response to client requests (that may also update the database).

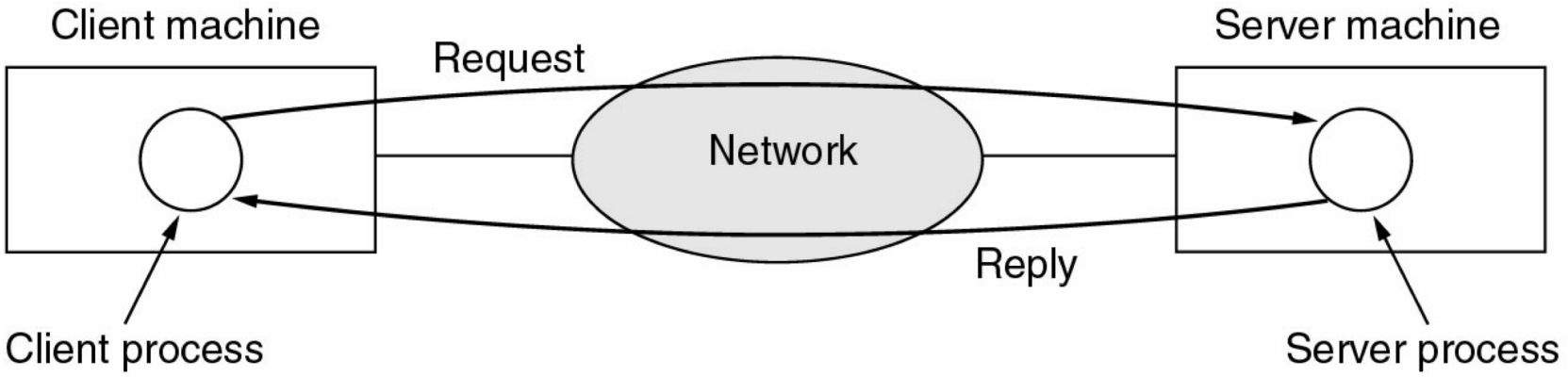
In this scenario:

- Server: Remote Web
- Client: user's personal computer, or to be more precise, the browser

Client Server Model (cont'd)

2 processes are involved in a client/server scenario:

- one on the client machine
- one on the server machine



Other popular uses of networks:

- communication, e.g., email, VoIP, and e-commerce

2 - Home Applications

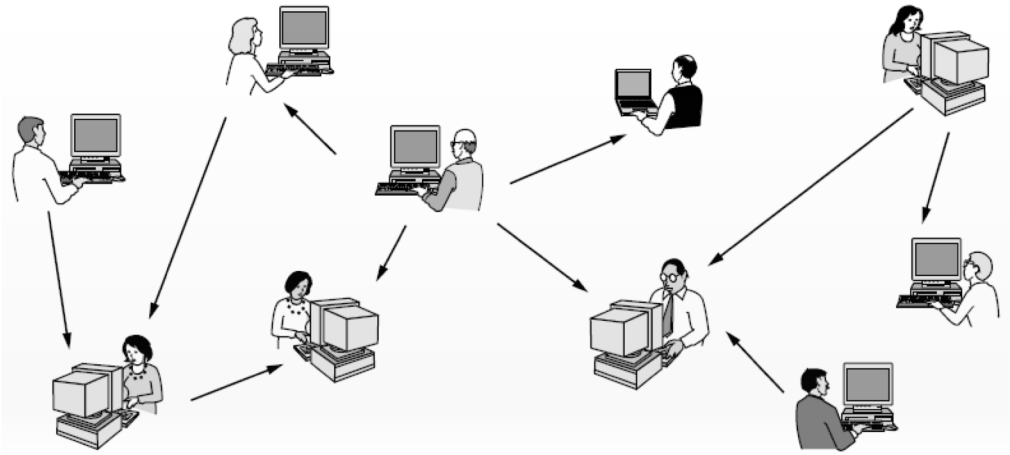
Homes contain many networked devices, e.g., computers, TVs, connected to the Internet by cable, DSL, wireless, etc.

Usage: Home users communicate, e.g., social networks, consume content, e.g., video, and transact, e.g., auctions through networks

Not all network usages follow a client/server model. Some applications use the **peer-to-peer** model in which there are no fixed clients and servers

Peer-to-peer Networks

- No fixed servers or clients
- Individuals who form a loose group can communicate with others in the group
- example usage: sharing music and data (Napster, BitTorrent)
- BitTorrent: no central database of content, but individual local databases. Each user maintains their own database locally and provides a list of other nearby people who are members of the system.



3 - Mobile Users

Tablets, laptops, and smart phones are popular devices; Network technology: WiFi hotspots and 3G+ cellular provide wireless connectivity.

Usage: Mobile users communicate, e.g., voice and texts, consume content, e.g., video and Web, and use sensors, e.g., GPS.

Wireless and mobile are related but different:

Wireless	Mobile	Typical applications
No	No	Desktop computers in offices
No	Yes	A notebook computer used in a hotel room
Yes	No	Networks in unwired buildings
Yes	Yes	Store inventory with a handheld computer

Social Issues of Networks

- Network neutrality – no network restrictions:
  - communications should not be differentiated by their content or source or who is providing the content
  - some network operators block content for their own reasons.
  - treat different companies differently

- Content ownership, e.g., DMCA (The Digital Millennium Copyright Act) takedowns
  - Peer to peer networks, sharing music and movie content
  - Napster shut down (2001), LimeWire shut down(2010),…
  - Example method of tracking: Automated systems search P2P networks, send warnings to network operators
- Privacy, e.g., Web tracking and profiling (cookies)

- Theft, e.g., botnets (pool of malware compromised machines- e.g. DDOS) and phishing (masquerade as originating from a trustworthy party)

Network Hardware

Networks can be classified by their scale:

Scale	**Type**
Vicinity	PAN (Personal Area Network)
Building	LAN (Local Area Network)
City	MAN (Metropolitan Area Network)
Country	WAN (Wide Area Network)
Planet	The Internet (network of all networks)

Wide Area Networks-terminology

- WAN spans a large geographical area : connects devices over a country
- Example: ISP, VPN

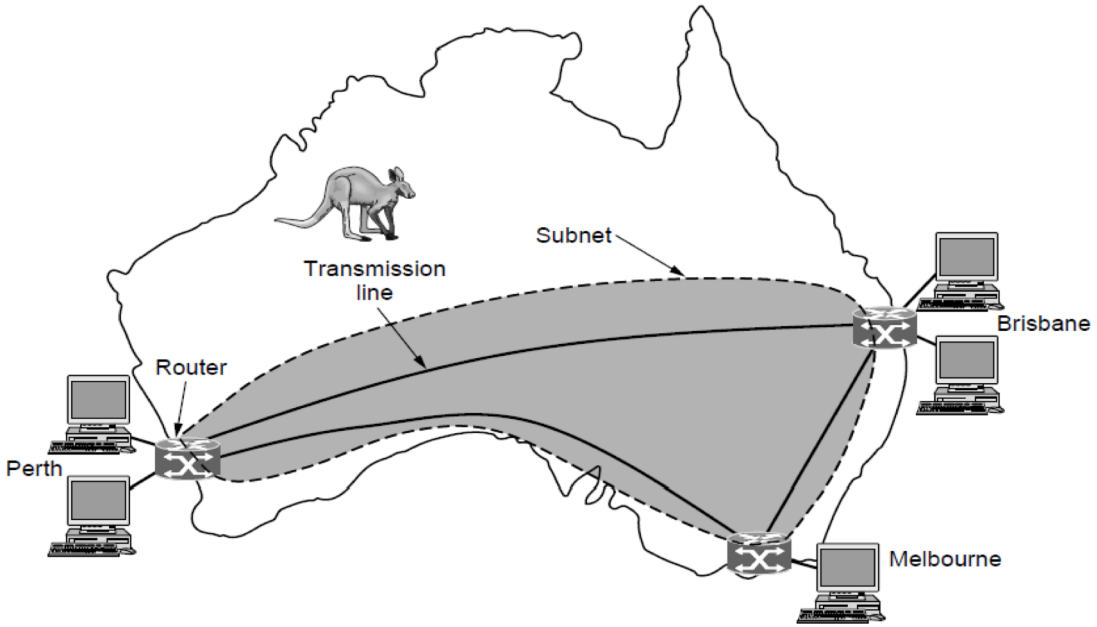
WAN components: **Hosts** + communication **subnet**

- **Subnet** = **transmission lines** + **switching elements**

1. **Transmission lines**: move bits between machines (copper wire, optical fiber, or even radio links). Most companies do not own transmission, so instead they **lease** the lines from a telecommunications company.

