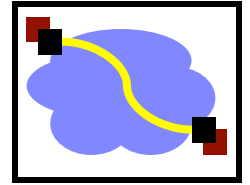# 416 Distributed Systems

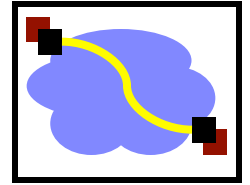Networks review; Day 1 of 2

Jan 6, 2017

Distributed Systems vs. Networks

- Low level (c/go)
- Run forever
- Support others
- Adversarial environment
- Distributed & concurrent
- Resources matter

- And have it implemented/run by vast numbers of different people with different goals/skills

Keep an eye out for…

- Modularity, Layering, and Decomposition:
  - Techniques for dividing the work of building systems
  - Hiding the complexity of components from each other
  - Hiding implementation details to deal with heterogeneity
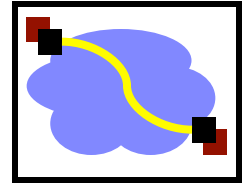- Naming/lookup/routing
- Resource sharing and isolation

- Models and assumptions about the environment and components

Today's Lecture
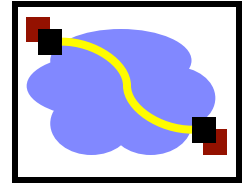
- Network links and LANs

- Layering and protocols

- Internet design

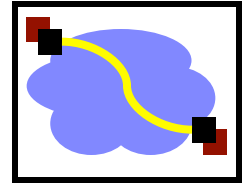# Basic Building Block: Links

Node   Link   Node

- Electrical questions
  - Voltage, frequency, …
  - Wired or wireless?

- Link-layer issues: How to send data?
  - When to talk – can either side talk at once?
  - What/how to say – low-level format?
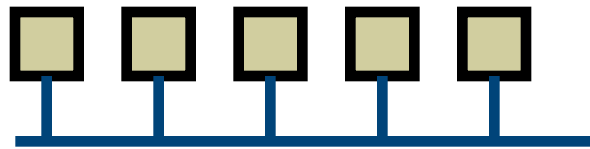
Model of a communication channel
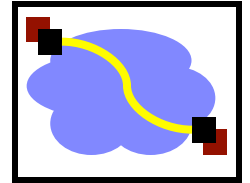
- Latency - how long does it take for the first bit to reach destination

- Jitter - how much variation in latency?

- Capacity - how many bits/sec can we push through? (often termed "bandwidth")

- Loss / Reliability - can the channel drop packets?
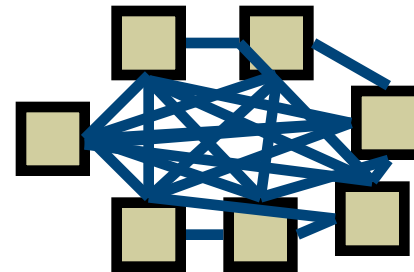
- Reordering

Basic Building Block: Links
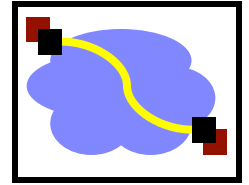
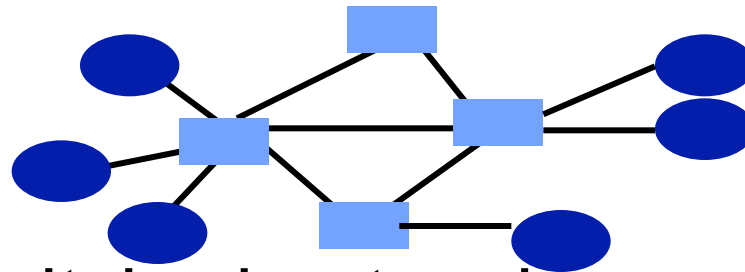- … But what if we want more hosts?

One wire

Wires for everybody!

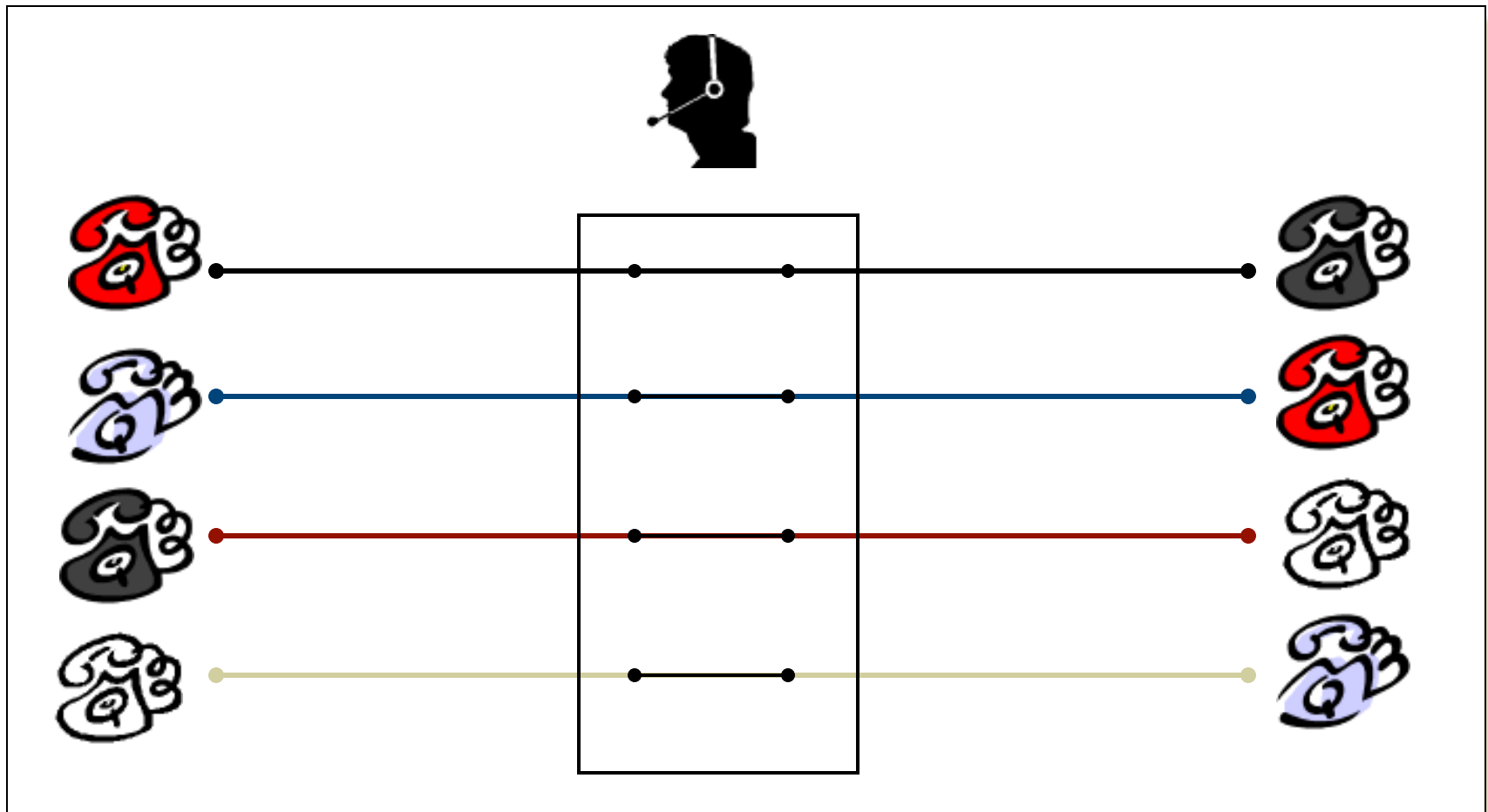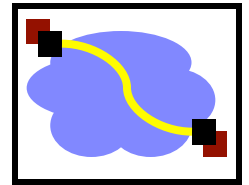- Scalability?!

Multiplexing
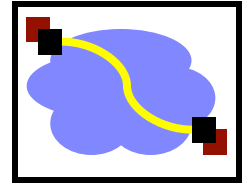
- Need to share network resources

- How? Switched network
  - Party "A" gets resources sometimes
  - Party "B" gets them sometimes
- Interior nodes act as "Switches"

- What mechanisms to share resources?

