

The Internet: a “Nuts And Bolts” View



Billions of connected computing *devices*:

- *hosts* = end systems
- running *network apps* at Internet's “edge”



Packet switches: forward packets (chunks of data)

- *routers, switches*

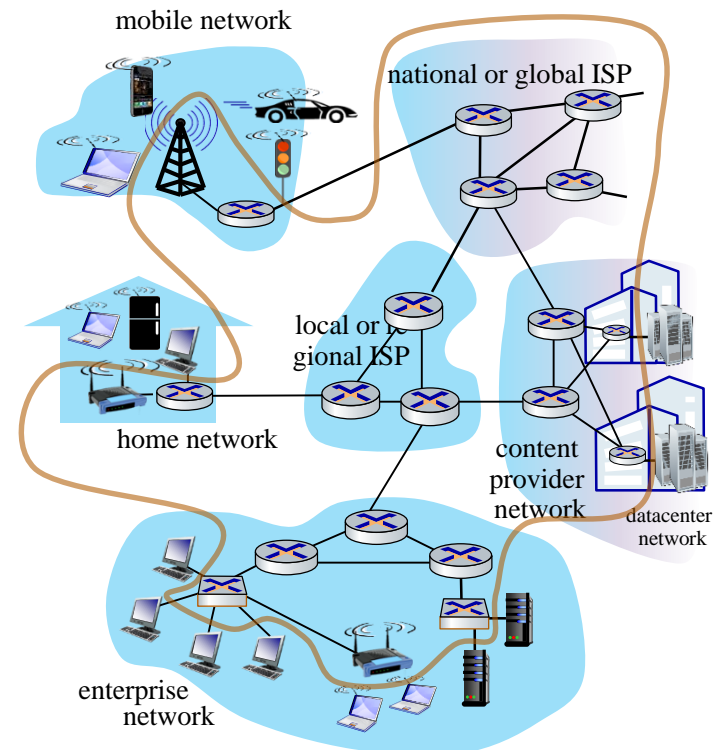


Communication links

- fiber, copper, radio, satellite
- transmission rate: *bandwidth*

Networks

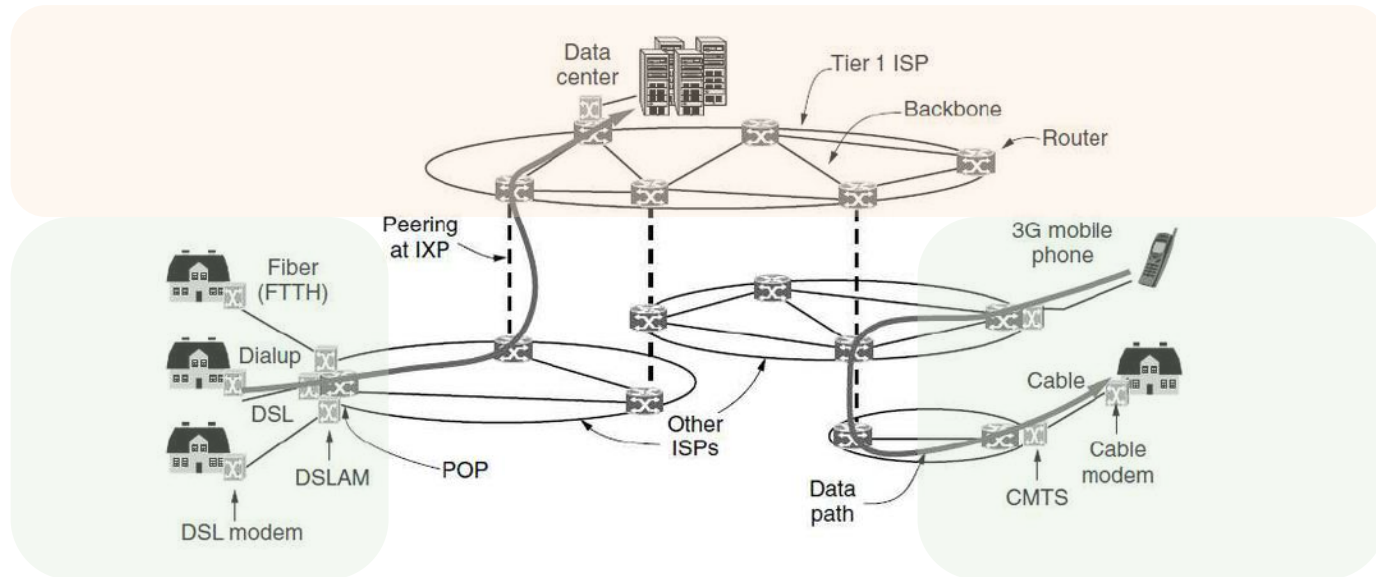
- collection of devices, routers, links: managed by an organization



Network Core/Edge

- *Network core:*

- interconnected routers
- network of networks



- *Network edge:*

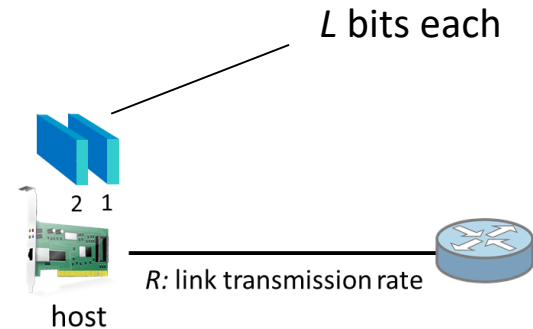
- hosts: clients and servers
- servers often in data centers

