Computer Networks

Chapter 1 – Computer Networks and the Internet

Edition8-1

Topics

- Internet: largest engineered system ever created by mankind
  - hundreds of millions of connected computers, communication links, and switches; with billions of users who connect via laptops, tablets, and smartphones; and with an array of new Internet-connected "**things**" including **game consoles**, **surveillance systems**, **watches**, **eye glasses**, **thermostats**, and **cars, …**
- Are there guiding principles and structure that can provide a foundation for understanding such an amazingly large and complex system?
- It is aim in this course to provide an introduction to computer networking, giving you the **principles and practical insights** you'll need to understand not only today's networks, but tomorrow's as well

Topics

We'll structure our overview of computer networks in this chapter:

- Basic terminology and concepts
- Basic hardware and software components that make up a network
- Network's edge: end systems, hosts, and network applications
- Network's core: Links and Switches that transport data
- Access networks and physical media that connect end systems to network core
- Internet: Network of networks architecture
- **Delay, loss, and throughput** of data in a computer network
- Key architectural principles: Protocol layering and Service models
- **Security** of computer network
- A brief history of computer networking

Contents

1.1 What Is the Internet?
1.2 The Network Edge
1.3 The Network Core
1.3’ The Name and Addresses
1.4 Delay, Loss, and Throughput in Packet-Switched Networks
1.5 Protocol Layers and Their Service Models
1.6 Networks Under Attack
1.7 History of Computer Networking and the Internet
1.8 Summary
Appendix

1.1 What Is the Internet?

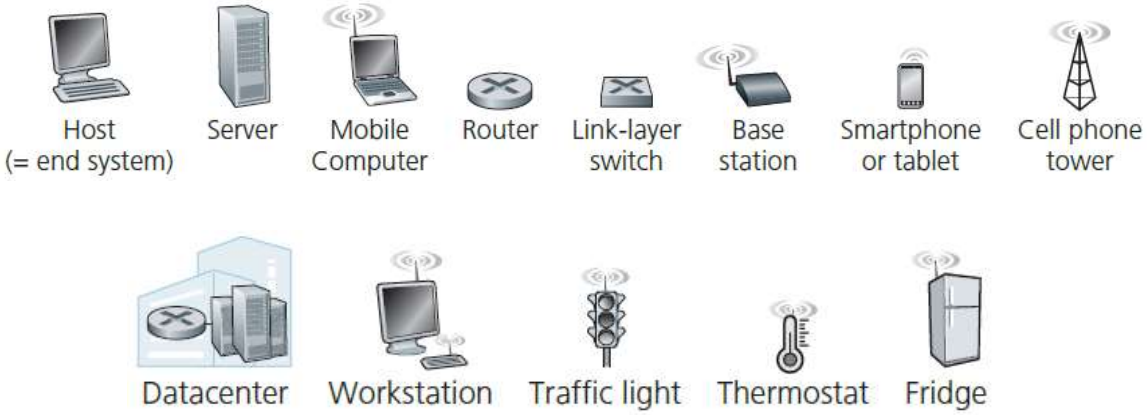
- Public Internet is our principal vehicle for discussing computer networks and their protocols

Internet

- A networking infrastructure that provides services to distributed applications
- Basic hardware and software components that make up the Internet

1.1.1 A Nuts-and-Bolts Description

- Basic practical details of Internet



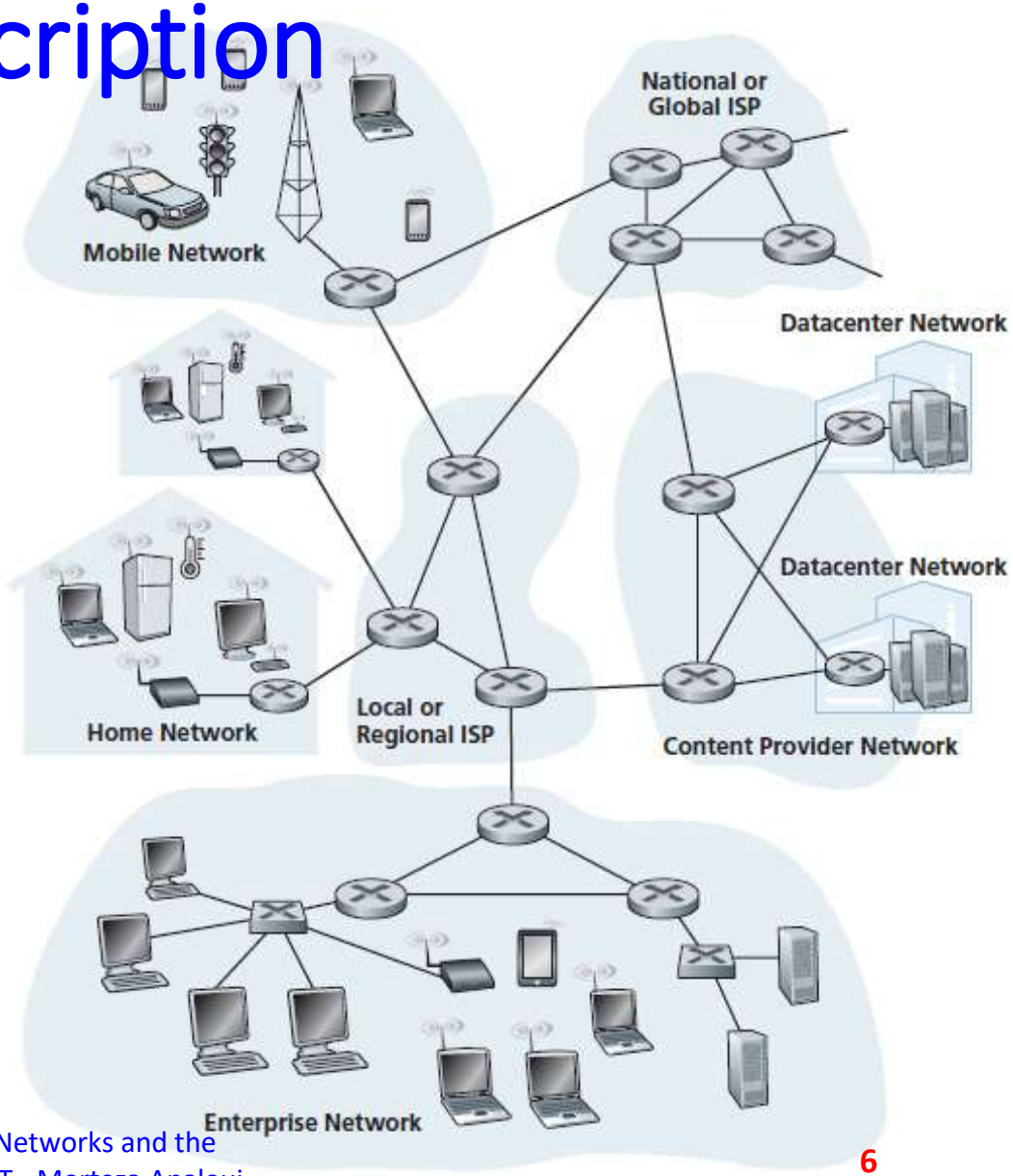
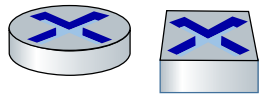


Figure 1.1 Some pieces of the Internet

Figure 1.1 Some pieces of the Internet

- Billions of connected computing devices:
  - hosts = end systems
  - APPs running at Internet's hosts on "edge"



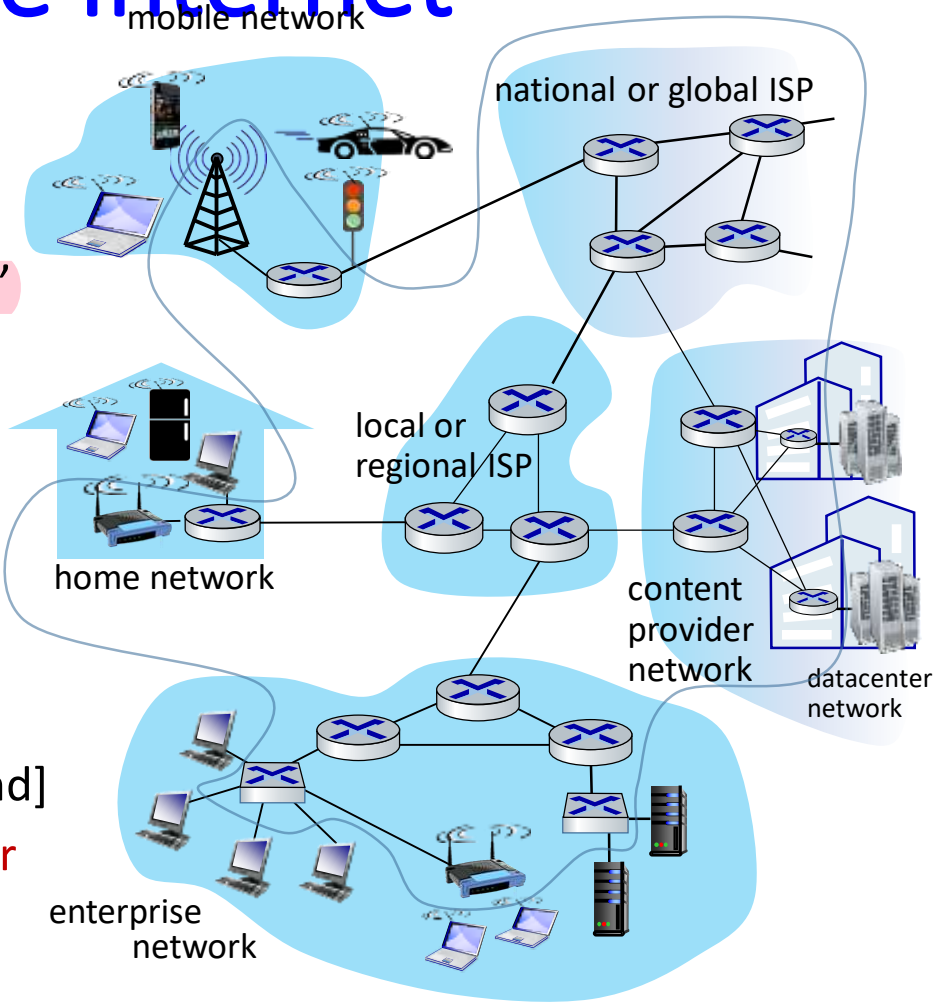
Packet switches: forward packets (chunks of data)

- routers, switches

Communication links

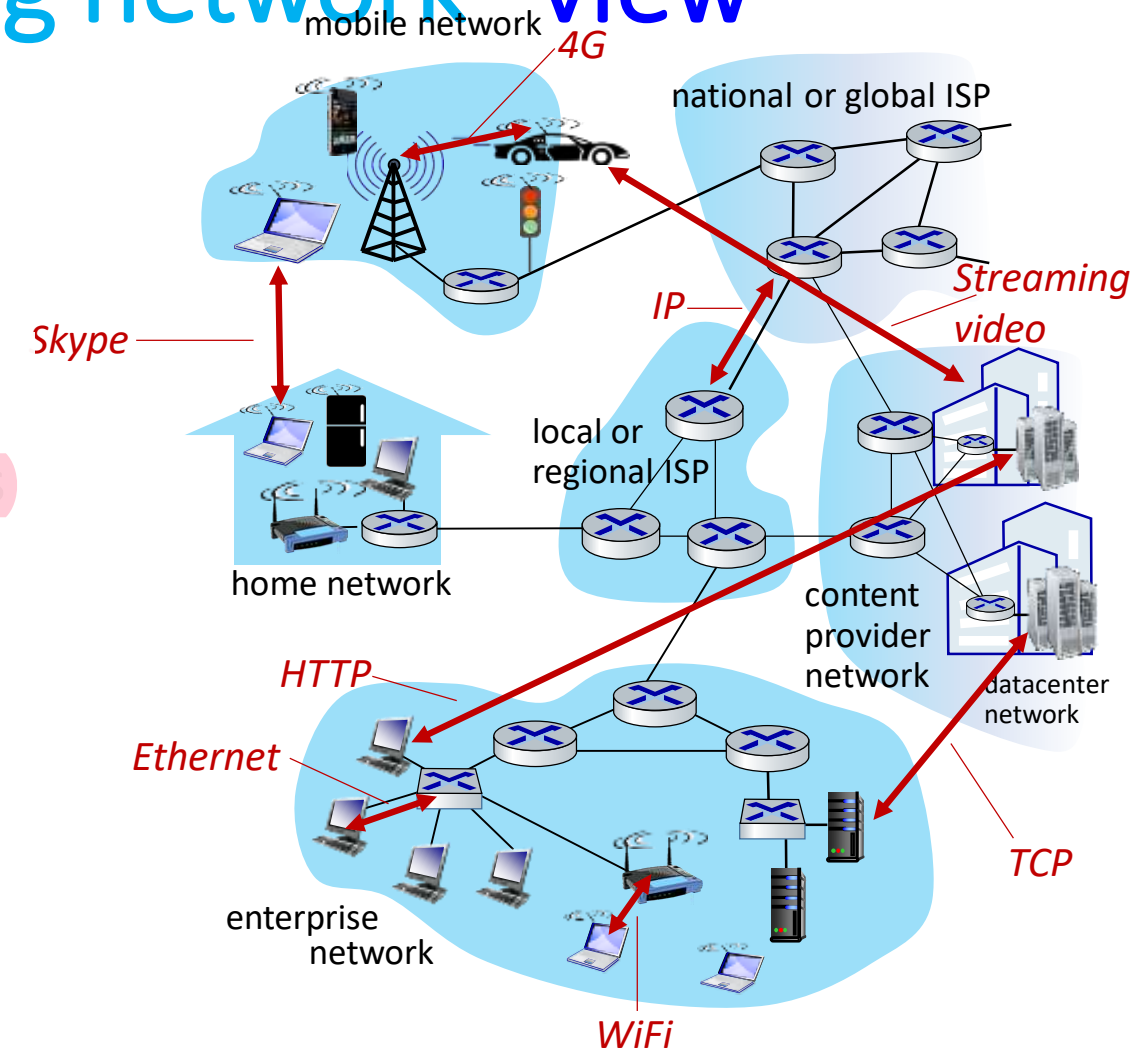
- fiber, copper, radio, satellite
- transmission rate = bandwidth [bit per second]
- annual global IP traffic 2019: 2 zettabytes per year (zetta:10^{21})

