Computer Networks CS348

Instructor: Joydeep Chandra

Course Outline

- Theory
  - Basics of Interworking
  - Modern Networking Applications and Technologies
  - Advanced Concepts in Design of Network based Applications
- Lab
  - Implementation of networking protocols
  - Use of simulators
  - Develop network based applications

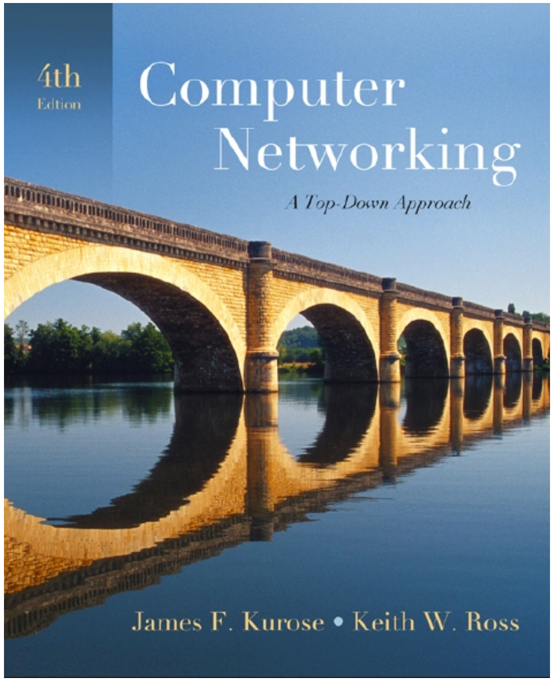
Evaluation

- Theory + Lab
  - Either 60-40 or 70-30 (TBD)
- Theory
  - Quizzes and Class Tests 20%
  - MidSem 30%
  - End Sem 50%
- Lab
  - Assignments 50%
  - MidSem 20%
  - EndSem 30%

Books and References

- Text Book
  - James F. Kurose and Keith W. Ross, Computer Networking: A Top-Down Approach
- References
  - Larry L. Peterson and Bruce S. Davie, Computer Networks, A Systems Approach
  - Tenenbaum and Wetherall, Computer Networks, Pearson India, 5^{th} Edition
  - Some research articles and papers

Chapter 1
Introduction



Computer Networking: A Top Down Approach ,
4th edition.
Jim Kurose, Keith Ross
Addison-Wesley, July 2007.

A note on the use of these ppt slides:

We're making these slides freely available to all (faculty, students, readers). They're in PowerPoint form so you can add, modify, and delete slides (including this one) and slide content to suit your needs. They obviously represent a *lot* of work on our part. In return for use, we only ask the following:

- If you use these slides (e.g., in a class) in substantially unaltered form, that you mention their source (after all, we'd like people to use our book!)
- If you post any slides in substantially unaltered form on a www site, that you note that they are adapted from (or perhaps identical to) our slides, and note our copyright of this material.

Thanks and enjoy! JFK/KWR

Outline

- Internet architecture
- Internet history
- Today's Internet
- Internet in a nutshell (protocols in practice)

Internet Architecture

- http://www.nap.edu/html/coming_of_age/
- http://www.ietf.org/rfc/rfc1958.txt

Why did the Internet win?

- Packet switching over circuit switching
- End-to-end principle and “Hourglass” design
- Layering of functionality
- Distributed design, decentralized control
- Superior organizational process

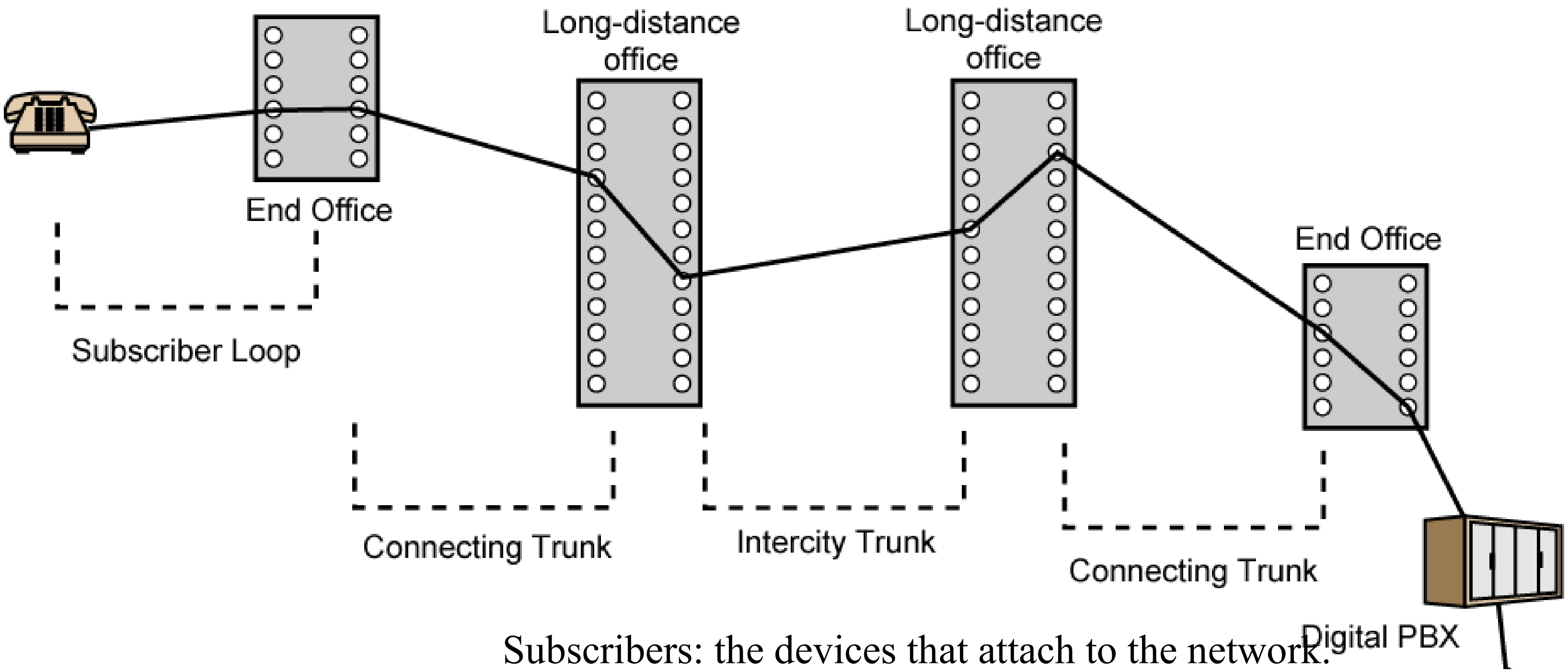
Circuit Switching

- Circuit switching:
  - There is a dedicated communication path between two stations (end-to-end)
  - The path is a connected sequence of links between network nodes. On each physical link, a logical channel is dedicated to the connection.
- Communication via circuit switching has three phases:
  - Circuit establishment (link by link)
    - Routing & resource allocation (FDM or TDM)
  - Data transfer
  - Circuit disconnect
    - Deallocate the dedicated resources
- The switches must know how to find the route to the destination and how to allocate bandwidth (channel) to establish a connection.

Circuit Switching Properties

- Inefficiency
  - Channel capacity is dedicated for the whole duration of a connection
  - If no data, capacity is wasted
- Delay
  - Long initial delay: circuit establishment takes time
  - Low data delay: after the circuit establishment, information is transmitted at a fixed data rate with no delay other than the propagation delay. The delay at each node is negligible.
- Developed for voice traffic (public telephone network) but can also applied to data traffic.
  - For voice connections, the resulting circuit will enjoy a high percentage of utilization because most of the time one party or the other is talking.
  - But how about data connections?

Public Circuit Switched Network



Subscribers: the devices that attach to the network.

Subscriber loop: the link between the subscriber and the network.

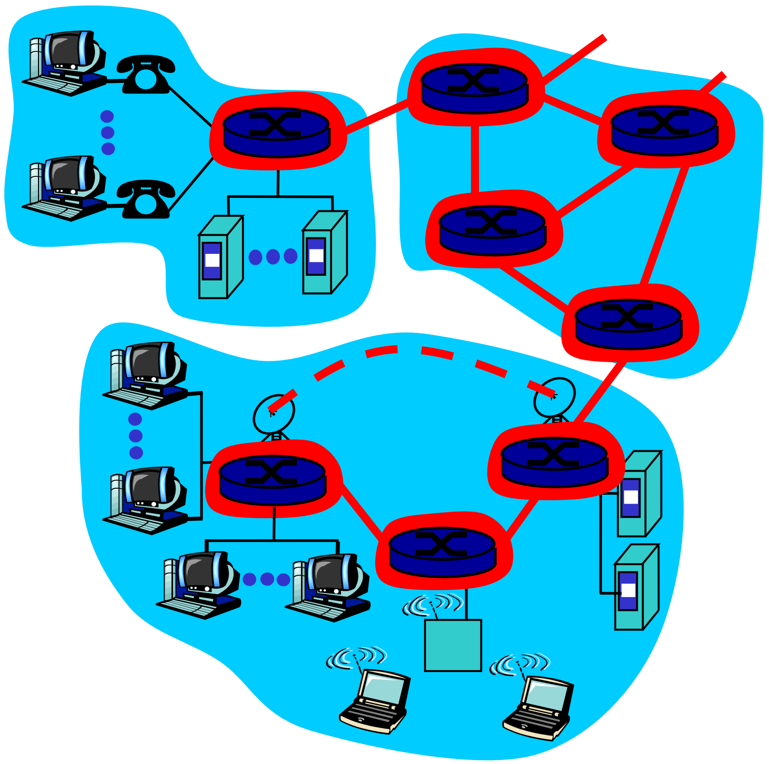
Exchanges: the switching centers in the network.

End office: the switching center that directly supports subscribers.

Trunks: the branches between exchanges. They carry multiple voice-frequency circuits using either FDM or synchronous TDM.

Packet vs. circuit switching

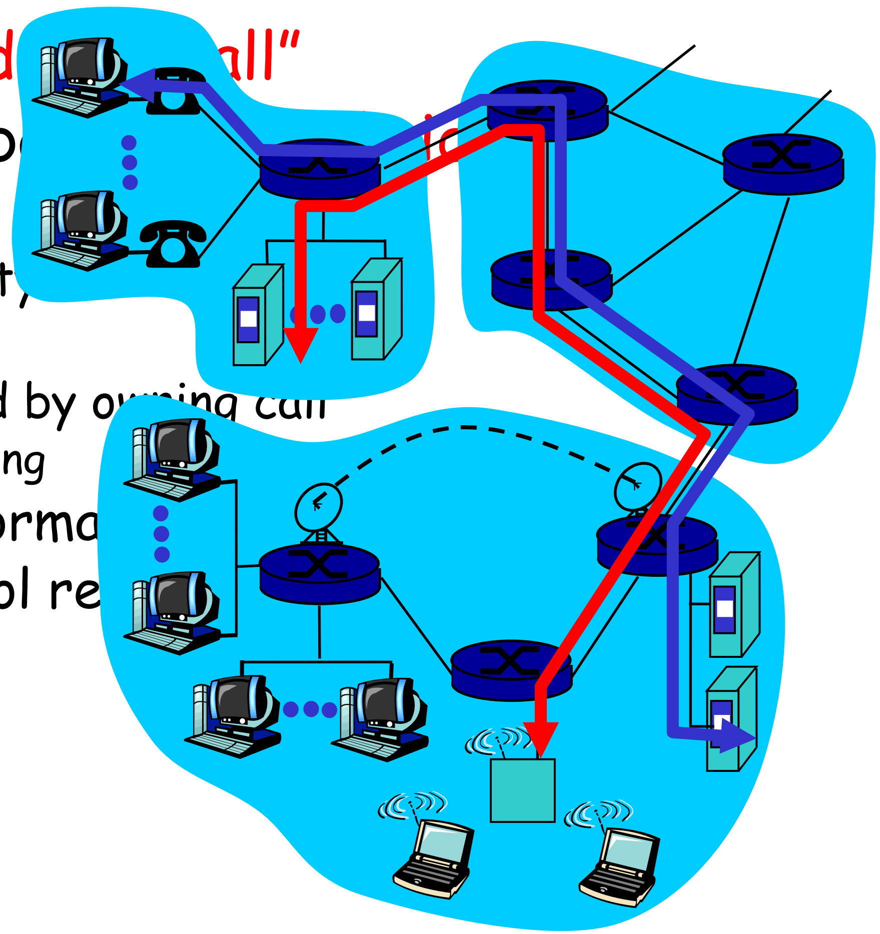
- mesh of interconnected routers
- *the* fundamental question: how is data transferred through net?
  - circuit switching: dedicated circuit per call: telephone net
  - packet-switching: data sent thru net in discrete "chunks"



Circuit Switching

End-end resources reserved … all”

- network resources (e.g., b…
  “pieces”
  - link bandwidth, switch capacit…
  - pieces allocated to calls
  - resource piece *idle* if not used by o… call
    - dedicated resources: no sharing
- circuit-like (guaranteed) performa…
- call setup and admission control re…

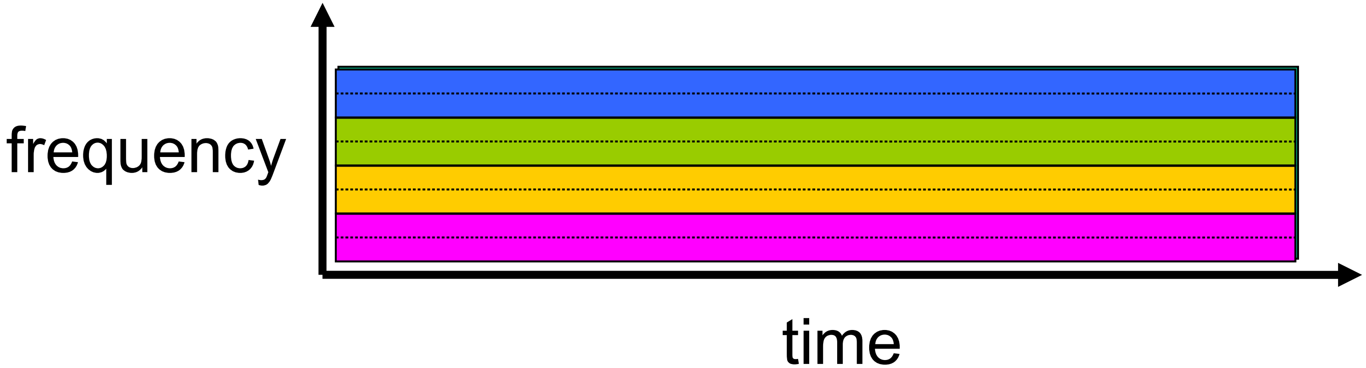


Circuit Switching: FDM and TDM

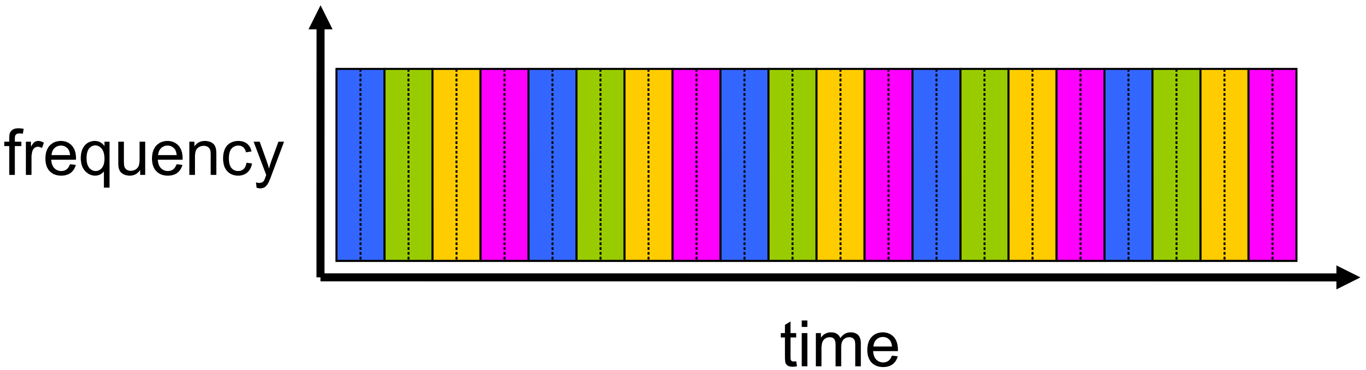
Example:

4 users

FDM



TDM



Numerical example

- How long does it take to send a file of 640,000 bits from host A to host B over a circuit-switched network?
  - All links are 1.536 Mbps
  - Each link uses TDM with 24 slots/sec
  - 500 msec to establish end-to-end circuit

Let's work it out!

Numerical example

- How long does it take to send a file of 640,000 bits from host A to host B over a circuit-switched network?
  - All links are 1.536 Mbps
  - Each link uses TDM with 24 slots/sec
  - 500 msec to establish end-to-end circuit

Let's work it out!

1,536,000/24 = 64000 bps per time slot

640000bits/64000bps = 10 sec.