- *Access network (last-mile network)*

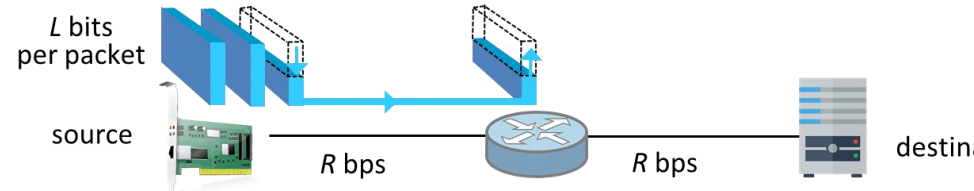
Network Core

Transmission delay:

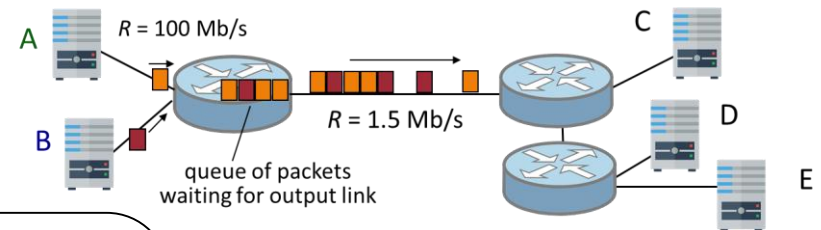
$$\text{packet transmission delay} = \text{time needed to transmit } L\text{-bit packet into link} = \frac{L \text{ (bits)}}{R \text{ (bits/sec)}}$$



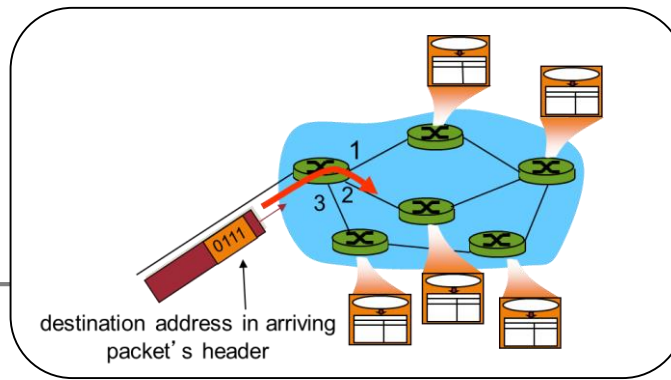
store and forward: entire packet must arrive at router before it can be transmitted on next link



queuing and loss: if arrival rate (in bits) to link exceeds transmission rate of link for a period of time