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



Internet: "communicating network" view

- Internet: "network of networks"
  - Interconnected ISPs
- Protocols are everywhere
  - control sending, receiving of messages
  - e.g., HTTP (Web), streaming video, Skype, TCP, IP, WiFi, 4G, Ethernet
- Internet standards
  - IETF (Internet Engineering Task Force) produces standard protocols
  - RFC: Request for Comments



Internet standards

- Hosts, packet switches run **protocols** that control sending and receiving of information within Internet
- **Transmission Control Protocol (TCP)** and **Internet Protocol (IP)** are two of most important protocols in Internet
- IP protocol specifies format of packets that are sent and received among routers and hosts
- Internet's protocols are collectively known as **TCP/IP**
- **Internet standards** are developed by Internet Engineering Task Force (IETF)
- IETF standards documents are called **requests for comments (RFCs)**
- RFCs are technical and detailed
- They define protocols such as TCP, IP, HTTP, SMTP, …
- There are currently nearly 9000 RFCs
- Other bodies also specify standards for network components, most notably for network links. **IEEE** 802 LAN Standards Committee, specifies Ethernet and wireless WiFi standards

Link – Switch -Segment - Packet

- Hosts are connected together by a network of **communication links** and **packet switches**
  - Many types of communication links, which are made up of different types of physical media, including coaxial cable, copper wire, optical fiber, and radio spectrum
- Different links can carry data at different rates (bits/second)
- When a host has message/file/object to send to another host, sending end host **segments message** and adds header bytes to each segment
- Resulting packages of information, known as **packets** , sent through network to destination hosts, where they are reassembled into original message

Forwarding – Path (Route)

- A packet switch takes a packet arriving on one of its incoming communication links and forwards (sends) that packet on one of its outgoing communication links, toward their ultimate destinations
- Packet switches come in many shapes and flavors, two most prominent types in today's Internet are **routers** and **LAN switches**
- LAN switches are typically used in access networks (LANs), while routers are typically used in network core
- Sequence of communication links and packet switches traversed by a packet from sending host to receiving host is known as a **route** or **path** through network

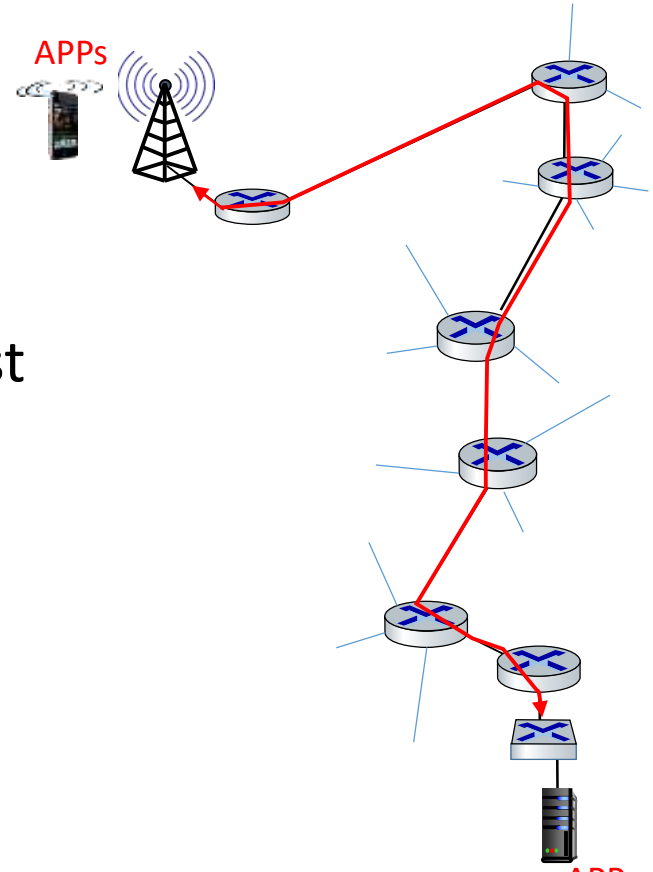


Figure Path between two LANs

Transportation network - Computer network

- Packet-switched networks (which transport packets) are in many ways similar to transportation networks of highways, roads, and intersections
- Cargo Transport: a factory needs to move a large amount of cargo to client building located thousands of kilometers away
  - At factory, cargo is segmented and loaded into a fleet of trucks
  - Each of trucks then independently travels through network of highways, roads, and intersections to destination building
  - At destination, cargo is unloaded and grouped with rest of cargo arriving from same shipment
- So: packets are analogous to trucks, communication links are analogous to highways and roads, packet switches are analogous to intersections, and hosts are analogous to buildings
- Truck takes a path through transportation network, a packet takes a path through a computer network

Internet Service Providers (ISPs)

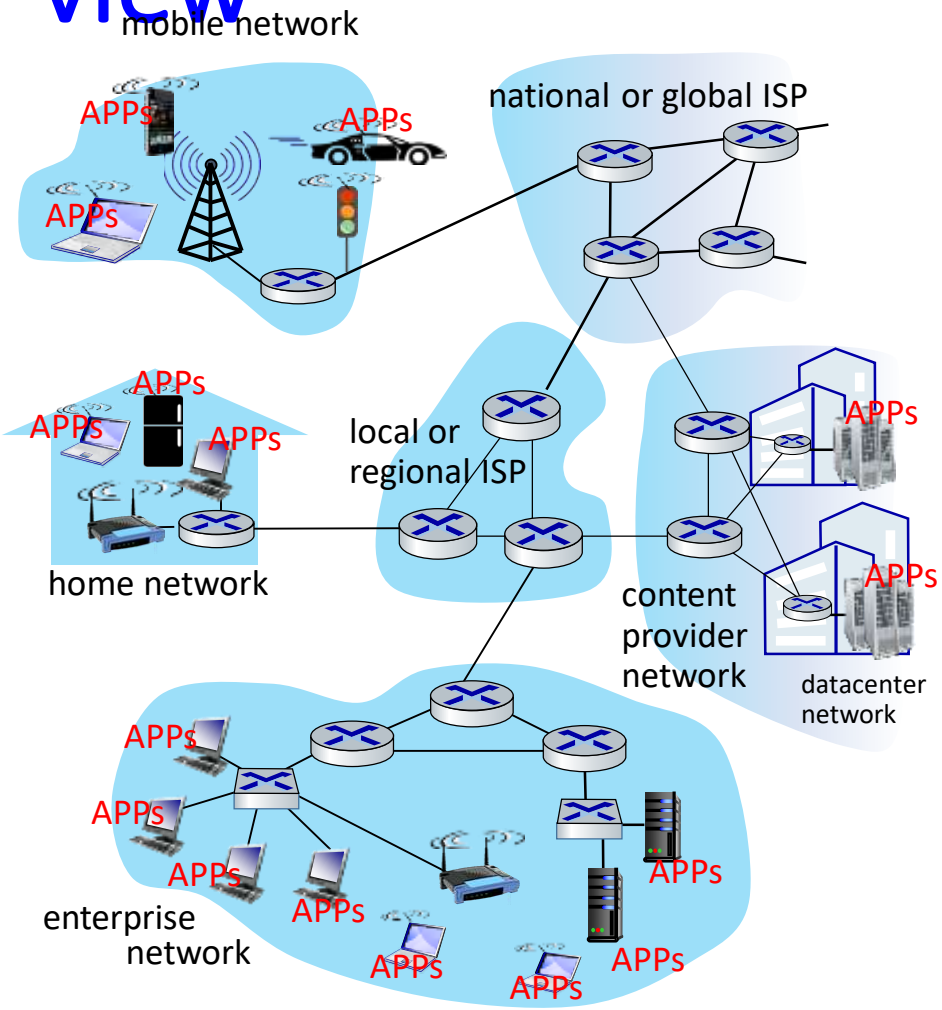
- Hosts access Internet through local (regional) **Internet Service Providers (ISPs)**, including
  - residential ISPs
  - corporate ISPs
  - university ISPs
  - ISPs that provide WiFi access in airports, hotels, coffee shops, and other public places
  - cellular data ISPs (providing mobile access to our smartphones and other devices)
- Each local ISP is in itself a network of packet switches and communication links
- Local ISPs provide a variety of types of network access to end systems, including
  - residential broadband access such DSL
  - local area network access
  - mobile wireless access
- Local ISPs also provide Internet access to servers of content providers, service providers, …
- Local ISPs are interconnected through national and international upper-tier ISPs and these upper-tier ISPs are connected directly to each other
- An upper-tier ISP consists of high-speed routers interconnected with high-speed fiber-optic links
- Each ISP network, whether upper-tier or lower-tier (regional), is managed independently

1.1.2 A Services Description

- Internet is an infrastructure that provides services to applications
- APPs: e-mail, Web, …(traditional), messaging, online courses, real-time road-traffic, music streaming, movie and television streaming, online social media, video conferencing, multi-person games, location-based recommendation systems, cloud services, …
- Network APPs are **distributed applications**, (multiple hosts exchange data with each other)
- APPs run on hosts (not on packet switches in network core)
- Packet switches are not concerned with communicating APPs

Internet: "platform for APPs" view

- Unexplored frontier of applications and services that programmers and entrepreneurs are able to explore
- Internet is an infrastructure that provides services to distributed applications (they involve multiple end systems):
  - Web, streaming video, multimedia teleconferencing, email, games, e-commerce, social media, inter-connected appliances, …
- Provides programming interface to distributed applications:
  - "interfaces=sockets" allowing sending/receiving apps to "connect" to, use Internet transport service
  - provides service options

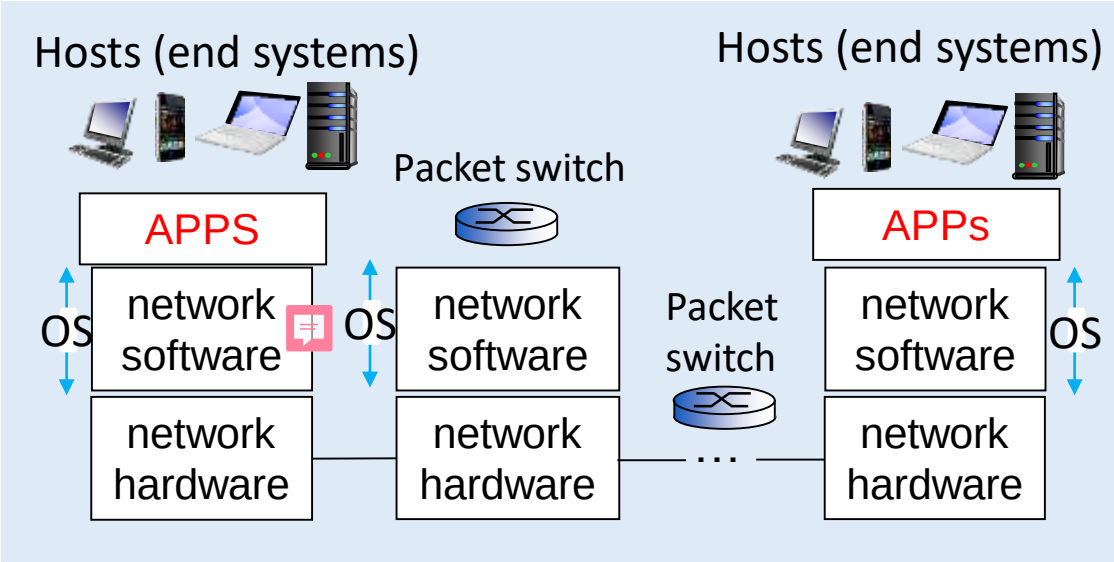


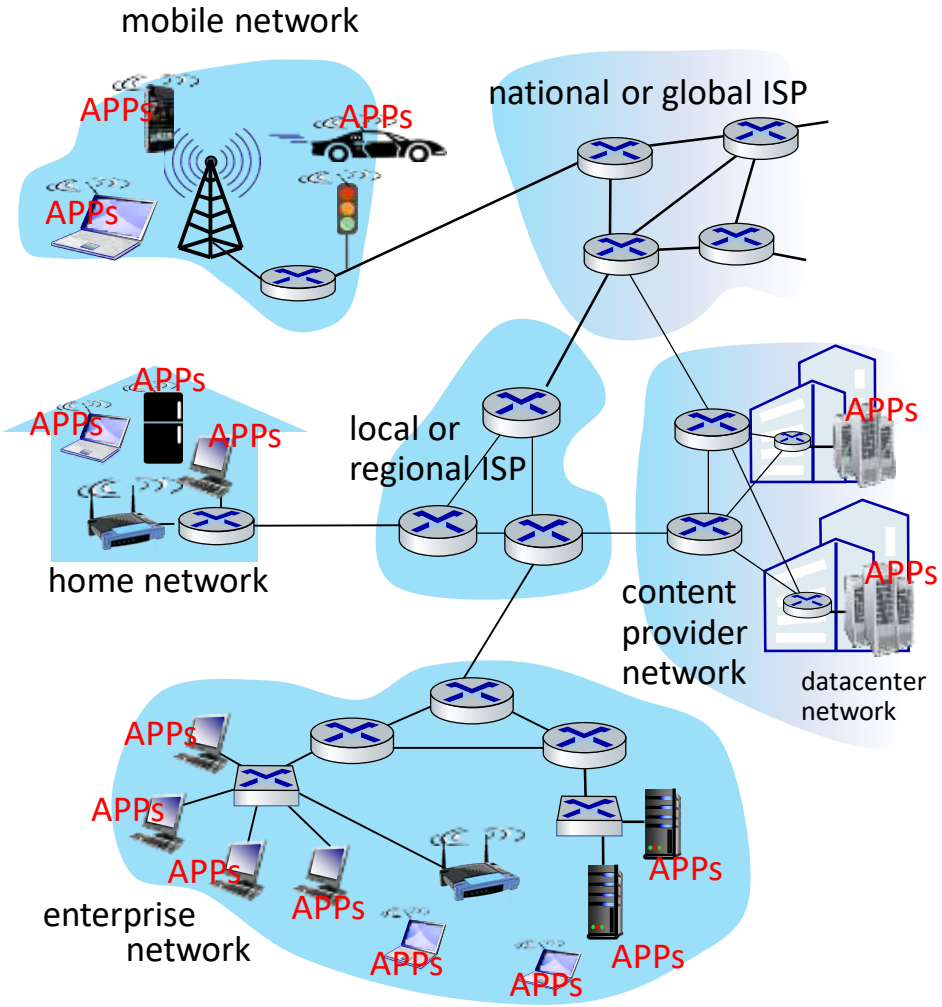
Internet: "platform for APPs" view

- **Q:** How does one program running on a host instruct Interne to deliver its messages to another program running on another host?

