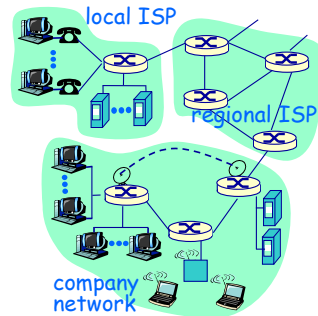


A Whirlwind Introduction to the Internet Overview

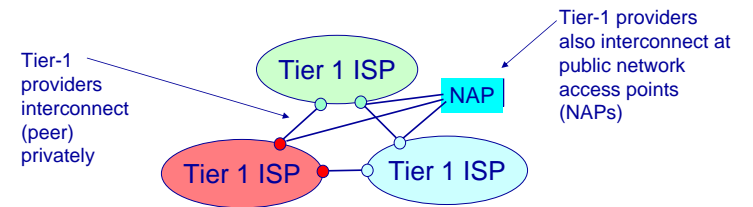
- ◆ What's the Internet
- ◆ Network core
- ◆ Network edge
- ◆ Access nets, physical media
- ◆ Internet Structure & ISPs
- ◆ Performance: loss, delay
- ◆ Protocol layers, service models



1

Internet Structure: Network of Networks

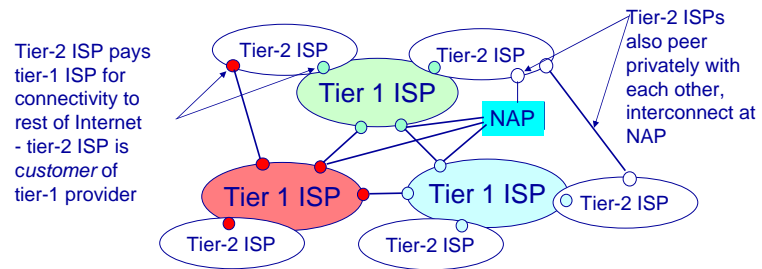
- ◆ Roughly Hierarchical
- ◆ At center: "tier-1" ISPs (e.g., UUNet, Level 3, Sprint, AT&T), national/international coverage
 - » treat each other as equals



2

Internet Structure: Network of Networks

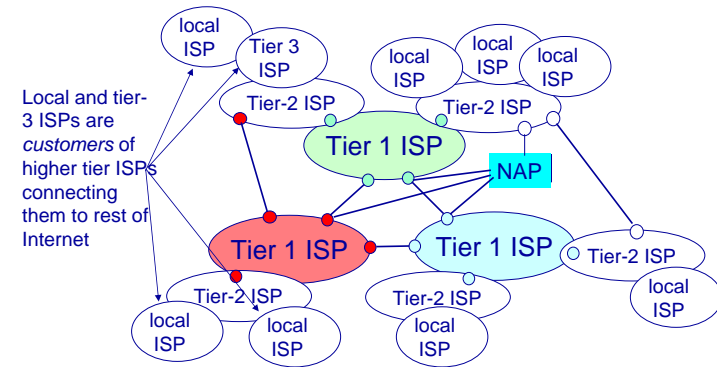
- ◆ “Tier-2” ISPs: smaller (often regional) ISPs
 - » Connect to one or more tier-1 ISPs, possibly other tier-2 ISPs



3

Internet Structure: Network of Networks

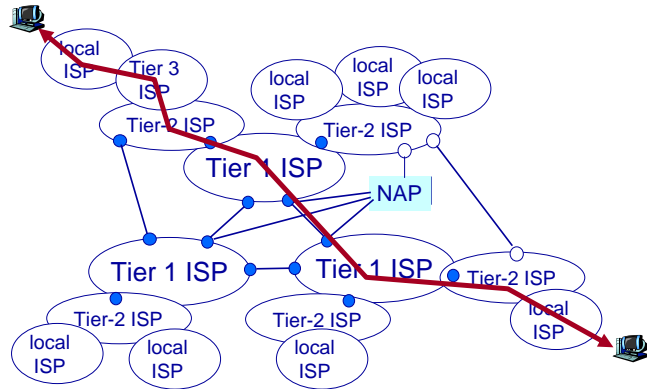
- ◆ “Tier-3” ISPs and local ISPs
 - » last hop (“access”) network (closest to end systems)



4

Internet Structure: Network of Networks

- ◆ A packet passes through many networks!