Total time = 500 msec + 10 sec = 10.5 sec

Case study: Circuit Switching

- 1890-current: Phone network
  - Fixed bit rate
  - Mostly voice
  - Not fault-tolerant
  - Components extremely reliable
  - Global application-level knowledge throughout network
  - Admission control at local switching station (dial-tone)

Network Core: Packet Switching

each end-end data stream divided into *packets*

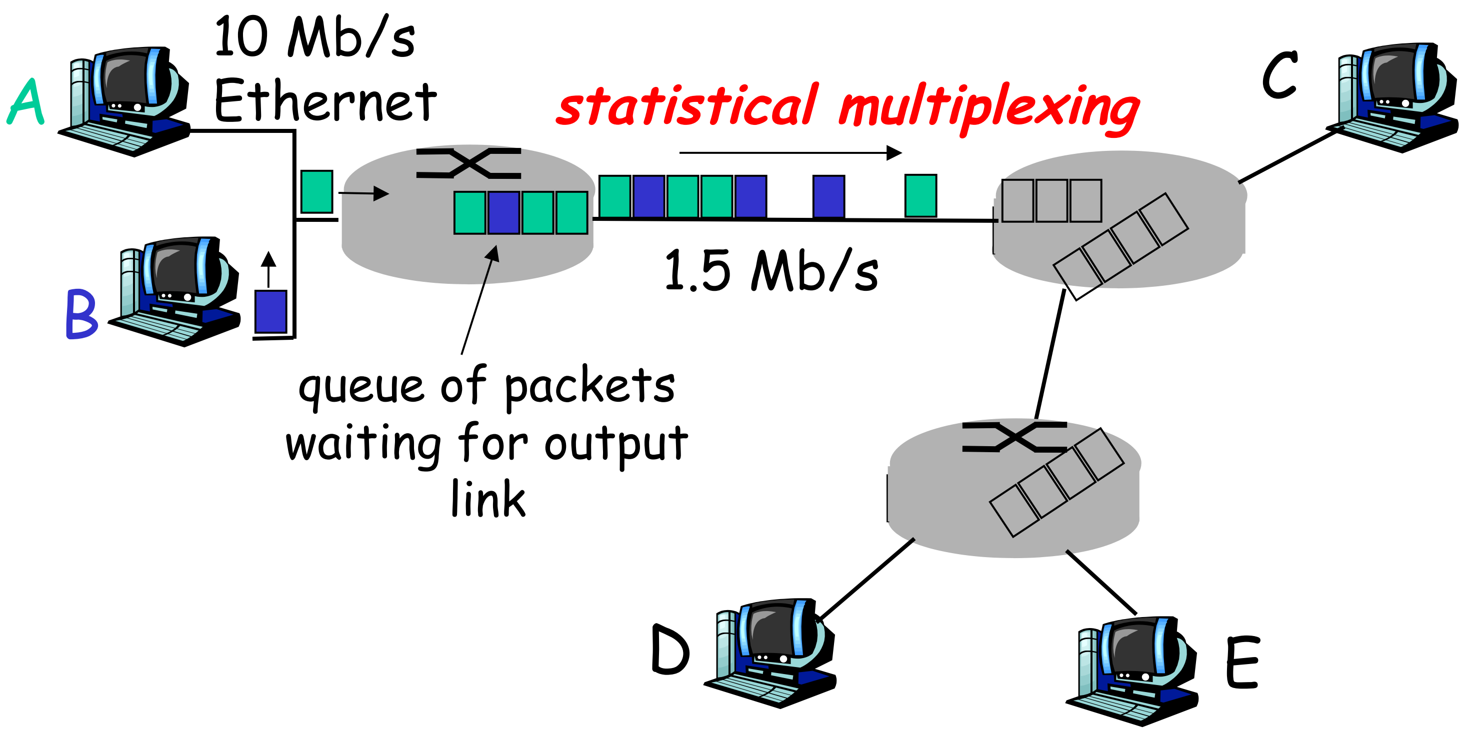
- user A, B packets *share* network resources
- each packet uses full link bandwidth
- resources used *as needed*

Bandwidth division into "pieces"
Dedicated allocation
Resource reservation

resource contention:

- aggregate resource demand can exceed amount available
- congestion: packets queue, wait for link use
- store and forward: packets move one hop at a time
  - Node receives complete packet before forwarding

Packet Switching: Statistical Multiplexing



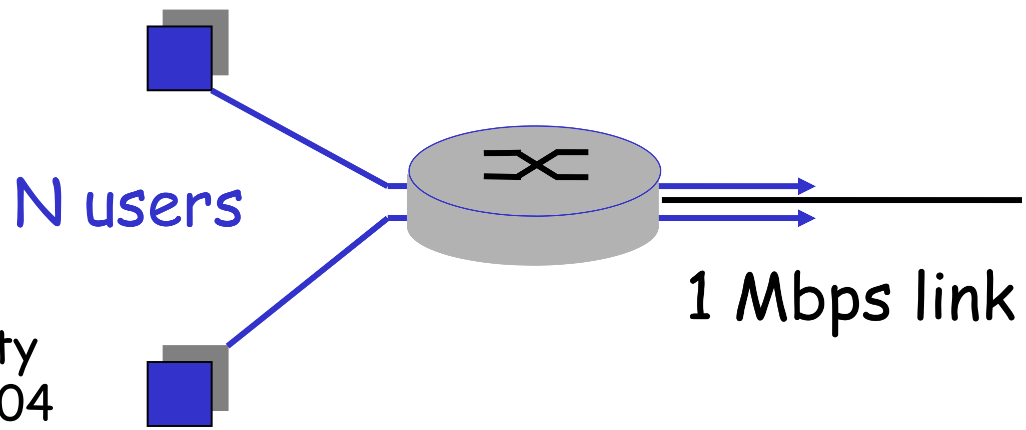
Sequence of A & B packets does not have fixed pattern, shared on demand → ***statistical multiplexing***.

TDM: each host gets same slot in revolving TDM frame.

Packet switching versus circuit switching

Packet switching allows more users to use network!

- N users over 1 Mb/s link
- each user:
  - 100 kb/s when "active"
  - active 10% of time
- circuit-switching:
  - 10 users
- packet switching:
  - with 35 users, probability > 10 active less than .0004
  - Allows more users to use network
  - "Statistical multiplexing gain"



Q: how did we get value 0.0004?

Packet switching versus circuit switching

Is packet switching a "slam dunk winner?"

- Great for bursty data
  - resource sharing
  - simpler, no call setup
- Bad for applications with hard resource requirements
  - Excessive congestion: packet delay and loss
  - Need protocols for reliable data transfer, congestion control
  - Applications must be written to handle congestion

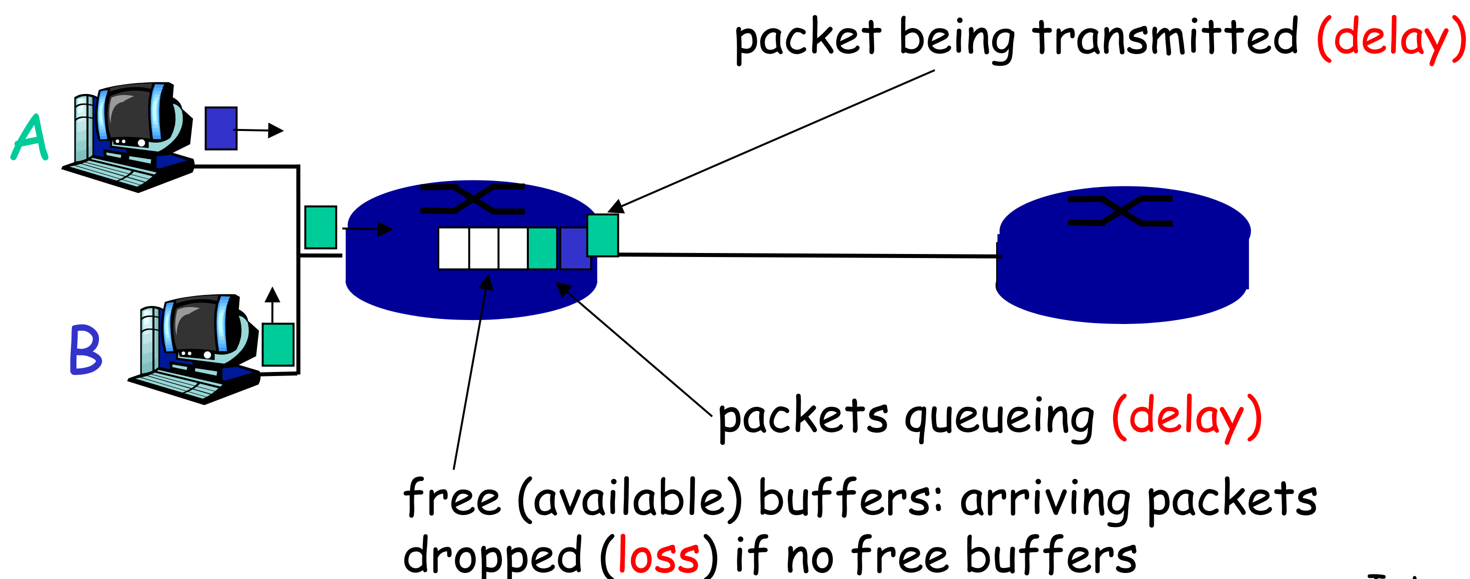
Q: How to provide circuit-like behavior?

- bandwidth guarantees needed for audio/video apps
- still an unsolved problem (chapter 7)
- Common practice: over-provision

Problems with packet switching

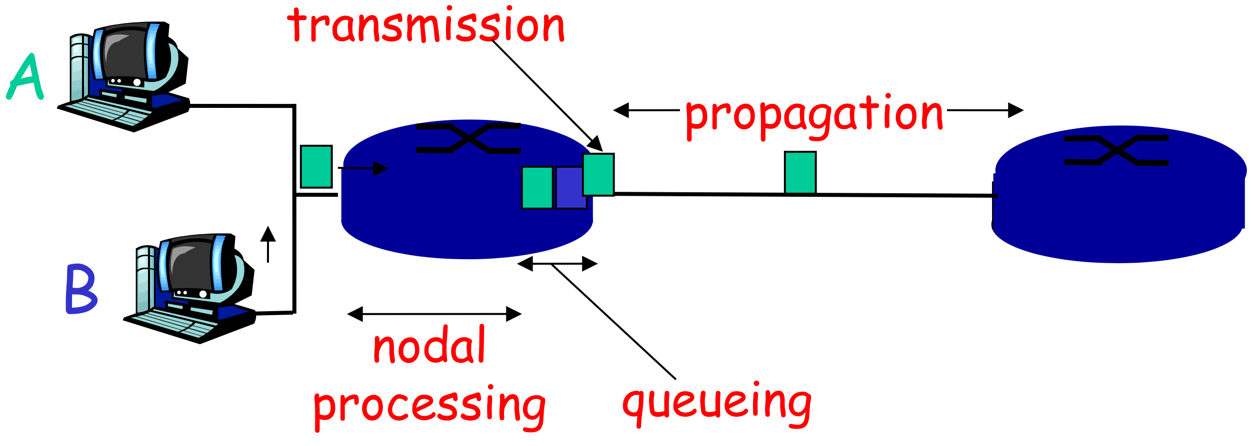
Packet loss and queuing delay
packets *queue* in router buffers

- packet arrival rate to link exceeds output link capacity
- packets queue, wait for turn
- when packet arrives to full queue, packet is dropped (aka lost)
  - lost packet may be retransmitted by previous node, by source end system, or not retransmitted at all



Four sources of packet delay

- 1. nodal processing:
  - check bit errors
  - determine output link
- 2. queueing
  - time waiting at output link for transmission
  - depends on congestion level of router



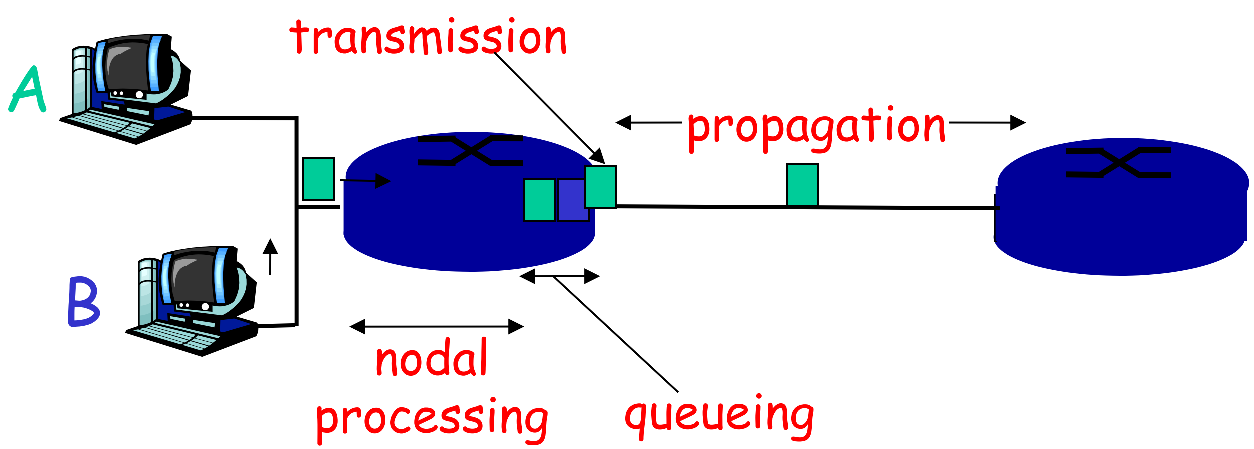
Delay in packet-switched networks

3. Transmission delay:

- R=link bandwidth (bps)
- L=packet length (bits)
- time to send bits into link = L/R

4. Propagation delay:

- d = length of physical link
- s = propagation speed in medium (~2×10^8 m/sec)
- propagation delay = d/s

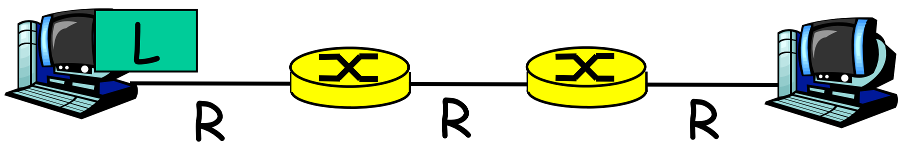


Nodal delay

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

- d_{proc} = processing delay
  - typically a few microsecs 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 microsecs to hundreds of msecs

Transmission delay example



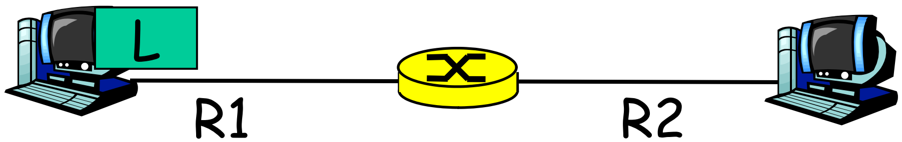
- Packet switching
  - Store-and-forward
  - Packet completely received before being transmitted to next node
- Takes L/R seconds to transmit (push out) packet of L bits on to link or R bps
- Entire packet must arrive at router before it can be transmitted on next link: *store and forward*
- delay = 3L/R (assuming zero propagation delay)

more on delay shortly …

Example:

- L = 7.5 Mbits
- R = 1.5 Mbps
- delay = 15 sec

Problem



- Consider a packet of length L that begins at system A, travels over one link to a packet switch over a second link to a destination end system.
- Let d_i, s_i and R_i denote the length, propagation speed and transmission rate of link i, for i=1,2.
- The packet switch delays each packet by d_{proc}
- Assuming no queuing delays, what is the total end-to-end delay?

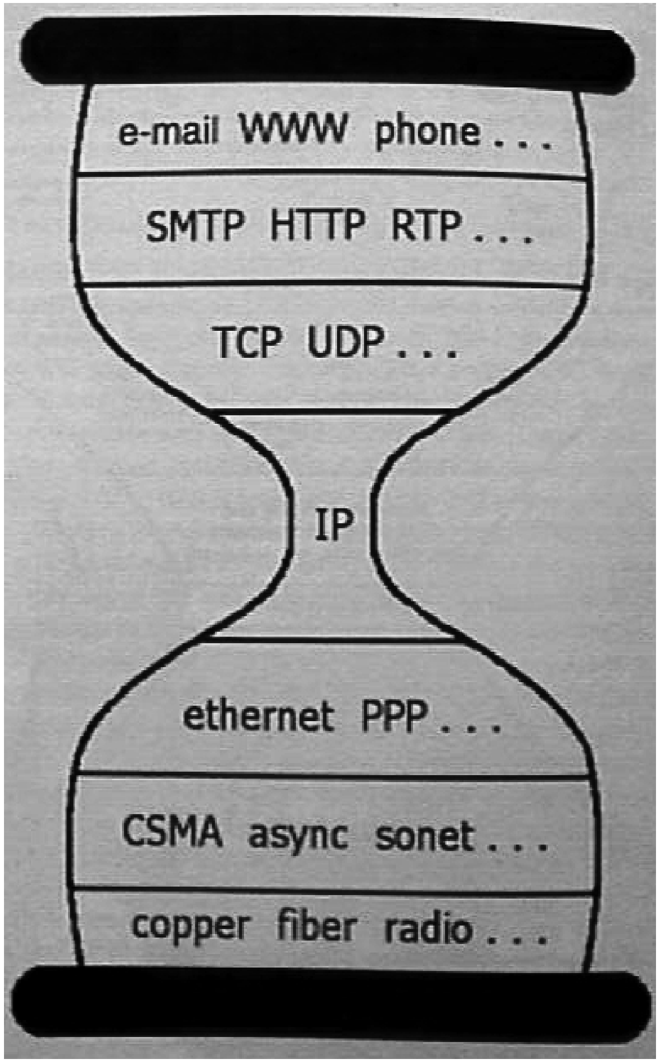
Case study: Packet Switching

- 1970/80s-current: Internet network
  - Variable bit rate
  - Mostly data
  - Fault-tolerant
  - Components not extremely reliable (versus phone components)
  - Distributed control and management

Why did the Internet win?