- **A:** Operating System (OS) in hosts provide a socket interface that specifies how a program running on one host asks Internet infrastructure to deliver messages to a specific destination program running on another host
- Internet socket interface is a set of APIs that APPs use to send and receive messages/files/objects through Internet
- Internet delivery (transport) comes with multiple services to applications. Services such as: Reliable data transfer, Security, …

Internet: "system" View





1.1.3 What Is a Protocol?

A Human Analogy

- First offer a greeting
- Then ask for time of day
- Last say thanks

- Transmitted and received messages, and actions taken when these messages are sent or received or other events occur (such as no reply within some given amount of time), play a central role in a human protocol
- It takes two (or more) communicating entities using same protocol in order to accomplish a task

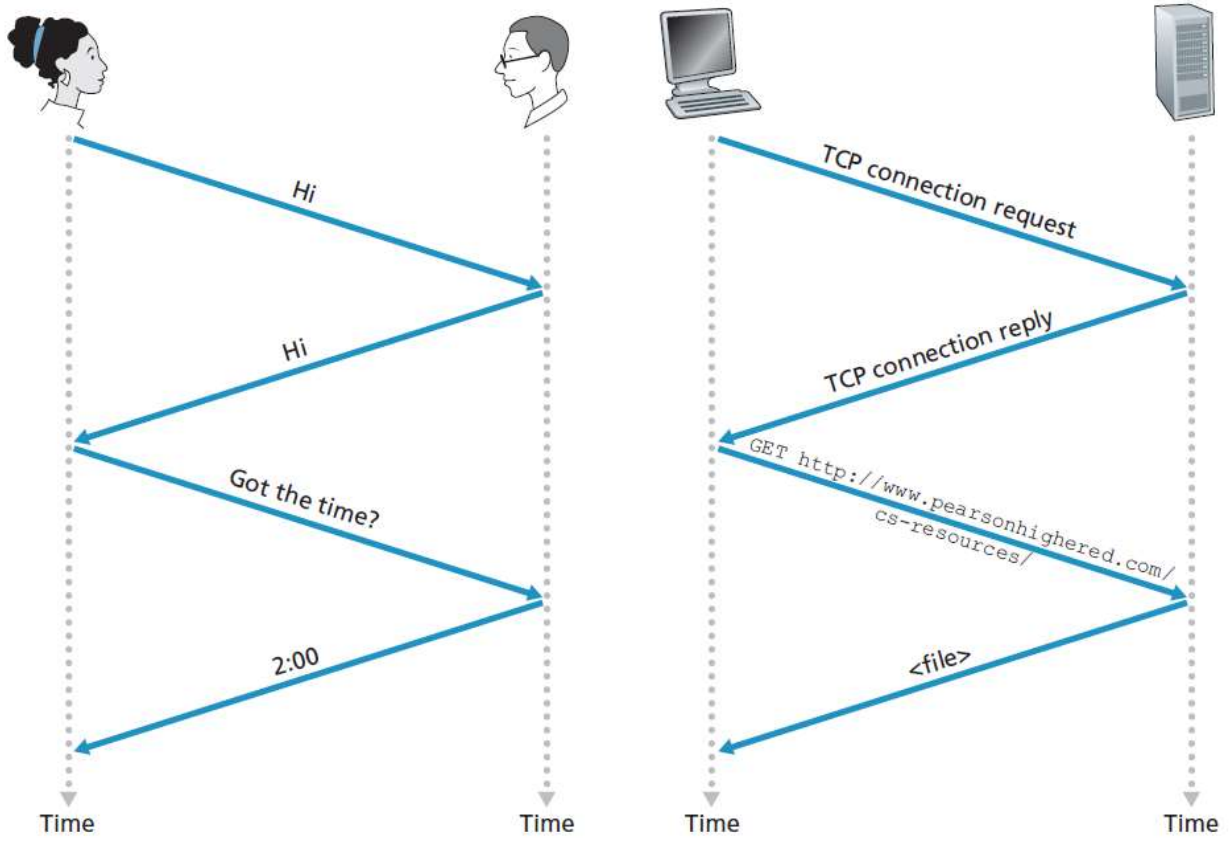


Figure 1.2 A human protocol and a computer network protocol

Network Protocols

- A network protocol is similar to a human protocol, except that entities exchanging messages and taking actions are hardware or software components of some device (for example, computer, smartphone, tablet, router, or other network-capable device)
- All activity in Internet that involves two or more communicating remote entities is governed by a protocol
- A **protocol** defines format and order of messages exchanged between two or more communicating entities, as well as actions taken on transmission and/or receipt of a message or other event

Contents

1.1 What Is the Internet?
1.2 The Network Edge
1.3 The Network Core
1.3’ The Name and Addresses
1.4 Delay, Loss, and Throughput in Packet-Switched Networks
1.5 Protocol Layers and Their Service Models
1.6 Networks Under Attack
1.7 History of Computer Networking and the Internet
1.8 Summary
Appendix

1.2 The Network Edge

End systems include:

- desktop computers (e.g., desktop PCs, Macs, and Linux boxes)
- Server machines (e.g., Web and e-mail server computers)
- mobile devices (e.g., laptops, smartphones, and tablets)
- printers, …,
- “things” (e.g., refrigerator, TV set, sensors, …)

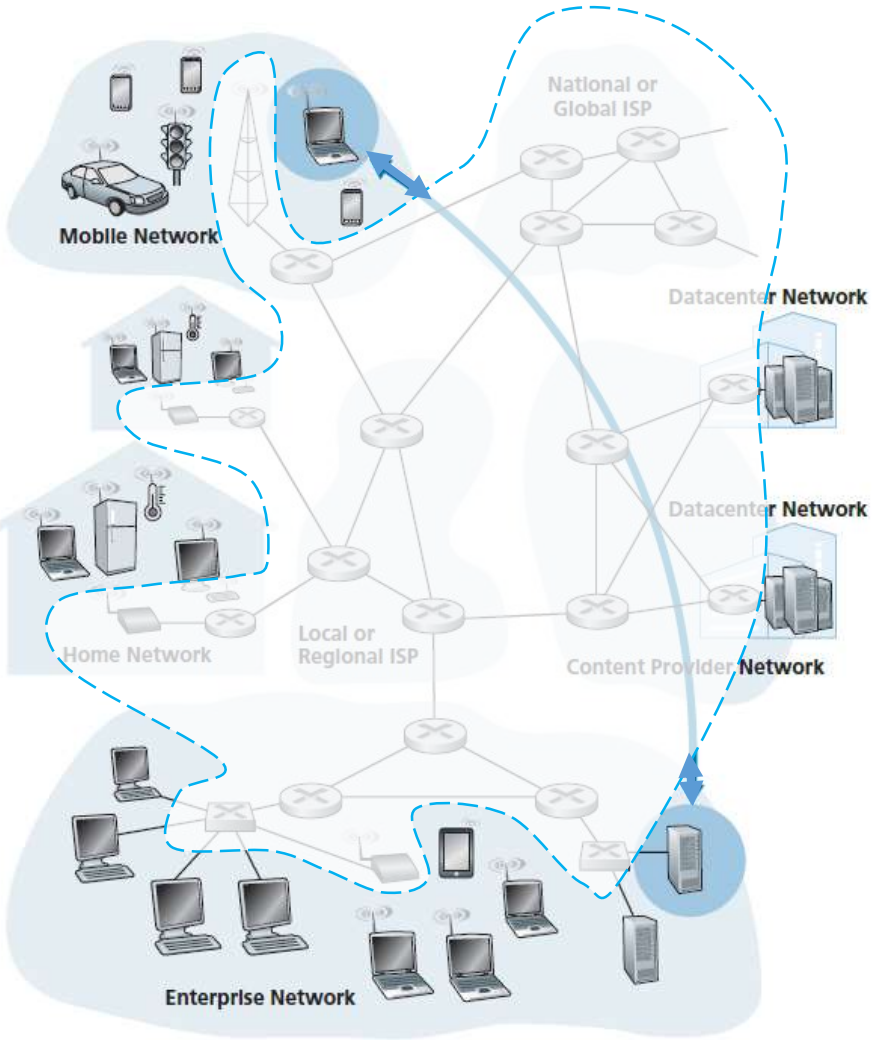


Figure 1.3 End-system interaction

End system hosting APPs

- They are end systems because they sit at the edge of network
- End systems are also referred to as hosts because they host (that is, run) application programs such as a Web browser program, a Web server program, an e-mail client program, or an e-mail server program
- Powerful computers are used to run server APPs (then they are called server machines or simple servers)
- Most of servers reside in **data centers**

1.2.1 Access Networks

- **Access network**: network that physically connects an end system to first router (edge router)
- Figure 1.4 shows several types of access network:
  - Home (WiFi)
  - Enterprise
  - Wide-area mobile wireless,
  - Internal network of data centers

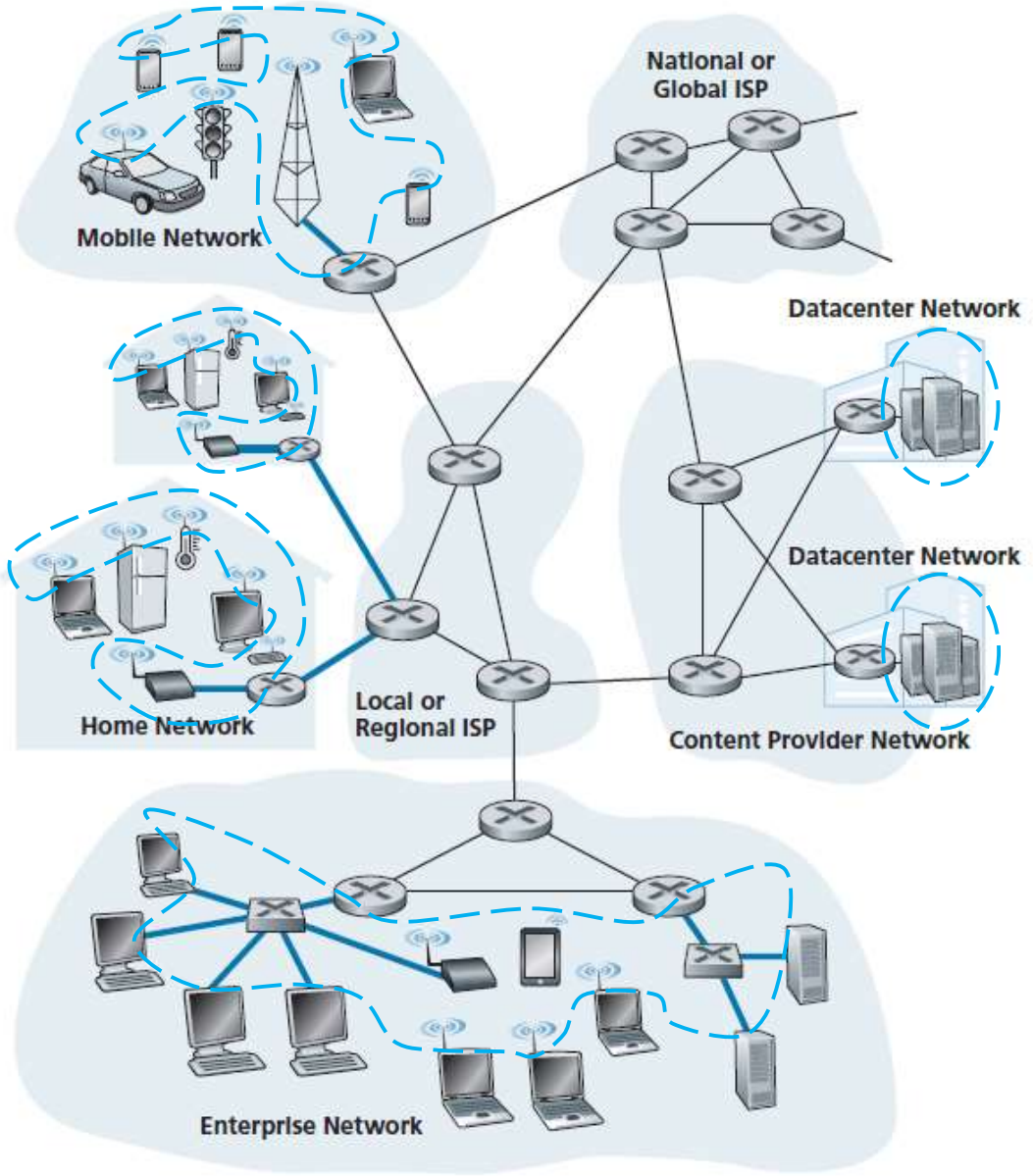
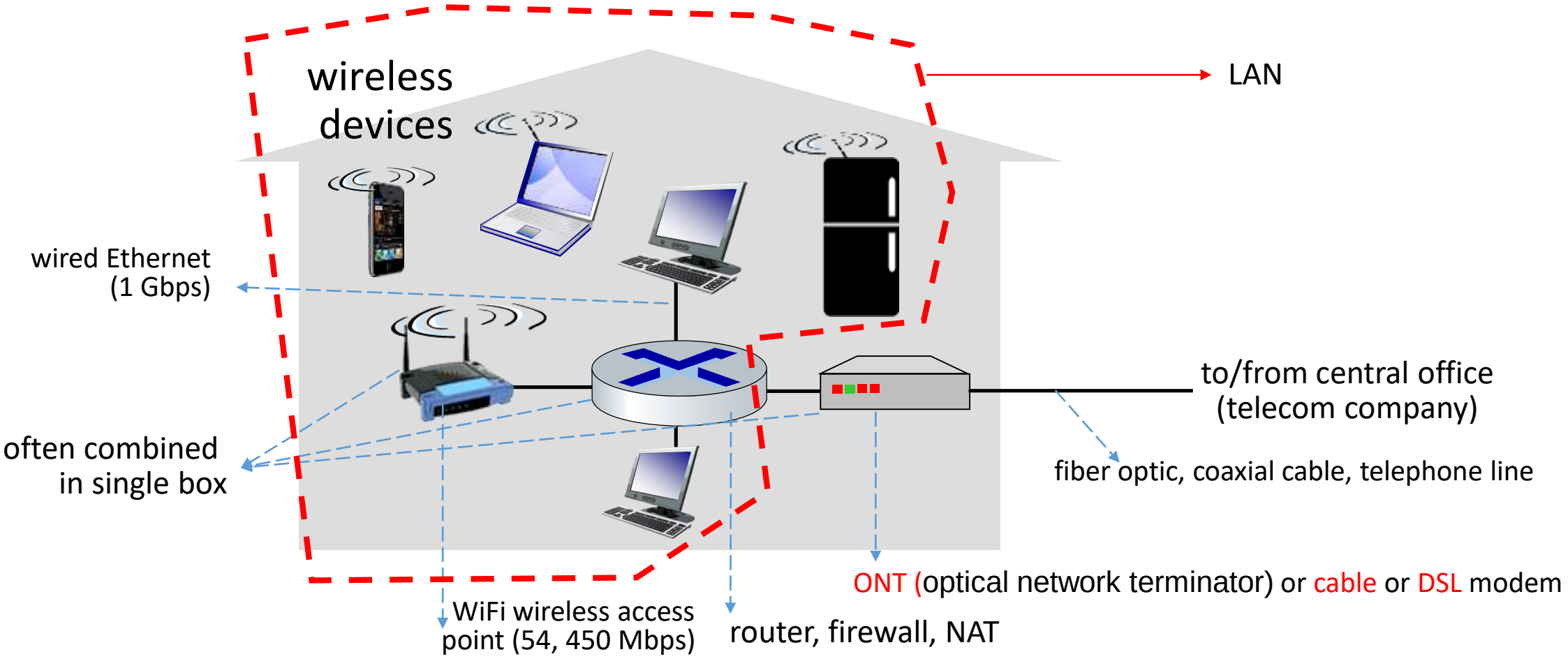