Packet Switching & Routing





Introduction

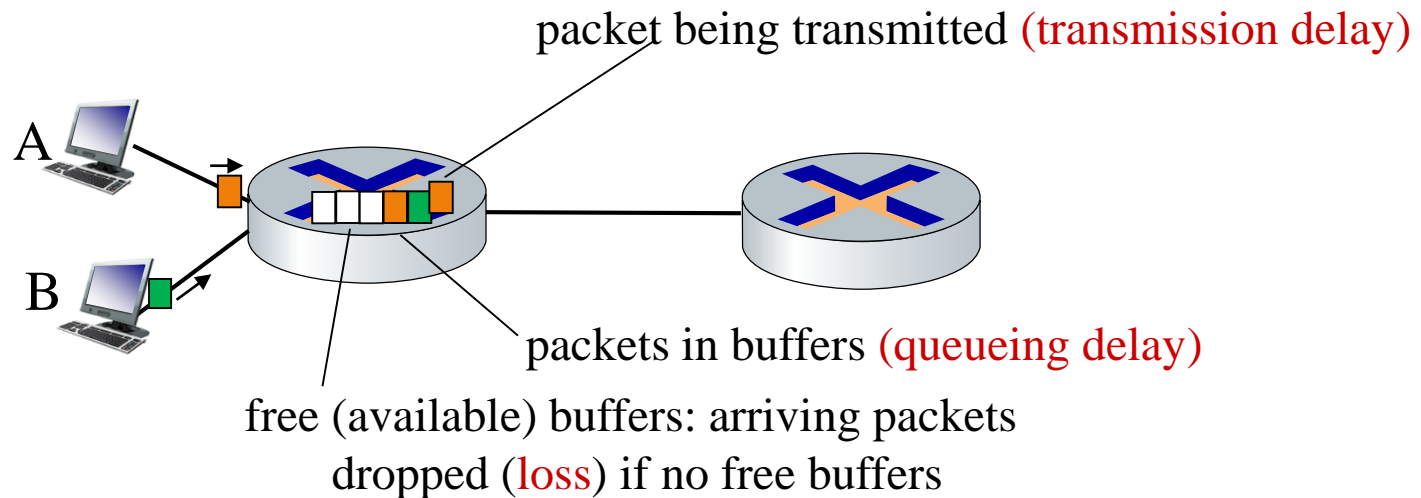
CSE351

Computer Networks

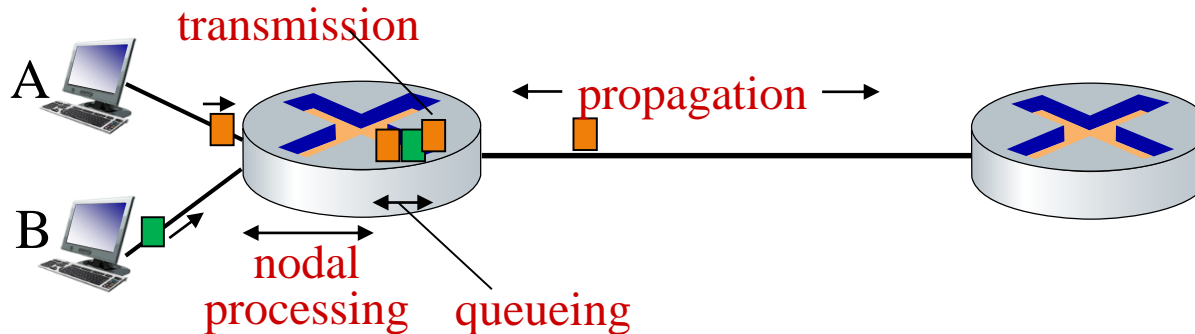
How do loss and delay occur?

packets *queue* in router buffers

- Packet arrival rate to link (temporarily) exceeds output link capacity
- Packets queue, wait for turn
- Packets are dropped in the tail part (drop-tail queue)



Packet Delay: four sources



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

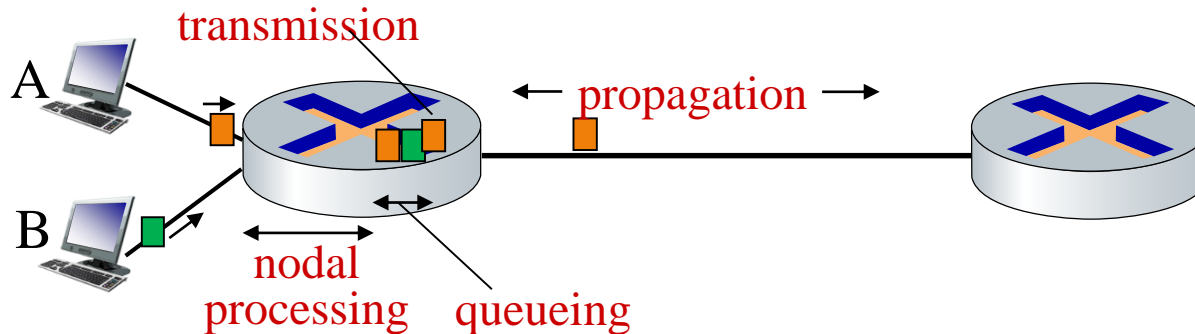
d_{proc} : nodal processing

- check bit errors
- determine output link
- typically < msec

d_{queue} : queueing delay

- time waiting at output link for transmission
- depends on congestion level of router

Packet Delay: four sources



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

d_{trans} : transmission delay:

- L : packet length (bits)
- R : link transmission rate (bps)

$$d_{\text{trans}} = L/R$$

d_{trans} and d_{prop}
very different

d_{prop} : propagation delay:

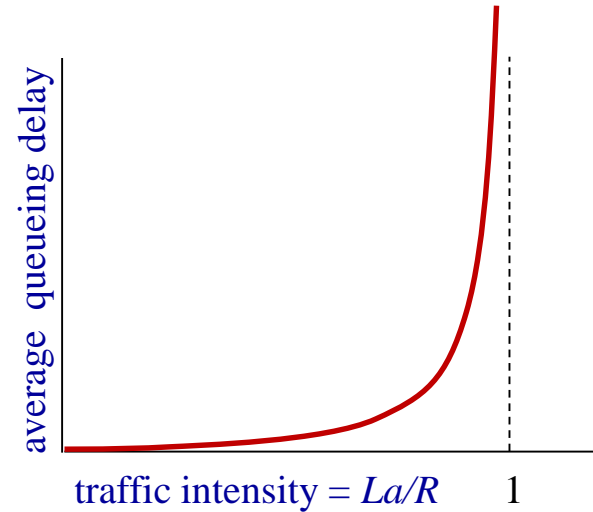
- d : length of physical link
- s : propagation speed ($\sim 2 \times 10^8$ m/sec)

$$d_{\text{prop}} = d/s$$

* Check out the online interactive exercises:
http://gaia.cs.umass.edu/kurose_ross

Queueing Delay (revisited)

- R : link bandwidth (bps)
- L : packet length (bits)
- a : average packet arrival rate
- $La/R \sim 0$: avg. queueing delay small
- $La/R \rightarrow 1$: avg. queueing delay large
- $La/R > 1$: more “work” arriving than can be serviced - average delay infinite!