2. **Switching elements** (routers, switches): are specialized computers that connect two or more transmission lines. When data arrive on an incoming line, the switching element must choose an outgoing line on which to forward them.

- In a WAN, the hosts and subnet are owned and operated by different people
- In a WAN, an endpoint can be a complete LAN in addition to individual computers
- Example WAN connecting three branch offices in the image =>



Network Software

Outline:

- Protocol layers
- Design issues for the layers
- Connection-oriented vs. connectionless service
- Service primitives
- Relationship of services to protocols

Network Software (cont'd)

- To reduce their design complexity, most networks are organized as a stack of layers or levels, each one built upon the one below it.

- The purpose of each layer is to offer certain services to the higher layers while shielding those layers from the details of how the offered services are actually implemented. And the implementation changes should not impact other layers.
- Abstraction, encapsulation

- A protocol is an agreement between the communicating parties on how communication is to proceed.

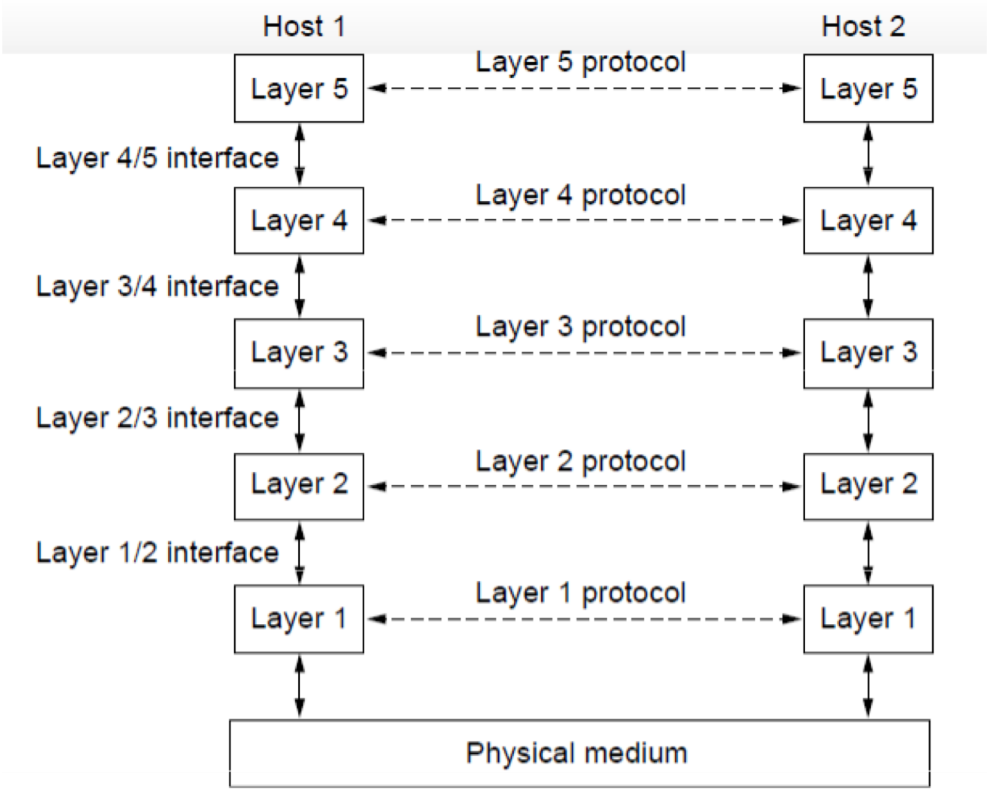
- **Layer N Protocol**: When **layer N on one machine** carries on a conversation with **layer N on another machine**, the rules and conventions used in this conversation are collectively known as the layer-N protocol.

- 2 hosts in a network communicate using the common network protocol

Protocol Layers (1)

Protocol layering is the main structuring method used to divide up network functionality.

- Each protocol instance talks **virtually** to its peer
- Each layer communicates to the destination only by using the one below
- Lower layer services are accessed by an interface
- At bottom, messages are carried by the (physical) medium



Protocol Layers (cont'd)

- Between each pair of adjacent layers is an **Interface**.
- **The interface** defines which operations and services the lower layer makes available to the upper one

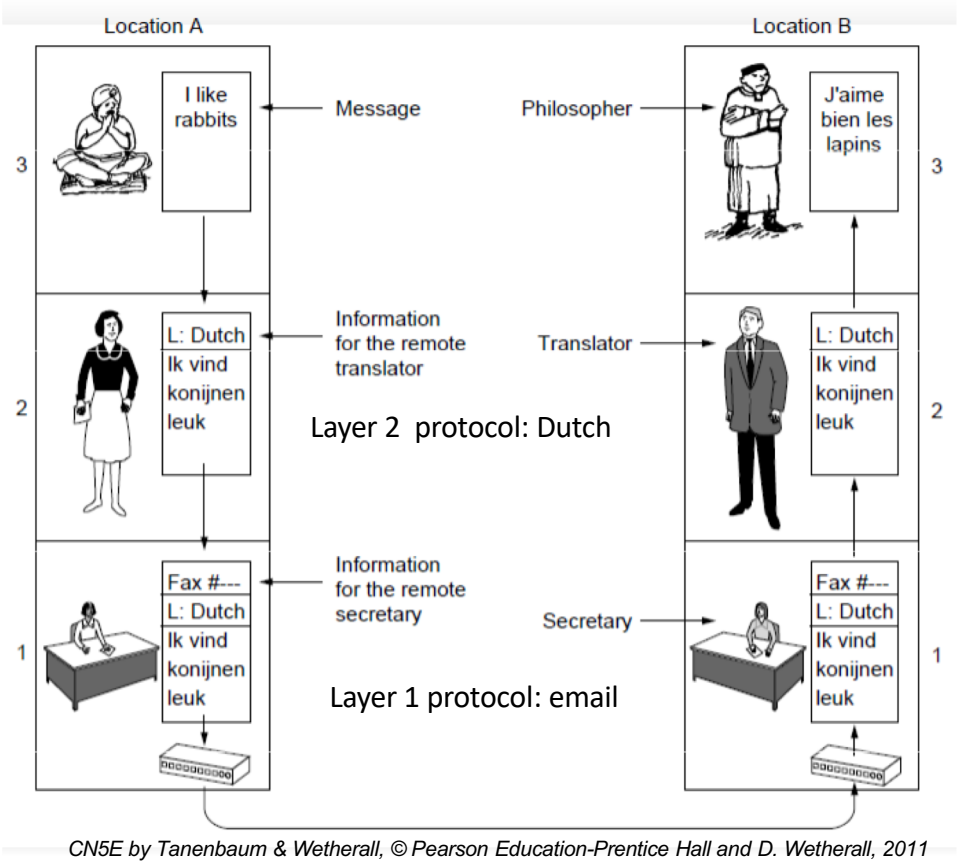
- A set of layers and protocols is called a **network architecture**.