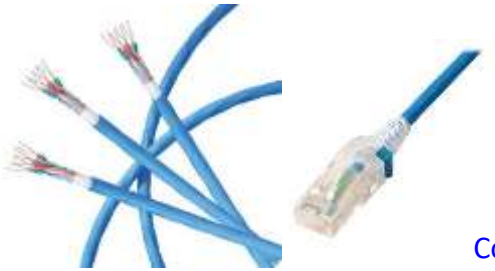
Figure 1.4 Access networks

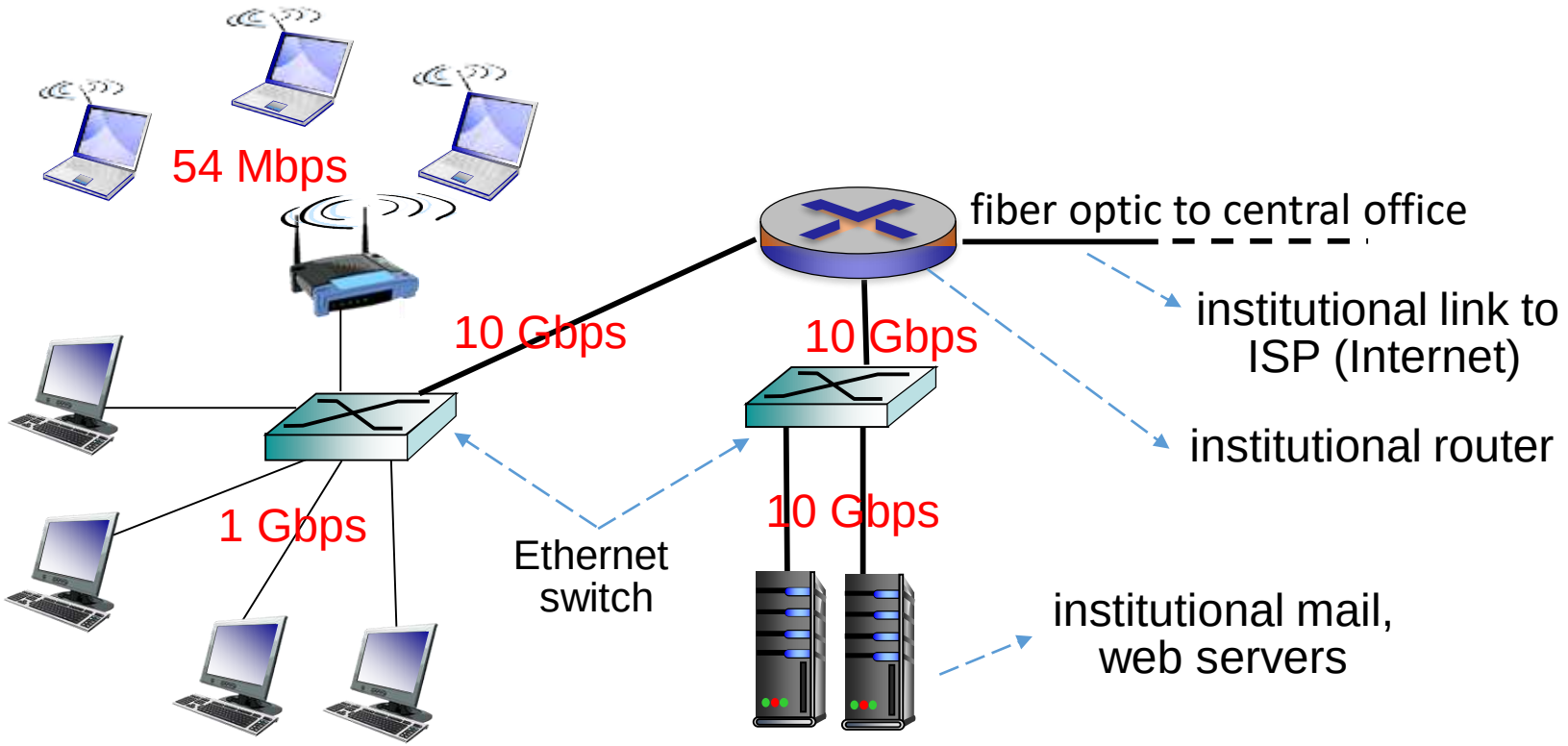
Access networks technologies - WiFi



Access networks technologies - Enterprise

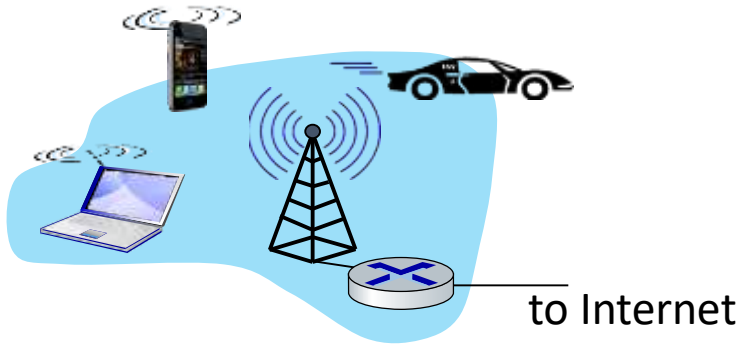
- Companies, Universities, etc.
- Link-layer switches have been used to set up an enterprise network
- Twisted pair cable
- Categories 5, 6 (10Gbps)





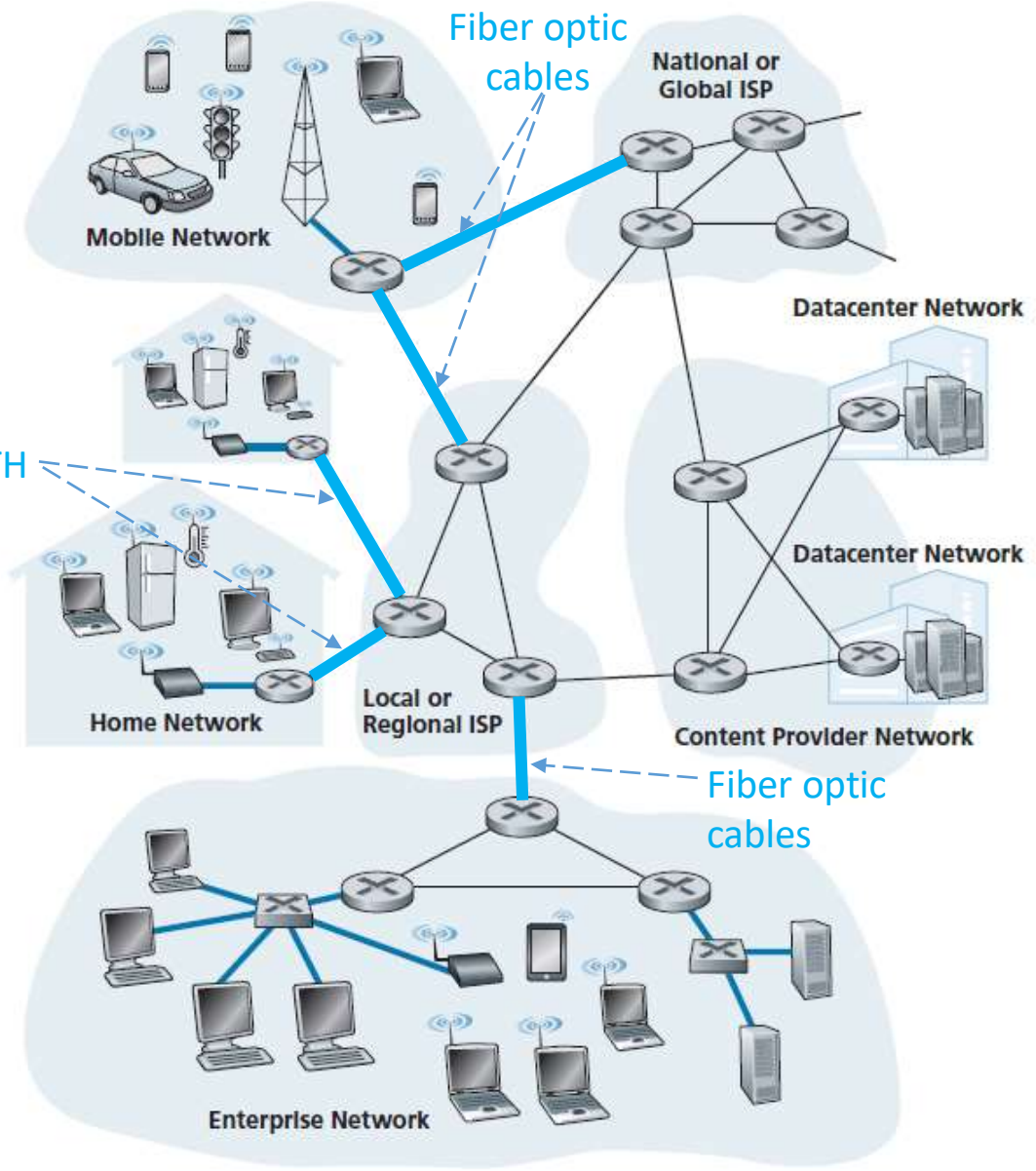
Access networks technologies–Mobile network

- Wide-area cellular access networks
  - provided by mobile, cellular network operator (100's km)
  - 10's Mbps
  - 4.5G cellular networks (5G coming)



Access link technologies

- **Access link** technology: Connecting access networks to Internet



Fiber optic cable:

- glass fiber carrying light pulses, each pulse a bit
- high-speed operation:
  - high-speed point-to-point transmission (10's-100's Gbps)
- low error rate:
  - repeaters spaced far apart
  - immune to electromagnetic noise

Home Access link: DSL

- digital subscriber line (DSL)
  - use *existing* telephone line to central office DSLAM
  - data over DSL phone line goes to Internet
  - voice over DSL phone line goes to telephone net
  - 24-52 Mbps dedicated downstream transmission rate
  - 3.5-16 Mbps dedicated upstream transmission rate