# In the Old Days…Circuit Switching

Packet Switching

- Source sends information as self-contained packets that have an address.
    - Source may have to break up single message in multiple

- Each packet travels independently to the destination host.
    - Switches use the address in the packet to determine how to forward the packets
    - Store and forward

- Analogy: a letter in surface mail.
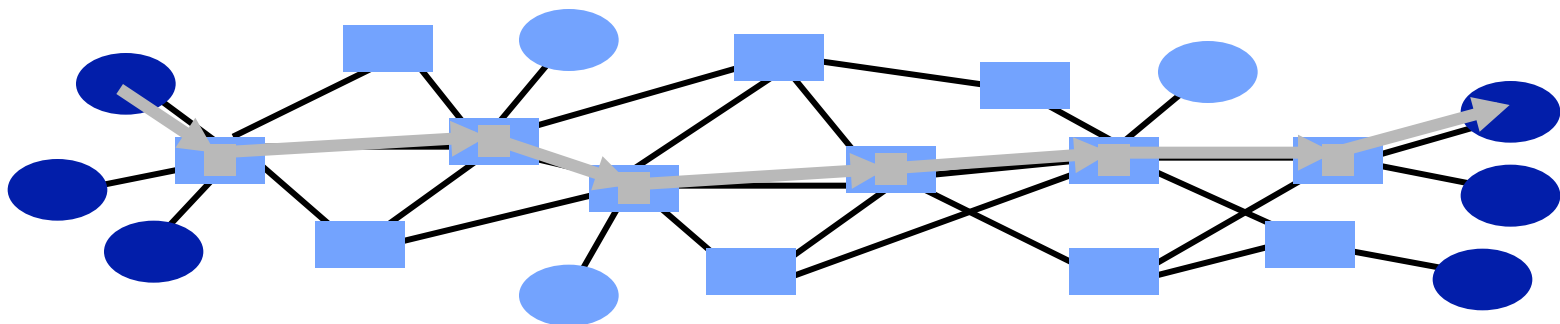
Packet Switching – Statistical Multiplexing

Packets
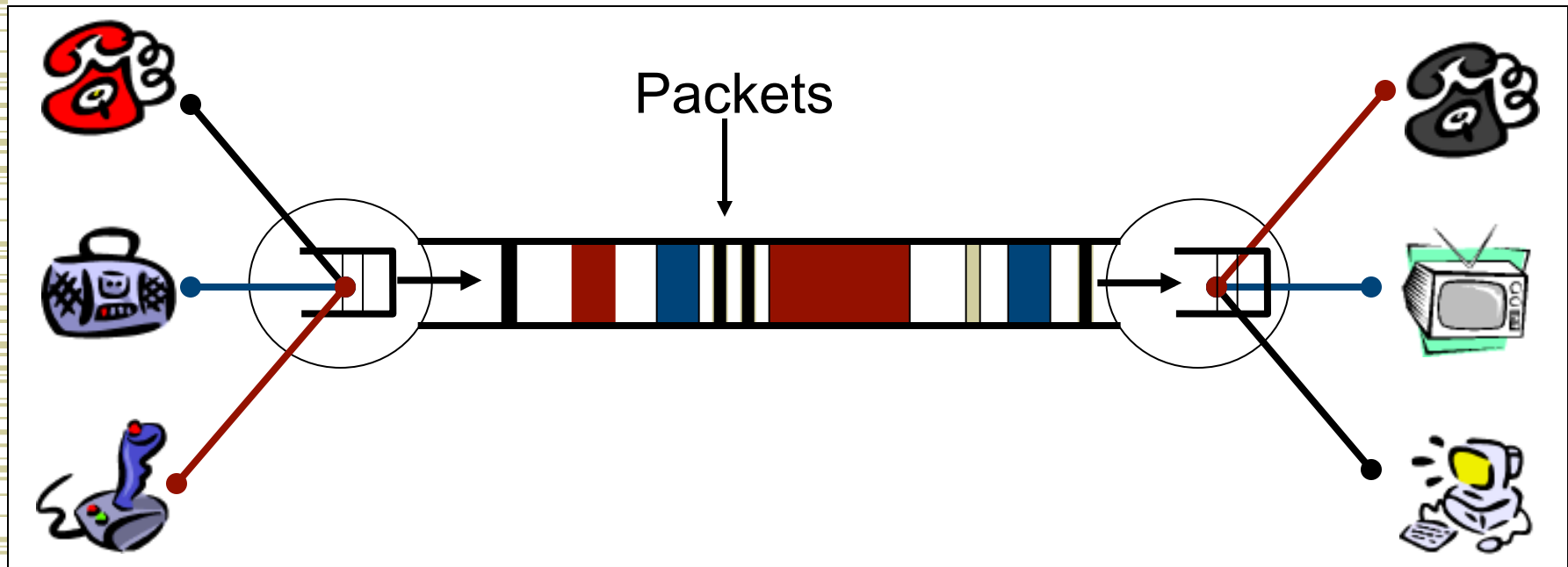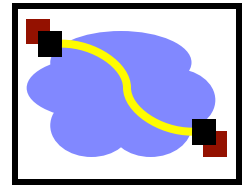
- Switches arbitrate between inputs
- Can send from *any* input that's ready
  - Links never idle when traffic to send
  - (Efficiency!)

What if Network is Overloaded?

Problem: Network Overload
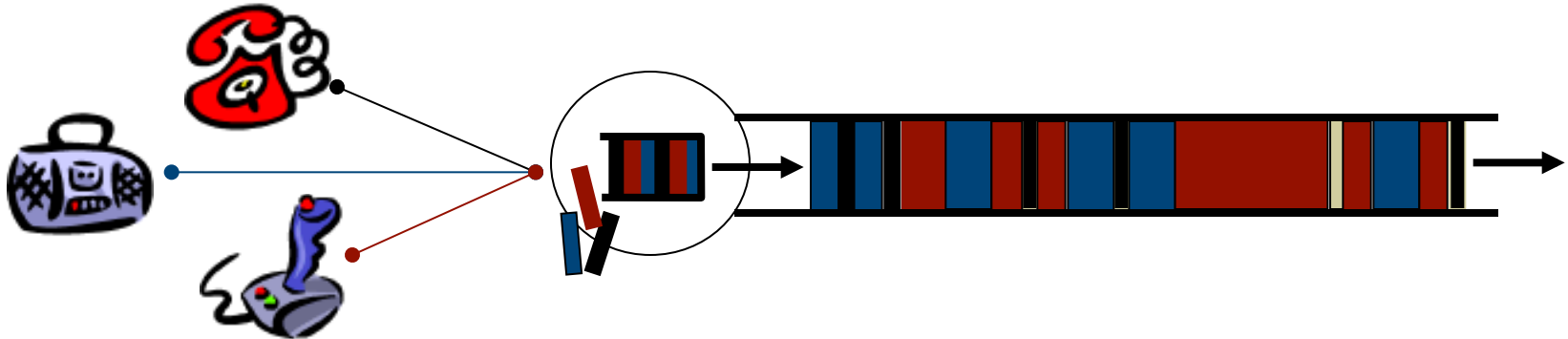
Solution: Buffering and Congestion Control

- Short bursts: buffer
- What if buffer overflows?
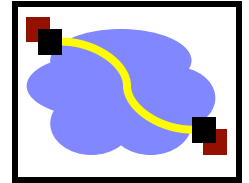 - Packets dropped
  - Sender adjusts rate until load = resources → "congestion control"
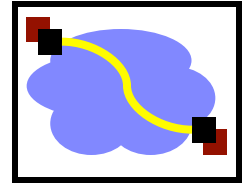
Example: Ethernet Packet

- Sending adapter encapsulates IP datagram (or other network layer protocol packet) in Ethernet frame