Real Internet Delay

ping www.google.com

```
PING www.google.com (172.217.160.68): 56 data bytes
64 bytes from 172.217.160.68: icmp_seq=0 ttl=44 time=71.642 ms
64 bytes from 172.217.160.68: icmp_seq=1 ttl=44 time=82.027 ms
64 bytes from 172.217.160.68: icmp_seq=2 ttl=44 time=67.633 ms
64 bytes from 172.217.160.68: icmp_seq=3 ttl=44 time=67.492 ms
64 bytes from 172.217.160.68: icmp_seq=4 ttl=44 time=80.396 ms
64 bytes from 172.217.160.68: icmp_seq=5 ttl=44 time=83.657 ms
```

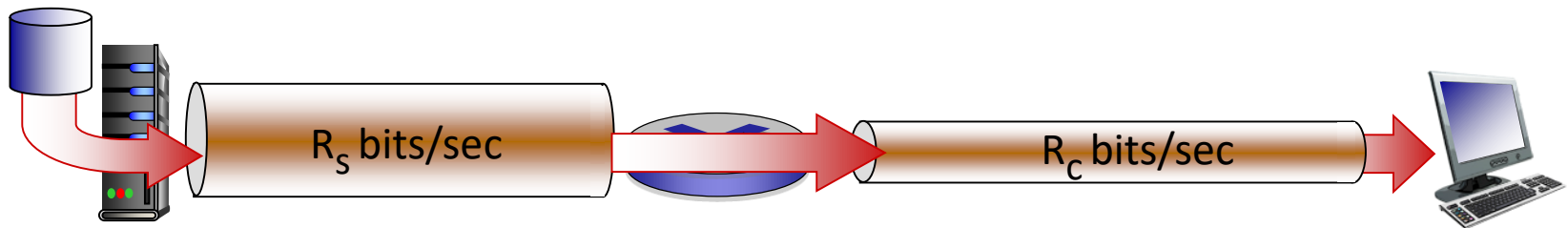
traceroute www.google.com

```
traceroute to www.google.com (172.217.160.68), 64 hops max, 52 byte packets
 1  router.asus.com (192.168.2.1)  2.360 ms  1.030 ms  1.131 ms
 2  router.asus.com (192.168.1.1)  1.624 ms  4.172 ms  1.429 ms
 3  * * *
 4  10.240.88.237 (10.240.88.237)  4.098 ms  3.047 ms *
 5  10.204.92.105 (10.204.92.105)  2.368 ms  4.382 ms  3.328 ms
 6  1.213.64.165 (1.213.64.165)  2.583 ms
    1.208.64.161 (1.208.64.161)  2.580 ms
    1.213.64.165 (1.213.64.165)  2.552 ms
 7  1.214.58.177 (1.214.58.177)  9.980 ms
    1.209.58.181 (1.209.58.181)  9.874 ms
    1.214.58.205 (1.214.58.205)  9.172 ms
 8  1.208.104.61 (1.208.104.61)  10.285 ms
    1.213.104.61 (1.213.104.61)  9.528 ms  10.819 ms
 9  1.208.148.141 (1.208.148.141)  12.972 ms  10.580 ms  8.417 ms
10  203.233.117.81 (203.233.117.81)  10.343 ms
    203.248.208.229 (203.248.208.229)  18.325 ms
    210.120.117.113 (210.120.117.113)  9.409 ms
```

Hop-by-hop ping

Throughput and Bottleneck

- *Throughput*: rate (bits/time unit) at which bits transferred between sender/receiver
 - *instantaneous*: rate at given point in time
 - *average*: rate over longer period of time
 - $R_s > R_c$ What is average end-to-end throughput?