5

Network Maps

Just how big are Tier-1 ISPs...?

6

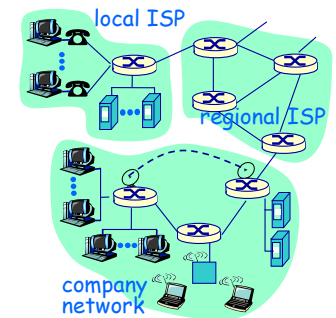
Network Maps

Just how big is the Internet...?

15

A Whirlwind Introduction to the Internet Overview

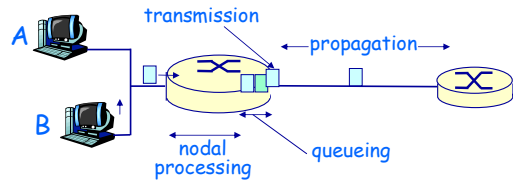
- ◆ What's the Internet
- ◆ Network core
- ◆ Network edge
- ◆ Access nets, physical media
- ◆ Internet Structure & ISPs
- ◆ Performance: loss, delay
- ◆ Protocol layers, service models



18

Understanding the Performance of the Internet

Delay in packet-switched networks

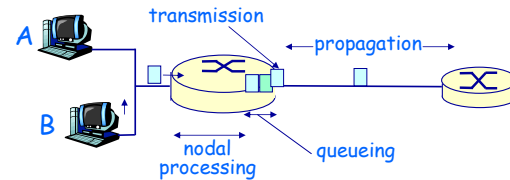


- ◆ Packets experience variable delays along path from source to destination
- ◆ Four sources of delay at *each* hop
 - » Nodal processing:
 - ❖ Check for bit errors
 - ❖ Determine the output interface to forward packet on
 - » Queuing:
 - ❖ Time spent waiting at outbound interface for transmission
 - ❖ Duration depends on the level of congestion at the interface
 - » Transmission
 - » Propagation

19

Understanding the Performance of the Internet

Delay in packet-switched networks



Beware: s and R are very different quantities!

- ◆ Transmission delay = time to “put bits onto the link” = L/R
 - » R = link bandwidth (bps)
 - » L = packet length (bits)
- ◆ Propagation delay = d/s
 - » d = length of physical link
 - » s = signal propagation speed in medium ($\sim 2 \times 10^8$ m/sec)

20

Transmission Delay

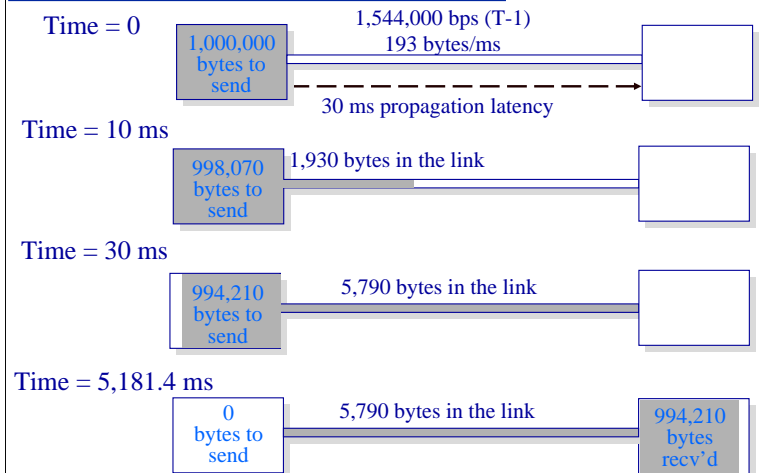
Telecommunications transmission speed alphabet soup

- ◆ DS-1/T-1 = 1.544 Mbps
- ◆ DS-3/T-3 = 44.736 Mbps
- ◆ OC-1 = 51.84 Mbps
- ◆ OC- n = $n \times$ OC-1
 - » OC-3 = $3 \times$ OC-1 (155.52 Mbps)
 - » OC-12 = $12 \times$ OC-1 (622.08 Mbps)
 - » OC-48 = $48 \times$ OC-1 (2,488.32 Mbps/2.5 Gbps)
 - » OC-192 = $192 \times$ OC-1 (9,953.28 Mbps/10 Gbps)