Preamble	Dest. Address	Source Address		Data	CRC

↑ Type
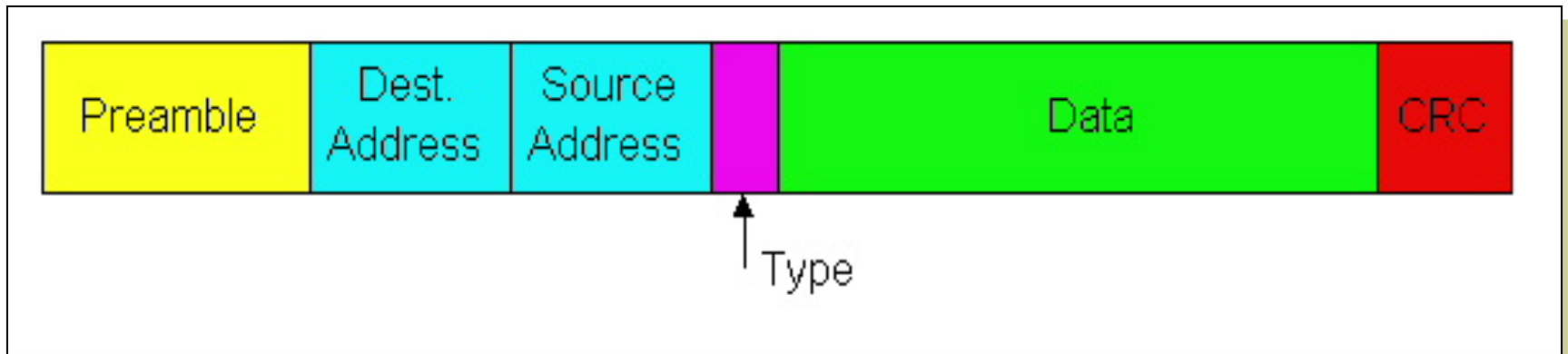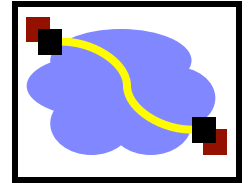
Ethernet Frame Structure

- Each protocol layer needs to provide some hooks to upper layer protocols
  - Demultiplexing: identify which upper layer protocol packet belongs to
  - E.g., port numbers allow TCP/UDP to identify target application
  - Ethernet uses Type field
- Type: 2 bytes
  - Indicates the higher layer protocol, mostly IP but others may be supported such as Novell IPX and AppleTalk
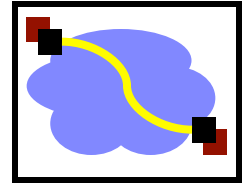
Ethernet Frame Structure (cont.)

- **Addresses:** 6 bytes
  - Each adapter is given a globally unique address at manufacturing time
    - Address space is allocated to manufacturers
      - 24 bits identify manufacturer
      - E.g., 0:0:15:* → 3com adapter
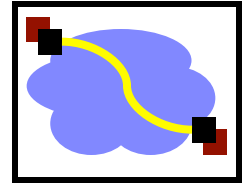   - Frame is received by all adapters on a LAN and dropped if address does not match
  - Special addresses
    - Broadcast – FF:FF:FF:FF:FF:FF is "everybody"
    - Range of addresses allocated to multicast
      - Adapter maintains list of multicast groups node is interested in

Packet Switching

- Source sends information as self-contained packets that have an address.
  - Source may have to break up single message in multiple

- Each packet travels independently to the destination host.
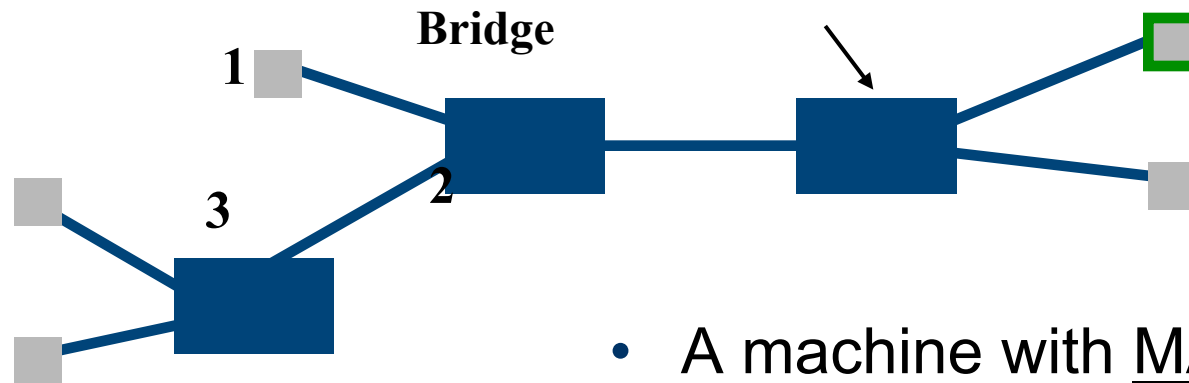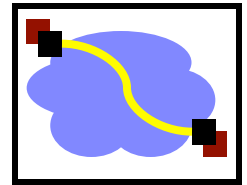 - Switches use the address in the packet to determine how to forward the packets
  - Store and forward

- Analogy: a letter in surface mail.

Frame Forwarding

Bridge

1

3

2
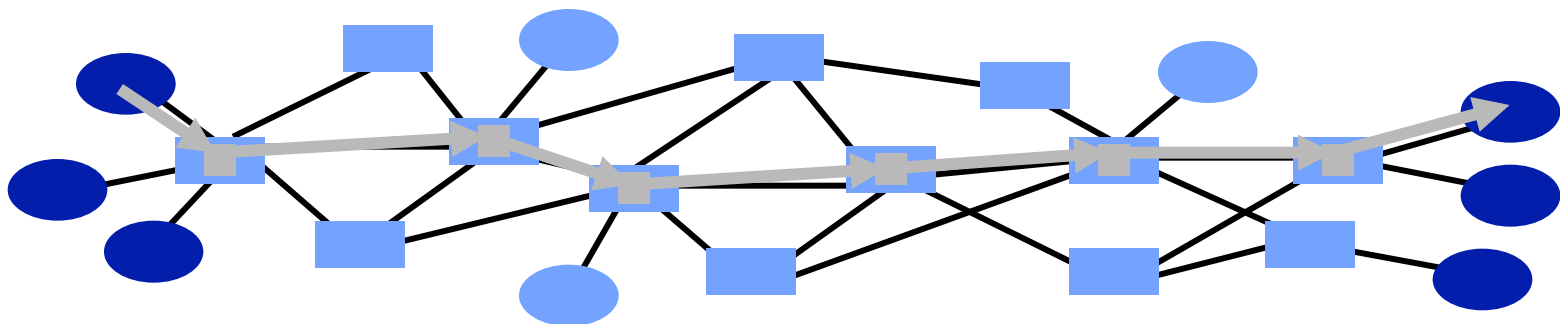
MAC Address	Port	Age
A21032C9A591	1	36
99A323C90842	2	01
8711C98900AA	2	15
301B2369011C	2	16
69551001190	3	11

- A machine with <u>MAC Address</u> lies in the direction of number <u>port</u> of the bridge

- For every packet, the bridge "looks up" the entry for the packet's destination MAC address and forwards the packet on that port.
  - Other packets are broadcast – why?

- Timer is used to flush old entries

Learning Bridges
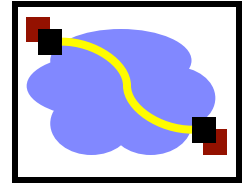
- Manually filling in bridge tables?
  - Time consuming, error-prone
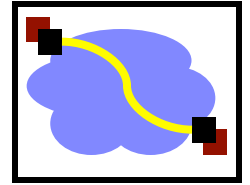- Keep track of source address of packets arriving on every link, showing what segment hosts are on
  - Fill in the forwarding table based on this information
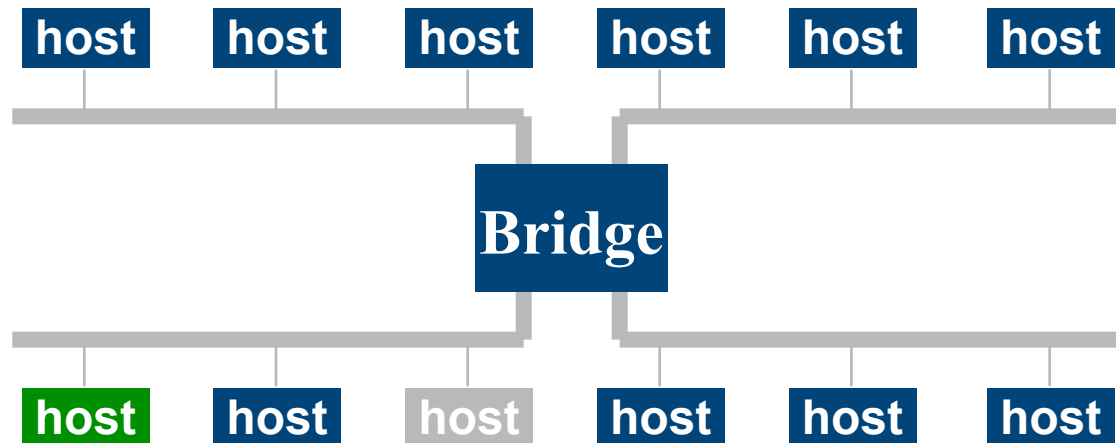
Today's Lecture

- Network links and LANs

- Layering and protocols

- Internet design

Internet

- An inter-net: a network of networks.
  - Networks are connected using routers that support communication in a hierarchical fashion
  - Often need other special devices at the boundaries for security, accounting, ..