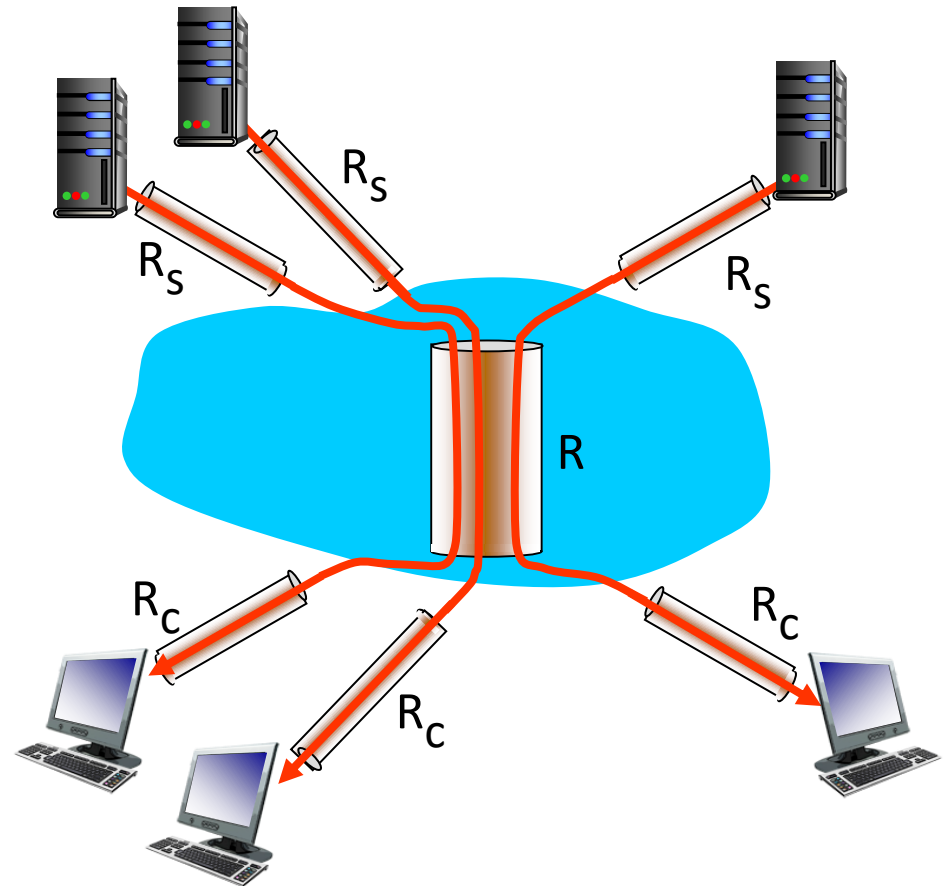


□ *Bottleneck link*

- link on e2e path that constrains e2e throughput

Throughput and Bottleneck

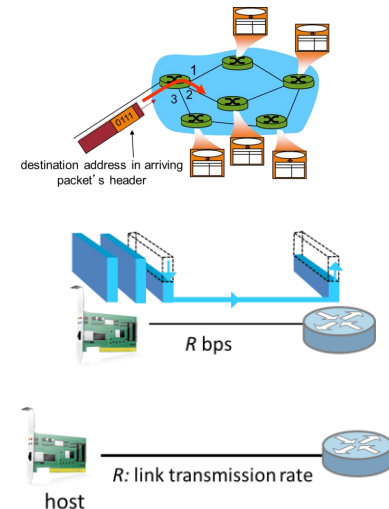
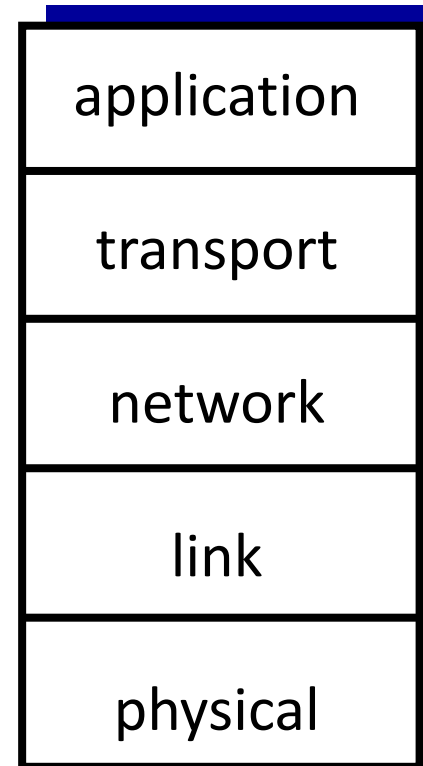
- Per-connection end-end throughput:
 $\min(R_c, R_s, R/10)$
- In practice: R_c or R_s is often bottleneck



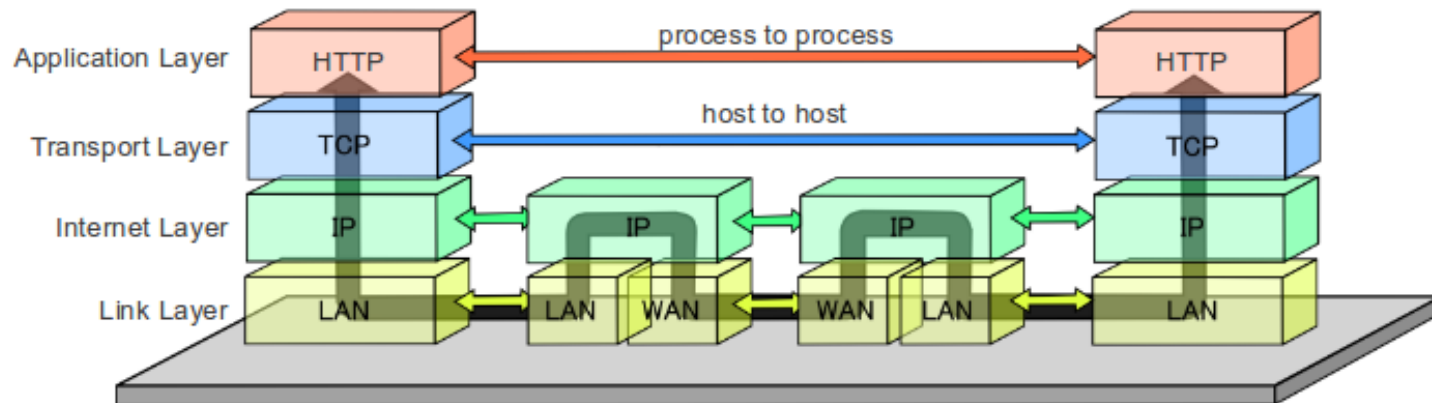
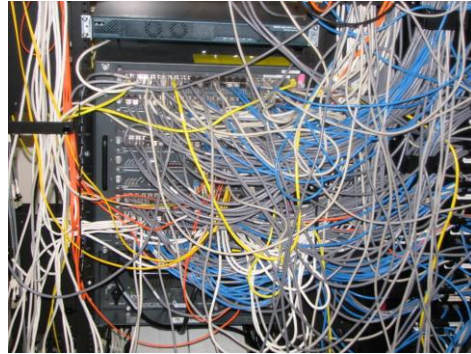
10 connections (fairly) share
backbone link of R bits/sec