- Packet switching over circuit switching
- End-to-end principle and "Hourglass" design
- Layering of functionality
- Distributed design, decentralized control
- Superior organizational process

End-to-end principle and Hourglass design



End-to-end principle

- J. H. Saltzer, D. P. Reed and D. D. Clark "End-to-end arguments in system design", Transactions on Computer Systems, Vol. 2, No. 4, 1984
- http://www.acm.org/pubs/citations/journals/tocs/1984-2-4/p277-saltzer/

Hourglass design

- D. Clark, “The design philosophy of the DARPA Internet”, SIGCOMM 1988, August 16 - 18, 1988.

http://www.acm.org/pubs/citations/proceedings/comm/52324/p106-clark/

End-to-end principle

- Where to put the functionality?
  - In the network? At the edges?
- End-to-end functions best handled by end-to-end protocols
  - Network provides basic service: data transport
  - Intelligence and applications located in or close to devices at the edge
  - Violate principle as a performance enhancement
- Leads to innovation at the edges
  - Phone network: dumb edge devices, intelligent network
  - Internet: dumb network, intelligent edge devices

Hourglass design

- End-to-end principle leads to "Hourglass" design of protocols
- Only one protocol at the Internet level
  - Minimal required elements at narrowest point
- IP - Internet Protocol
  - http://www.rfc-editor.org/rfc/rfc791.txt
  - http://www.rfc-editor.org/rfc/rfc1812.txt
  - Unreliable datagram service
  - Addressing and connectionless connectivity
  - Fragmentation and assembly

Hourglass design

- Simplicity allowed fast deployment of multi-vendor, multi-provider public network
  - Ease of implementation
  - Limited hardware requirements (important in 1970s)
    - Is it relevant now with today's semiconductor speeds?
  - Eventual economies of scale
- Designed independently of hardware
  - No link-layer specific functions
  - Hardware addresses decoupled from IP addresses
  - IP header contains no data/physical link specific information
  - Allows IP to run over any fabric

Hourglass design

- Waist expands at transport layer
- Two dominant services layered above IP
- TCP – Transmission Control Protocol
  - Connection-oriented service
  - http://www.rfc-editor.org/rfc/rfc793.txt
- UDP – User Datagram Protocol
  - Connectionless service
  - http://www.rfc-editor.org/rfc/rfc768.txt

Hourglass design

- TCP – Transmission Control Protocol
  - Reliable, in-order byte-stream data transfer
    - Acknowledgements and retransmissions
  - Flow control
    - Sender won't overwhelm receiver
  - Congestion control
    - Senders won't overwhelm network
- UDP – User Datagram Protocol
  - Unreliable data transfer
  - No flow control
  - No congestion control

Hourglass design

- What uses TCP?
  - HTTP, FTP, Telnet, SMTP, NNTP, BGP, IMAP, POP
- What uses (mainly) UDP?
  - SNMP, NTP, NFS, RTP (streaming media, IP telephony, teleconferencing), multicast applications
  - Many protocols can use both
- Check out /etc/services on *nix or C:\WIN*\system32\services
- IANA
  - http://www.iana.org/assignments/port-numbers

Hourglass design

- Question?
  - Are TCP, UDP, and IP enough?
  - What other functionality would applications need?

Hourglass design

- Security?
  - IPsec/SSL/TLS
- Quality-of-service?
  - RSVP, int-serv, diff-serv
- Reliable, out-of-order delivery service?
  - SCTP
- Handling greedy sources?
- Accounting and pricing support?

End-to-end principle and the Hourglass design

- The good
  - Basic network functionality allowed for extremely quick adoption and deployment using simple devices
- The bad
  - New network features and functionality are impossible to deploy, requiring widespread adoption within the network
  - IP Multicast, QoS

Why did the Internet win?

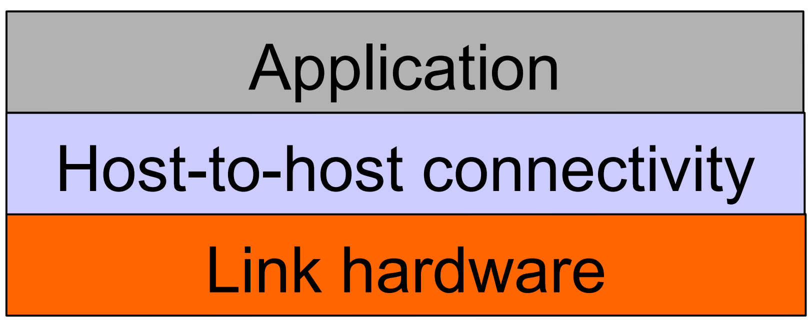
- Packet switching over circuit switching
- End-to-end principle and “Hourglass” design
- Layering of functionality
- Distributed design, decentralized control
- Superior organizational process

Layering

- Modular approach to network functionality
  - Simplifies complex systems
    - Each layer relies on services from layer below and exports services to layer above
  - Hides implementation
  - Eases maintenance and updating of system
    - Layer implementations can change without disturbing other layers (black box)

Layering

- Examples:
  - Topology and physical configuration hidden by network-layer routing
    - *Applications require no knowledge of routes*
    - New applications deployed without coordination with network operators or operating system vendors

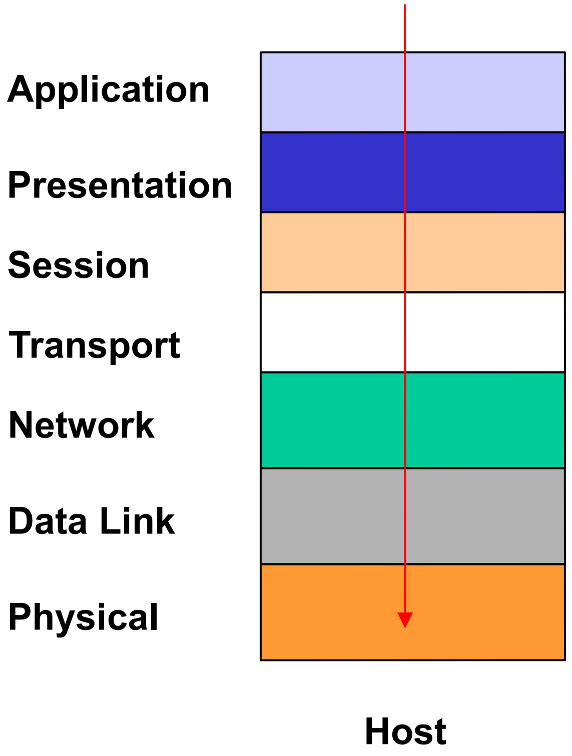


Layering essential in Protocols

- Set of rules governing communication between network elements (applications, hosts, routers)
- Protocols specify:
  - Interface to higher layers (API)
  - Interface to peer
    - Format and order of messages
    - Actions taken on receipt of a message
  - Interface defines interaction

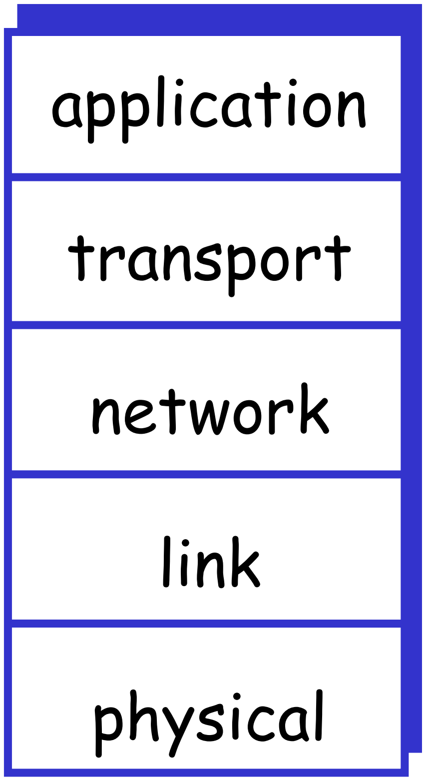
Layering: OSI Model

- Physical
  - how to transmit bits
- Data link
  - how to transmit frames
- Network
  - how to route packets host-to-host
- Transport
  - how to send packets end2end
- Session
  - how to tie flows together
- Presentation
  - byte ordering, formatting
- Application: everything else

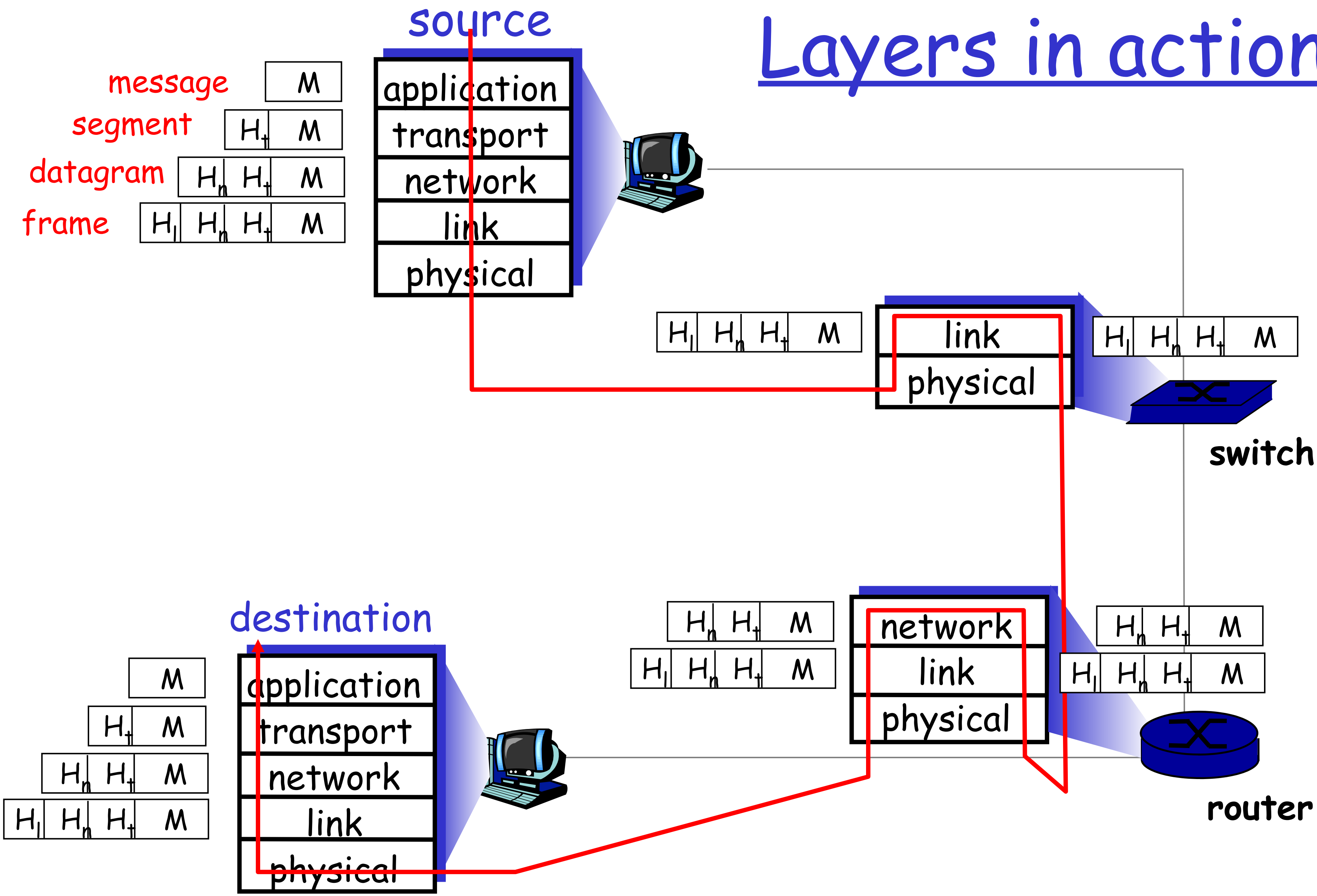


Layering: Internet protocols

- application: (L7 & L6 of OSI) supporting network applications
  - FTP, SMTP, HTTP
- transport: (L5 & L4 of OSI) host-host data transfer
  - TCP, UDP
- network: routing of datagrams from source to destination
  - IP, routing protocols
- link: data transfer between neighboring network elements
  - PPP, Ethernet
- physical: bits "on the wire"



Layers in action



Layering

- Is Layering always good?
  - Sometimes not..
    - Layer N may duplicate lower level functionality (e.g., error recovery)
    - Layers may need same info (timestamp, MTU)
    - Strict adherence to layering may hurt performance

Why did the Internet win?

- Packet switching over circuit switching
- End-to-end principle and "Hourglass" design
- Layering of functionality
- Distributed design, decentralized control
- Superior organizational process

Distributed design and control

- Requirements from DARPA
  - Must survive a nuclear attack
- Reliability
  - Intelligent aggregation of unreliable components
  - Alternate paths, adaptivity
- Distributed management & control of networks
  - Allows individual networks to independently develop without large amounts of coordination
  - Exceptions: TLDs and TLD servers, IP address allocation (ICANN)

Superior organizational process

- IAB/IETF process allowed for quick specification, implementation, and deployment of new standards
  - Free and easy download of standards
  - Rough consensus and running code
  - 2 interoperable implementations
  - Bake-offs
  - http://www.ietf.org/
- ISO/OSI
  - Comparison to IETF left as an exercise

Problem 1

- Consider the queuing delay in a router buffer (preceding an outbound link). Suppose all packets are L bits, the transmission rate is R bps and that N packets simultaneously arrive at the buffer every LN/R seconds. Find the average queuing delay of a packet

Internet history

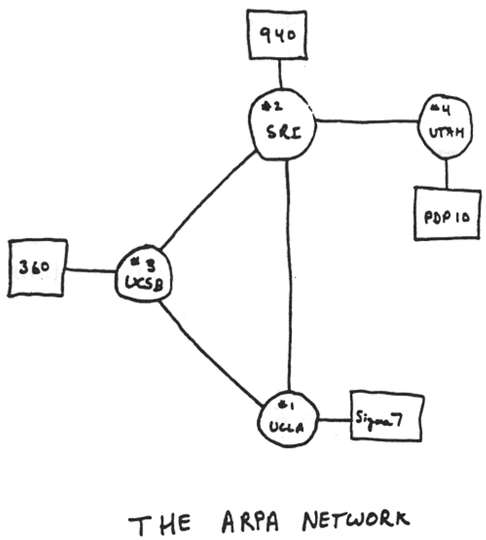
How old is the Internet?

- Guesses?
- Hint
  - It used to be the case that everyone in this class remembered the “pre-Internet” days

Internet History

1961-1972: Early packet-switching principles

- 1961: Kleinrock - queueing theory shows effectiveness of packet-switching
- 1964: Baran - packet-switching in early military nets
- 1967: ARPAnet conceived by Advanced Research Projects Agency
- 1969: first ARPAnet node operational
- 1972:
  - ARPAnet public demonstration
  - NCP (Network Control Protocol) first host-host protocol
  - first e-mail program
  - ARPAnet has 15 nodes



Internet History

1972-1980: Internetworking, new and proprietary nets

- 1970's: proprietary network architectures developed: DECnet, SNA, XNA
- 1974: Cerf and Kahn - architecture for interconnecting networks
- 1976: Ethernet at Xerox PARC
- 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
- 1983: smtp e-mail protocol defined
- 1983: DNS defined for name-to-IP-address translation
- 1985: ftp protocol defined
- 1988: TCP congestion control

- Late 1980s, Early 1990s: new national networks: Csnet, BITnet, NSFnet, Minitel
  - 100,000 hosts connected to confederation of networks

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 1990s: 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

2007:

- ~500 million hosts
- Voice, Video over IP
- P2P applications: BitTorrent (file sharing) Skype (VoIP), PPLive (video)
- more applications: YouTube, gaming
- wireless, mobility

Internet in a nutshell (protocols in practice)

A day in the life of an Internet host…

- Booting
  - Dynamically configure network settings
    - DHCP request
      - UDP (unreliable datagrams)
      - IP and data-link broadcast

Datalink broadcast header	IP broadcast 255.255.255.255	UDP header	DHCP request Host's datalink (MAC) address 00:50:7e:0d:30:20

    - DHCP response from listening server
      - IP address of host
      - Netmask (i.e. 255.255.255.0) to determine network ID
      - Default router

Datalink header 00:50:7e:0d:30:20	IP of Host	UDP Header	DHCP reply Host's network settings

A day in the life of an Internet host…

- Web request http://www.yahoo.com/index.html
  - Step #1: Locate DNS server

    if (netmask & IP_{Host} == netmask & IP_{DNS}) {

      DNS server on local network

      ARP for hardware address of IP_{DNS}

    } else {

      DNS server on remote network

      ARP for hardware address of $IP_{DefaultRouter}$

    }

    - ARP (Address Resolution Protocol)
      - IP address to hardware address mapping
      - Request broadcast for all hosts on network to see
      - Reply broadcast for all hosts to cache

A day in the life of an Internet host…

- Step #2: ARP request and reply

Datalink header broadcast	ARP request: Who has MAC address of IP addr "X"? (X=next-hop router, dns server) MAC address of requestor

Datalink header MAC of requestor or broadcast addr	ARP reply: MAC address of "X" is a:b:c:d:e:f

A day in the life of an Internet host…

- Step #3: DNS request/reply
  - UDP, IP, data-link header
  - DNS request to local DNS server from host

Datalink header (DNS server or next-hop router)	IP of DNS Server	UDP Header	DNS request www.yahoo.com "A" record request

  - DNS reply from local DNS server to host

Datalink header (host)	IP of host	UDP Header	DNS reply www.yahoo.com is 216.115.105.2

A day in the life of an Internet host…

- Step #4: TCP connection establishment
  - TCP 3-way handshake (SYN, SYN-ACK, ACK)
  - Session establishment to support reliable byte stream