- The Internet: the interconnected set of networks of the Internet Service Providers (ISPs)
  - About 17,000 different networks make up the Internet
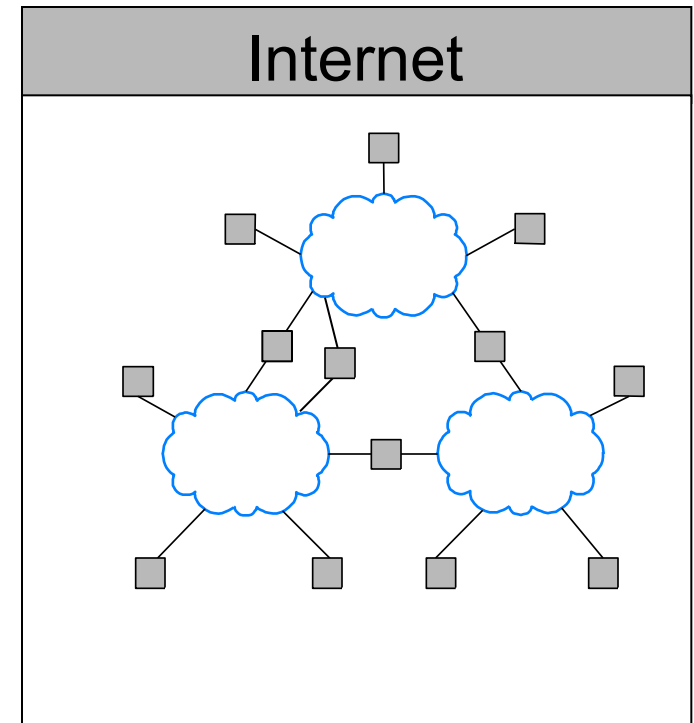
Internet

Challenges of an internet
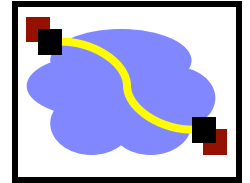
- Heterogeneity
    - Address formats
    - Performance – bandwidth/latency
    - Packet size
    - Loss rate/pattern/handling
    - Routing
    - Diverse network technologies → satellite links, cellular links, carrier pigeons
    - In-order delivery

How To Find Nodes?
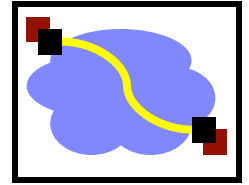
Computer 1   **Internet**   **Computer 2**

Need naming and routing

Naming

What's the IP address for www.cmu.edu?

→

It is 128.2.11.43

←

Computer 1   **Local DNS Server**

Translates human readable names to logical endpoints

Routing

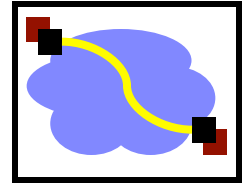Routers send packet towards destination

H: Hosts

R: Routers
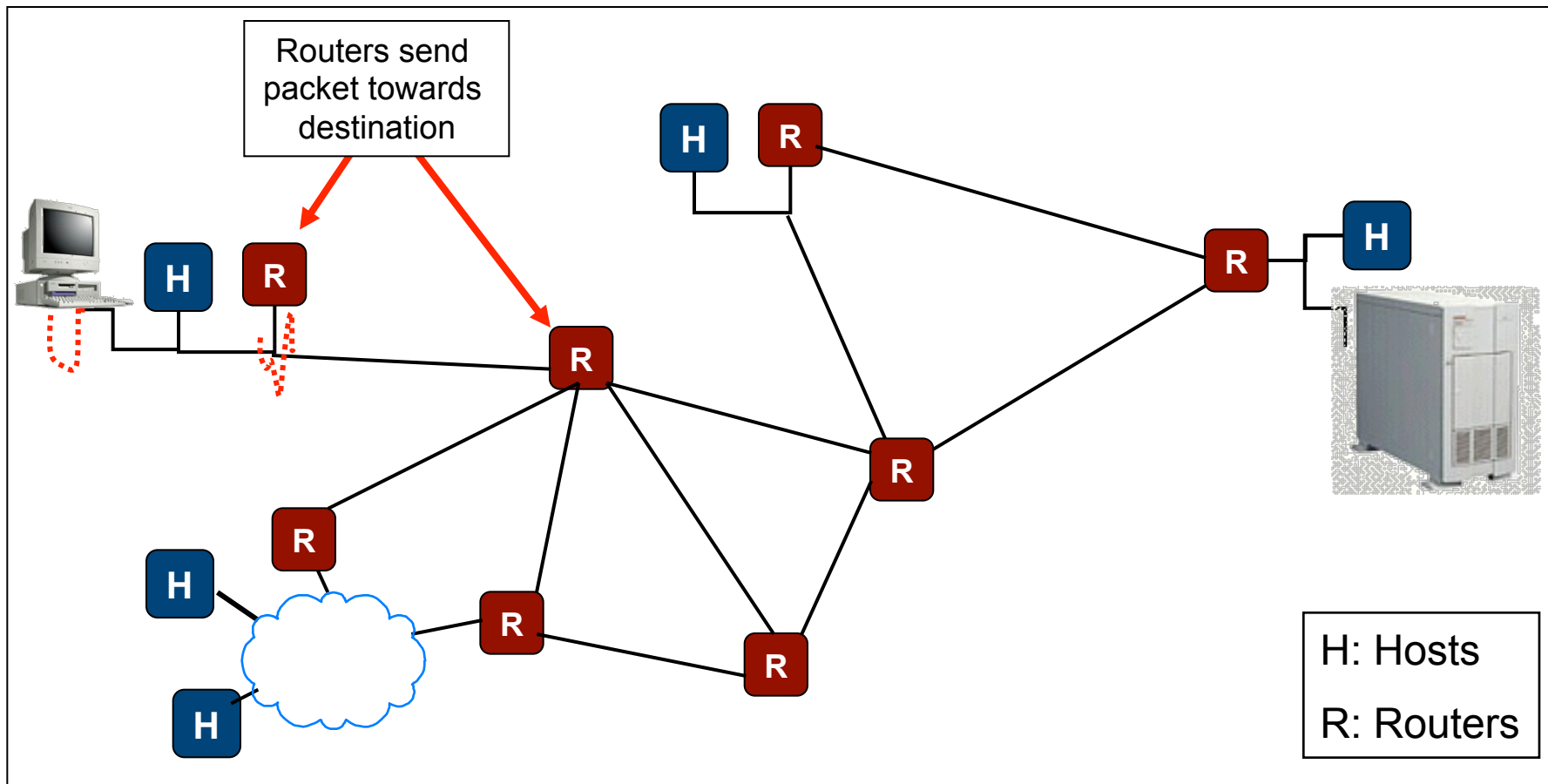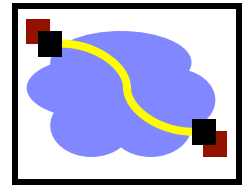
Network Service Model
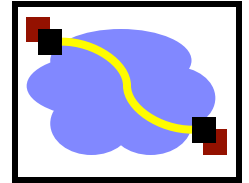
- ## What is the *service model?*
  - ### Ethernet/Internet: *best-effort* – packets can get lost, etc.
- ## What if you want more?
  - ### Performance guarantees (QoS)
  - ### Reliability
    - Corruption
    - Lost packets
  - ### Flow and congestion control
  - ### Fragmentation
  - ### In-order delivery
  - ### Etc…
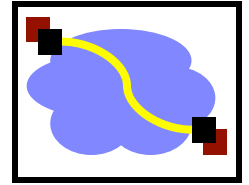
Failure models

- Fail-stop:
  - When something goes wrong, the process stops / crashes / etc.
- Fail-slow or fail-stutter:
  - Performance may vary on failures as well
- Byzantine:
  - Anything that can go wrong, will.
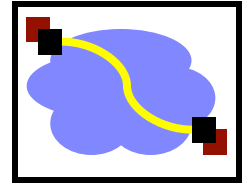 - Including malicious entities taking over your computers and making them do whatever they want.
- These models are useful for proving things;
- The real world typically has a bit of everything.