Internet Protocol Stacks (Layering)

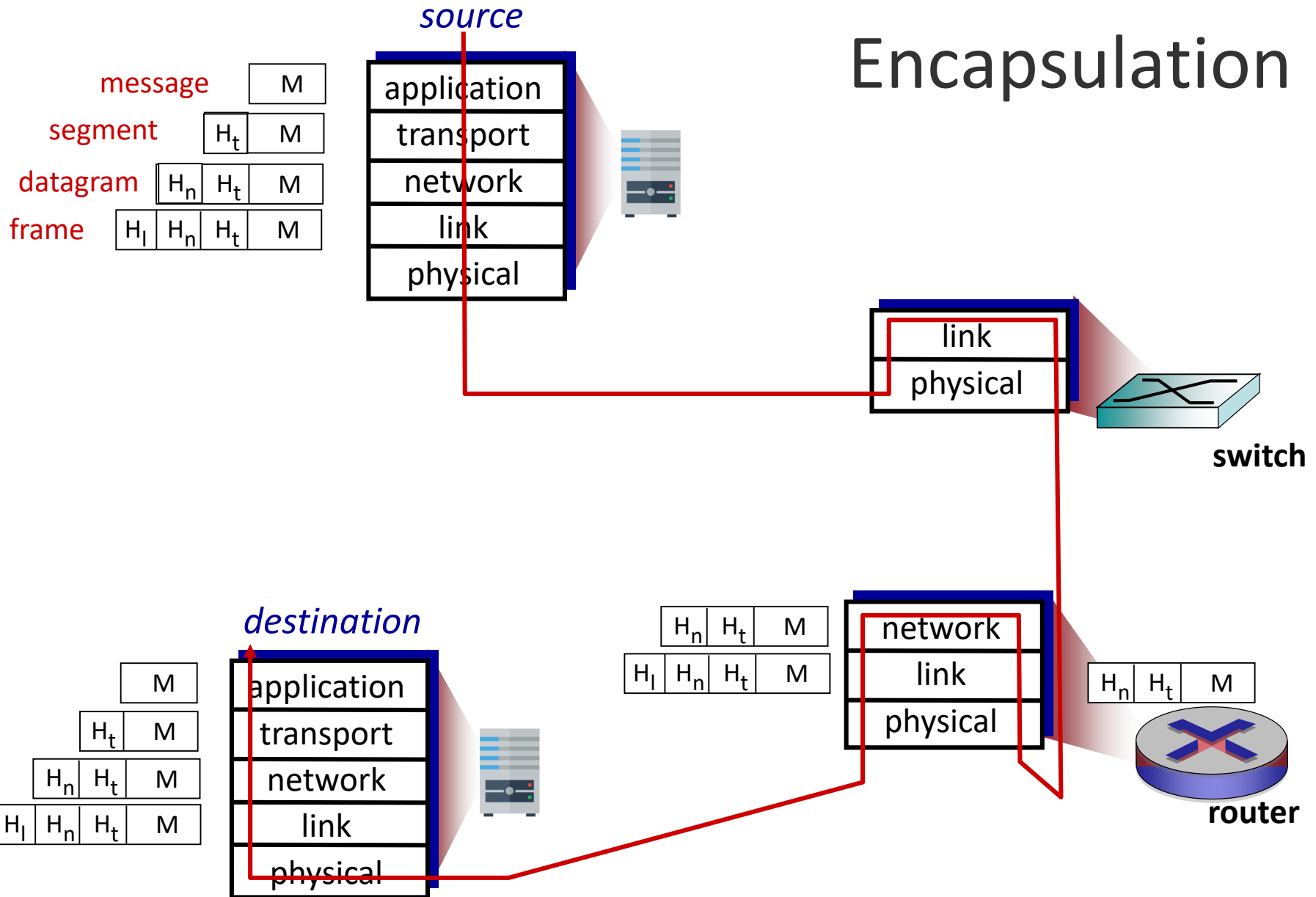
- *application*: supporting network applications
 - FTP, SMTP, HTTP
- *transport*: process-process data transfer
 - TCP, UDP
- *network*: routing of datagrams from source to destination
 - IP, routing protocols
- *link*: data transfer between neighboring network elements
 - Ethernet, 802.11 (WiFi), Bluetooth
- *physical*: bits “on the wire”