- A list of the protocols used by a certain system, one protocol per layer, is called a **protocol stack**.

Protocol Layers (cont'd)

Example: the philosopher-translator-secretary architecture
Each protocol at different layers serves a different purpose
Each layer may add some information intended only for its peer. This information is not passed up to the layer above.
Layer 2 peers agree to change the protocol from Dutch to French. No changes to the other layers

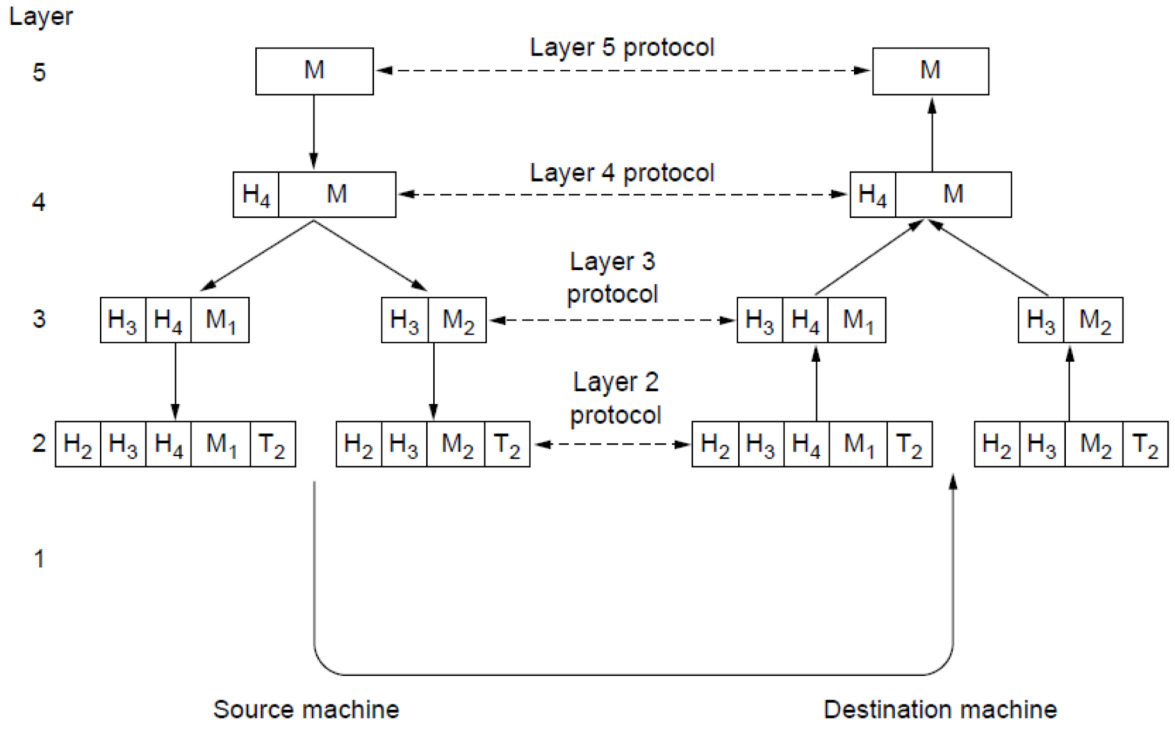
Urdu, English



Chinese, French

Protocol Layers (cont'd)

Each lower layer adds its own header (with control information) to the message to transmit and removes it on receive



Layers may also **split** and **join** messages, etc.

Protocol Layers (cont'd)

- In network protocol stack, a message, M, is produced by an application process running in layer 5 and given to layer 4 for transmission.
- Layer 4 puts a header in front of the message to identify the message and passes the result to layer 3. The header includes **control information**, such as **addresses**, to allow layer 4 on the destination machine to deliver the message.
- Other examples of control information used in some layers are **sequence numbers** (in case the lower layer does not preserve message order), sizes, and times.
- In many networks, no limit is placed on the size of messages transmitted in the layer 4 protocol but there is nearly always a limit imposed by the layer 3 protocol.
- Consequently, layer 3 must break up the incoming messages into smaller units, packets, prepending a layer 3 header to each packet.

Design Issues for the Layers

Different issues occur in networks design and operations.

Each layer solves a particular problem but must include mechanisms to address a set of recurring design issues

Issue	Example mechanisms at different layers
Reliability despite failures	Codes for error detection/correction (§3.2, 3.3) Routing around failures (§5.2)
Network growth and evolution	Addressing (§5.6) and naming (§7.1) Protocol layering (§1.3)
Allocation of resources like bandwidth	Multiple access (§4.2) Congestion control (§5.3, 6.3)
Security against various threats	Confidentiality of messages (§8.2, 8.6) Authentication of communicating parties (§8.7)

Connection-Oriented vs. Connectionless

The service provided by a layer to the upper one can be of either kind.

1- Connection-oriented Service

- Must be set up for ongoing use (and torn down after use), e.g., phone call
- Includes negotiation
- the service user first **establishes** a **connection**, uses the connection, and then **releases** the connection.

- acts like a tube: the sender pushes objects (bits) in at one end, and the receiver takes them out at the other end. **Order** is preserved.
- **Circuit** is established and used to the end of the connection
- Circuit : a connection with associated resources (Fixed bandwidth, QOS metrics, max size, …)

	Service	Example
Connection-oriented	Reliable message stream	Sequence of pages
	Reliable byte stream	Movie download
	Unreliable connection	Voice over IP
Connection-less	Unreliable datagram	Electronic junk mail
	Acknowledged datagram	Text messaging
	Request-reply	Database query

Connection-Oriented vs. Connectionless(2)

2- Connectionless Service

- messages are handled separately, e.g., postal delivery
- **Each** message (letter) carries the full destination **address**
- Each Message is routed through the intermediate nodes inside the system independent of all the subsequent messages.
- Might take different paths to the destination
- Might be delivered **out of order**
- Example: request-reply, **datagram**

	Service	Example
Connection-oriented	Reliable message stream	Sequence of pages
	Reliable byte stream	Movie download
	Unreliable connection	Voice over IP
Connection-less	Unreliable datagram	Electronic junk mail
	Acknowledged datagram	Text messaging
	Request-reply	Database query

Reliable vs. Unreliable Services

- **Reliability of the service**:
- In a **reliable** service, the receiver **acknowledges** the receipt of each message so the sender is sure that it arrived**. Overhead and delay.**

- Connection-oriented and connectionless can be either Reliable or Unreliable

- Connection-oriented Services:
  - **Reliable** connection-oriented
    - Acknowledgement (Ack)
    - Example: TCP, file transfer by FTP
  - **Unreliable** Connection-oriented service:
    - No Acknowledgement
    - Example: VOIP

Reliable vs. Unreliable Services

Connectionless Services:

1. Unreliable Datagram Service: (not acknowledged) connectionless service (mail)

- Example: Request-reply (e.g. UDP):
  - the sender transmits a single datagram containing a request; the reply contains the answer. Used for communication in the client-server model

2. **Reliable (Acknowledged) Datagram service:**
   - **Reliable** through acknowledgment (text messages)
   - RUDP- Reliable UDP