21

Transmission & Propagation Example

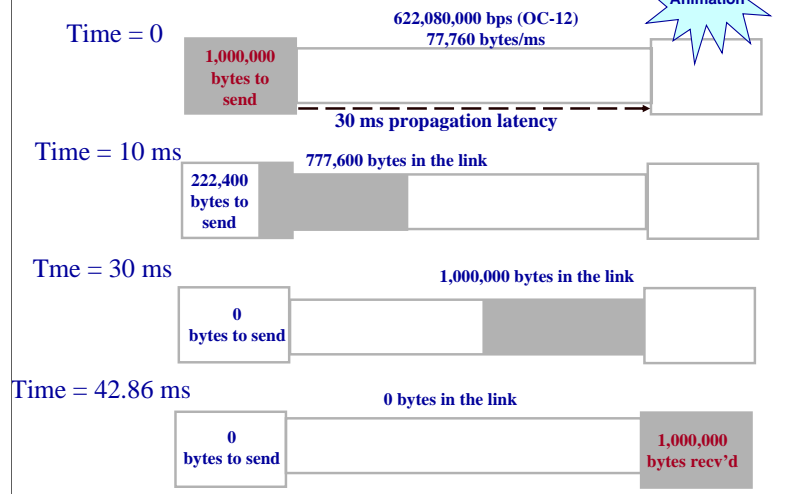
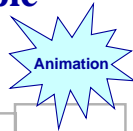
Transmission on a "slow" link



22

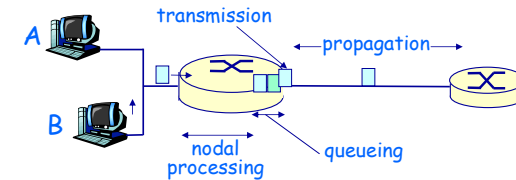
Transmission & Propagation Example

Transmission on a "fast" link



Understanding the Performance of the Internet

Delay in packet-switched networks



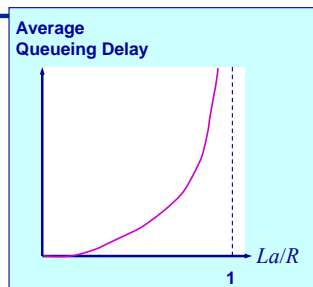
- ◆ Typical transmission delay: 120 μ s
 - » 1,500 byte packet on a 100 Mbps Ethernet
- ◆ Typical processing delay:
 - » ??
- ◆ Typical propagation delay:
 - » $\leq 1 \mu$ s on a small campus
 - » ≈ 25 -30 ms to the West coast (and back)
- ◆ Typical queuing delay:
 - » ??

Delay in Packet-Switched Networks

Queuing delay & traffic intensity

- ◆ Understand queuing delay in terms of *traffic intensity* La/R

- » R = link transmission speed (bps)
- » L = packet length (bits/packet)
- » a = average packet arrival rate (packets/second)



- ◆ If $La/R \sim 0$: Average queuing delay small
- ◆ As $La/R \Rightarrow 1$: Delays become large
- ◆ If $La/R > 1$: Work arrives faster than it can be serviced
 - » Average delay goes to infinity (with infinite buffers)!
 - » With finite buffers???

25

Packet loss

- ◆ Queue (buffer) preceding link in buffer has finite capacity
- ◆ When packet arrives to full queue, packet is dropped (lost)
- ◆ Lost packet may be retransmitted by source end system (connection-oriented like TCP), or not retransmitted at all (connection-less like UDP).