Datalink header (next-hop router)	IP of 216.115.105.2	TCP Header SYN

Datalink header (host)	IP of host	TCP Header SYN-ACK

Datalink header (next-hop router)	IP of 216.115.105.2	TCP Header ACK

A day in the life of an Internet host…

- Step #5: HTTP request and reply
  - HTTP (application data), TCP, IP, data-link header
  - HTTP request

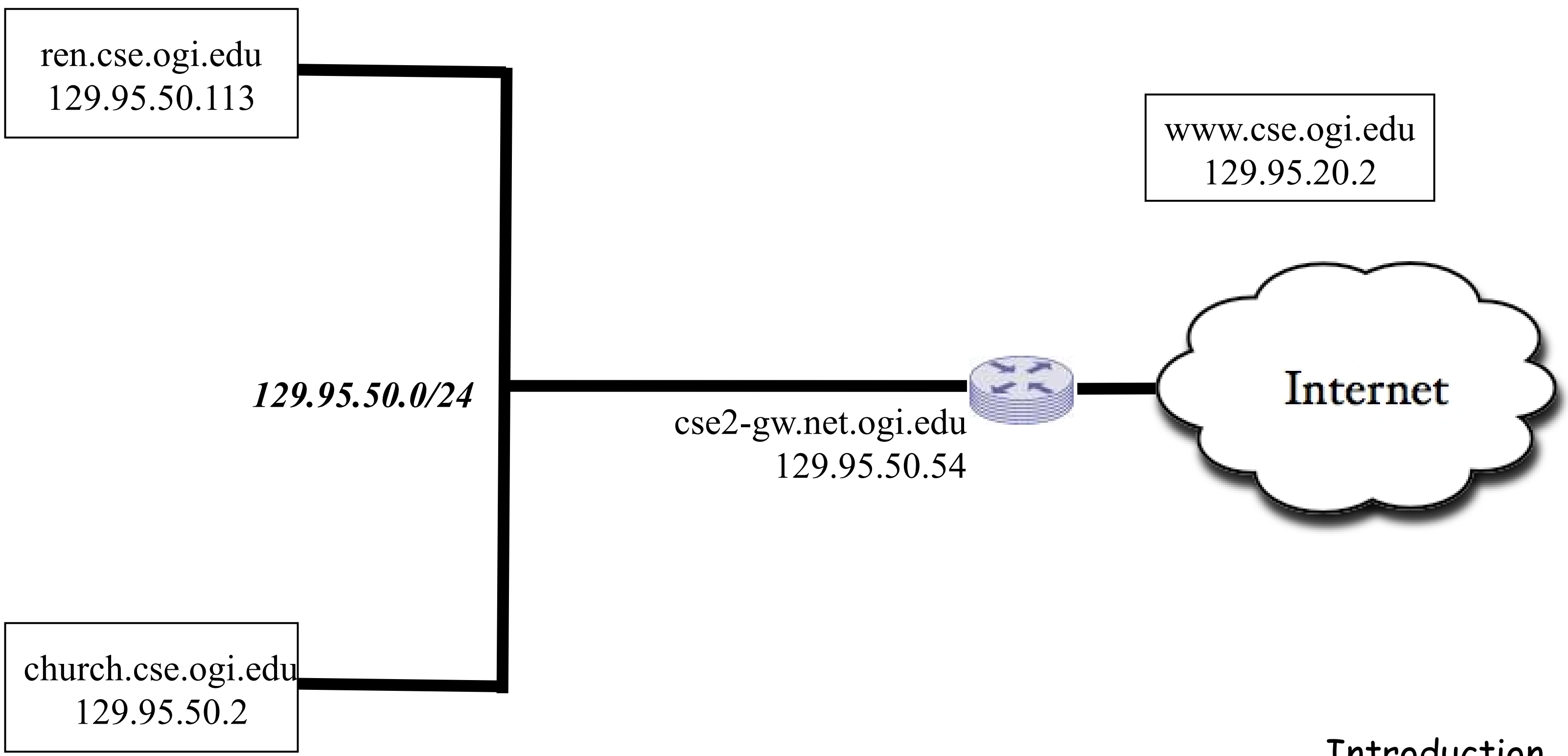
Datalink header (next-hop router)	IP of 216.115.105.2	TCP Header	HTTP request GET /index.html HTTP/1.0

  - HTTP reply

Datalink header (host)	IP of host	TCP Header	HTTP reply HTTP/1.0 200 OK Date: Mon, 24 Sep 2001 Content-Type: text/html <html> …. </html>

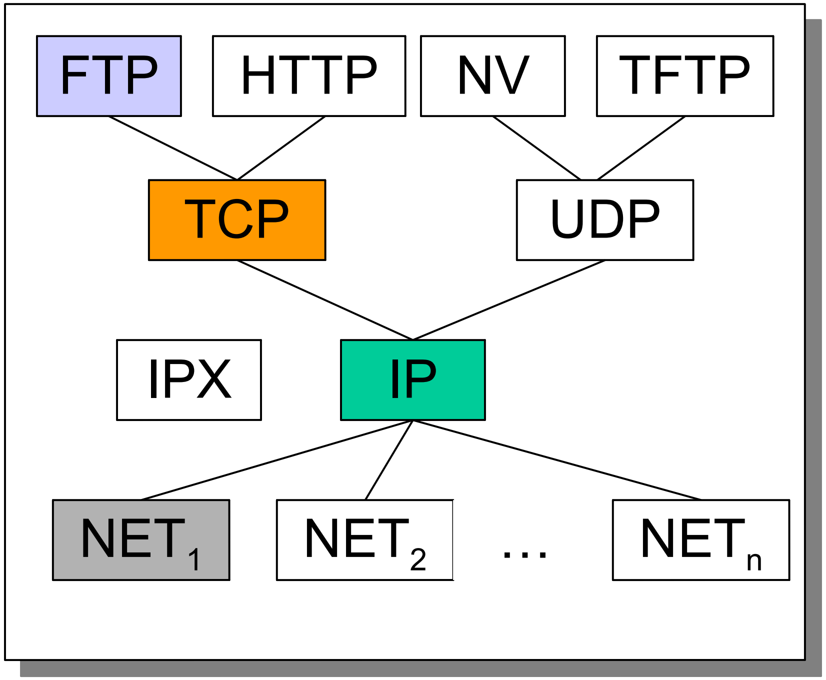
tcpdump example

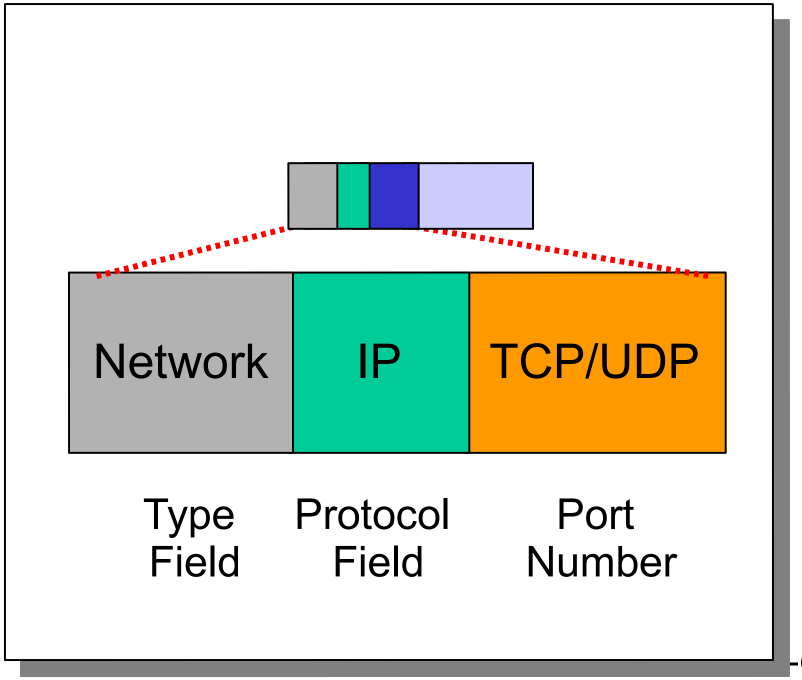
- http://thefengs.com/wuchang/work/courses/cs594/trace.txt



A day in the life of an Internet host…

- Role of TCP and UDP?
- Demultiplex at end hosts.
  - Which process gets this request?





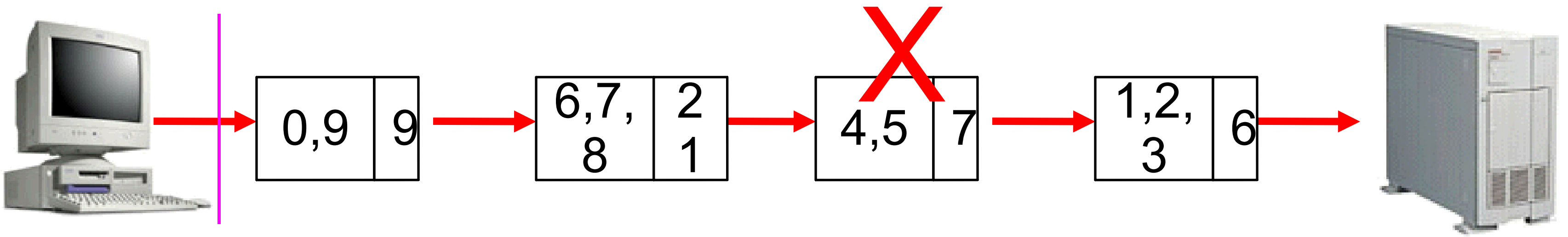
A day in the life of an Internet host….

- What about….
  - Reliability
    - Corruption
    - Lost packets
  - Flow and congestion control
  - Fragmentation
  - Out-of-order delivery
- The beauty of TCP, IP, and layering
  - All taken care of transparently

What if the Data is Corrupted?

Problem: Data Corruption

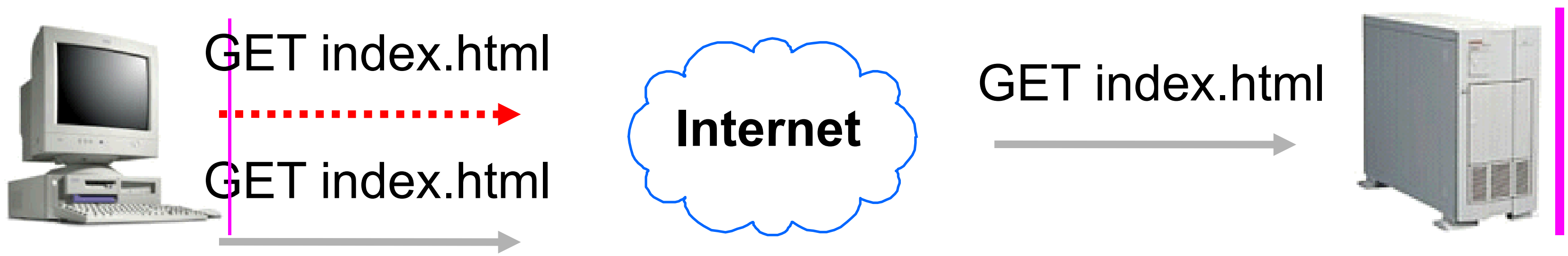
Solution: Add a *checksum*



What if the Data is Lost?

Problem: Lost Data

Solution: Timeout and Retransmit

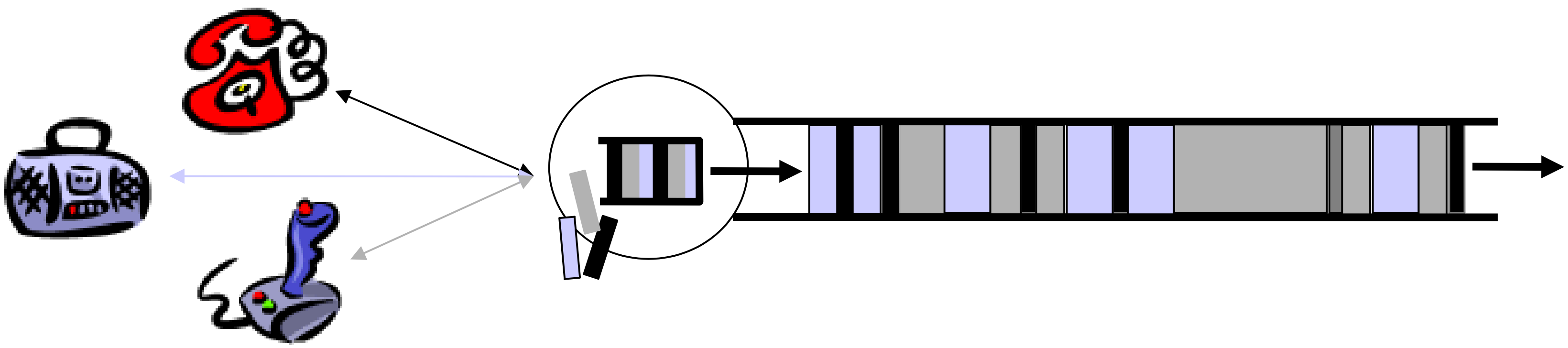


What if receiver has no resources (flow control)?

Problem: Overflowing receiver buffers

Solution: Receiver advertised window

What if Network is Overloaded?



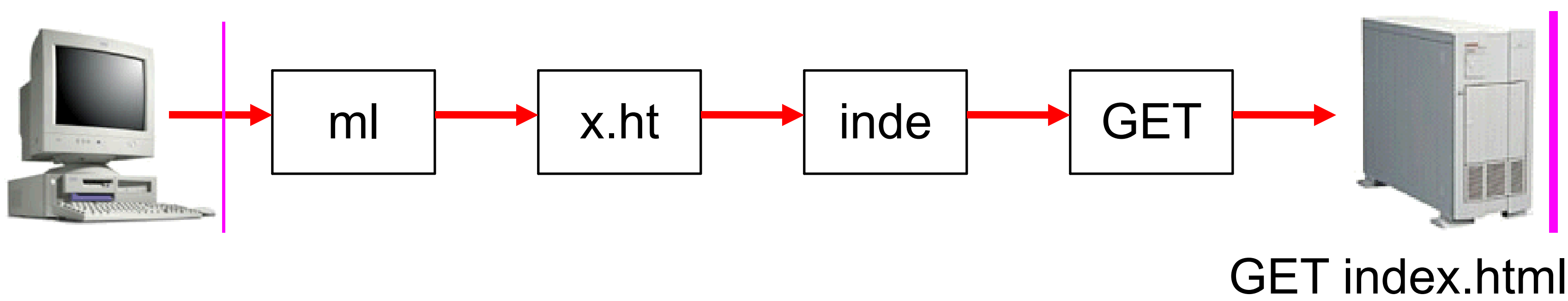
- Short bursts: buffer
- What if buffer overflows?
  - Packets dropped and retransmitted
  - Sender adjusts rate until load = resources
- Called "Congestion control"

What if the Data Doesn't Fit?

Problem: Packet size

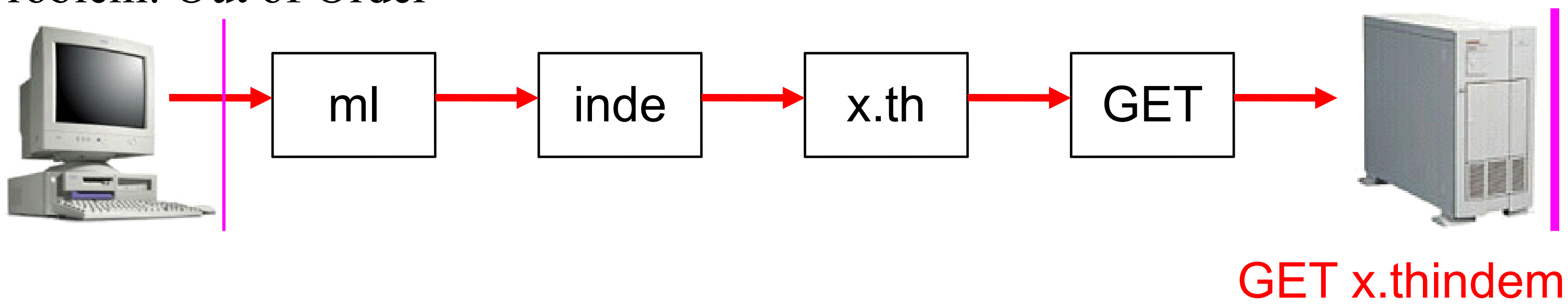
- On Ethernet, max IP packet is 1.5kbytes
- Typical web page is 10kbytes

Solution: Fragment data across packets

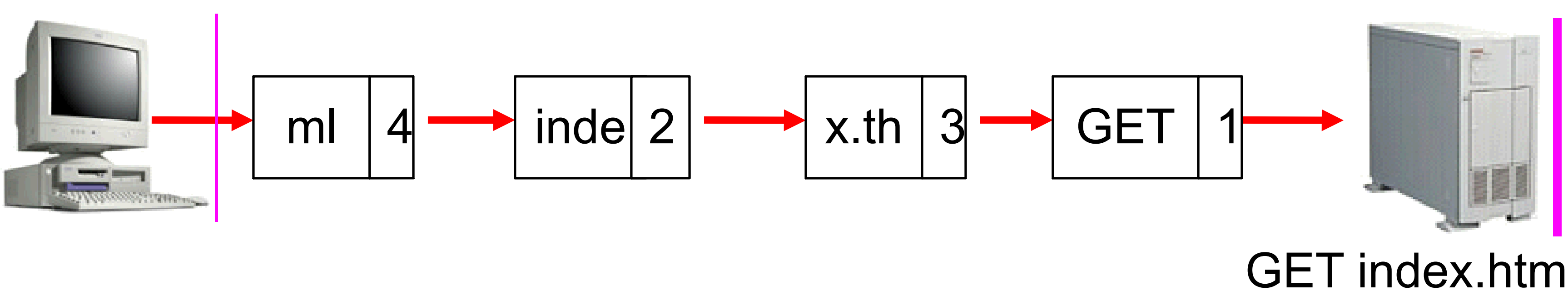


What if the Data is Out of Order?