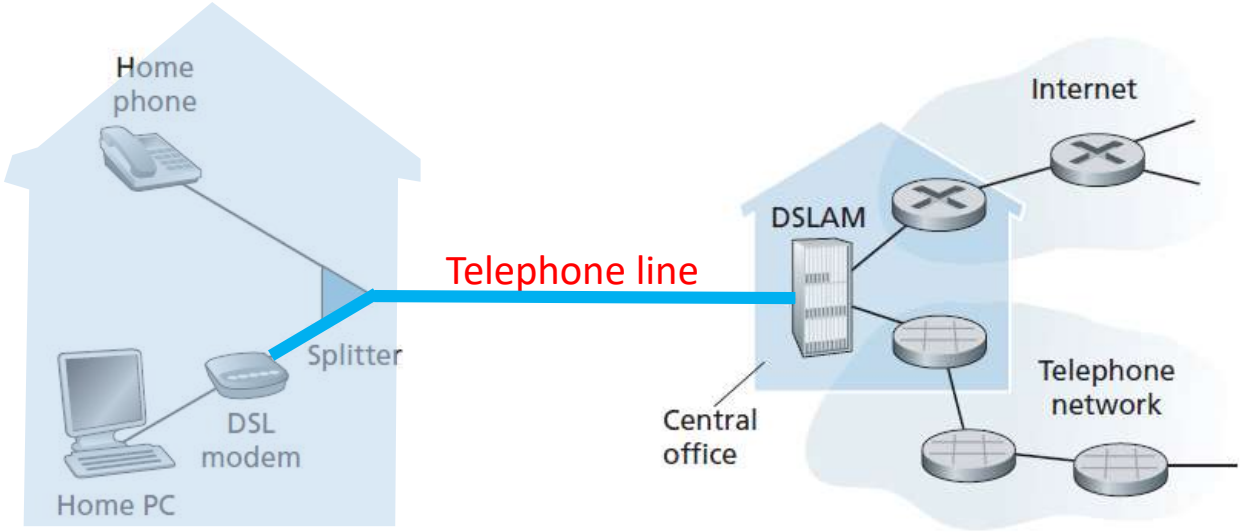


Figure 1.5 DSL Internet Access Link

Home Access link: Cable

Coaxial cable: 100s Mbps

- **Cable Internet access:** uses cable television company's existing cable
- Fiber optics connect cable head end to neighborhood-level junctions, from which traditional coaxial cable is then used to reach individual houses and apartments
- Each neighborhood junction typically supports 500 to 5,000 homes
- Fiber and Coaxial cable are employed: hybrid fiber coax (HFC)

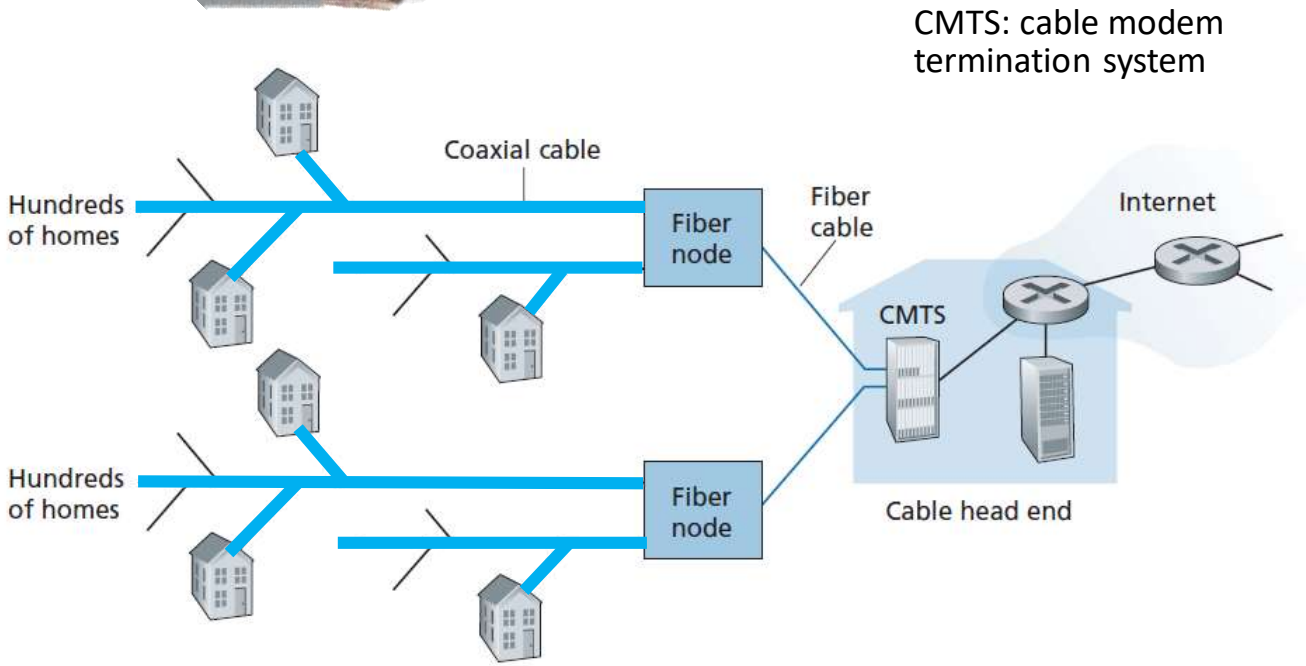
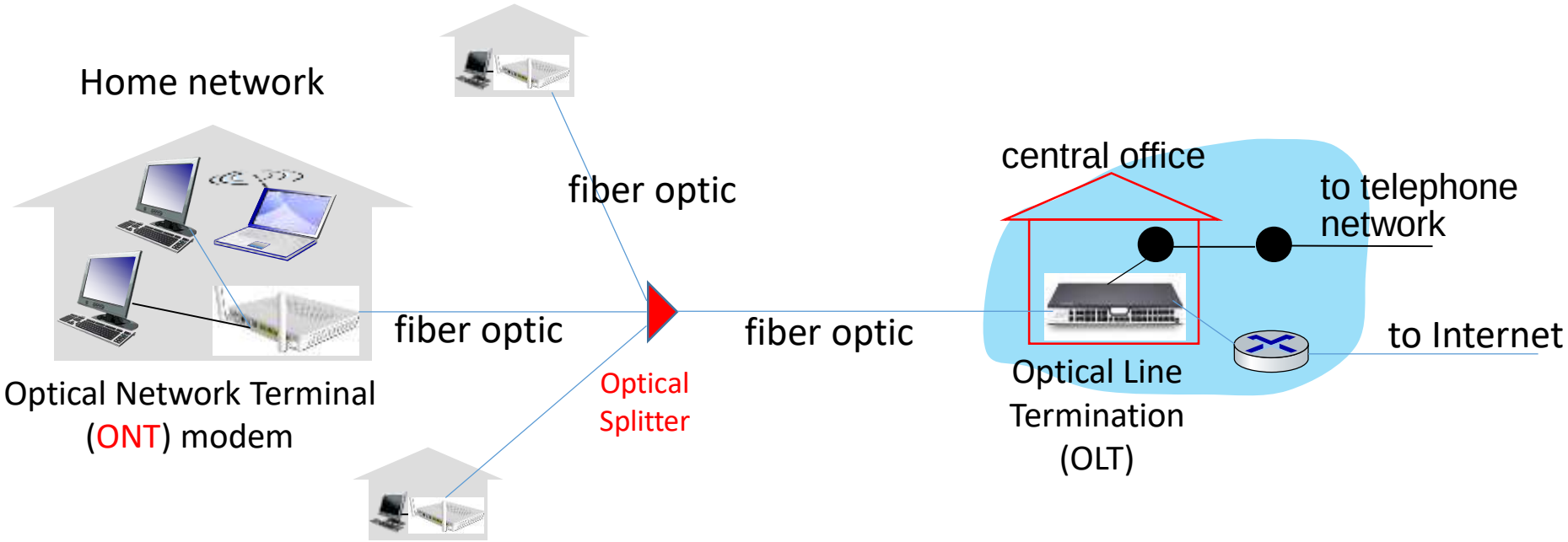


Figure 1.6 A hybrid fiber-coaxial access network

- HFC: 40 Mbps–1.2 Gbps downstream transmission rate, 30-100 Mbps upstream transmission rate

Home Access link: FTTH

- FTTH can potentially provide Internet access rates in gigabits per second range
- Most FTTH ISPs provide different rate offerings
- Telecom Company of Iran provides up to 50 Mbps for each home



1.2.2 Physical Media

- Radio
  - terrestrial microwave
  - up to 45 Mbps channels
- Wireless LAN (WiFi)
  - Up to 100's Mbps
- wide-area (e.g., cellular)
  - 4G cellular: ~ 10's Mbps
- Satellite
  - up to 45 Mbps per channel
  - 270 msec end-end delay
  - geosynchronous versus low-earth-orbit

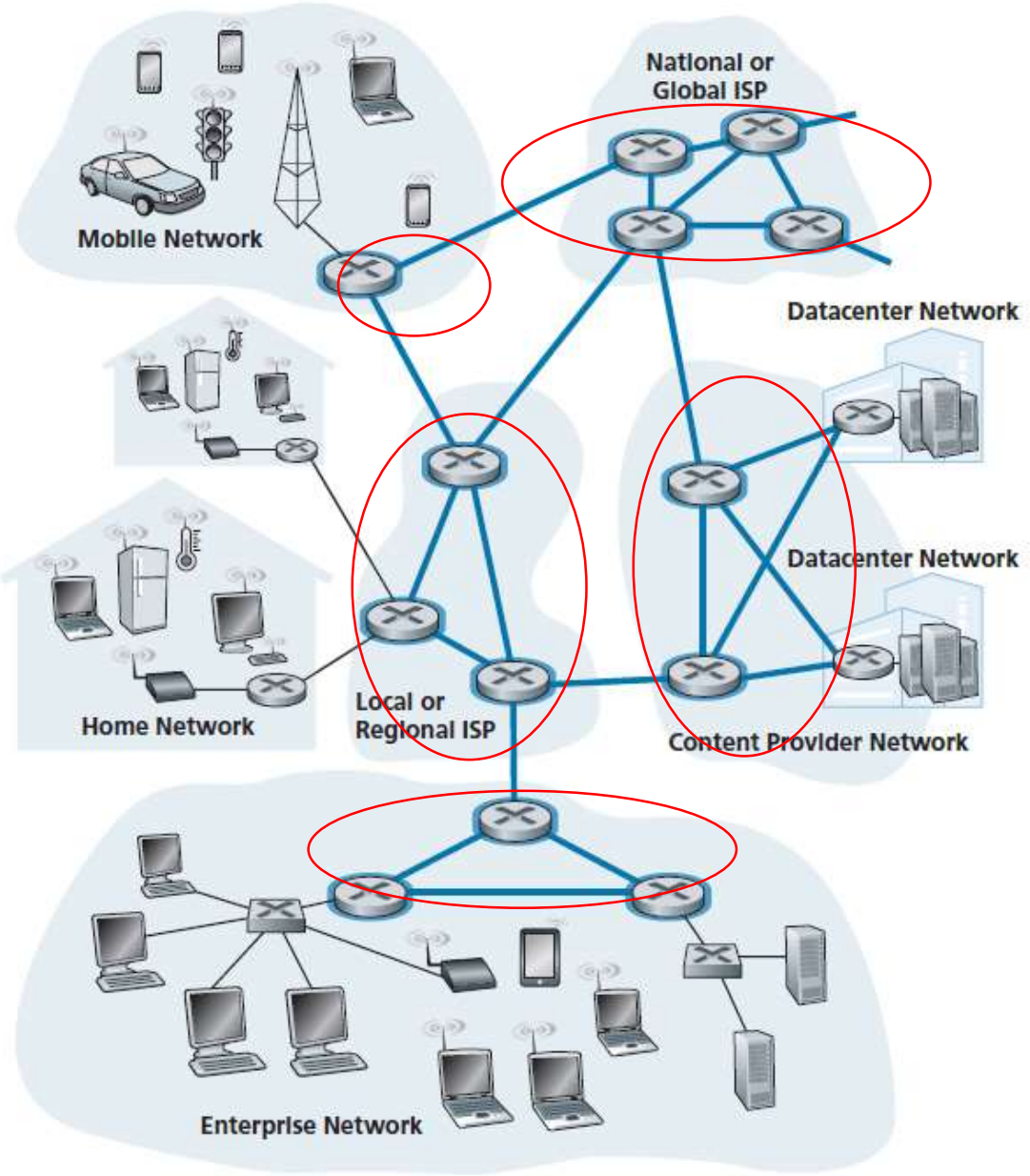
- Fiber optic
- Twisted pair
- Coaxial Cable

Contents

- 1.1 What Is the Internet?
- 1.2 The Network Edge
- 1.3 The Network Core
- 1.3’ The Name and Addresses
- 1.4 Delay, Loss, and Throughput in Packet-Switched Networks
- 1.5 Protocol Layers and Their Service Models
- 1.6 Networks Under Attack
- 1.7 History of Computer Networking and the Internet
- 1.8 Summary
- Appendix

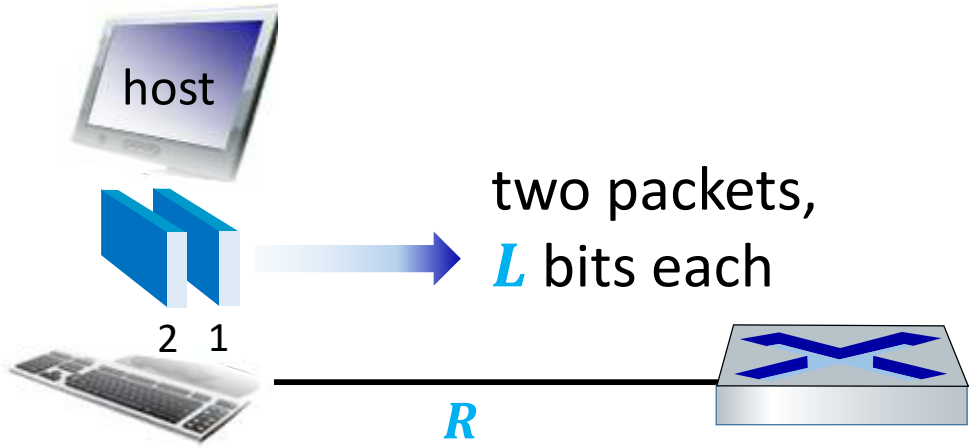
1.3 The Network Core

- **Figure 1.10** The network core
- Mesh of packet switches that interconnects access networks