- Deciding which model to use is important!

Fancier Network Service Models

- What if network had reliable, in-order, mostly no-corruption, stream-oriented communication (i.e. TCP)

- Programmers don't have to implement these features in every application

- But note limitations: this can't turn a byzantine failure model into a fail-stop model...

What if the Data gets Corrupted?

Problem: Data Corruption

GET index.html

Internet

GET inrex.html

Solution: Add a *checksum*

0,9	9
6,7,8	21
4,5	7
1,2,3	6
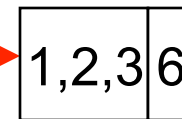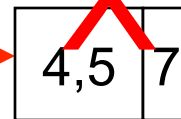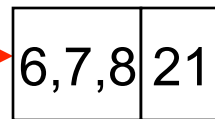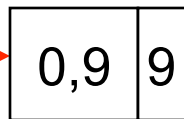
What if the Data gets Lost?

Problem: Lost Data

GET index.html

Internet

Solution: Timeout and Retransmit
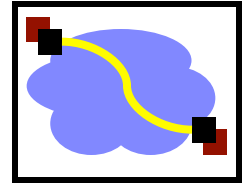
GET index.html

GET index.html

Internet

GET index.html

What if the Data is Out of Order?

Problem: Out of Order

ml	→	inde	→	x.ht	→	GET

GET x.htindeml

Solution: Add Sequence Numbers

ml	4	→	inde	2	→	x.ht	3	→	GET	1

GET index.html

Networks [including end points] Implement Many Functions

- Link

- Multiplexing

- Routing

- Addressing/naming (locating peers)

- Reliability

- Flow control

- Fragmentation

- Etc….

What is Layering?
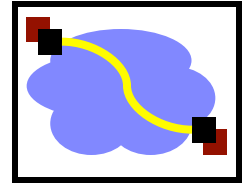
- Modular approach to network functionality

- Example:

Application
Application-to-application channels
Host-to-host connectivity
Link hardware

What is Layering?

Peer Layer ⬌ Peer Layer

User A   User B

Application

Transport

Network

Link

Host   Host

Modular approach to network functionality
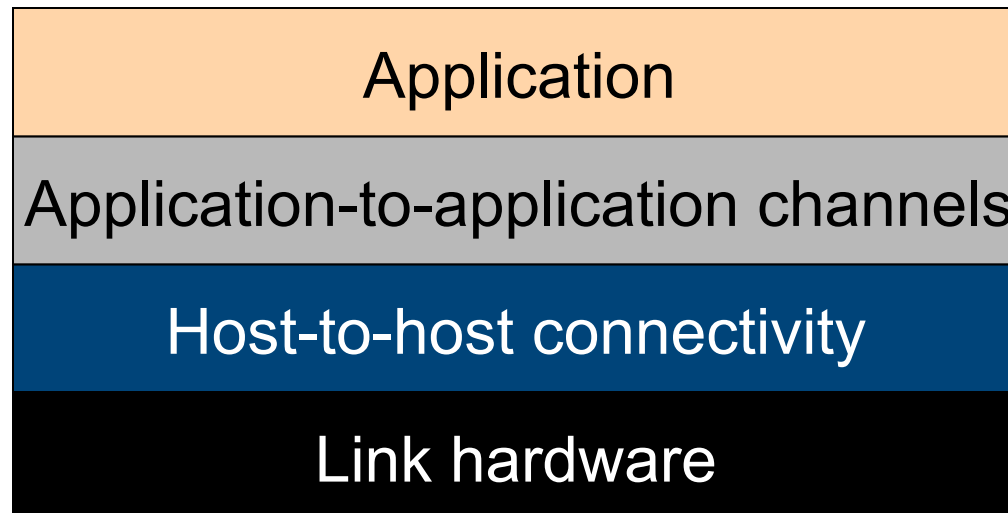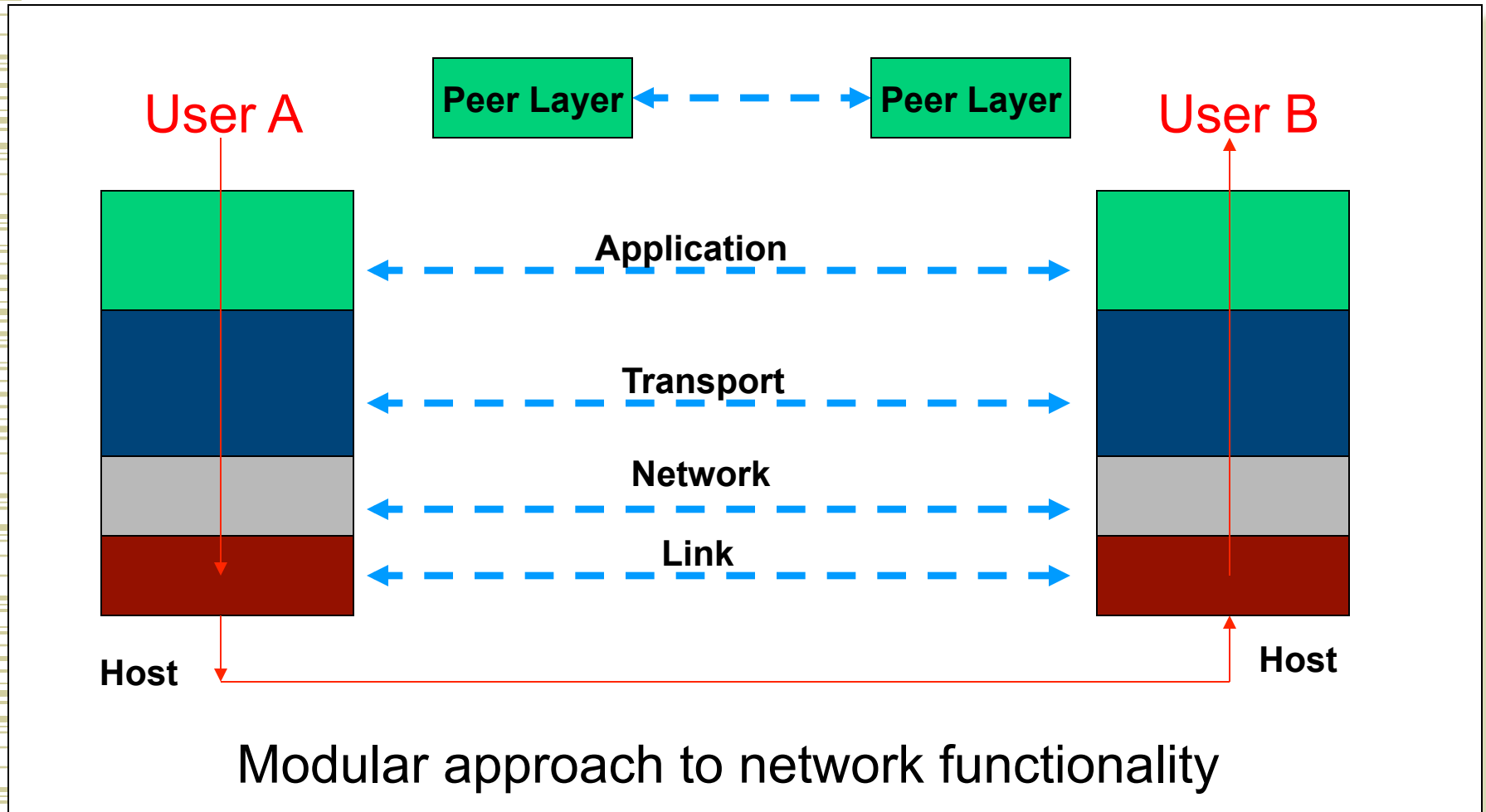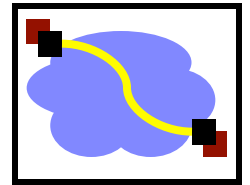
Layering Characteristics

- Each layer relies on services from layer below and exports services to layer above

- Interface defines interaction with peer on other hosts

- Hides implementation - layers can change without disturbing other layers (black box)

What are Protocols?