Problem: Out of Order



Solution: Add Sequence Numbers



The rest of the course

- From birds-eye view, we will now focus on specific components
- Review these lectures for perspective when looking at the components
- Mostly classical material with some references to newer technologies

Acknowledgements

- Material taken from course slides by Srini Seshan's Computer Networking course at http://www.cs.cmu.edu/~srini/15-744/S01/

Extra slides

Layering

- Need for exposing underlying layers for optimal application performance
  - D. Tennenhouse and D. Clark. Architectural Considerations for a New Generation of Protocols. SIGCOMM 1990.
  - Application Layer Framing (ALF)
    - Enable application to process data as soon as it can
    - Expose application processing unit (ADU) to protocols
  - Integrated Layer Processing (ILP)
    - Layering convenient for architecture but not for implementations
    - Combine data manipulation operations across layers

Residential access: cable modems

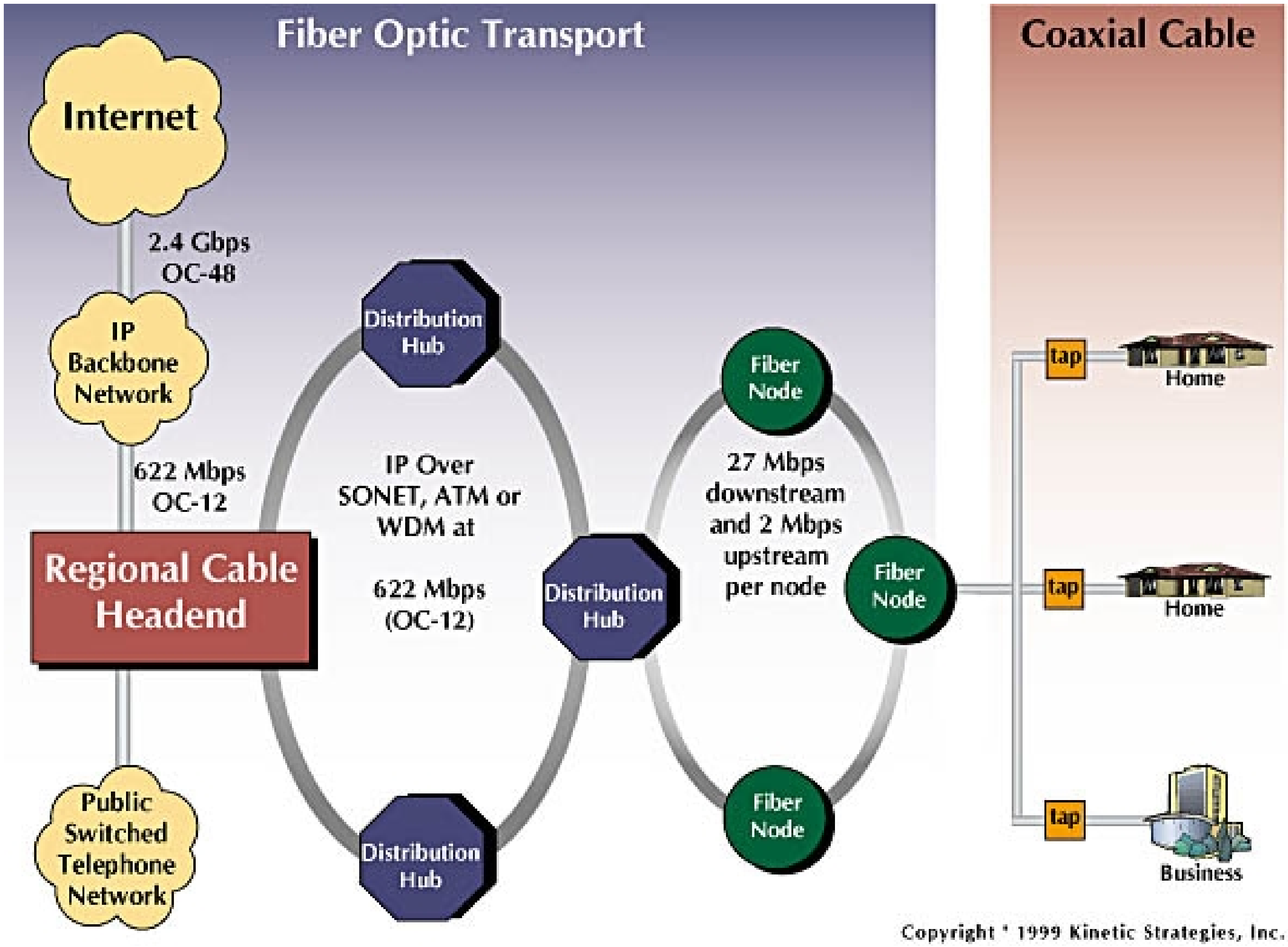
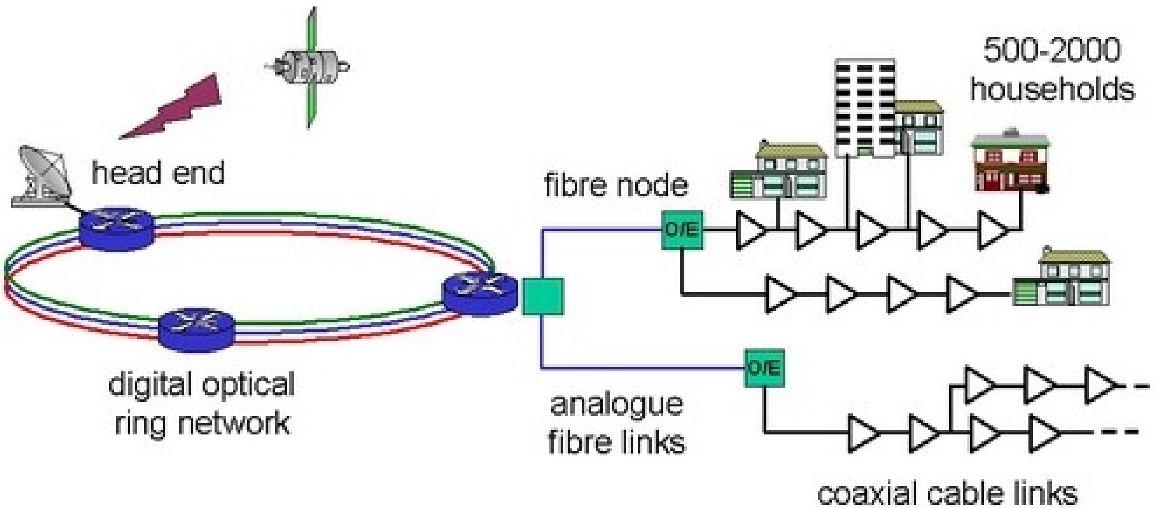
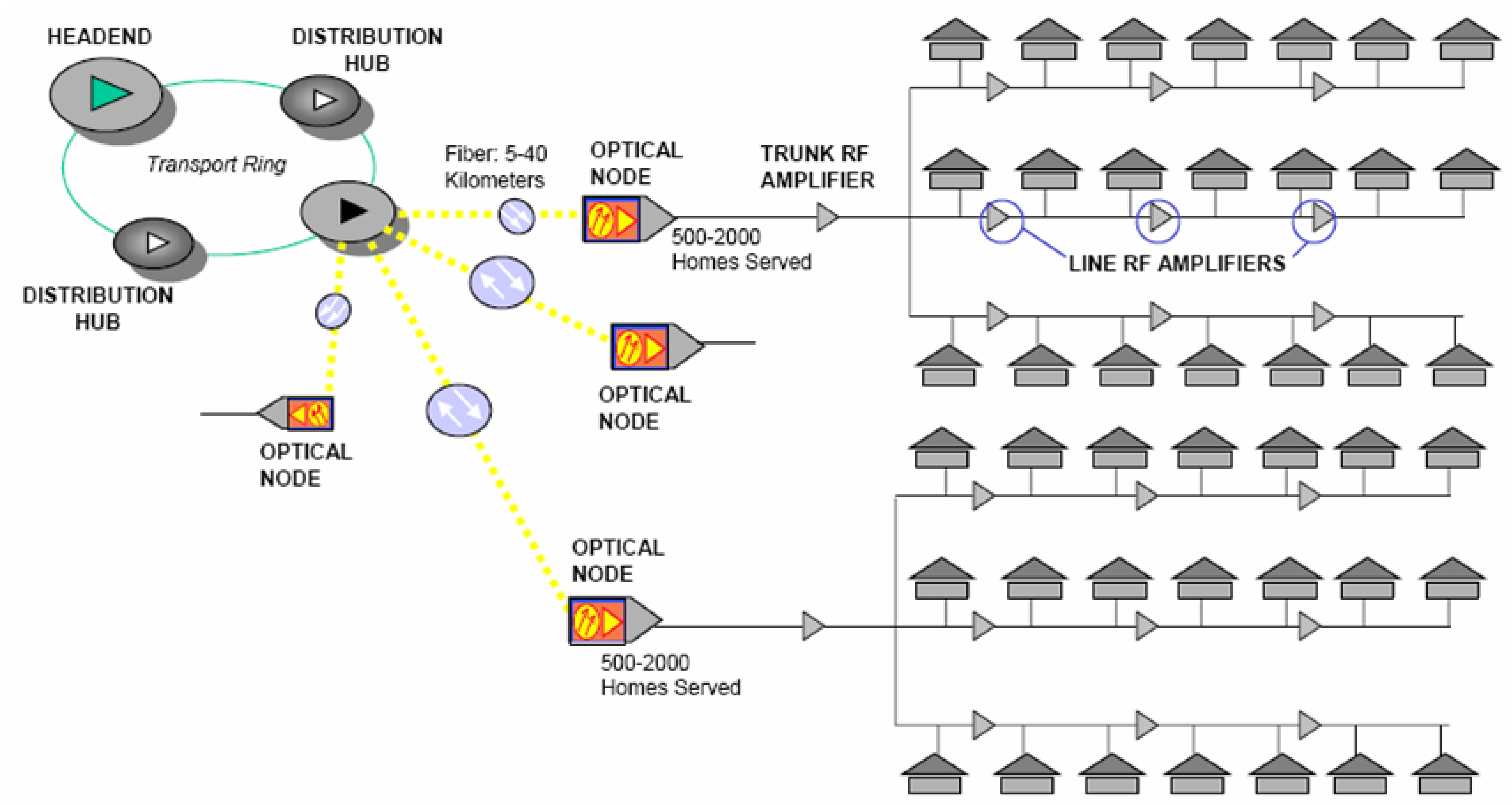


Diagram: http://www.cabledatacomnews.com/cmic/diagram.html

Cable Network Architecture: Overview



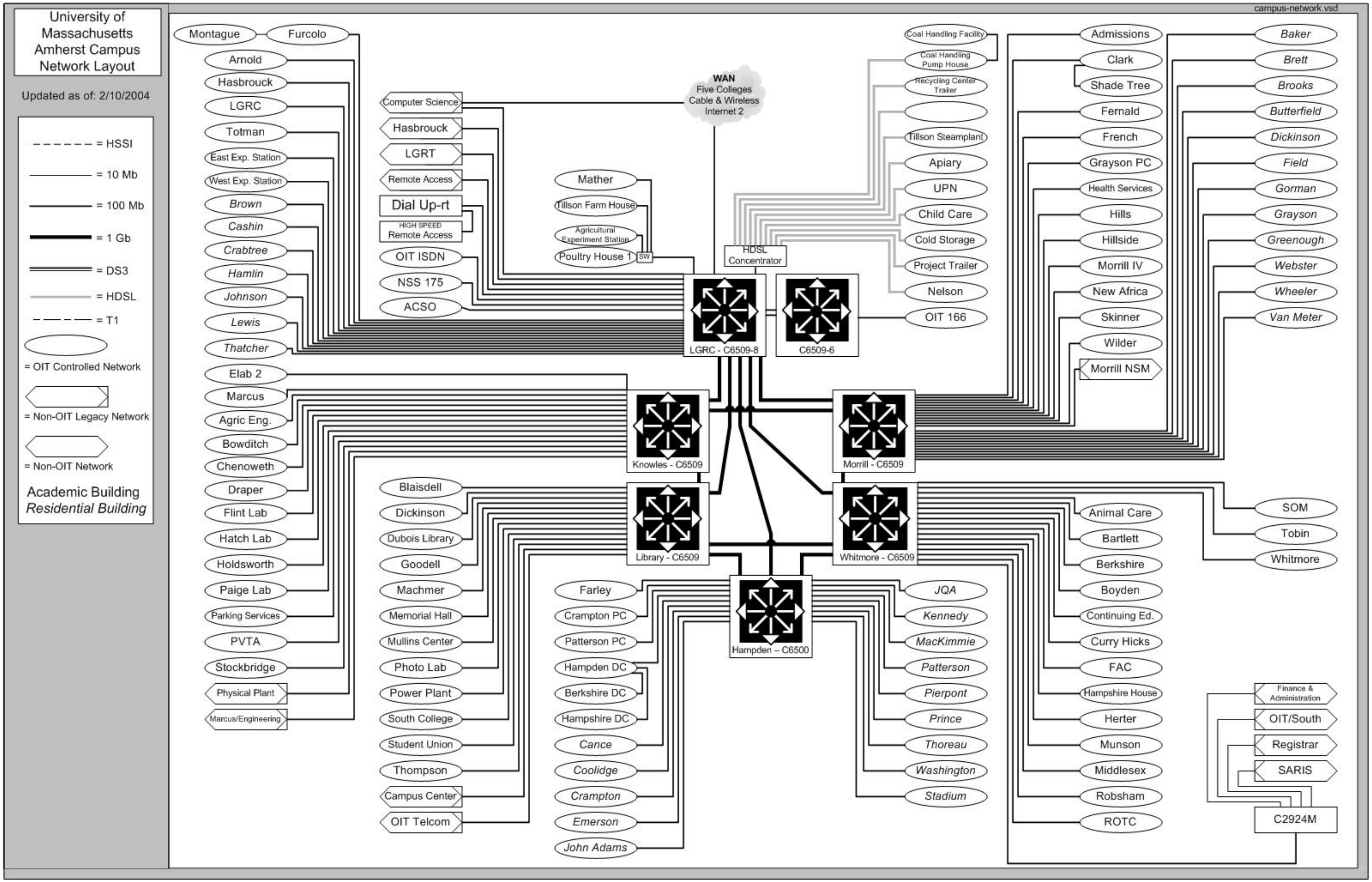
Residential access: cable modems



Residential access: cable modems

- HFC: hybrid fiber coax
  - asymmetric: up to 27Mbps downstream, 2 Mbps upstream
  - Limited upstream bandwidth due to multiple noise sources vs. downstream case with one controllable noise source (headend)
- network of cable and fiber attaches homes to ISP router
  - homes share access to router
- deployment: available via cable TV companies

UMass Campus Network



Internet History

- Those who ignore the past are doomed to repeat it
  http://www.worldcom.com/about_the_company/cerfs_up/
- Where did it come from?
- Who built it?
- Why does it work?
- Most of the original designers (old-timers) still around and active…
  - internet-history-request@postel.org

Internet timeline

- 1961 Kleinrock proposes packet switching
- 1962 Licklider proposes “galactic” network
  - Goes to DARPA as head of CS research
- 1966 Roberts proposes galactic network using packet switching
  - Goes to ARPA to build it (ARPANET)
- 1968 RFQs to build routers (Interface Message Processors)
- 1968 Kahn separates hardware addresses from network addresses
  - ARPANET to run over any hardware
- 1969 Crocker initiates RFC notes to document protocols
  - Freely available
- 1969 First node of ARPANET UCLA (September)
- 1969 4-node ARPANET at UCLA, SRI, Utah, UCSB (December)
  - Initial hosts.txt name database
- 1970 Crocker develops NCP (host-to-host protocol for applications)
  - Precursor to TCP
- 1972 Tomlinson develops e-mail (@)

Internet timeline

- 1972 Issues with NCP and ARPANET arise
  - NCP relied on ARPANET for end2end reliability (assumed no packet loss)
    - Can not work over satellite or packet radio links
  - NCP addressing tied to ARPANET
- 1973 Kahn redesigns protocols
  - Communication on a “best-effort” basis
    - Least-common denominator
    - End points in charge of retransmission, reassembly, flow control
  - No per-flow state in gateways between networks
    - Simple, avoids adaptation and recovery from failure
  - Addressing
    - 8-bit network number, 24 bit host number
    - Fails to forsee development of the LAN
    - Later split into Class A (national), B (regional), and C (LAN)
- 1974 Kahn, Cerf develop TCP (with IP included) (December)
  - IP later separated for unreliable applications, UDP added
- 1981 RFCs for TCP and IP
  - Initial applications: file transfer, e-mail, voice/video, login

Internet timeline

- 1978-1983: NCP replaced by TCP/IP
  - Implementations of TCP/IP on many platforms (Clark)
  - Mandate from to switch all users on ARPANET from NCP to TCP/IP (1980)
    - Not well received
    - One-day shutoff of NCP in mid-1982 makes people angry, but not sufficiently convincing
    - January 1983: NCP banned from ARPANET "Flag Day" -> The Internet is born
    - Some older computers allowed to operate with old NCP for a short time
    - Full transition takes several months, finishes at end of 1983
    - "I survived the TCP/IP transition" buttons (Y2K bug?)
  - Will there be an "IPv6 day?"