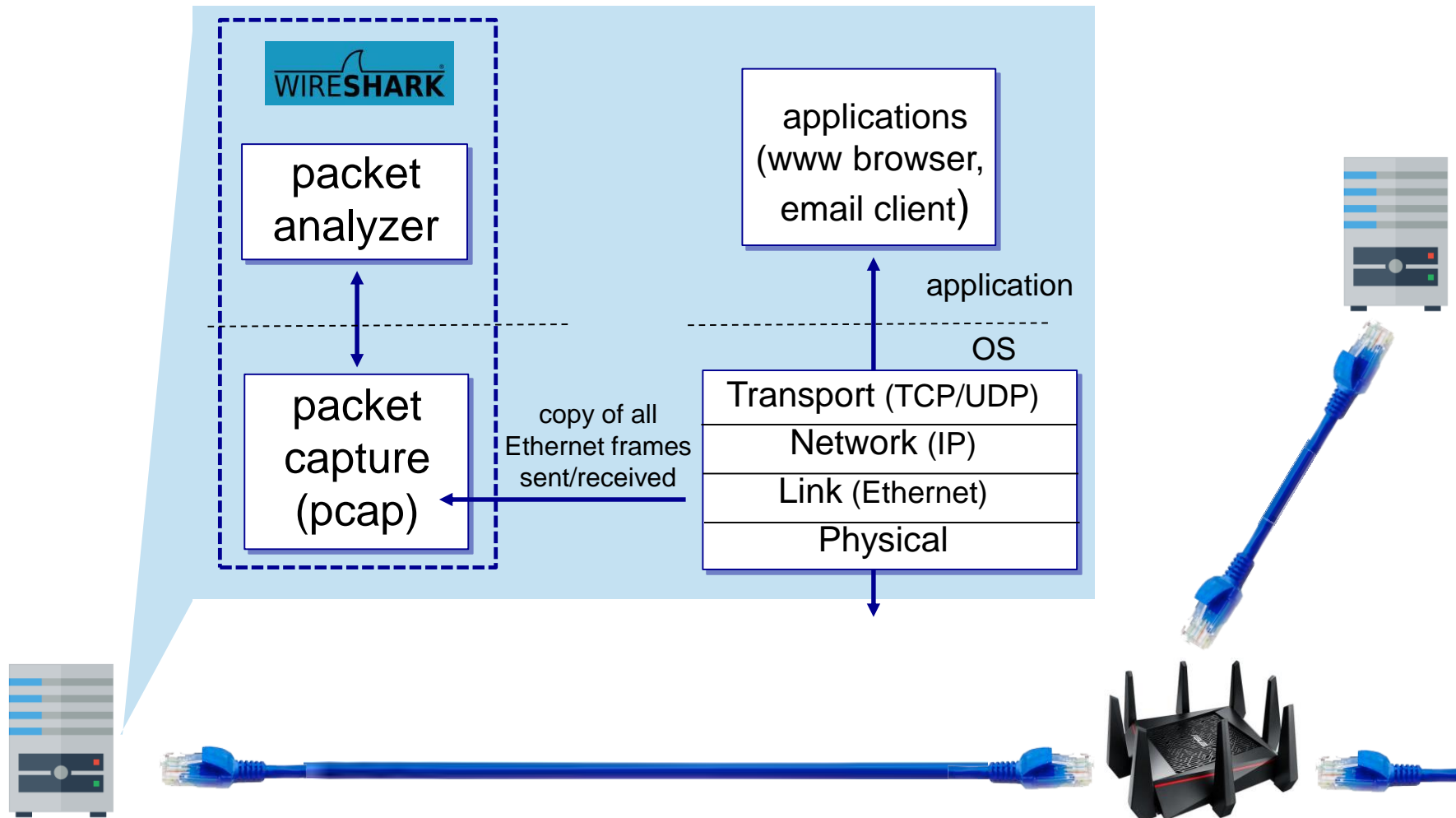
Communication over the Internet



Encapsulation



Packet Capturing



Packet Capturing

The image shows a Wireshark packet capture window titled "tv-netflix-problems-2011-07-06.pcap". The interface includes a menu bar (File, Edit, View, Go, Capture, Analyze, Statistics, Telephony, Wireless, Tools, Help) and a toolbar. A display filter is set to "Apply a display filter ... <Ctrl-/>".

The packet list pane shows several packets. Packet 348 is selected, showing a DNS standard query response from 192.168.0.1 to 192.168.0.21. The packet details pane shows the following information:

- > Frame 349: 489 bytes on wire (3912 bits), 489 bytes captured (3912 bits)
- > Ethernet II, Src: Globalsec_00:3b:0a (f0:ad:4e:00:3b:0a), Dst: Vizio_14:8a:e1 (00:19:9d:14:8a:e1)
- > Internet Protocol Version 4, Src: 192.168.0.1, Dst: 192.168.0.21
- > User Datagram Protocol, Src Port: 53 (53), Dst Port: 34036 (34036)
- > Domain Name System (response)
 - [Request In: 348]
 - [Time: 0.034338000 seconds]
 - Transaction ID: 0x2188
 - > Flags: 0x8180 Standard query response, No error
 - Questions: 1
 - Answer RRs: 4
 - Authority RRs: 9
 - Additional RRs: 9
 - > Queries
 - > cdn-0.nflximg.com: type A, class IN
 - > Answers
 - > Authoritative nameservers

The packet bytes pane shows the raw data of the selected packet, with the first few bytes highlighted in blue:

```
0020 00 15 00 35 84 f4 01 c7 83 3f 21 88 81 80 00 01 ...5....?!....
0030 00 04 00 09 00 09 05 63 64 6e 2d 30 07 6e 66 6c .....c dn-0.nfl
0040 78 69 6d 67 03 63 6f 6d 00 00 01 00 01 c0 0c 00 ximg.com .....
0050 05 00 01 00 00 05 29 00 22 06 69 6d 61 67 65 73 .....). ".images
0060 07 6e 65 74 66 6c 69 78 03 63 6f 6d 09 65 64 67 .netflix .com.edg
0070 65 73 75 69 74 65 03 6e 65 74 00 c0 2f 00 05 00 esuite.n et../...
```

The status bar at the bottom indicates "Packets: 10299 · Displayed: 10299 (100.0%) · Load time: 0:0.182 | Profile: Default".