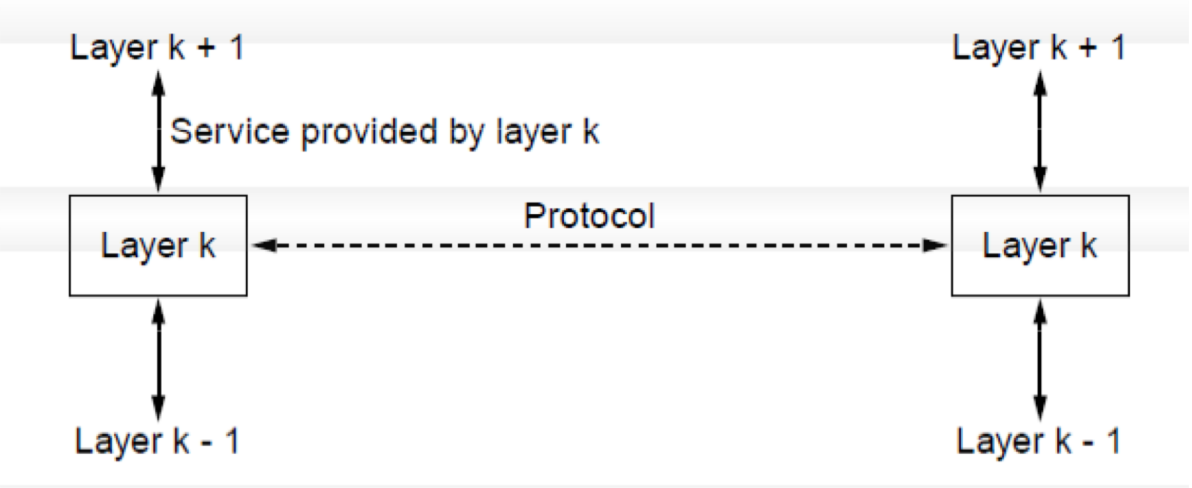
Reliable VS. Unreliable Services

- Both are needed
- Reliable communication may not be available in a given layer (Ethernet) => the only option is unreliable service
- But, packets can occasionally be damaged in transit. What to do?
- So if the service is unreliable, it is up to higher protocol levels to recover from this problem. Many reliable services are built on top of an unreliable datagram service.
- If delays might not be acceptable (Multimedia), sending Ack is not a proper option, then unreliable service is a better choice.

Relationship of Services to Protocols

- A layer provides a **service** to the one above [vertical]
  - A set of primitives (set of available operation)
  - Says nothing at all about how these operations are implemented.

- A layer talks to its peer using a **protocol** [horizontal]
  - is a set of rules around the format and meaning of the messages that are exchanged by the peer entities within a layer.
  - Entities use protocols to implement their service definitions.



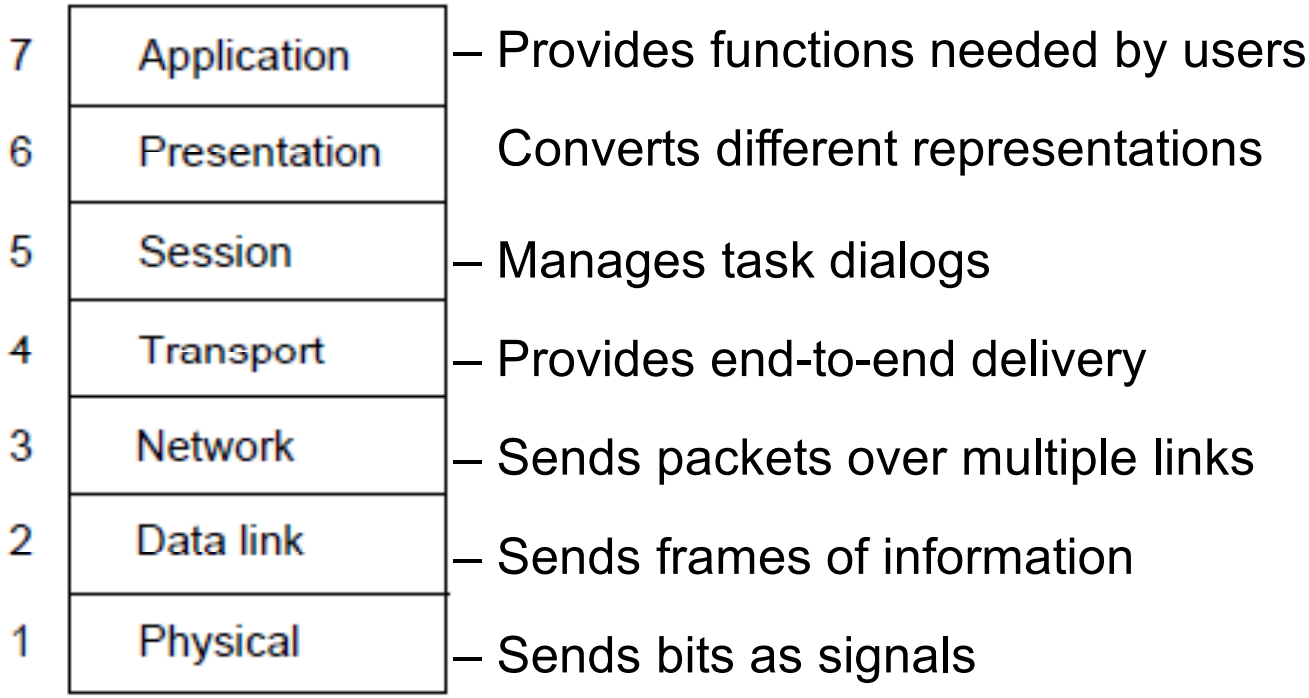
Reference Models

Reference models describe the **layers** in a **network architecture**

- OSI reference model (valid model, protocols not in use)
  - Short for Open Systems Interconnection model
- TCP/IP reference model (model not is use, protocols in use)
  - TCP: Transmission Control Protocol
  - IP: Internet Protocol

OSI (Open Systems Interconnection) Reference Model

A principled, international standard, seven layer model to connect different systems



OSI Layers Functionalities

Physical (Layer 1 in OSI Model)

- Layer 1 conveys the bit stream - electrical impulse, light or radio signal — through the network at the electrical and mechanical level.
- It provides the hardware means of sending and receiving data on a carrier, including defining cables, cards and physical aspects.

Example: **Ethernet**, FDDI (fiber), RJ45.

OSI Layers Functionalities

Data Link (Layer 2)

- At OSI Model, Layer 2 data packets - called **frames** - are encoded and decoded into bits.
- It handles errors in the physical layer, flow control and frame synchronization.
- Moves frames from one end of a wire to the other (**one hop**)
- The data link layer is divided into two sub layers:
  1. The Media Access Control (MAC) layer and
  2. the Logical Link Control (LLC) layer.

MAC sublayer:

- The MAC sub layer controls how a computer on the network gains access to the data and permission to transmit it.
- Also Layer 2 Addressing is another task of MAC Sublayer. MAC Address is a unique value assigned to NIC by manufacturer and is 48 bits of length.
- Example MAC address: 3c:15:c2:e2:44:33

LLC Sublayer:

- The LLC layer controls frame synchronization, flow control and error checking.

Example of Data link protocols: **Ethernet**, ATM, IEEE 802.5(token ring), Frame Relay.

OSI Layers Functionalities

Network (Layer 3)

- Gets layer 3 data units, called **packets**, from the source host all the way to the destination host through multiple hops.
- Network layer is the lowest layer that deals with end-to-end transmission.

- Layer 3 provides switching and routing technologies, creating logical paths, known as virtual circuits, for transmitting data from node to node.
- Routing and forwarding are functions of this layer, as well as addressing, internetworking, error handling, congestion control and packet sequencing.

- Every machine on the internet has an IP address (layer 3 address). It is used to deliver the packet to the destination.
- **Internet** network layer protocol: IP
- IP protocol is best-effort (i.e., not guaranteed) to deliver packets.

- IPv4 address is 32 bits of length. Example: 19.117.63.203

- IPv6 address is 128 bits of length. Example: 2001 : db8 : 3333 : 4444 : CCCC : DDDD : C0A8 : 0102

Example: AppleTalk DDP, IP, IPX, ICMP.