Internet timeline

- 1982-1985 Application protocols
  - SMTP (1982)
  - Mockapetris develops DNS (1983)
  - telnet (1983)
  - ftp (1985)
- 1980s Jealous non-interoperable competitors
  - DOE: MFENet (Magnetic Fusion Energy scientists)
  - DOE: HEPNet (High Energy Physicists)
  - NASA: SPAN (Space physicists)
  - NSF: CSNET (CS community)
  - NSF: NSFNet (Academic community) 1985
  - AT&T: USENET with Unix, UUCP protocols
  - Academic networks: BITNET (Mainframe connectivity)
  - Xerox: XNS (Xerox Network System)
  - IBM: SNA (System Network Architecture)
  - Digital: DECNet
  - UK: JANET (Academic community in UK) 1984

Internet timeline

- 1986-1995 NSFNet (Jennings/Wolff with funding assist from Al Gore)
  - Network for academic/research community
  - Selects TCP/IP as mandatory for NSFNet
  - Builds out wide area networking infrastructure
  - Develops strategy for developing and handing it over eventually to commercial interests
    - Prohibit commercial use of NSFNet to encourage commercial backbones
    - Leads to PSINet, UUNET, ANS, CO+RE backbone development
- 1989 WWW
  - Tim Berners-Lee develops initial web browser supporting URLs, HTTP, HTML

Internet timeline

- Early 1990s Privatization
  - ARPANET decommissioned (1990)
  - NSFNet decommissioned (1995) ($200 million spent from 1986-1995)
- Early 1990s Architectural issues
  - Address depletion
    - Multi-class addressing to break 8/24 network/host split in address bits
  - Routing table explosion
    - Hierarchy and CIDR
  - Congestion
    - TCP congestion control
- 1994 Andreessen
  - Mosaic web browser

Packet switching

- Kleinrock, MIT (July 1961)
  - Theoretical feasibility of communications using packets instead of circuits
  - L. Kleinrock, "Information Flow in Large Communication Nets", RLE Quarterly Progress Report, July 1961.
  - L. Kleinrock, Communication Nets: Stochastic Message Flow and Delay, Mcgraw-Hill (New York), 1964.

Conceptual "Internet"

- J.C.R. Licklider, W. Clark, MIT (August 1962)
  - "On-line Man Computer Communication"
  - "Galactic network" concept of globally interconnected set of computers
  - Licklider goes to DARPA as head of computer research program (Oct. 1962)

ARPANET

- Roberts, (1966)
  - Puts idea of galactic computer network and packet switching together
  - Goes to DARPA as program manager
    - Plans for building “ARPANET” based on system
    - L. Roberts, "Multiple Computer Networks and Intercomputer Communication", ACM Gatlinburg Conf., October 1967.

ARPANET

- Structure and specification (August 1968)
  - RFQ to build IMPs (Interface Message Processors)
    - Packet switches which route packets
    - BBN (Bolt, Beranek, and Newman) wins contract
  - Kahn at BBN updates ARPANET design
    - Run over any fabric (separation of hardware and network addresses)
    - Support for multiple independent networks
- First node UCLA (Sept. 1969)
  - 4 node ARPANET (Dec. 1969) SRI, UCSB, Utah
  - Initial hostname/address database (flat file: hosts.txt)

RFCs

- 1969: Crocker establishes RFC series of notes
  - Official protocol documentation
    - Printed on paper and snail mailed at first
    - Then available via ftp and now http
    - Open and free access to RFCs mandated
    - Effective, positive feedback loop
    - Key to quick development process ("time-to-market")
    - Has changed considerably as of late...
- Jon Postel RFC editor and protocol number assignment

NCP

- Crocker
  - Connectivity implemented
  - Require a host-to-host protocol standard for two ends to talk to each other
  - NCP (Network Control Protocol) defined (Dec. 1970)
  - Precursor to TCP
  - Deployed from 1971-1972
  - Allows applications to be developed on top of network

E-mail

- BBN's Tomlinson (Mar. 1972)
  - Time-shared systems at the time allow users to leave messages for each other
  - Extended to remote systems
  - Writes first e-mail application to send and read
  - Infamous "@" used

Internetting

- ARPANET not the only network in town...
  - International Network Working Group (Sept. 1973)
  - Goal: run protocols over packet satellite net, packet radio net, and wired ARPANET
  - Problems
    - NCP can only address networks connected to IMPs on ARPANET
    - NCP relied on ARPANET for end2end reliability
    - NCP assumed no packet loss: applications halt upon loss
    - NCP had no end-end host error control
  - Kahn redesigns protocols for internetworking

Internetting

- Kahn's Architecture
  - Each network stands alone
    - No changes required to connect to Internet
    - Communication between networks handled by gateways
  - Communication on a "best-effort" basis
    - Least-common denominator
    - Source in charge of retransmission
    - Host-to-Host flow control (sliding windows and acks)
  - Black boxes interconnecting networks (gateways and routers) have no per-flow information
    - Simple, avoids complicated adaptation and recovery from failure
  - No global control at the operations level

Internetting

- Other issues
  - Host-to-Host data pipelining (multiple packets en route)
  - Gateway interprets IP headers for routing and performs fragmentation to other networks
  - end2end checksums, reassembly of fragments, duplicate detection at end-hosts (much of TCP's virtual circuit model)
  - Global addressing via 32-bit address (IP's limitation)
    - 8-bit network number, 24 bit host number
    - Fails to forsee development of the LAN
      - Later split into Class A (national), B (regional), and C (LAN)
  - Interfaces to operating systems
    - R. Kahn, Communications Principles for Operating Systems. Internal BBN memo, Jan. 1972.

Internetting

- Kahn brings in Cerf (Stanford) to help implement ideas on multiple OS platforms
  - V. Cerf, R. Kahn "A protocol for packet network intercommunication" IEEE Transactions on Communications, May 1974
  - TCP draft produced (includes IP) Dec. 1974
- ARPA sponsors 3 groups to implement on hosts
  - Stanford (Cerf), BBN (Tomlinson), UCL (Kirstein)
  - All interoperate
- IP later separated (not all apps need reliability)
  - UDP added

Internetting

- IP
  - Internet Protocol (Sept. 1981) Postel
  - http://www.rfc-editor.org/rfc/rfc791.txt
- TCP
  - Transmission Control Protocol (Sept. 1981) Postel
  - http://www.rfc-editor.org/rfc/rfc793.txt
- Initial applications
  - Goal is resource sharing of systems on ARPANET
    - File transfer
    - Remote login (telnet)
    - E-mail
    - Packet voice, packet video (late 1970s)

Application protocols

- SMTP
  - Simple Mail Tranfer Protocol (Aug. 1982) Postel
    - http://www.rfc-editor.org/rfc/rfc821.txt
- DNS
  - Hostnames server, SRI (Mar. 1982) Harrenstien
    - http://www.rfc-editor.org/rfc/rfc811.txt
  - Current hierarchical architecture (Aug. 1982) Su, Postel
    - http://www.rfc-editor.org/rfc/rfc819.txt
  - Domain Name System standard (Nov. 1983) Mockapetris
    - http://www.rfc-editor.org/rfc/rfc882.txt
    - http://www.rfc-editor.org/rfc/rfc882.txt

Application protocols

- Telnet
  - Telnet protocol (May 1983) Postel, Reynolds
    - http://www.rfc-editor.org/rfc/rfc854.txt
- FTP
  - File transfer protocol (Oct. 1985) Postel, Reynolds
    - http://www.rfc-editor.org/rfc/rfc959.txt

NSFNet

- Structure
  - 6 nodes with 56kbs links
    - Jointly managed exchange points
    - Statistical, non-metered peering agreements
    - Cost-sharing of infrastructure
  - Seek out commercial, non-academic customers
    - Help pay for and expand regional academic facilities
    - Economies of scale
    - Prohibit commercial use of NSFNet to encourage commercial backbones
    - Leads to PSINet, UUNET, ANS, CO+RE backbone development

TCP/IP software proliferation

- Widespread dispersal leads to critical mass
- Case study: Berkeley Unix
  - Unix TCP/IP available at no cost (DoD)
  - Incorporates BBN TCP/IP implementation
  - Large-scale dissemination of code base
  - Eventual economies of scale

Privatization

- Commercial interconnection
  - US Federal Networking Council (1988-1989)
  - MCI Mail allowed
- ARPANET decommissioned (1990)
- NSFNet decommissioned (1995)
  - 21 nodes with multiple T3 (45Mbs) links
  - Regional academic networks forced to buy national connectivity from private long haul networks
  - TCP/IP supplants and marginalizes all others to become THE bearer service for the Internet
  - Total cost of NSF program?

$200 million from 1986-1995

Growing pains

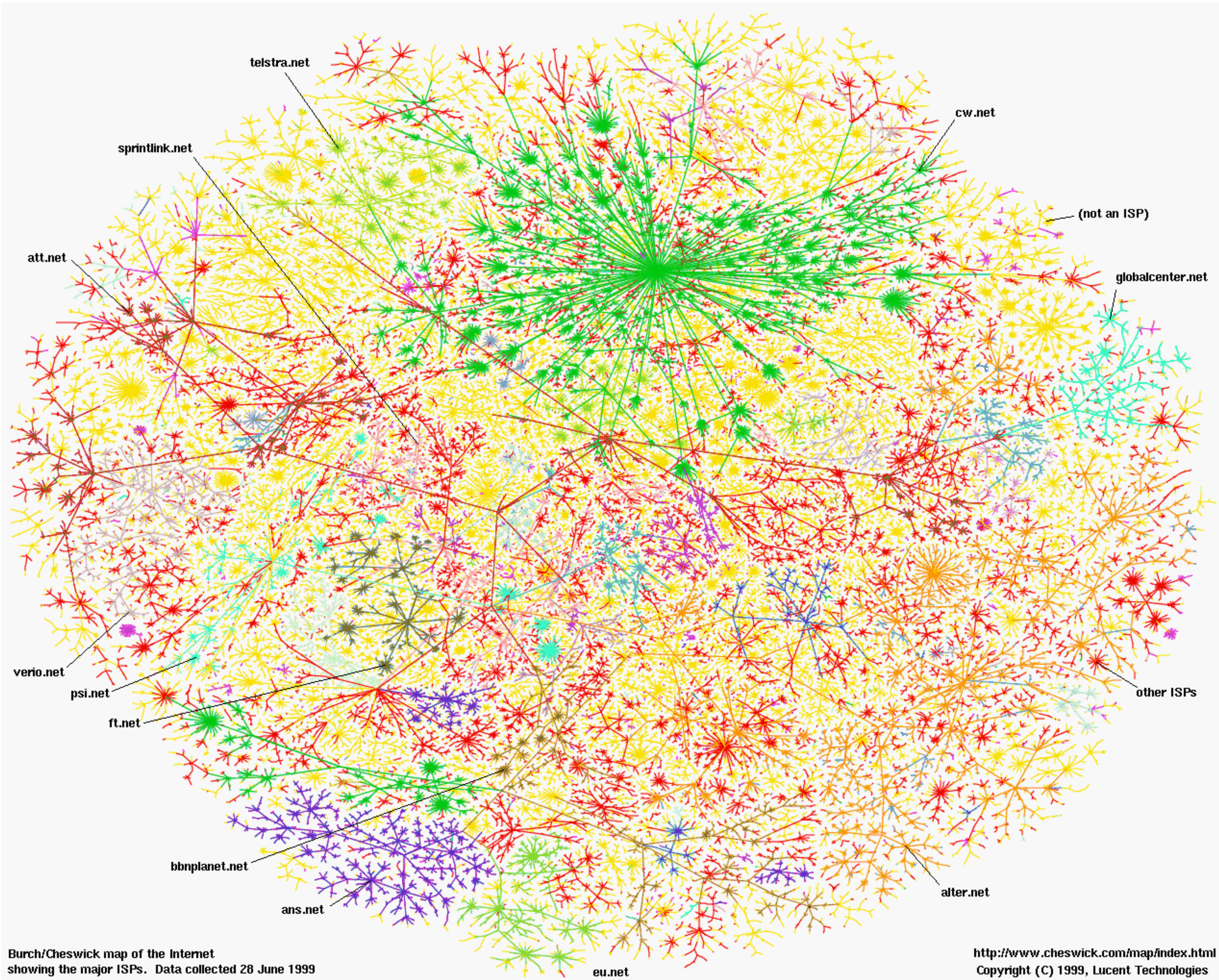
- Address depletion
  - Multi-class addressing to break up 8-bit network/24-bit host
- Explosion of networks
  - Routing initially flat, each node runs the same distributed routing algorithm
  - Moved to hierarchical model to match commercial reality (IGP, EGP)
    - Reduces table size, distributes control (a bit)
  - Classless addressing (CIDR)
    - Reduces table size
- Congestion
  - Network "brown-outs", congestion collapse
  - Add congestion control to TCP protocol, not IP

WWW

- CERN (European Organization for Nuclear Research)
  - Berners-Lee, Caillau work on WWW (1989)
  - First WWW client (browser-editor running under NeXTStep)
  - Defines URLs, HTTP, and HTML
  - Berners-Lee goes to MIT and LCS to start W3C
    - Responsible for evolving protocols and standards for the web
  - http://www.w3.org/People

WWW

- NCSA (National Center for Supercomputing Applications)
  - Federally funded research center at University of Illinois at Urbana-Champaign
  - Andreessen: Mosaic and eventually Netscape (1994)
  - http://www.dnai.com/~thomst/marca.html


telstra.net
cw.net
sprintlink.net
(not an ISP)
att.net
globalcenter.net
verio.net
psi.net
ft.net
other ISPs
bbnplanet.net
ans.net
alter.net
Burch/Cheswick map of the Internet
showing the major ISPs. Data collected 28 June 1999
eu.net
http://www.cheswick.com/map/index.html
Copyright (C) 1999, Lucent Technologies

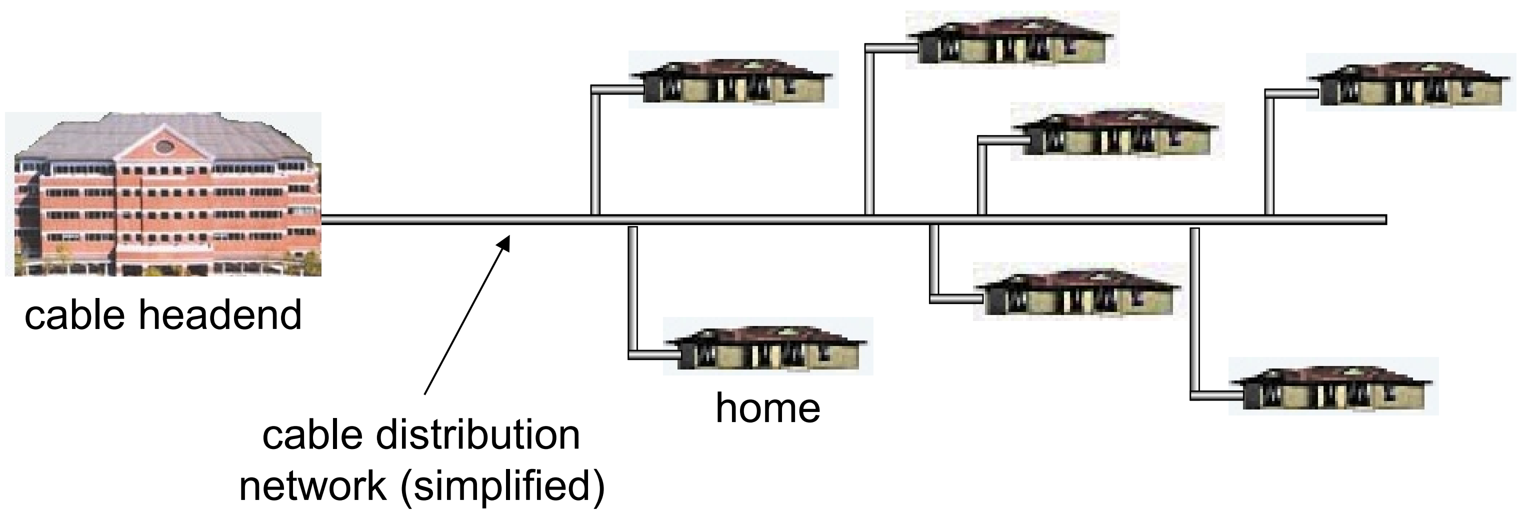
Today's Internet

Residential access

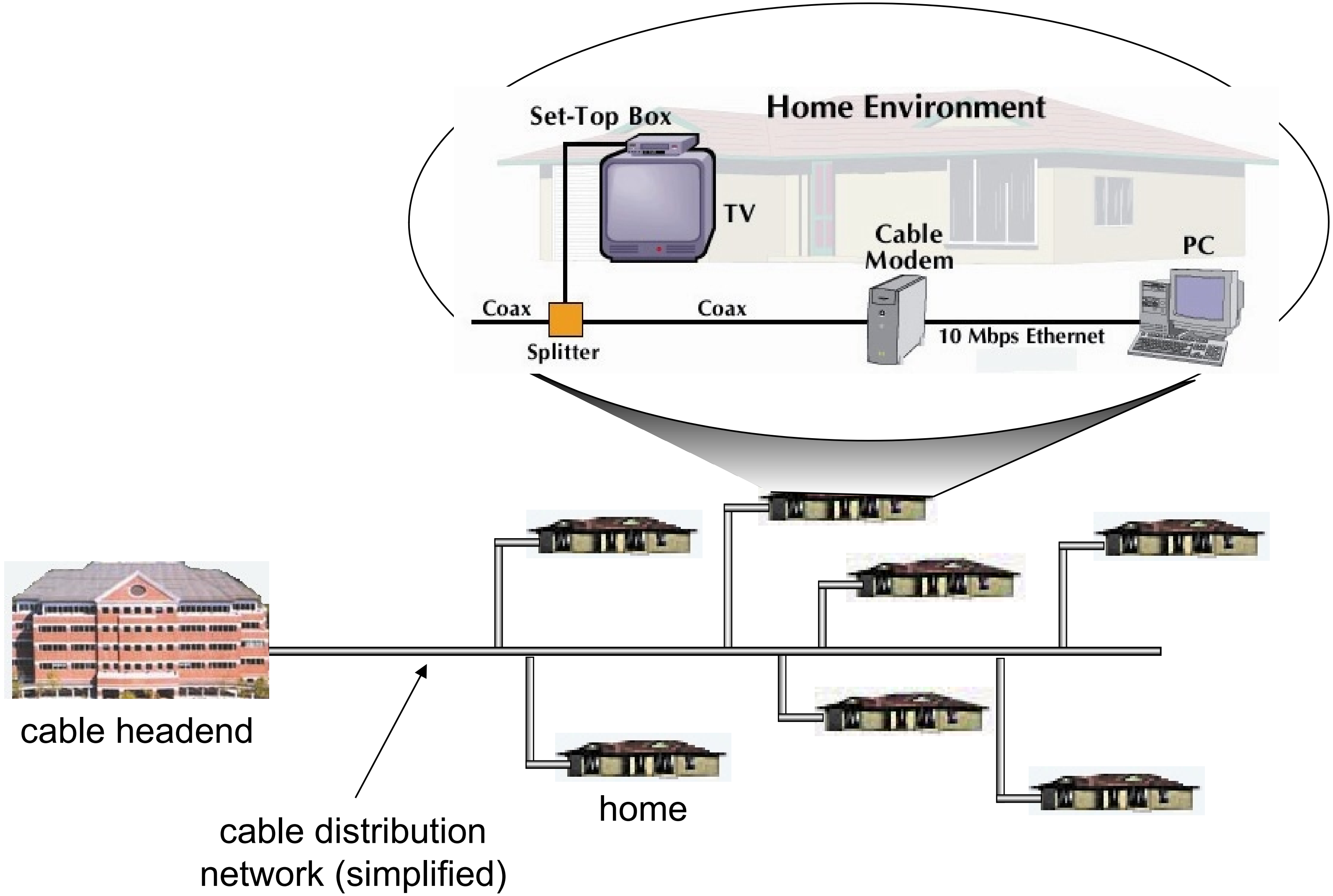
- Driven by networks already in place to the home
  - Most common
    - Cable TV lines
    - Phone lines
  - Less common
    - Satellite television
    - Power lines