1.3.1 Packet Switching

- To send a message from a source to a destination host, source breaks long messages into smaller chunks of data known as **packets**
- Packet travels through communication links and **packet switches** (**routers** and **link-layer switches**)
  - Packets forwarded from one packet switch to next packet switch, across links on path from source to destination
- Packets are transmitted over each communication link at a rate equal to full transmission rate of link
- Packet transmission time: $\frac{L(bits)}{R(bits\ per\ sec)}$

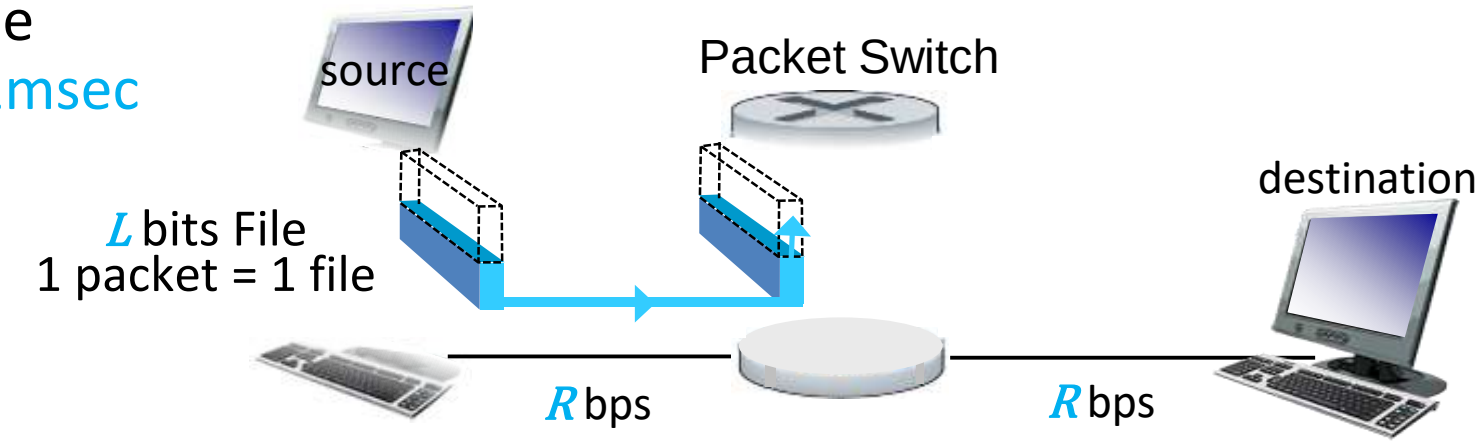


Store-and-Forward Transmission

- Store-and-forward transmission: packet switch receives entire packet before it can begin to transmit first bit of packet onto outbound link

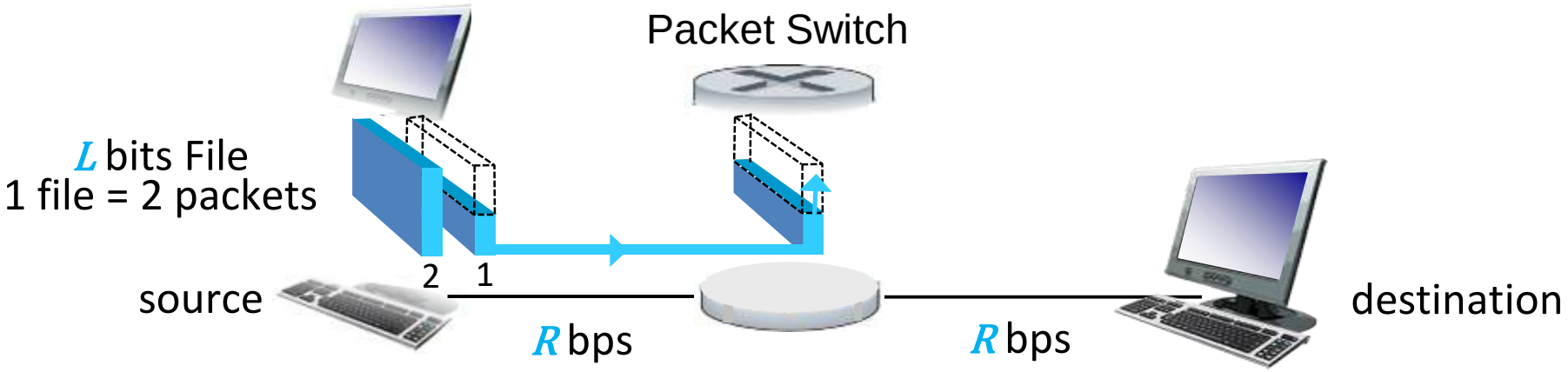
Example: one file = one packet, L = 10 Kbits, R = 100 Mbps

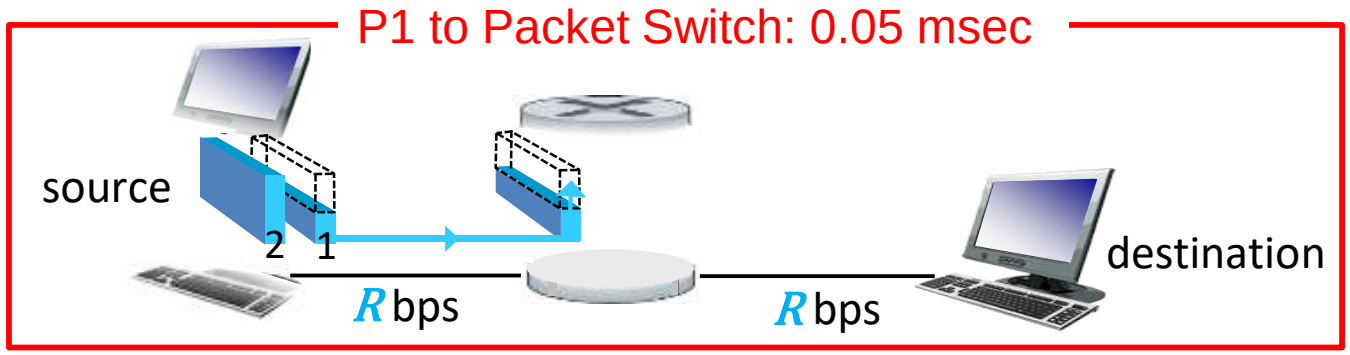
- One-hop transmission delay = L/R = 0.1 msec
- Store and forward end to end delay: $L/R + L/R$ = 0.2 msec, assuming zero propagation delay, no queue
- Circuit switch e-e-delay=0.1msec

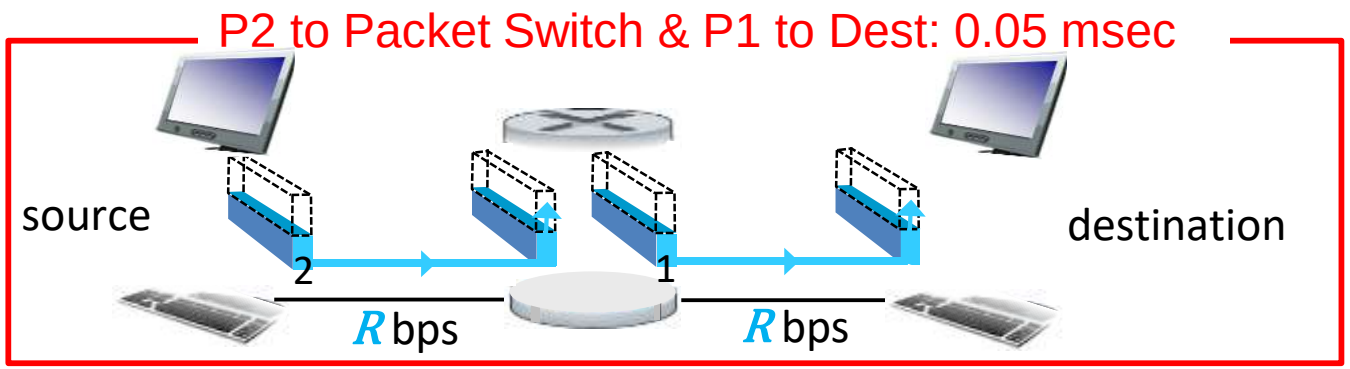


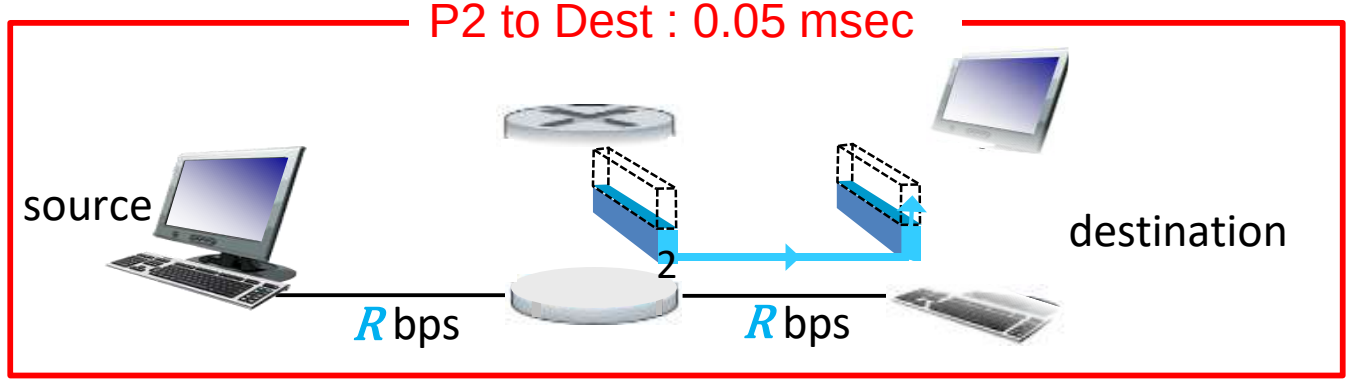
Example: one file = two packets, one PS

- Two packets, each of L = 5 Kbits, L/R=0.05 msec
- End-end delay ?







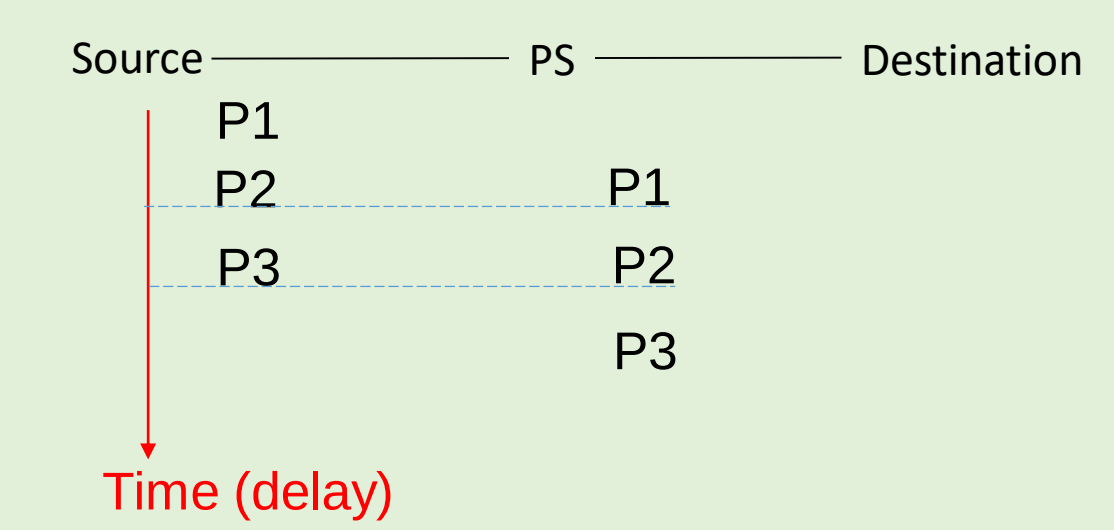


Store and Forward
e-e-d=3*0.05=0.15msec

Example: one file = n packets, one PS

- n = 3 packets & 1 Packet Switch

e-e-d = 4*(L/3)/R = 0.133 msec



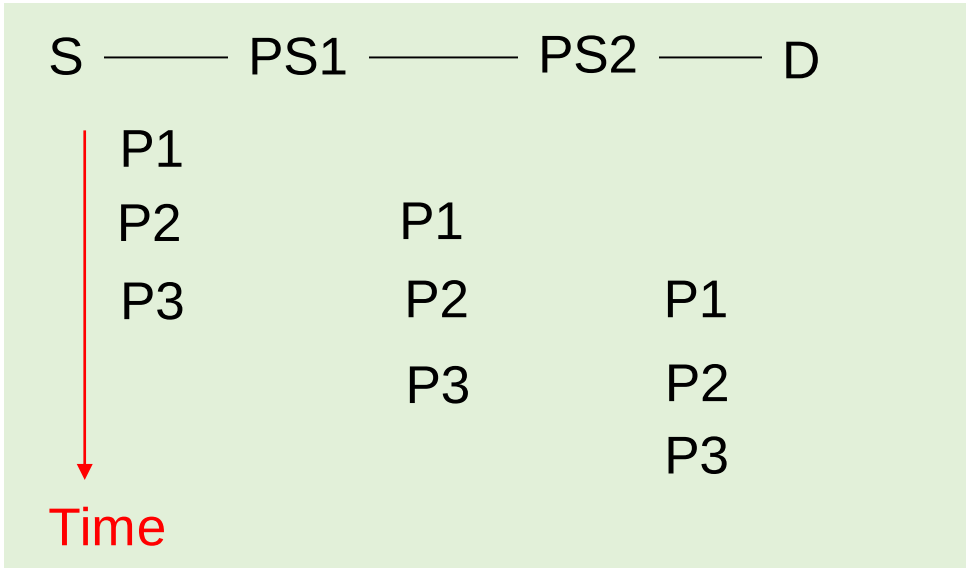
- n packets & 1 Packet Switch

$$e-e-d = \frac{n+1}{n}\frac{L}{R} = \left(\frac{1}{n}+1\right)\frac{L}{R}\ [sec] \qquad e-e-d \approx \frac{L}{R} \quad for\ n \gg 1$$