OSI Layers Functionalities

Transport (Layer 4)

- OSI Model, Layer 4, provides transparent transfer of layer 4 data unit – called **segment** - between end systems, or hosts (end-to-end delivery), from a specific service(application) at the source to a specific service at the destination
- It is responsible for end-to-end error recovery and flow control.
- It ensures complete data transfer.
- Layer 4 address is called Port and is 16 bits of length. (0 to 65535)
- Applications to application delivery by sockets.
- Socket = IP Address + Port 208.80.154.224:80
- IP: a specific machine on the internet
- Port : a specific service on that machine

Example protocols: TCP, UDP.

Session (Layer 5)

This layer establishes, manages and terminates connections between applications. The session layer sets up, coordinates, and terminates conversations, exchanges, and dialogues between the applications at each end. It deals with session and connection coordination.

Example: NFS, NetBios names, RPC, SQL.

OSI Layers Functionalities

Presentation (Layer 6)
This layer provides independence from differences in data representation (e.g., encryption) by translating from application to network format, and vice versa. The presentation layer works to transform data into the form that the application layer can accept. This layer formats and encrypts data to be sent across a network, providing freedom from compatibility problems. It is sometimes called the syntax layer.

Example: encryption, ASCII, EBCDIC, TIFF, GIF, PICT, JPEG, MPEG, MIDI.

Application (Layer 7)
OSI Model, Layer 7, supports application and end-user processes. Communication partners are identified, quality of service is identified, user authentication and privacy are considered, and any constraints on data syntax are identified. Everything at this layer is application-specific. This layer provides application services for file transfers, e-mail, and other network software services. Telnet and FTP are applications that exist entirely in the application level. Tiered application architectures are part of this layer.

Example: WWW browsers, NFS, SNMP, Telnet, HTTP, FTP

OSI

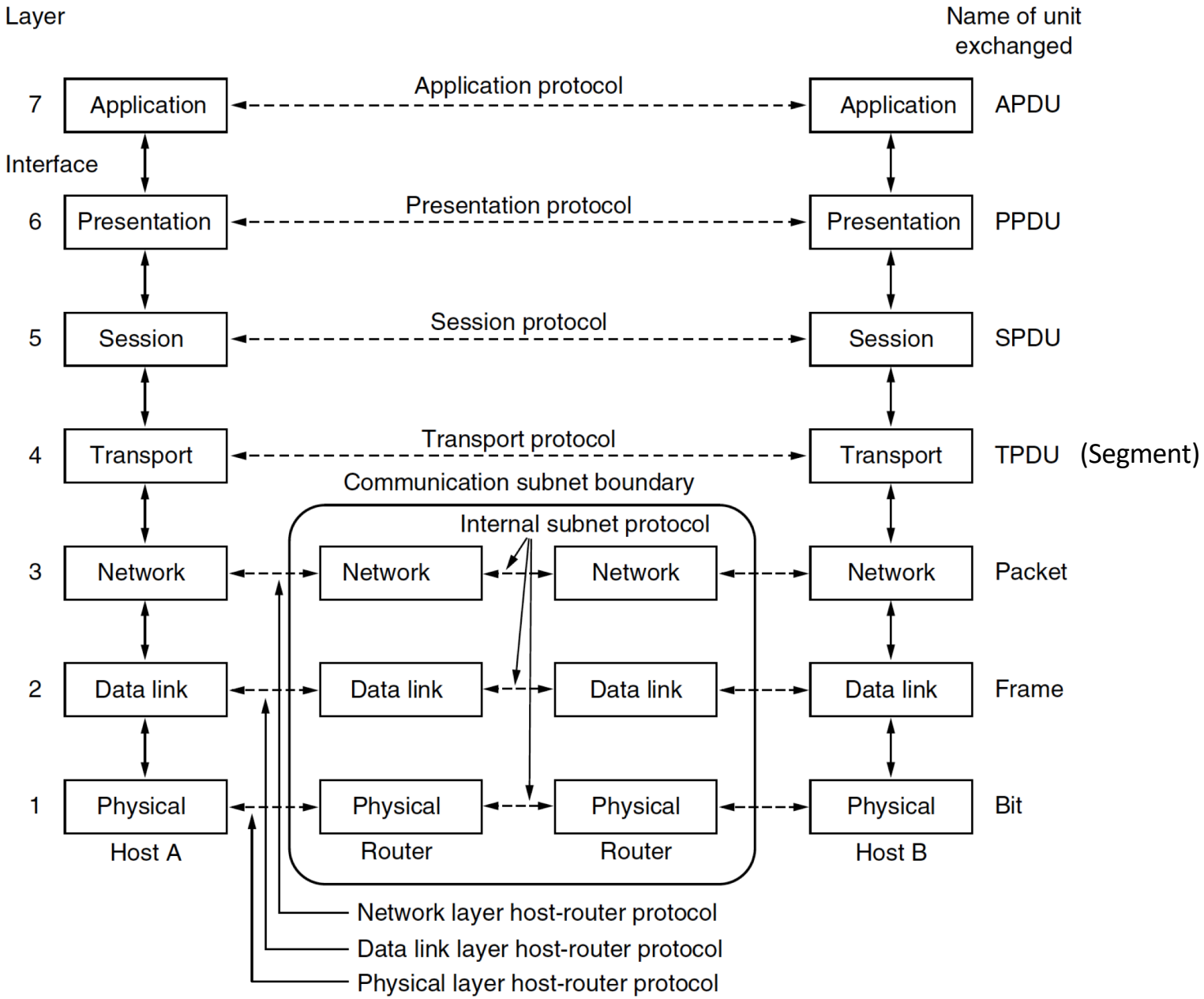
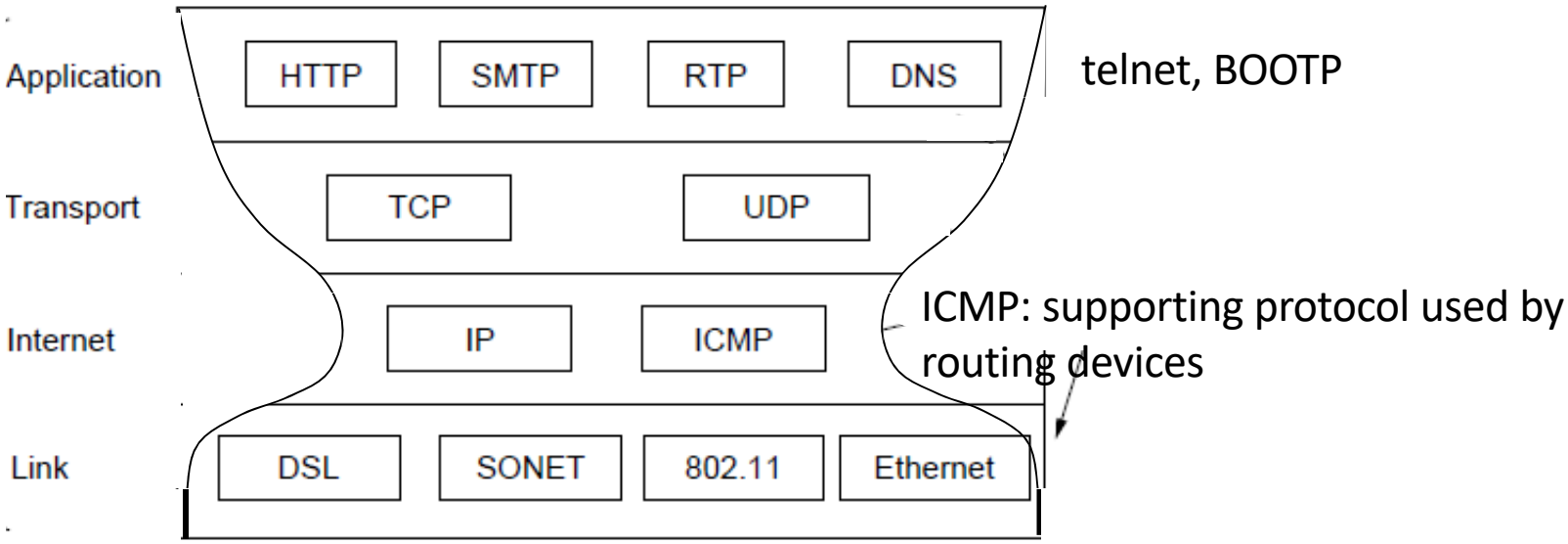


Figure 1-20. The OSI reference model.