- An agreement between parties on how communication should take place

- Module in layered structure

- Protocols define:
  - Interface to higher layers (API)
  - Interface to peer (syntax & semantics)
    - Actions taken on receipt of a messages
    - Format and order of messages
    - Error handling, termination, ordering of requests, etc.

- Example: Buying airline ticket

Friendly greeting

Muttered reply

Destination?

Honolulu

Thank you
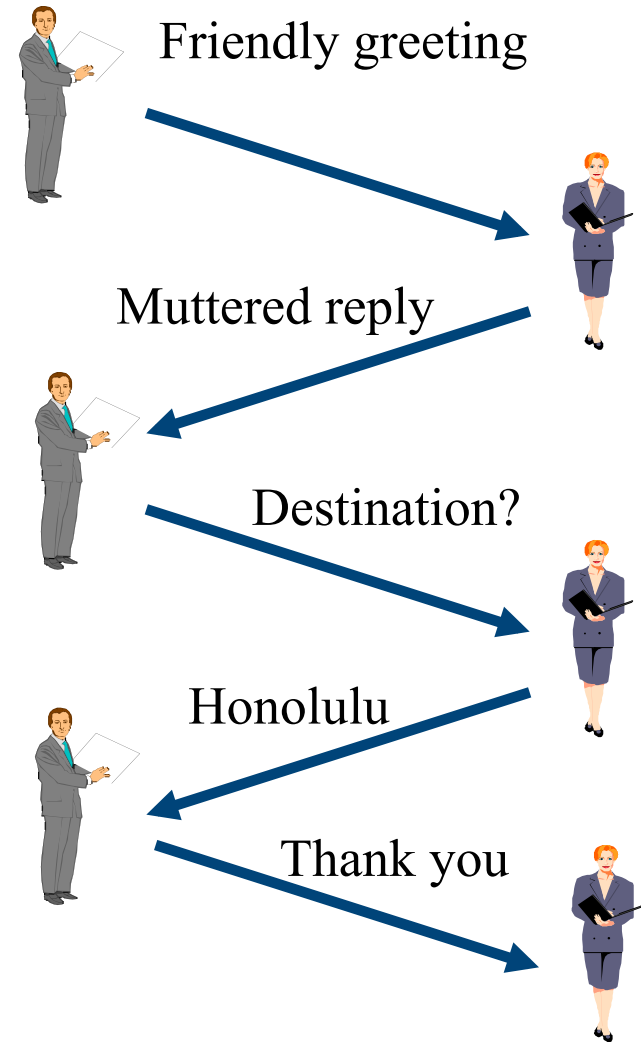
IP Layering

- ## Relatively simple

Application			
Transport			
Network			
Link			
Physical			
Host	**Bridge/Switch**	**Router/Gateway**	**Host**
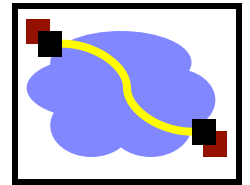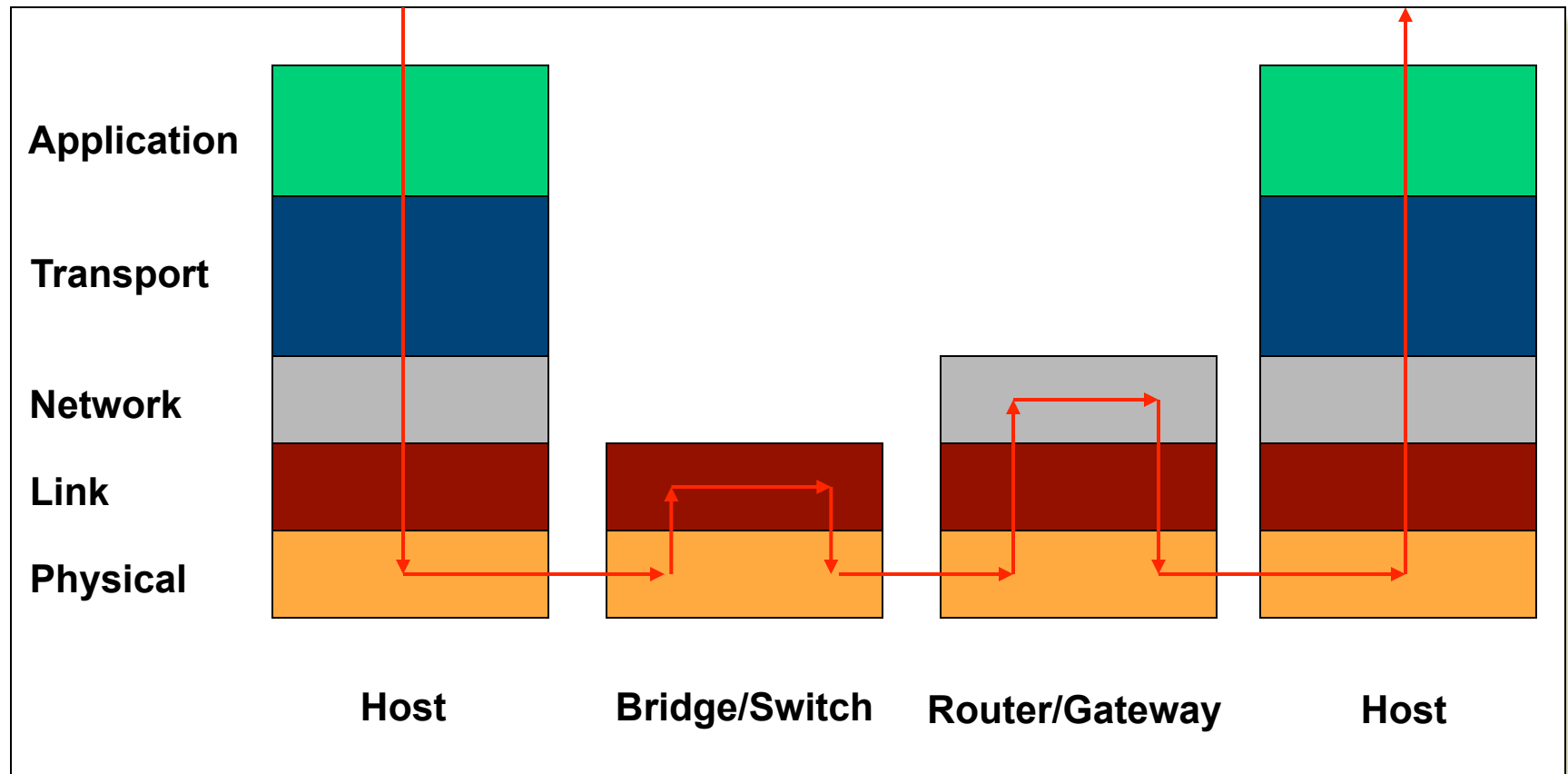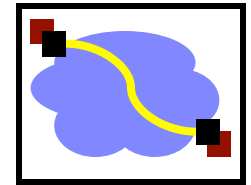
The Internet Protocol Suite