Residential access: cable

- Originally one-way distribution

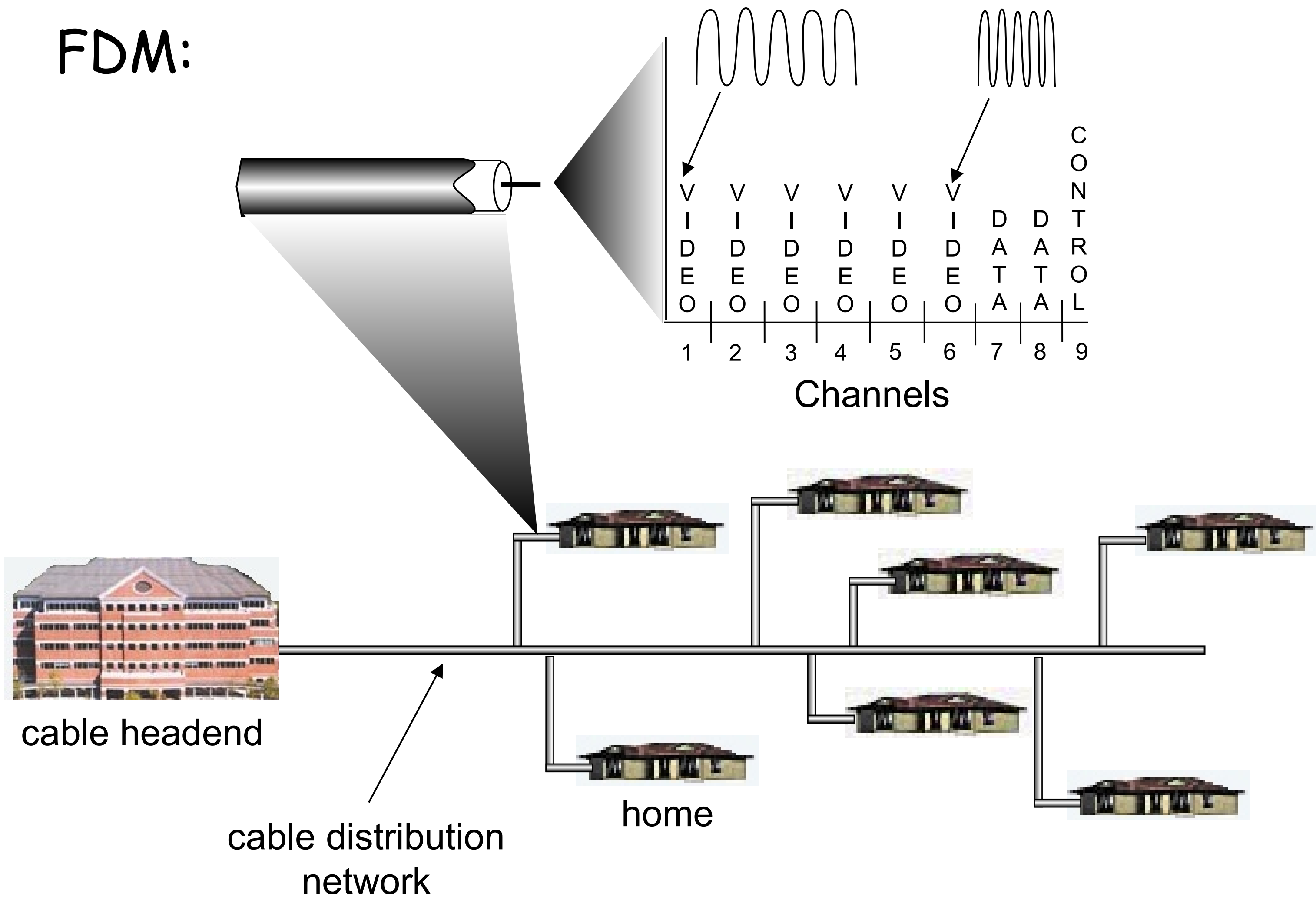


Residential access: cable



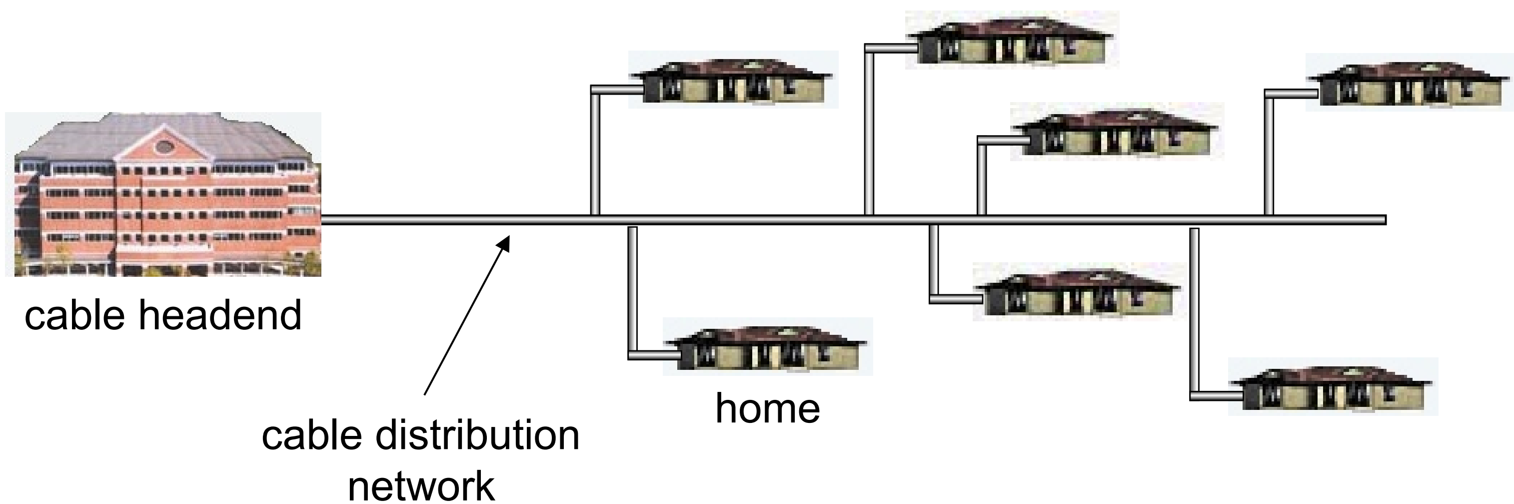
Residential access: cable

FDM:

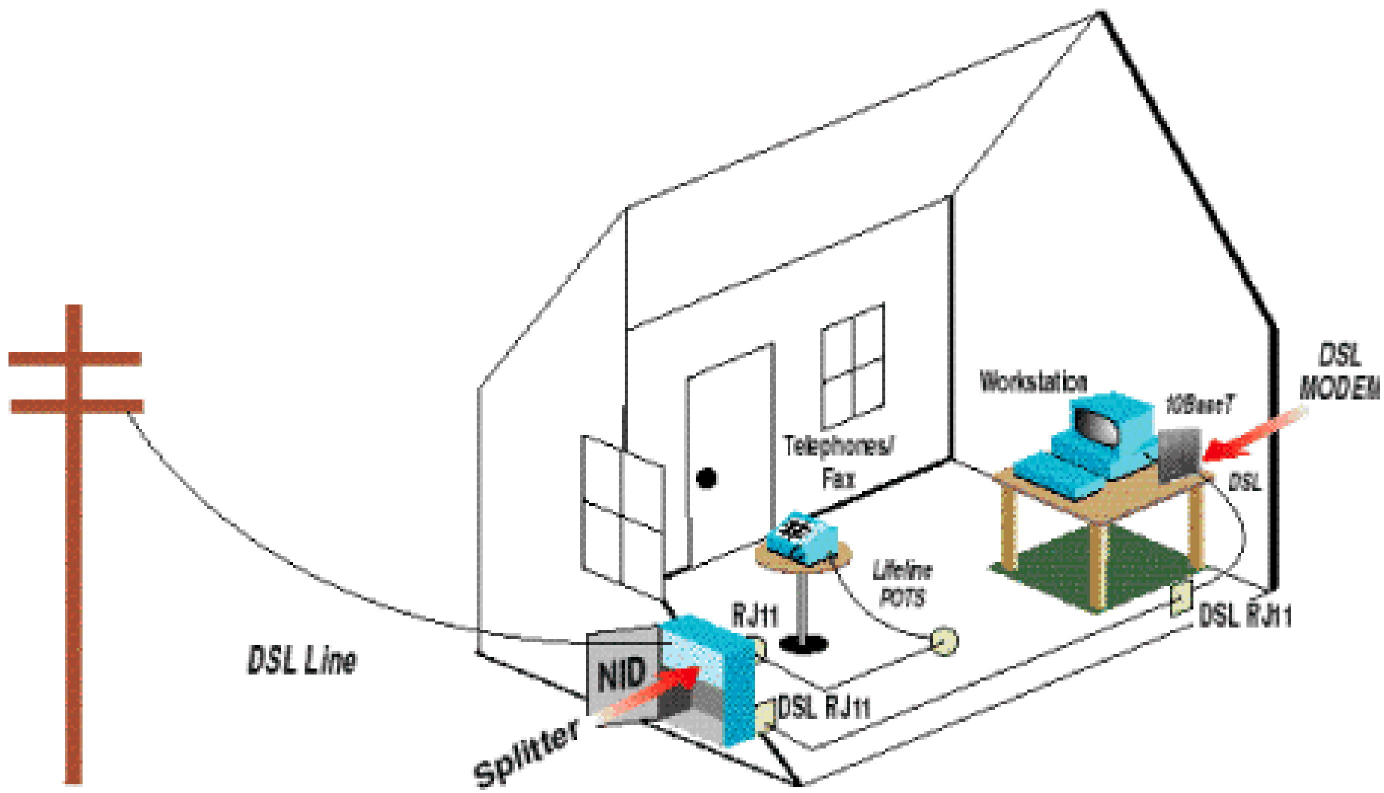


Residential access: cable

- Download faster than upload
  - Noise issues with one source vs. many



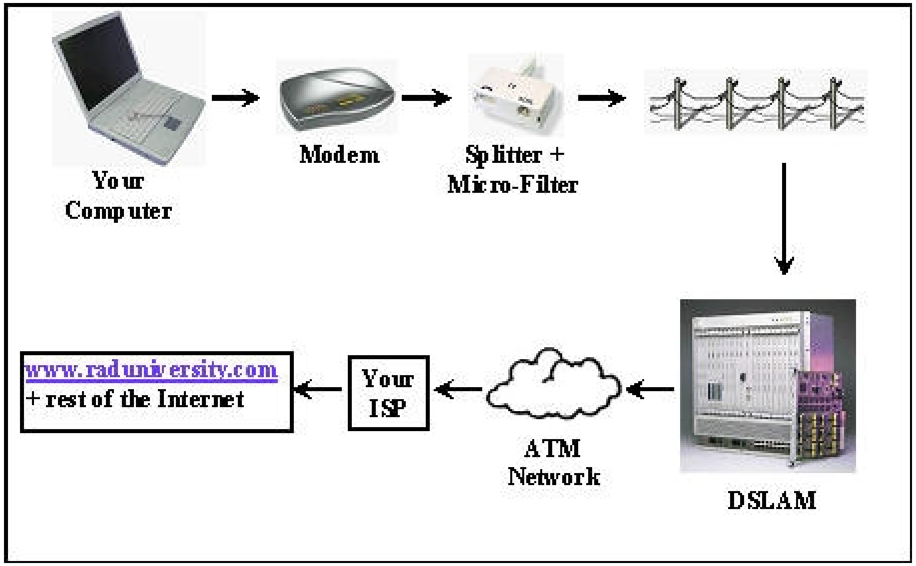
Residential access: DSL

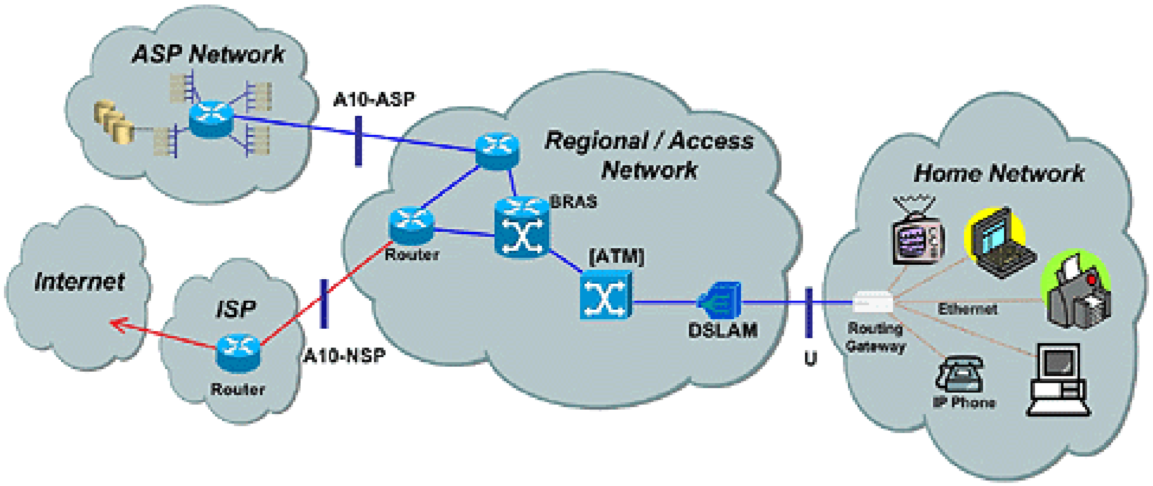


Residential access: DSL

- Uses high-frequency spectrum
  - Data superimposed onto voice using high-frequencies on existing telephone line
  - Voice transmitted in low-frequency spectrum
  - Transmissions may disrupt each other
    - Low-pass filters typically added to protect legacy devices and DSL modems from each other