26

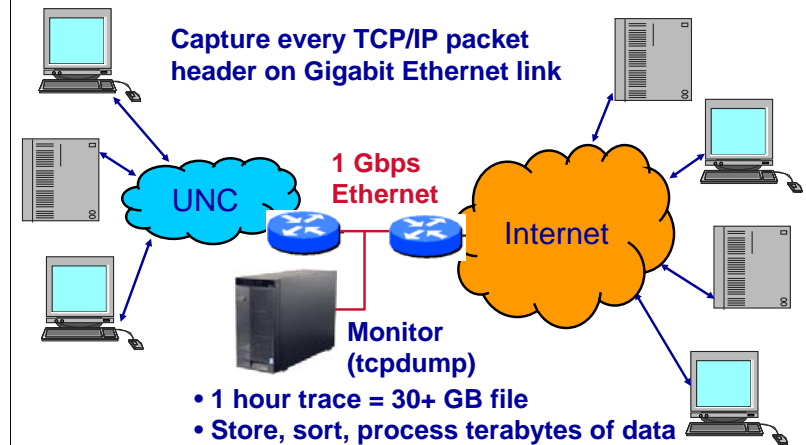
Per-Hop Delay (“nodal delay”)

$$d_{\text{nodal}} = d_{\text{proc}} + d_{\text{queue}} + d_{\text{trans}} + d_{\text{prop}}$$

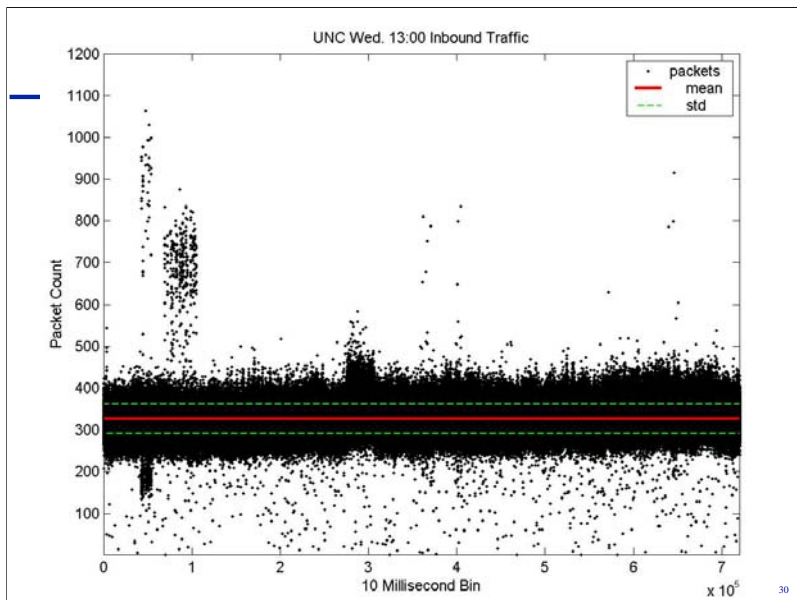
- ◆ d_{proc} = processing delay
 - » typically a few microseconds or less
- ◆ d_{queue} = queuing delay
 - » depends on congestion
- ◆ d_{trans} = transmission delay
 - » = L/R , significant for low-speed links
- ◆ d_{prop} = propagation delay
 - » a few microseconds to hundreds of msecs
- ◆ What dominates end-to-end delay?
 - » Processing, transmission, queuing delays encountered at each hop
 - » End-to-end delay is largely a function of the number of routers encountered along the path from source to destination

27

Internet Traffic Measurements & Models

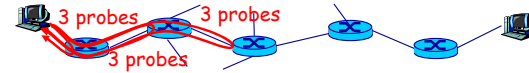


29



“Real” Internet delays and routes

- ◆ What do “real” Internet delay & loss look like?
- ◆ **Traceroute program**: provides delay measurement from source to router along end-end Internet path towards destination. For all i :
 - » sends three packets that will reach router i on path towards destination
 - » router i will return packets to sender
 - » sender times interval between transmission and reply.