Example: one file = n packets, two PSs

- n = 3 packets & 2 Packet Switches

  e-e-d = 5*(L/3)/R = 0.167 msec
  (no packetizing: 0.3 msec)



- n packets & m Packet Switches:

$$e-e-d = \frac{n+m}{n}\frac{L}{R} = \left(\frac{m}{n}+1\right)\frac{L}{R}\ [sec]$$

$$e-e-d \approx \frac{L}{R} \quad \text{for } n \gg m$$

- circuit-switching: e-e-d = L /R (packetizing doesn't help)
- Average network throughput $= \frac{L}{e-e-d} = \frac{1}{\left(\frac{m}{n}+1\right)\frac{1}{R}}$ [bps] $\approx R$ for $n \gg m$

Pros and Prone of Packetizing

Pros:

- Reducing end to end delay
- If there is a single bit error, whole message has not to be retransmitted (a single packet is retransmitted)
- Huge messages (such as 4k videos) are sent into network. Routers have to accommodate these huge messages
- Smaller messages have to queue behind enormous messages and suffer unfair delays

Prone:

- Packets have to be put in sequence at destination to produce original message
- Message segmentation results in many smaller packets. Since header size is usually same for all packets regardless of their size, with message segmentation total amount of header bytes are added to message

Queuing Delays and Packet Loss

- Hosts A and B are sending packets to Host E
- If, during a short interval of time, arrival rate of packets to router exceeds 15 Mbps, congestion will occur at router as packets queue in link's output buffer
- For example, if Host A and B each have sending windows five, then most of these packets will spend some time waiting in queue, some loss packet if buffer does not have enough space

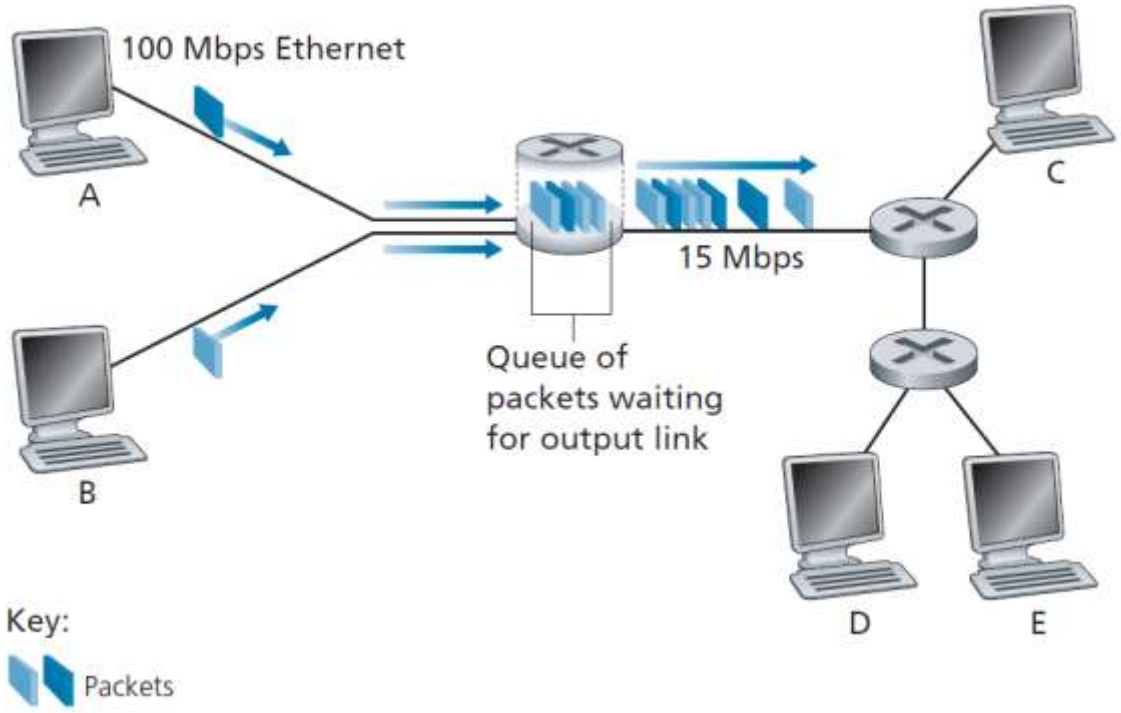


Figure 1.12 Packet switching

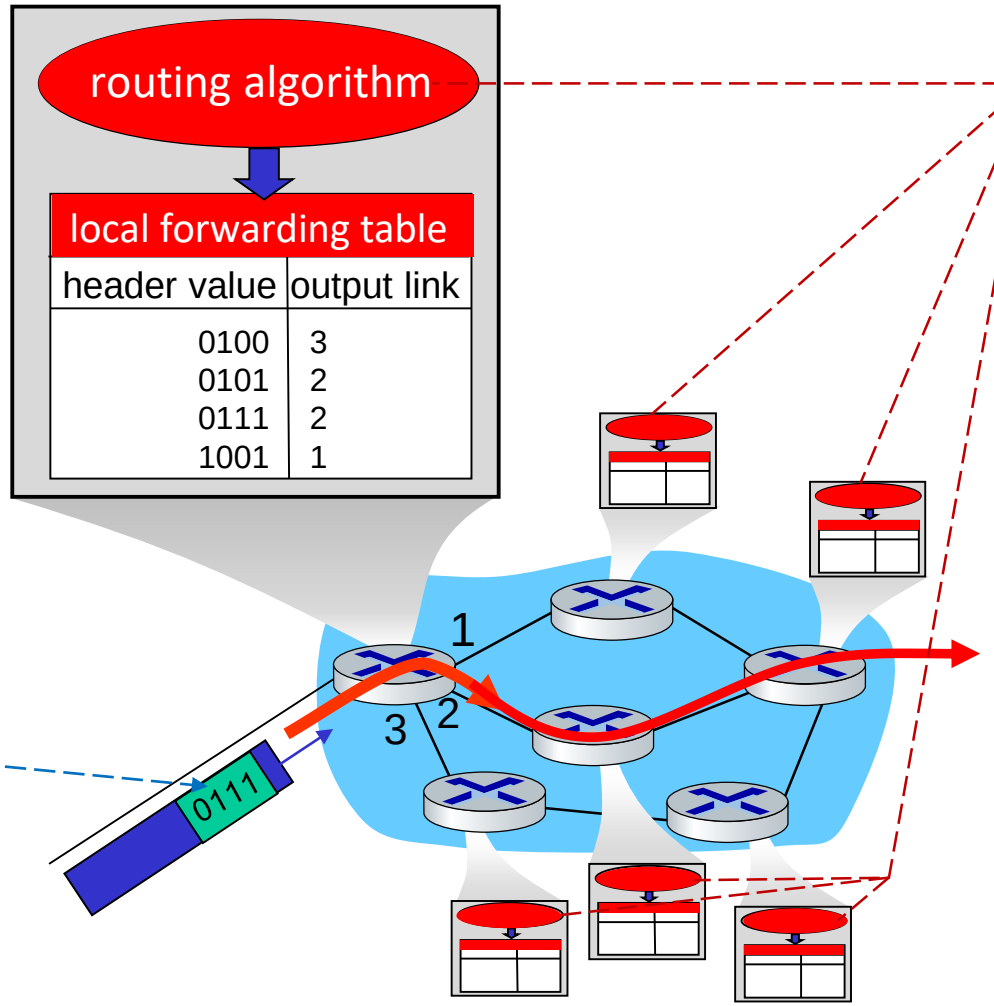
Forwarding Tables and Routing Protocols

- Each input port of router has a forwarding table that maps destination addresses (or portions of destination addresses) to that router's appropriate outbound links
- How do forwarding tables get set? (Chapter 5)
  - Routing protocols are used to automatically set forwarding tables
- Internet has a number of special routing protocols
- Each routing protocol includes a routing algorithm
- Routing algorithm of a routing protocol may, for example, determine shortest path from each router to each destination

Router's main functions

Forwarding:

local action: move arriving packets from router's input link to appropriate router output link



Routing:

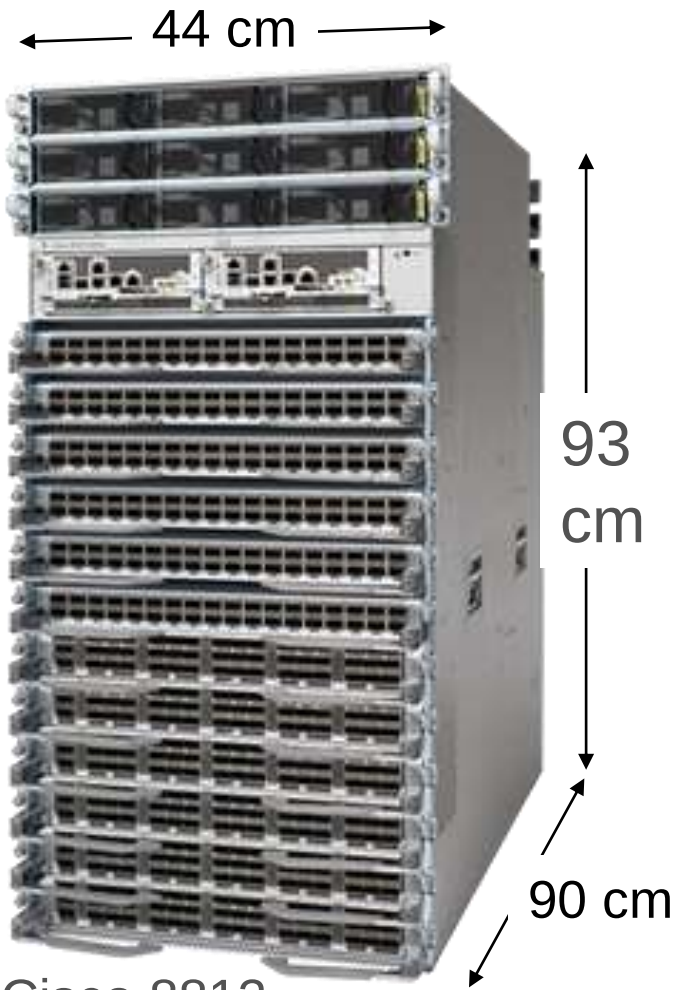
global action: determine source-destination paths taken by packets

routing algorithms

Packet Switch: Cisco 8800 Series Routers

- Cisco 8812 (single rack)
- Up to 648 ports
- Port bandwidth: 400-GbE (GbE=Gbps, Gigabit Ethernet)
- 648*400Gbps = 260 Tbps
- 36-port (400 GbE) line card
- 8 switch fabric cards