TCP/IP Reference Model

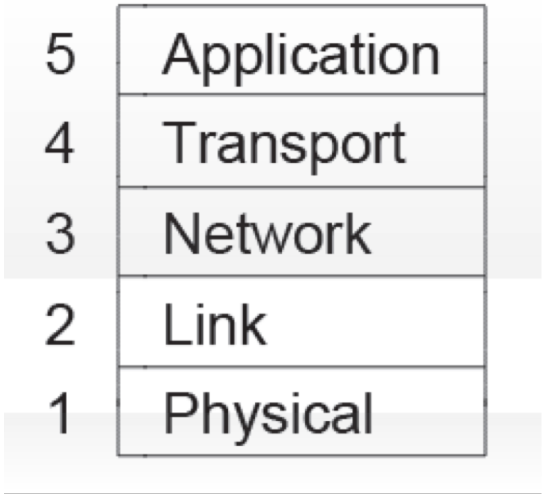
A four layer model derived from experimentation; omits some OSI layers and uses the IP as the network layer.



Protocols are shown in their respective layers

Model Used in this Book

It is based on the TCP/IP model but we call out the physical layer and look beyond Internet protocols.



Critique of OSI & TCP/IP

OSI:

- \+ Very influential model with clear concepts
- – Models, protocols and adoption all bogged down by politics and complexity
  - – Bad timing, Bad technology (session and presentation vs. data link and network), Bad implementations, Bad politics.

TCP/IP:

- \+ Very successful protocols that worked well and thrived

  One of the first implementations as a part of Unix implementation/ free
- – Weak model derived after the fact from protocols
- – not clearly distinguish the concepts of services, interfaces, and protocols.

Example Networks

- The Internet

Internet

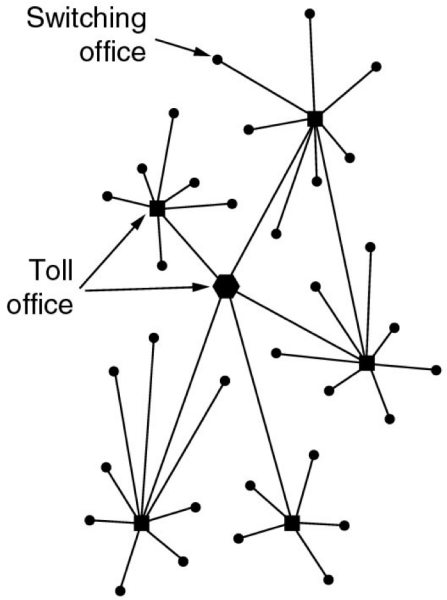
- Not really a network, but a vast collection of different networks that use certain common protocols and provide certain common services.
- Not planned by anyone, not controlled by anyone.
- Cold War: U.S. DoD (Department of Defense) wanted a command and control network, less vulnerable than telephone network

(a) Structure of the telephone system

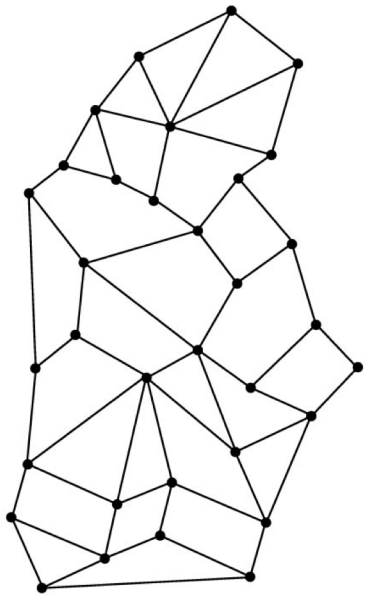
- Telephones-> switching offices -> toll offices
- Hierarchy, little redundancy, vulnerable to points of failures, distortion

(b) Paul Baran's (RAND Corporation) proposed distributed+fault-tolerant switching system

- packet-switching to solve for distortion from longer paths
- Idea wasn't implemented



(a)



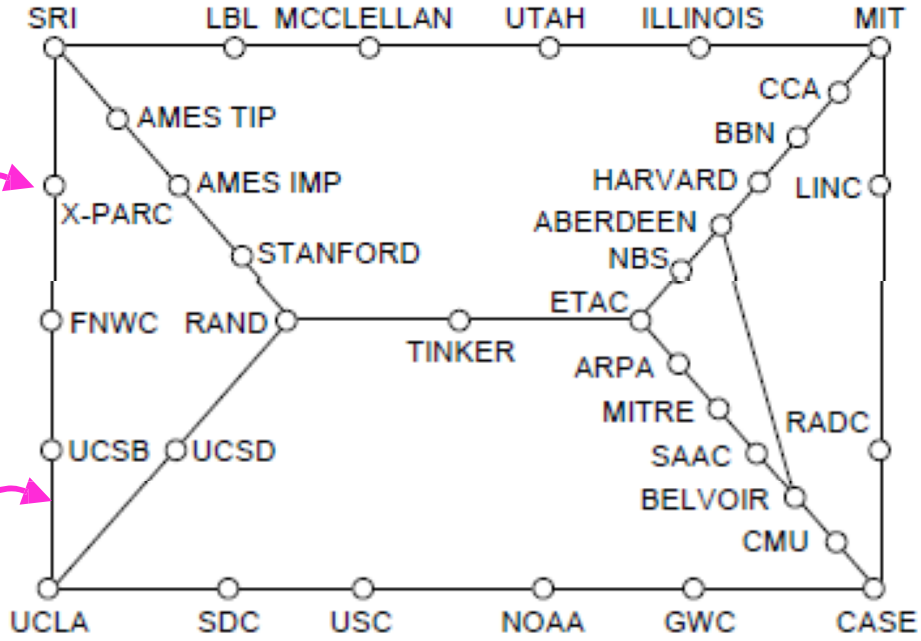
(b)

Internet

- Store and-forward packet-switching network (look at the next slide)
- ARPA- Advanced Research Projects Agency- by Pentagon
- Internet is a successor of ARPANET
- ARPANET : a decentralized, packet-switched network based on Baran's ideas
- Hosts connected to routers or IMPs
- Subnet: Minicomputers (**Interface Message Processors**) and 56-kbps links
- Reliability:
  - each IMP connected to 2+ others
- Subnet and software built

Nodes are IMPs, or early routers, linked to hosts

56 kbps links



ARPANET topology in Sept 1972.