Seeing Paths and Delays in the Internet

◆ www.traceroute.org

33

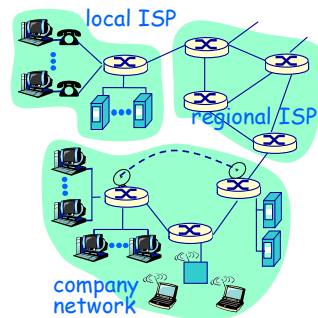
Understanding the Performance of the Internet Example: What is the delay to cs.utexas.edu?

```
>>> traceroute cs.utexas.edu

traceroute: Warning: cs.utexas.edu has multiple addresses; using
128.83.139.9
traceroute to cs.utexas.edu (128.83.139.9), 30 hops max, 38 byte packets
 1 ciscokid-cs.net.unc.edu (152.2.31.1)  0.418 ms  0.355 ms  0.356 ms
 2 unc7600.internet.unc.edu (128.109.36.254)  0.412 ms  0.495 ms  0.473
ms
 3 rtp7600-gw-to-unc7600-gw.ncren.net (128.109.70.33)  0.908 ms  0.941
ms  0.849 ms
 4 nlr-atl-to-rtp7600.ncren.net (128.109.70.106)  10.669 ms  10.381 ms
10.273 ms
 5 hous-atla-70.layer3.nlr.net (216.24.186.8)  34.444 ms  34.269 ms
34.280 ms
 6 192.124.229.6 (192.124.229.6)  33.767 ms  33.835 ms  33.815 ms
 7 192.124.229.10 (192.124.229.10)  36.995 ms  36.962 ms  37.005 ms
 8 192.124.229.82 (192.124.229.82)  37.149 ms  36.948 ms  37.146 ms
 9 ser9-v703.gw.utexas.edu (128.83.9.1)  37.112 ms  37.016 ms  37.124 ms
10 128.83.37.42 (128.83.37.42)  37.093 ms  37.113 ms  37.147 ms
11 cs.utexas.edu (128.83.139.9)  37.390 ms  37.245 ms  37.330 ms
```

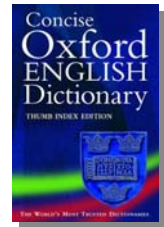
A Whirlwind Introduction to the Internet Overview

- ◆ What's the Internet
- ◆ Network core
- ◆ Network edge
- ◆ Access nets, physical media
- ◆ Internet Structure & ISPs
- ◆ Performance: loss, delay
- ◆ Protocol layers, service models



35

The Nuts & Bolts View What is a protocol?



Main Entry: pro-to-col

1: An original draft, minute, or record of a document or transaction

2a: A preliminary memorandum often formulated and signed by diplomatic negotiators as a basis for a final convention or treaty

b: The records or minutes of a diplomatic conference or congress that show officially the agreements arrived at by the negotiators

3a: A code prescribing strict adherence to correct etiquette and precedence (as in diplomatic exchange and in the military services)

b: A set of conventions governing the treatment and especially the formatting of data in an electronic communications system

4: A detailed plan of a scientific or medical experiment, treatment, or procedure

36

The Nuts & Bolts View

What is a protocol?

- ◆ Human protocols:
 - » “Do you have the time?”
 - » “I have a question”
 - » Introductions
- ◆ Network protocols:
 - » Machines rather than humans
 - » All communication activity in Internet governed by protocols

- ◆ Both:
 - » Specific messages sent
 - » Specific actions taken when messages (or other events) received

Protocols define format, order of messages sent and received among network entities, and actions taken on message transmission, receipt

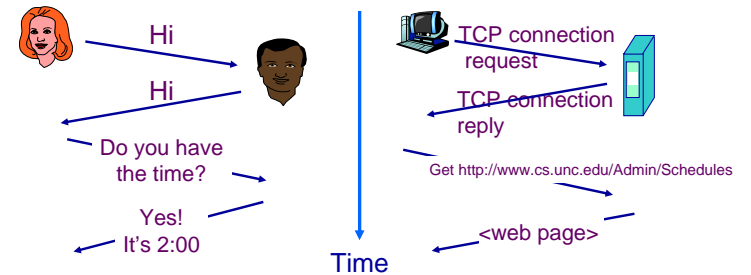
37

What is a protocol?

A specification for a set of message exchanges

◆ Example:

- » Human protocols: Get the time from a stranger
- » Computer protocols: Get the class time from a web server



38

A Whirlwind Introduction to the Internet Protocol “Layers”

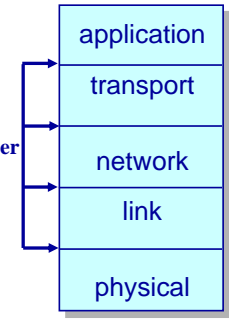
- ◆ Networks are complex!
 - » Composed of many “pieces”
 - ❖ Hosts, routers, links of various media, applications, protocols, hardware, software
 - ◆ Is there any hope of organizing the structure of the network?
 - » Or at least organizing our discussion of networks?
 - ◆ Solution!
 - » Decompose functions into a “stack” of function “layers”
 - » Each layer provides well-defined “services” to the layer above it in the stack...
 - » ...and uses the services provided by the layer below it
- ◆ Each layer can treat everything below it in the stack as a “black box”**