FTP	HTTP	NV	TFTP

TCP UDP

IP → **Waist**

NET$_1$ NET$_2$ … NET$_n$

Applications

UDP TCP

Data Link

Physical

The Hourglass Model
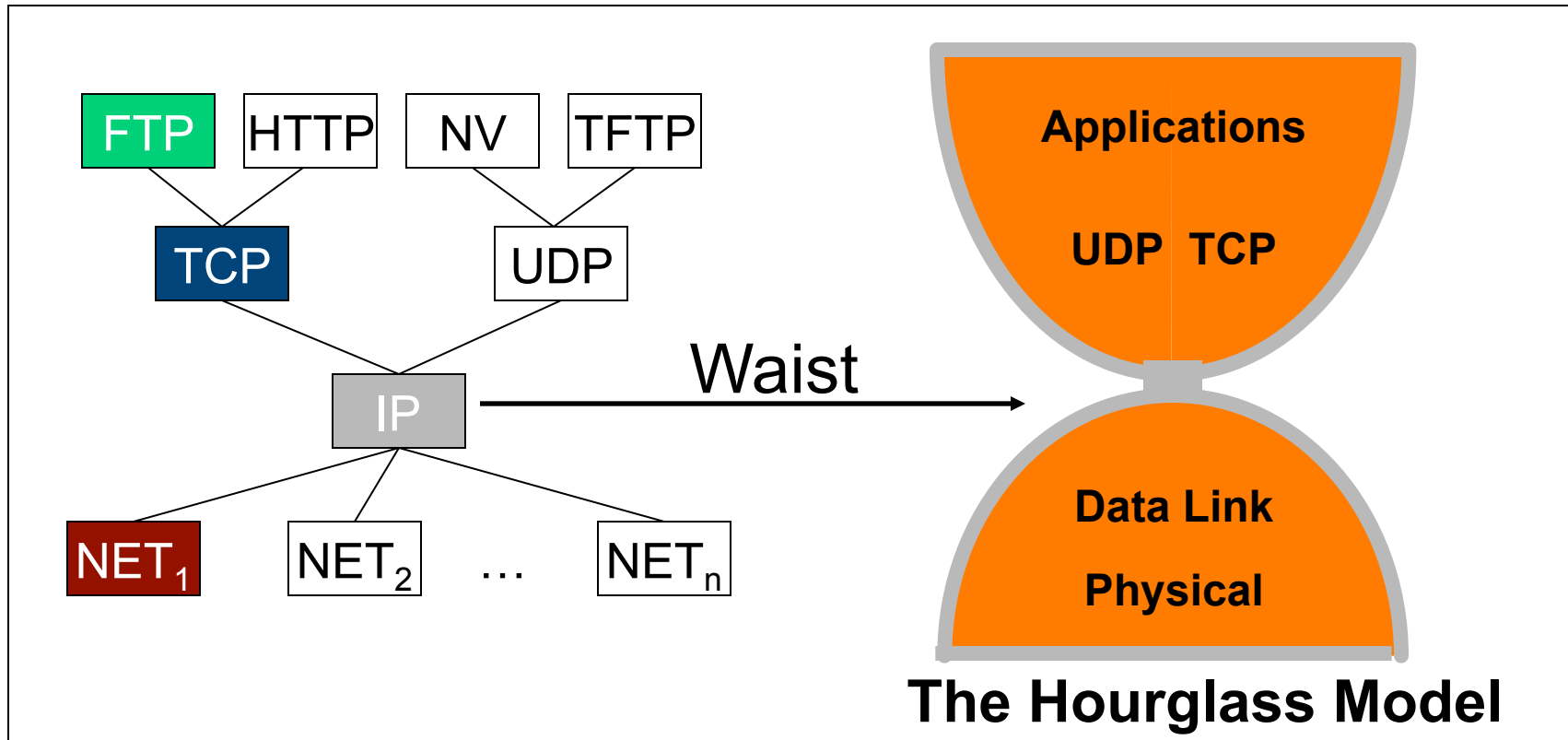
The "thin" waist facilitates interoperability

Layer Encapsulation

User A

User B

Get index.html

Connection ID

Source/Destination

Link Address
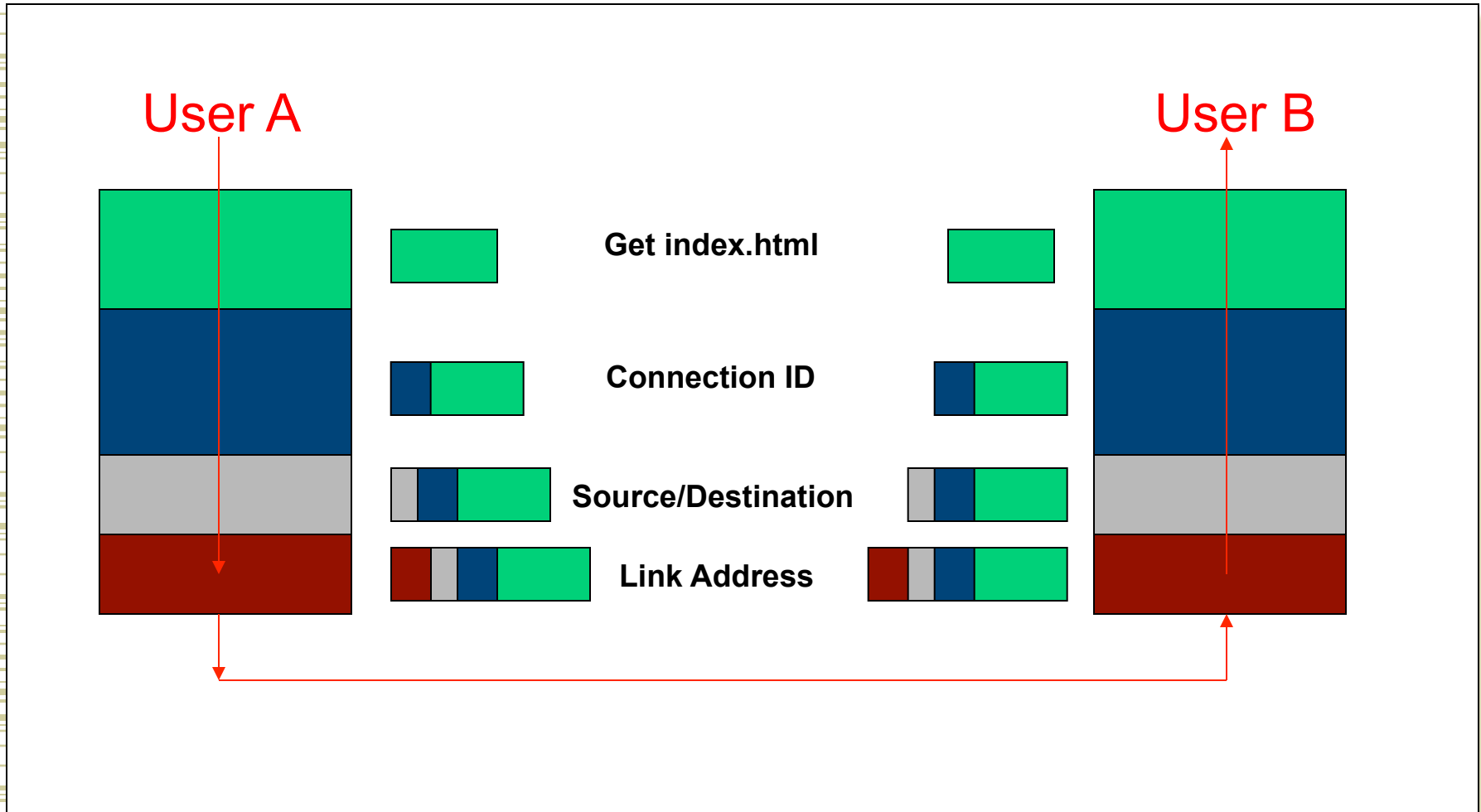
Multiplexing and Demultiplexing
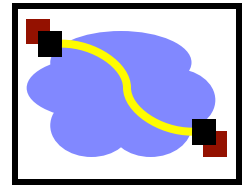
- There may be multiple implementations of each layer.
  - How does the receiver know what version of a layer to use?
- Each header includes a demultiplexing field that is used to identify the next layer.
  - Filled in by the sender
  - Used by the receiver
- Multiplexing occurs at multiple layers. E.g., IP, TCP, …
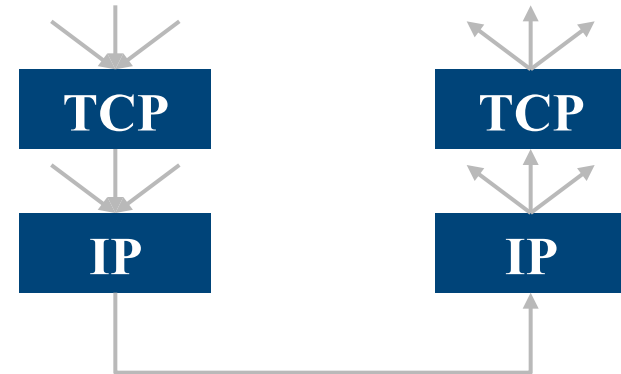
TCP				TCP	
IP				IP	

V/HL	TOS	Length	
ID		Flags/Offset	
TTL	Prot.	H. Checksum	
Source IP address			
Destination IP address			
Options..			