Internet

- **Store-and-forward switching:** the intermediate nodes receive a message in full before sending it on to the next node,
- **Cut-through switching**: the onward transmission of a message at a node starts before it is completely received by the node

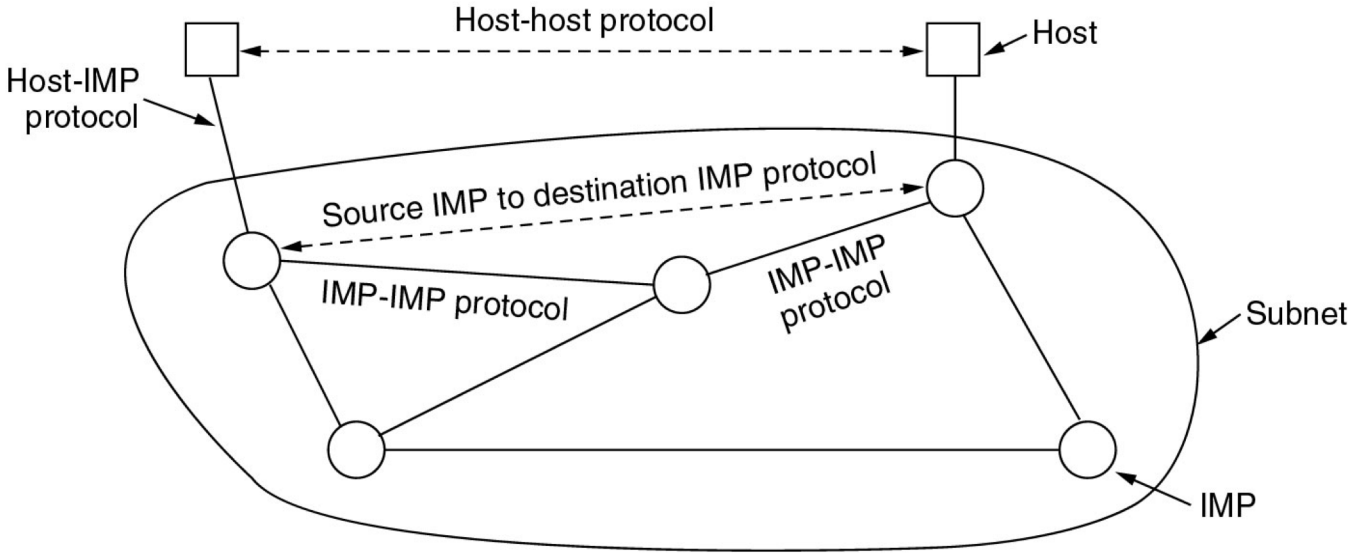
Internet: Formation of TCP/IP

- ARPANET eventually connected hundreds of universities and government installations, using leased telephone lines.
- Satellite and radio networks were added later, so the existing protocols had trouble interworking with them, so a new reference architecture was needed (new protocols).
- One major goal: the ability to connect multiple networks in a seamless way.
- This architecture later became known as the TCP/IP Reference Model, after its two primary protocols.

Internet

Original Arpanet Design

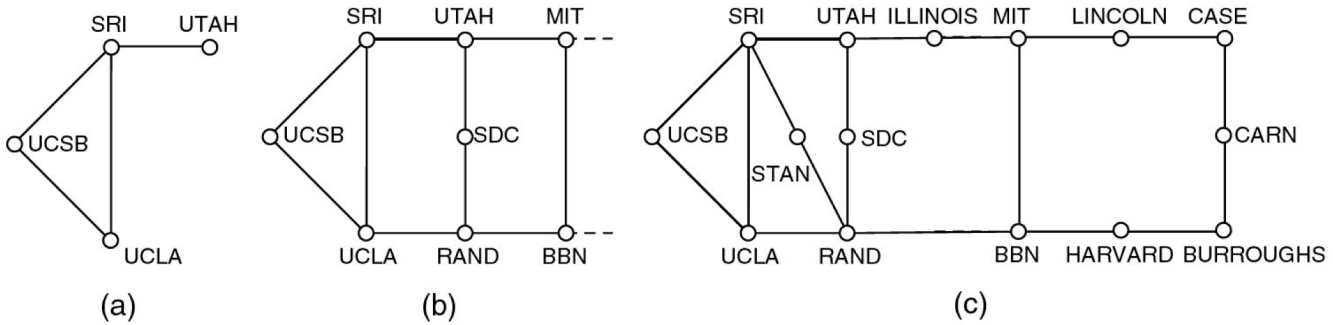
- Communication between nodes
- Host <-> IMP: Message size limit 8063 bits
- IMP <-> IMP: Packets size limit 1008 bits
- Splitting needed

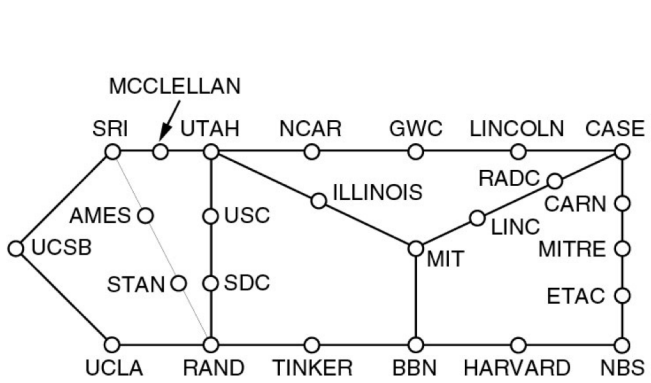


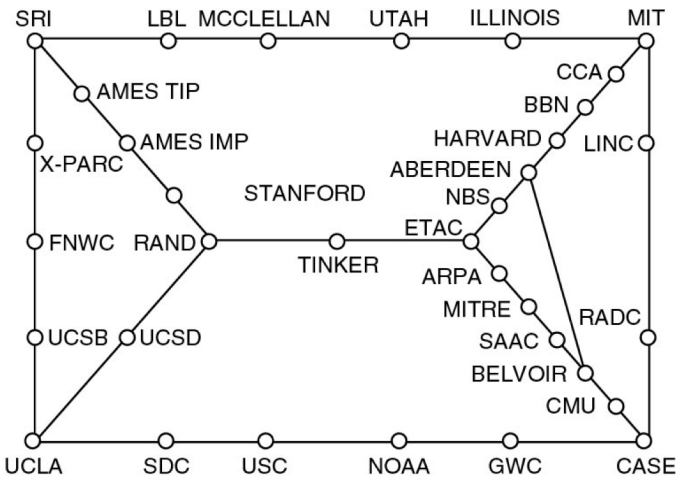
Internet

- Experiments: ARPANET protocols were not suitable for running over different networks (internetworks)
- Invention of the TCP/IP model and protocols