40

Protocol Layering in the Internet Internet protocol layers (“stack”)

- ◆ Application layer
 - » Supporting network applications
 - ❖ *ftp*, SMTP, HTTP
- ◆ Transport layer
 - » Host-host data transfer
 - ❖ TCP, UDP
- ◆ Network layer
 - » Routing of packets from source to destination
 - ❖ IP, routing protocols
- ◆ Link layer
 - » Data transfer between directly connected network elements
 - ❖ Ethernet, 802.11, SONET, ...
- ◆ Physical layer
 - » The insertion of individual bits “on the wire”

Different services
specified at each layer
interface

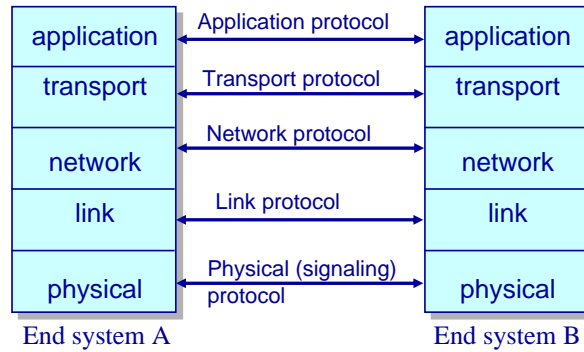


41

Protocol Layering in the Internet

Internet protocol layers (“stack”)

Each layer implements a protocol with its peer layer in a distributed system

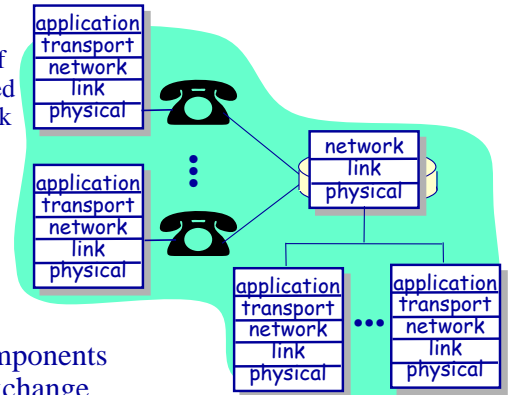


42

Protocol Layering in the Internet

Logical communication

- ◆ The implementation of each layer is distributed throughout the network
 - » Some layers just distributed on the end-systems



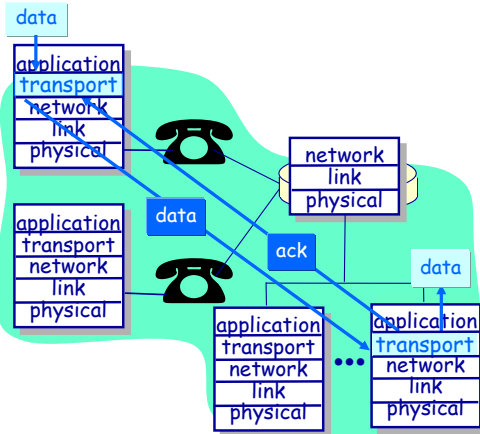
- ◆ The distributed components perform actions, exchange messages with peers

43

Logical Communication Example

The transport layer

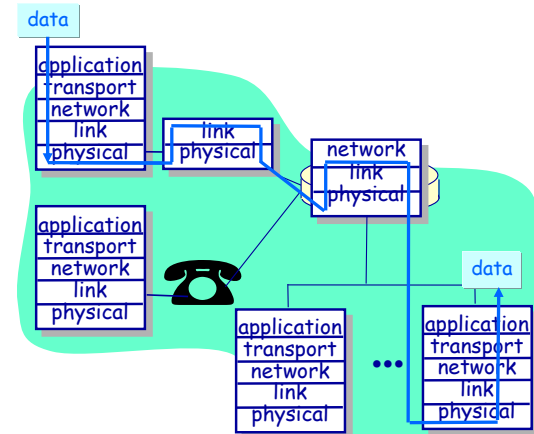
- ◆ Receive data from application
- ◆ Add transport-layer protocol information
- ◆ **Send to peer transport layer**
- ◆ Wait for peer transport layer to respond
- ◆ Peer transport delivers data to its application layer