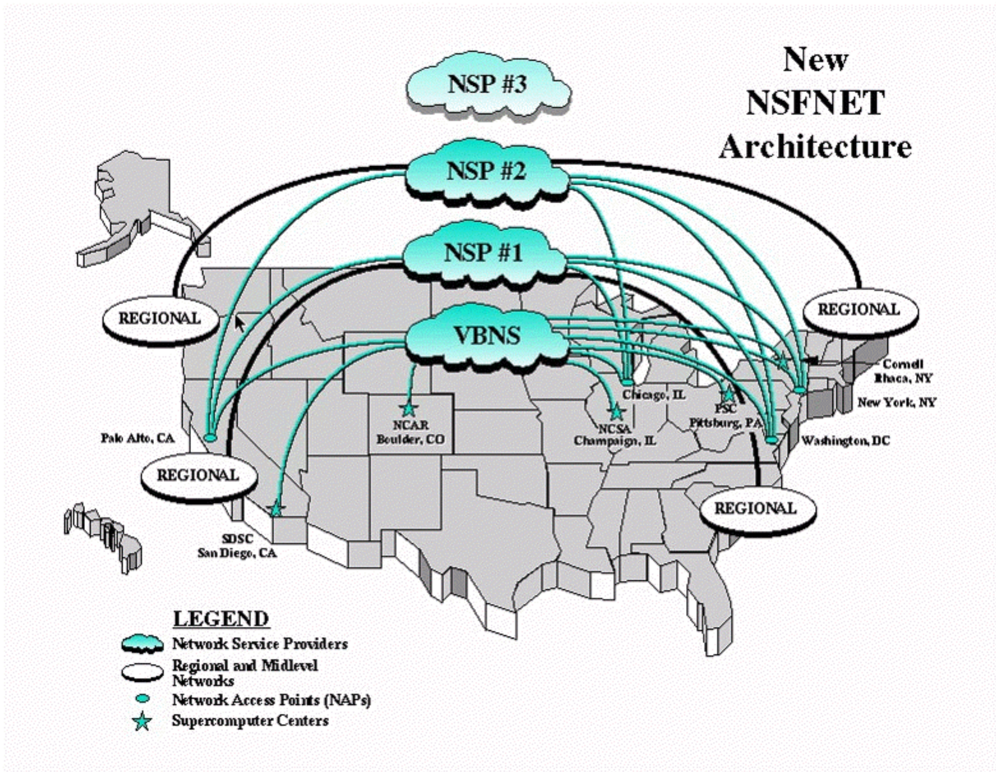
Residential access: DSL





Internet structure: network of networks

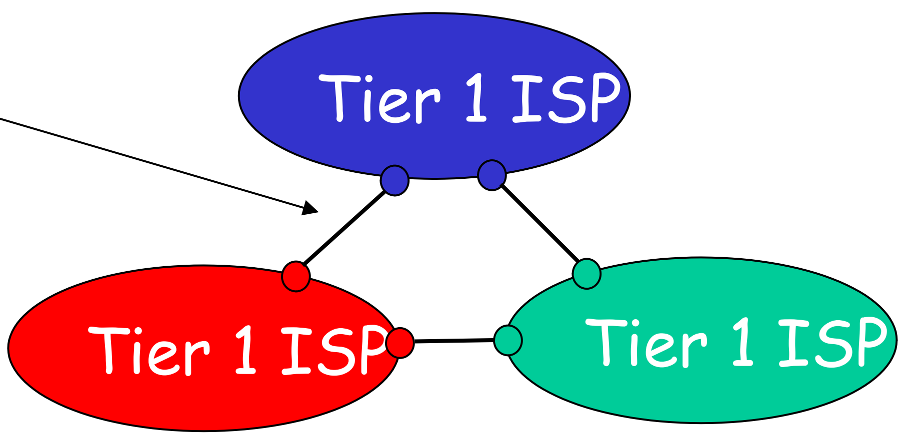
- Influenced by decommissioning of NSFNet
  - Academic network connecting NSF's supercomputing sites
  - NSF wanted out of the ISP business
  - Provided peering points for multiple competing commercial ISPs



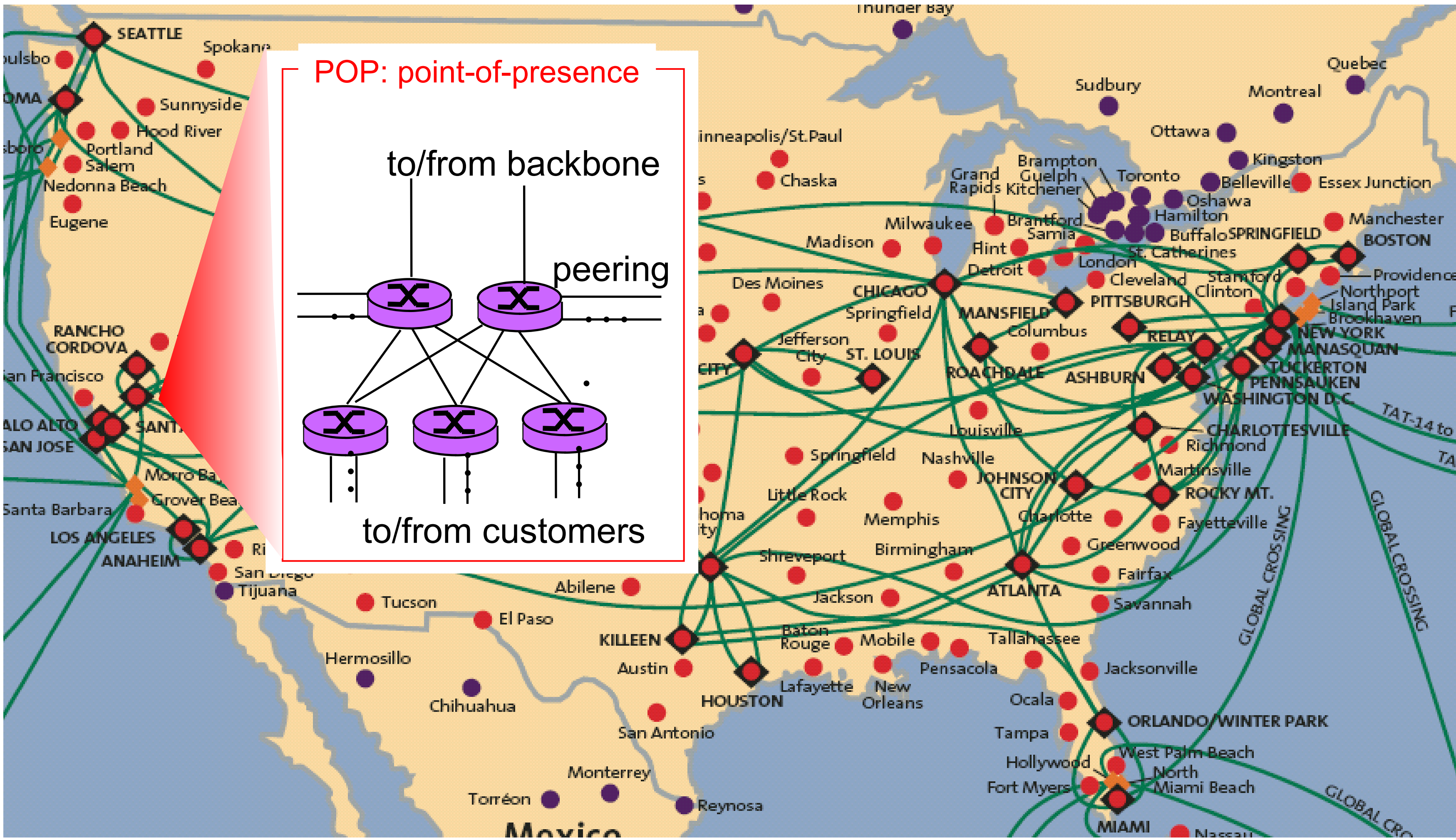
Internet structure: network of networks

- roughly hierarchical
- at center: "tier-1" ISPs (e.g., Verizon, Sprint, AT&T, Cable and Wireless), national/international coverage
  - treat each other as equals
  - Peers with every other network to reach Internet

Tier-1 providers interconnect (peer) privately



Tier-1 ISP: e.g., Sprint

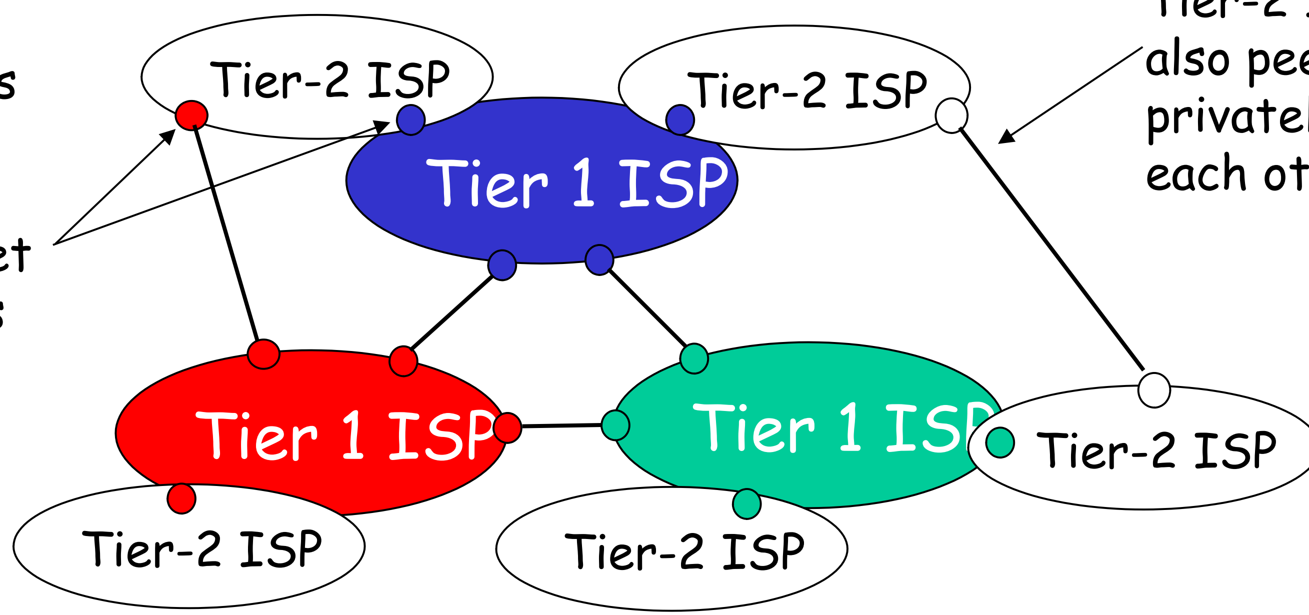


Internet structure: network of networks

- "Tier-2" ISPs: smaller (often regional) ISPs
  - Peers with some networks, but still purchases IP transit from tier-1 ISP to reach some portion of the Internet

Tier-2 ISP pays tier-1 ISP for connectivity to rest of Internet

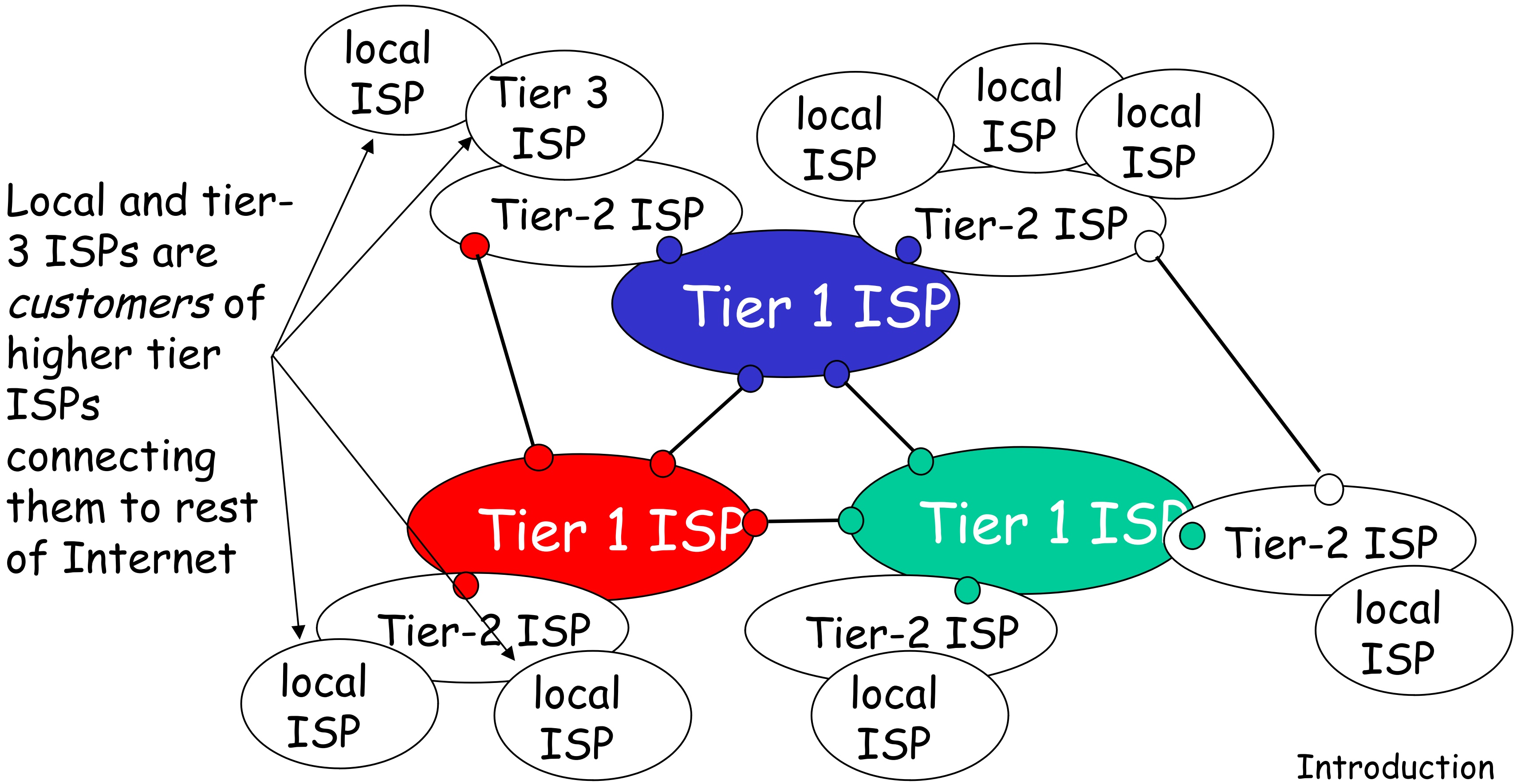
- tier-2 ISP is *customer* of tier-1 provider



Tier-2 ISPs also peer privately with each other.

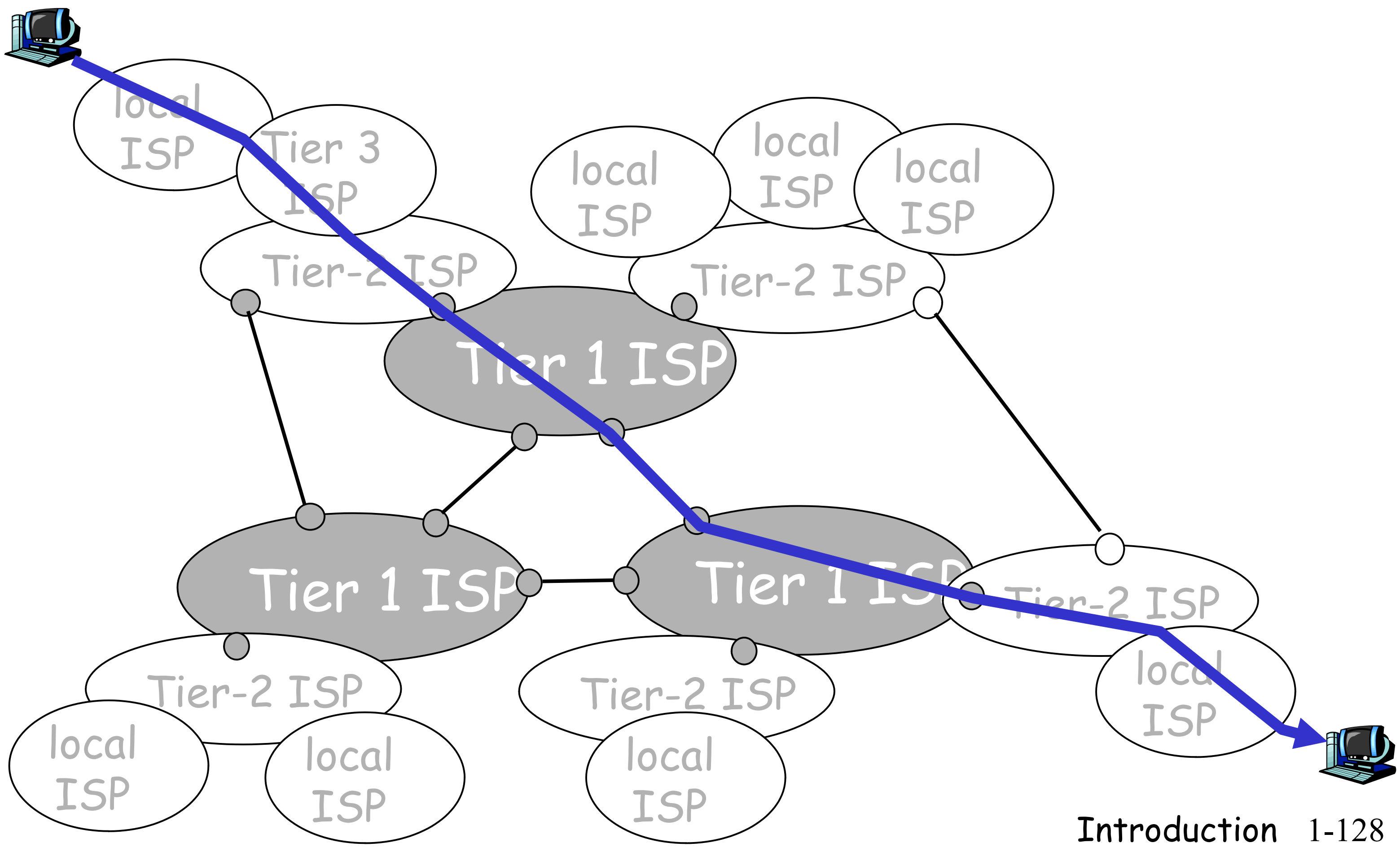
Internet structure: network of networks

- "Tier-3" ISPs and local ISPs
  - last hop ("access") network (closest to end systems)
  - solely purchases transit from other networks to reach Internet



Internet structure: network of networks

- a packet passes through many networks!



Introduction: Summary

Covered a "ton" of material!

- Internet overview
- what's a protocol?
- network edge, core, access network
  - packet-switching versus circuit-switching
- Internet/ISP structure
- performance: loss, delay
- layering and service models
- history

You now have:

- context, overview, "feel" of networking
- more depth, detail *to follow!*