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





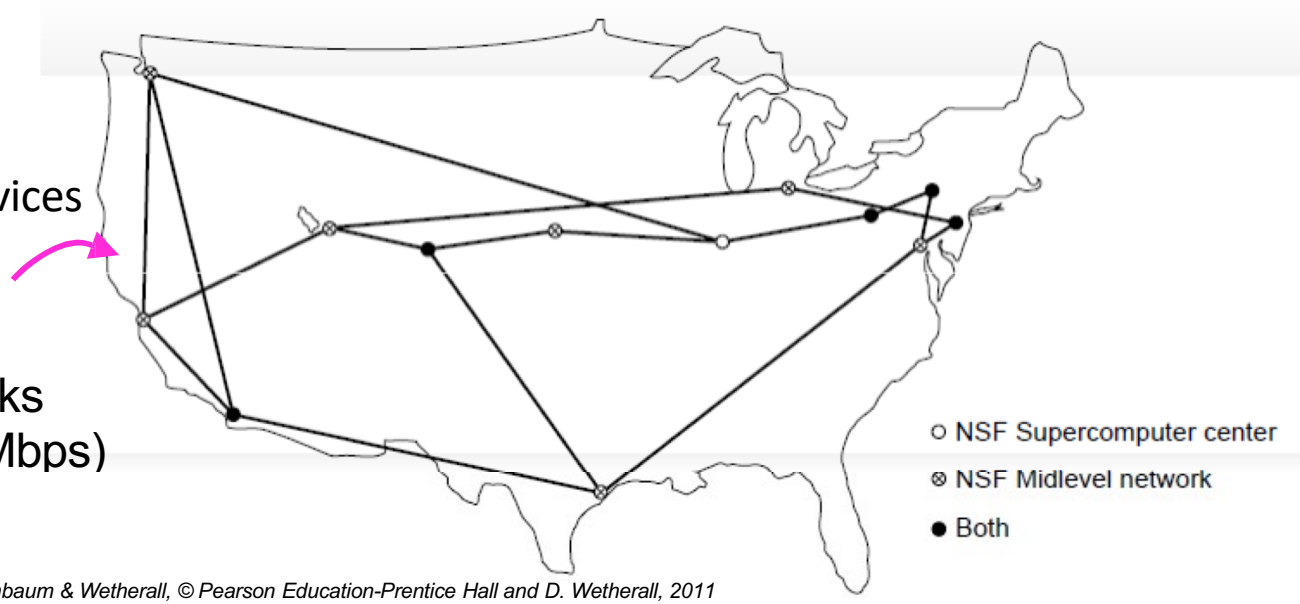


• Internet

- NSF- National Science Foundation
  - To get on the ARPANET a university had to have a research contract with the DoD.
  - NSF: At first funded the Computer Science Network (CSNET) in 1981. It connected CS departments and industrial research labs to the ARPANET via dial-up and leased lines.
  - NSF created a Successor to ARPANET called NSFNET
    - Backbone of 6 supercomputers
    - 56-kbps leased lines
    - TCP/IP from the start – first TCP/IP WAN
- The early Internet used NSFNET (1985-1995) as its backbone; universities, libraries, museums … connected to get on the Internet
- Connected to ARPANET

- ANS corporation
- Advanced Networks and Services
  - Commercialization

T1 links
(1.5 Mbps)



NSFNET topology in 1988

Internet

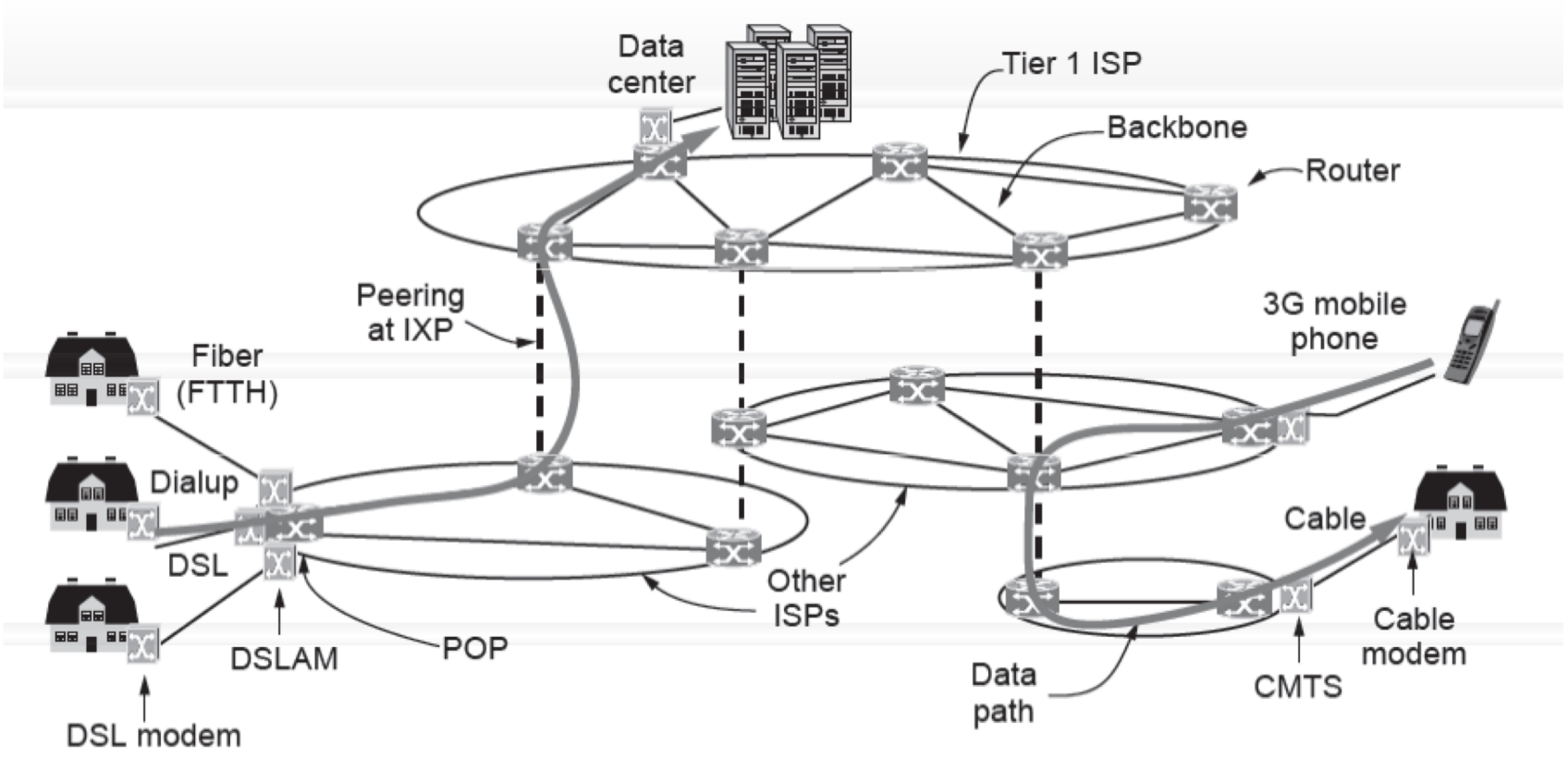
The modern Internet is more complex:

- ISP(Internet Service Provider) networks serve as the Internet backbone
- the user purchases Internet access or connectivity from an ISP
- The ISP's POP (Point of Presence): the location at which customer packets enter the ISP network
- Last leg of transmission: from user to ISP POP
- Hosts connect to ISP at the edge through (Limited to last mile BW):
  - Phone/Cable: max 64kbps
    - **Phone** lines (phone company as the ISP): **DSL** (DSLAM-Digital Subscriber Line Access Multiplexer), **Dial up** (MODEM: modulator demodulator)
    - **Cable TV system** (CMTS-Cable Modem Termination System)
    - Both reuse an existing infrastructure as last leg of transmission
  - **Broadband-**10 to 100 Mbps
    - FTTH (Fiber to the Home)
    - T3 lease lines (from office to nearest ISP)

Internet

- **POP**- Point of Presence: location at which customer packets enter the ISP network. From this point on, the system is fully digital and packet switched.
- ISPs connect or **peer** to exchange traffic at **IXPs** (Internet eXchange Points)
- Within each network routers switch packets
- Between networks, traffic exchange is set by business agreements in IXPs
- IXPs: root of routers, at least one per ISP – forming a LAN – (BGP routing)
- **The path a packet takes through the Internet depends on the peering choices of the ISPs.**
- Data centers (google) concentrate many servers ("the cloud") –with good connection to the internet at ISP POPs so fast connections can be made between the servers and the ISP backbones
- Most Internet traffic is content from data centers (esp. video)
- The Internet architecture continues to evolve
- **Intranets**: some companies have interconnected all their existing internal networks, often using the same technology as the Internet.
    - Typically accessible only on company premises or from company notebooks but otherwise work the same way as the Internet.

Internet



Architecture of the Internet

Network Standardization

Standards define what is needed for interoperability

Some of the many standards bodies:

Body	Area	Examples
ITU	Telecommunications	G.992, ADSL H.264, MPEG4
IEEE	Communications	802.3, Ethernet 802.11, WiFi
IETF	Internet	RFC 2616, HTTP/1.1 RFC 1034/1035, DNS
W3C	Web	HTML5 standard CSS standard