Internet History

(1962-1972: Early Packet-Switching Principles)

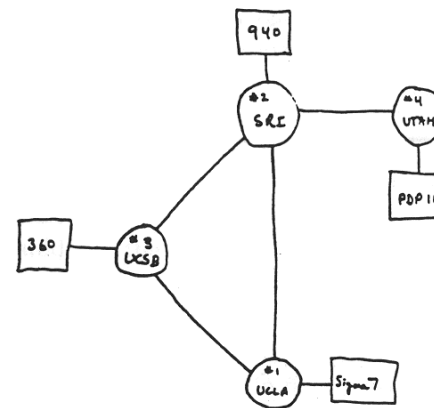
- **1961:** Kleinrock - queueing theory shows effectiveness of packet-switching



- **1964:** Baran - packet-switching in military nets
- **1967:** ARPAnet conceived by Advanced Research Projects Agency
- **1969:** first ARPAnet node operational

□ **1972:**

- ARPAnet public demo
- NCP (Network Control Protocol) first host-host protocol
- first e-mail program
- ARPAnet has 15 nodes



THE ARPA NETWORK

Internet History

(1972-1980: Internetworking, new and proprietary nets)

- 1970: ALOHAnet satellite network in Hawaii
- 1974: Cerf and Kahn - architecture for interconnecting networks
- 1976: Ethernet at Xerox PARC
- late70' s: proprietary architectures: DECnet, SNA, XNA
- late 70' s: switching fixed length packets (ATM precursor)
- 1979: ARPAnet has 200 nodes

Cerf and Kahn' s internetworking principles:

- minimalism, autonomy - no internal changes required to interconnect networks
- best effort service model
- stateless routers
- decentralized control

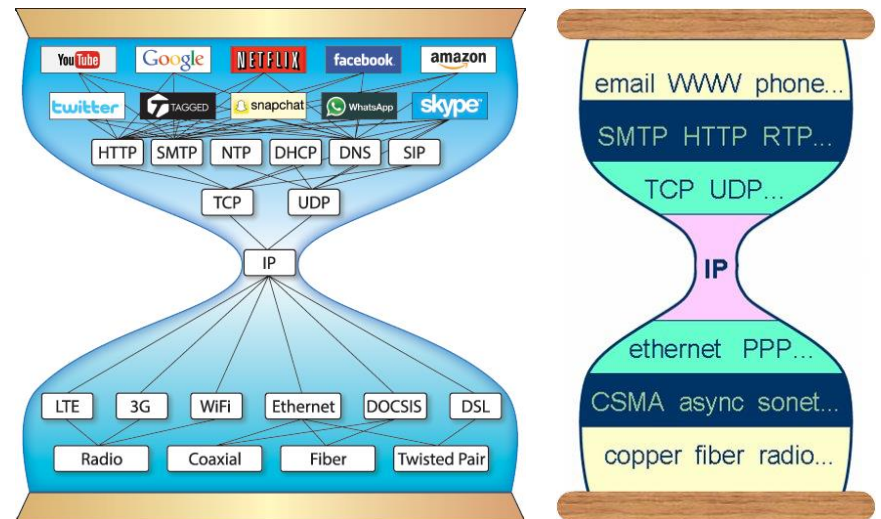
define today' s Internet
architecture

Internet History

(1980-1990: new protocols, a proliferation of networks)

- 1983: deployment of TCP/IP
- 1982: smtp e-mail protocol defined
- 1983: DNS defined for name-to-IP-address translation
- 1985: ftp protocol defined
- 1988: TCP congestion control

- new national networks: CSnet, BITnet, NSFnet, Minitel
- 100,000 hosts connected to confederation of networks



Drawing from Johann Schleier-Smith

Internet History

(1990, 2000's: commercialization, the Web, new apps)

- Early 1990's: ARPAnet decommissioned
- 1991: NSF lifts restrictions on commercial use of NSFnet (decommissioned, 1995)
- Early 1990's : Web
 - hypertext [Bush 1945, Nelson 1960's]
 - HTML, HTTP: Berners-Lee
 - 1994: Mosaic, later Netscape
 - late 1990's: commercialization of the Web
- Late 1990's – 2000's:
 - more killer apps: instant messaging, P2P file sharing
 - network security to forefront
 - est. 50 million host, 100 million+ users
 - backbone links running at Gbps

Internet History

(2005-present)

- ☐ ~5B devices attached to Internet (2016)
 - smartphones and tablets
- ☐ Aggressive deployment of broadband access
- ☐ Increasing ubiquity of high-speed wireless access
- ☐ Emergence of online social networks:
 - Facebook: ~ one billion users
- ☐ Service providers (Google, Microsoft) create their own networks
 - bypass Internet, providing “instantaneous” access to search, video content, email, etc.
- ☐ e-commerce, universities, enterprises running their services in “cloud” (e.g., Amazon EC2)