44

Protocol Layering in the Internet

Data flow through protocol layers

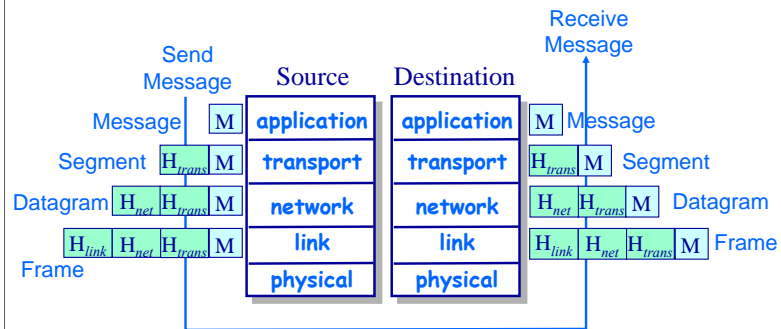


45

Protocol Layering in the Internet

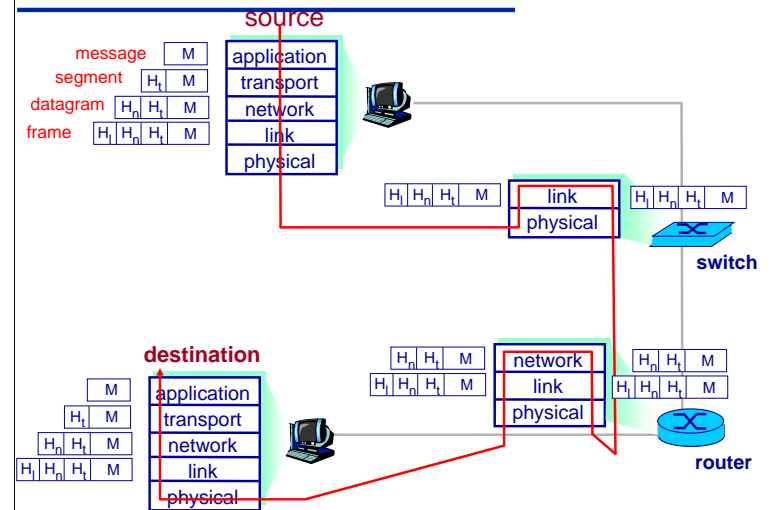
Protocol layering and data formats

- ◆ At sender, each layer takes data from above
 - » Adds header information to create new data unit
 - » Passes new data unit to layer below
- ◆ Process reversed at receiver



46

Encapsulation Flow in Network Layers

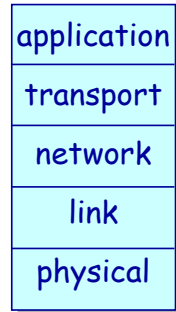


47

Protocol Layering in the Internet

Common logical functions in most layers

- ◆ Error control
 - » Make the logical channel between layers reliable (or simply more reliable)
- ◆ Flow control
 - » Avoid overwhelming a peer with data
- ◆ Segmentation and reassembly
 - » Partitioning large messages into smaller ones at the sender and reassembling them at the receiver
- ◆ Multiplexing
 - » Allowing several higher-level sessions to share a single lower-level connection
- ◆ Connection setup
 - » Handshaking with a peer



48

Why Layering?

Dealing with complex systems

- ◆ Explicit structure allows identification, relationship of complex system's pieces
 - » Layered reference model for discussion
- ◆ Modularization eases maintenance, updating of system
 - » Change of implementation of layer's service transparent to rest of system
- ◆ Layering considered harmful?

49

A Whirlwind Introduction to the Internet Summary

- ◆ Covered a “ton” of material
 - » Internet overview
 - » What’s a protocol?
 - » Network edge, core, access network
 - » ISPs
 - » Performance: loss, delay
 - » Layering and service models
- ◆ You now hopefully have:
 - » Context, overview, “feel” of networking
 - » More depth, detail *later* in course
- ◆ Something dangerous to mumble at parties!

50