Multiplexing and Demultiplexing
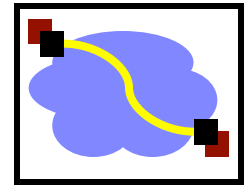
List of IP protocol numbers

From Wikipedia, the free encyclopedia

This is a list of IP numbers used in the *Protocol* field of the IPv4 hea

| Decimal	Hex	Keyword	
0	0x00	HOPOPT	IPv6 Hop-by-Hop Option
1	0x01	ICMP	Internet Control Message Prot
2	0x02	IGMP	Internet Group Management F
3	0x03	GGP	Gateway-to-Gateway Protocol
4	0x04	IP-in-IP	IP in IP (encapsulation)
5	0x05	ST	Internet Stream Protocol
6	0x06	TCP	Transmission Control Protocol
7	0x07	CBT	Core-based trees
8	0x08	EGP	Exterior Gateway Protocol
9	0x09	IGP	Interior Gateway Protocol (any their IGRP))
10	0x0A	BBN-RCC-MON	BBN RCC Monitoring
11	0x0B	NVP-II	Network Voice Protocol
12	0x0C	PUP	Xerox PUP
13	0x0D	ARGUS	ARGUS
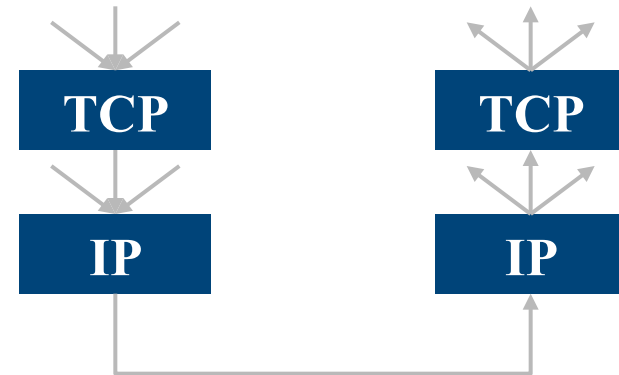
V/HL	TOS	Length	
ID		Flags/Offset	
TTL	Prot.	H. Checksum	
Source IP address			
Destination IP address			
Options..			

40

Protocol Demultiplexing
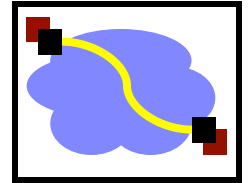
- Multiple choices at each layer

FTP	HTTP		NV	TFTP

```
    FTP   HTTP      NV    TFTP
      \   /          \    /
       TCP            UDP
          \          /
    IPX      IP
           / |  \
      NET₁  NET₂  …  NETₙ
```

Network | IP | TCP/UDP

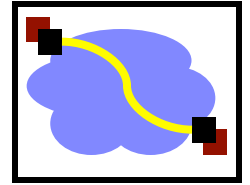Type Field   Protocol Field   Port Number
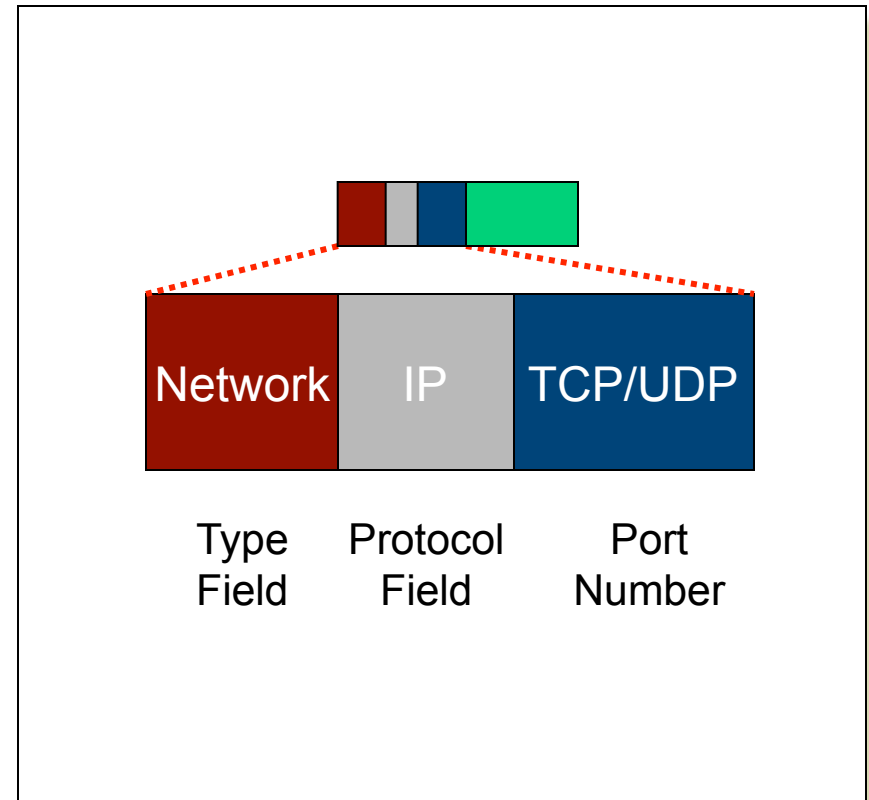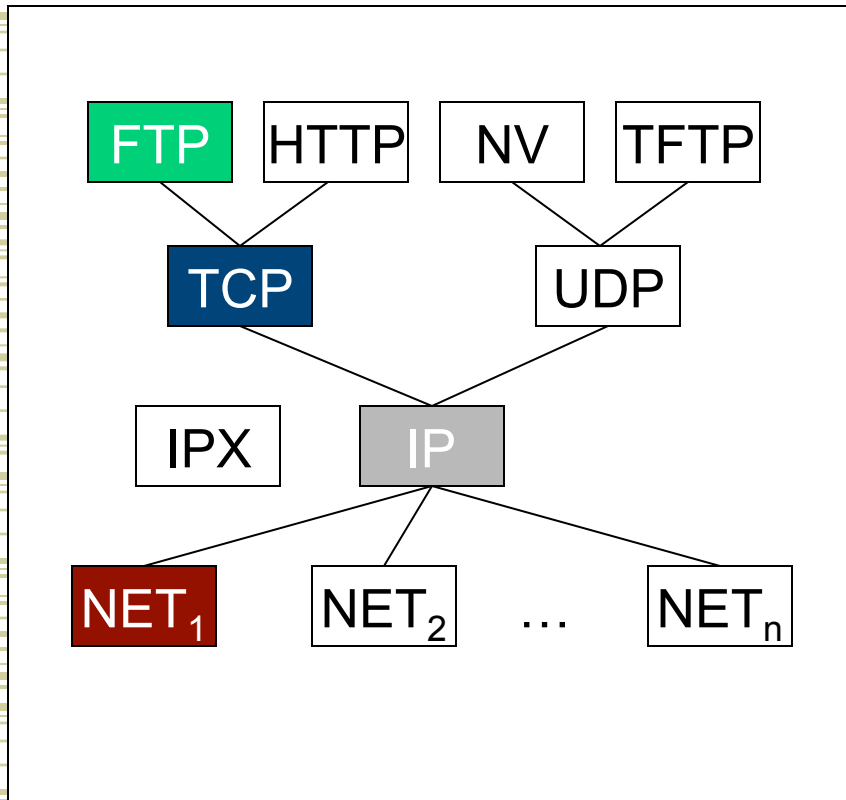
Today's Lecture

- Network links and LANs

- Layering and protocols

- Internet design

Goals [Clark88]

0 **Connect existing networks**

   initially ARPANET and ARPA packet radio network

1. Survivability

   ensure communication service even in the presence of network and router failures

2. Support multiple types of services
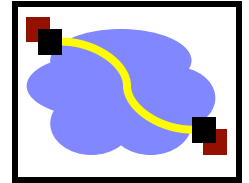
3. Must accommodate a variety of networks

4. Allow distributed management

5. Allow host attachment with a low level of effort

6. Be cost effective

7. Allow resource accountability

Goal 1: Survivability

- If network is disrupted and reconfigured…
  - Communicating entities should not care!
  - No higher-level state reconfiguration

- How to achieve such reliability?
  - Where can communication state be stored?

	Network	Host
Failure handing	Replication	"Fate sharing"
Net Engineering	Tough	Simple
Routing state	Maintain state	Stateless
Host trust	Less	More