36 (400-GbE) line card

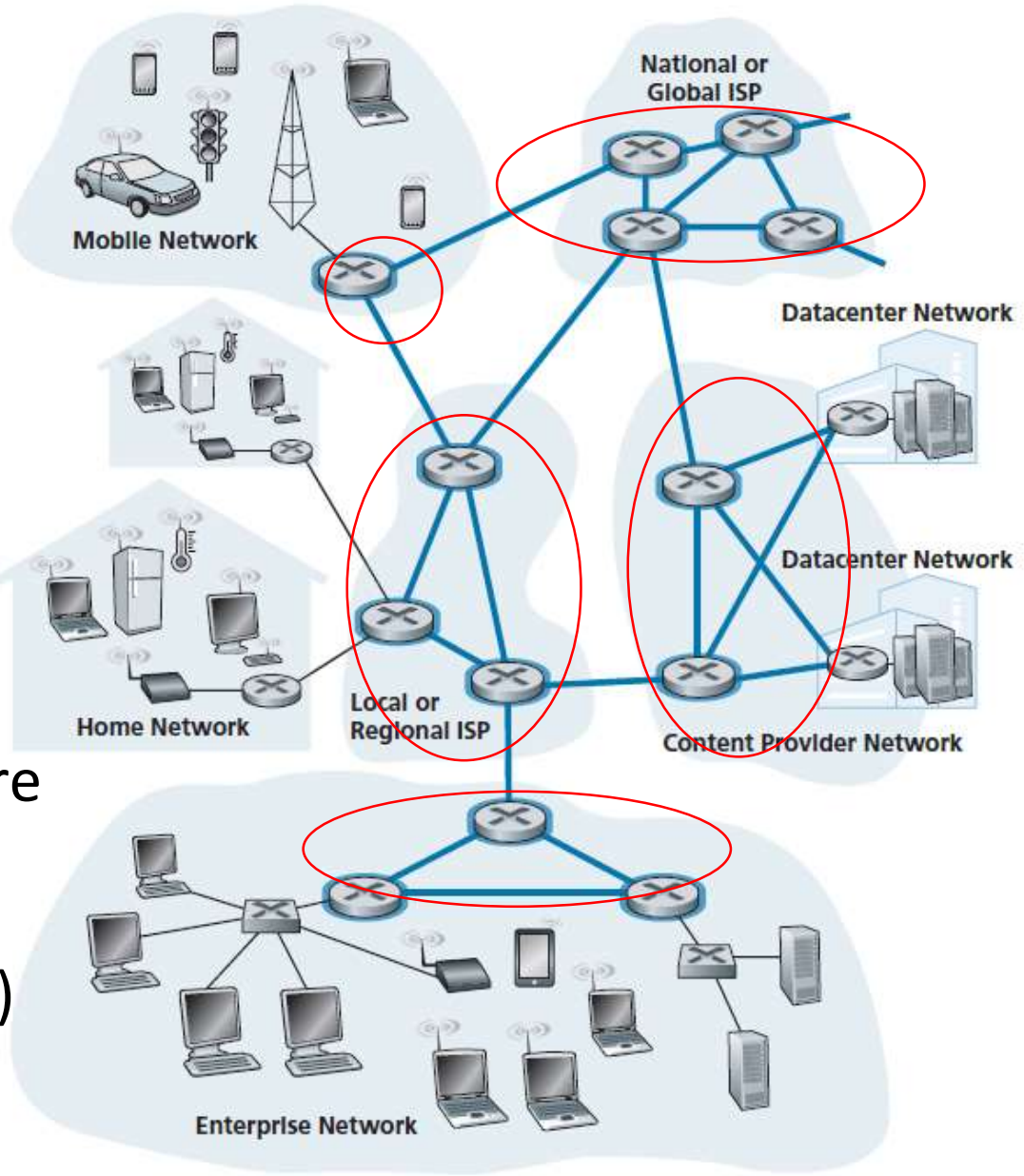


Cisco 8812

390 kg, 20 KW

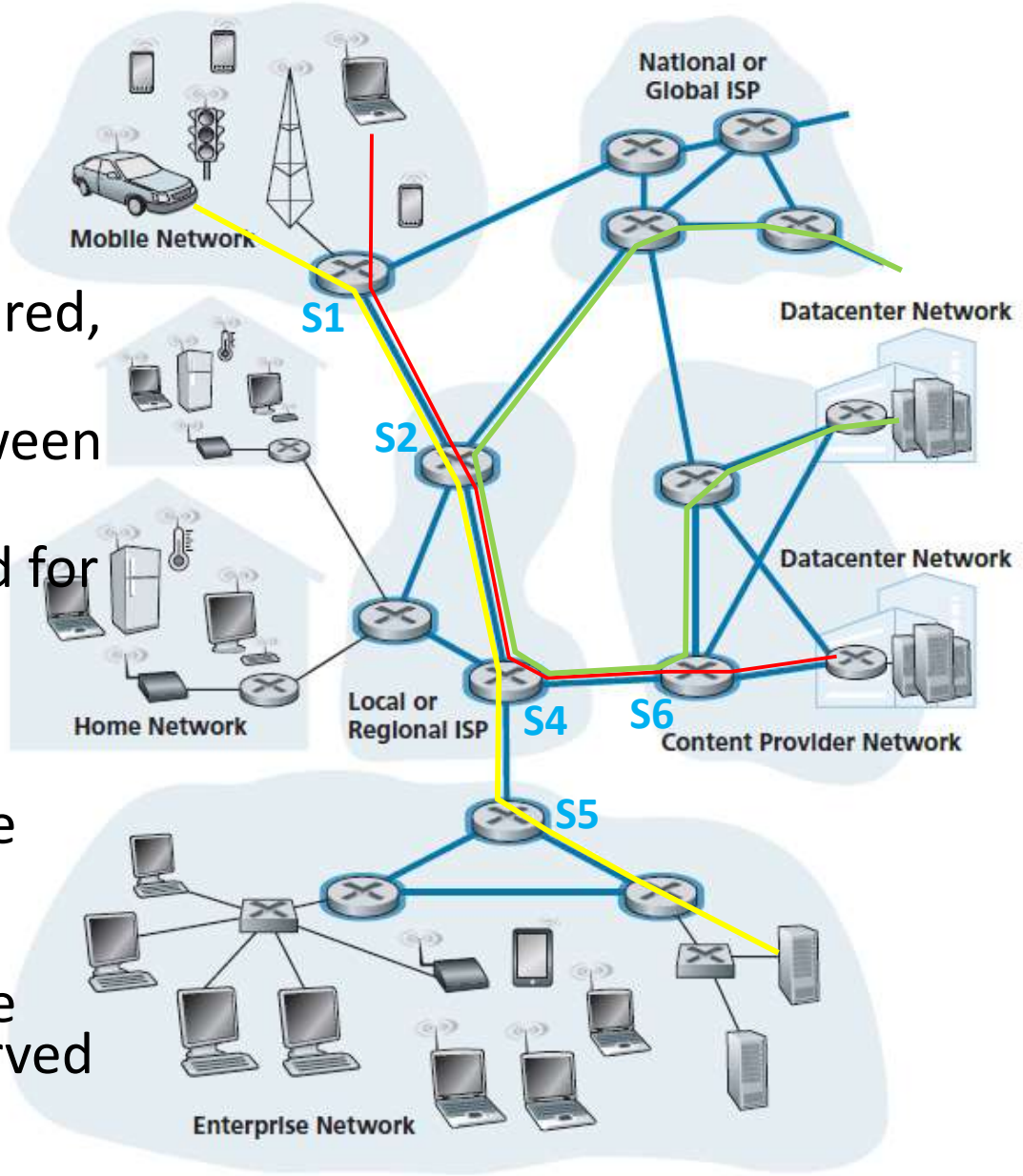
1.3.2 Circuit Switching

- Circuit-switched networks: resources needed along a path (link transmission rate) to provide for communication between end systems are reserved for duration of communication session between end systems
- Packet-switched networks: resources are not reserved; a session's messages use resources on demand and, as a consequence, may have to wait (queue) for access to a communication link



Resources reservation

- Circuit-switched networks: 3 paths (green, red, yellow)
- A path is set up before data exchange between APPs
- Part of each link transmission rate reserved for a path during path set up
- Reserve rate is freed just after path termination
- Link between circuit switches S1 and S2 are divided to n parts (circuits), one part is reserved for each path (red and yellow)
- Link between circuit switches S2 and S4 are divided to m parts (circuits), one part reserved for each path (green, red and yellow)



Circuit switching

- During a **path set up** between 2 hosts, circuit switches assign input and output circuits to that path
  - S2 switches (connects) one of its circuits to S1, one of its circuit to S4, and set up part of a path between 2 hosts

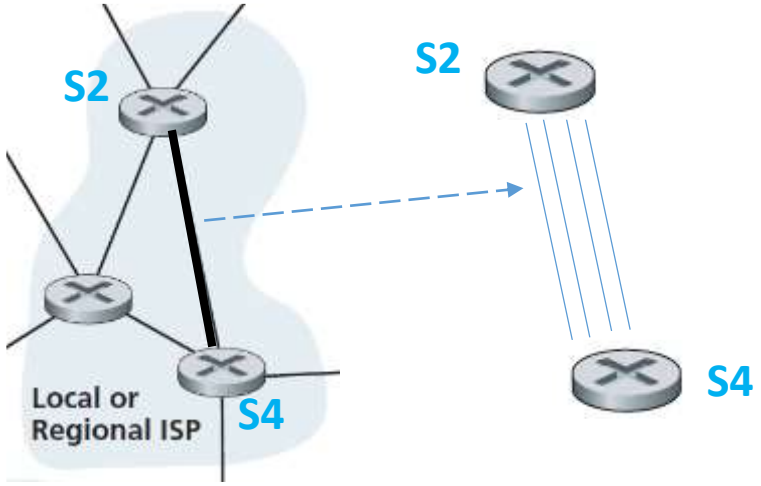
Performance:

- End to end bandwidth of a path is guarantee (circuits are dedicated to a path)
- No queue in circuit switches
- No store and forward (data move from input port to output port of each switch almost immediately)

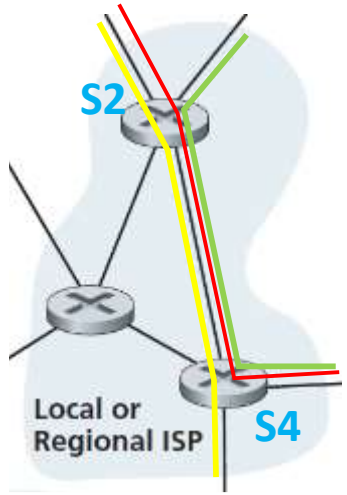
- commonly used in traditional telephone networks

Link partitioning

- Physical link between S2 and S4 is effectively divided into n partitions (circuits) using TDM or FDM or CDM technologies
- A partition is allocated to a particular path while it is up

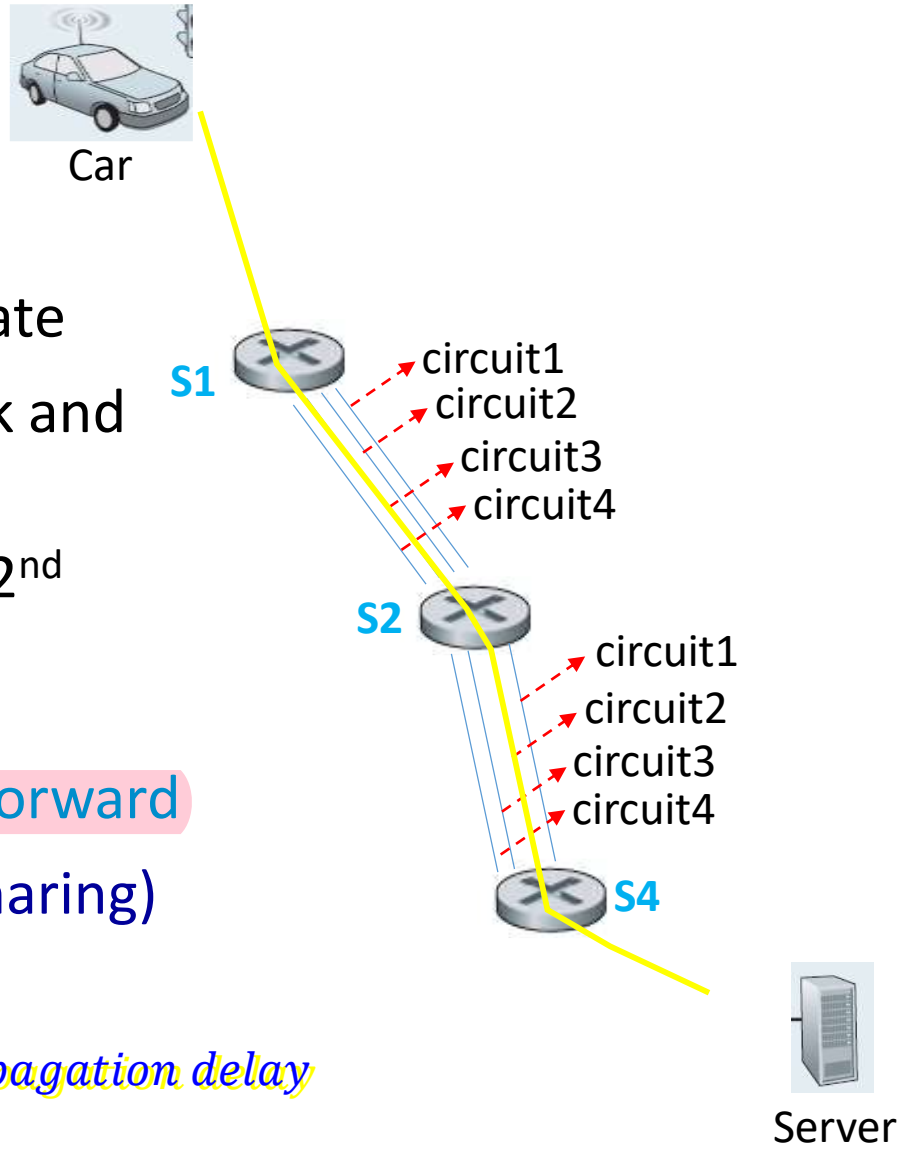


n =4



3 of 4 circuits are allocated to 3 paths, and one partition is free for future use

Example: Circuit switching

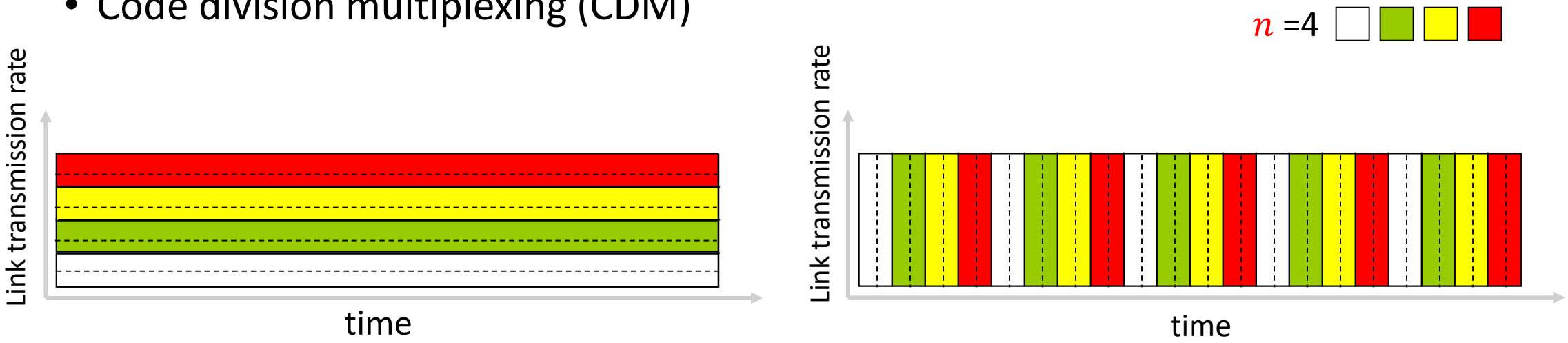


- In Figure, each link has four circuits, all have same rate
- Car-Server connection gets 3rd circuit in S1-to-S2 link and 2nd circuit in S2-to-S4 link
- Dedicated resources: 3rd circuit in S1-to-S2 link and 2nd circuit in S2-to-S4 link are dedicated to Car-Server connection (no sharing)
  - Bandwidth guarantee, No queue, No store and forward
- A circuit will be idle if not used by connection (no sharing)

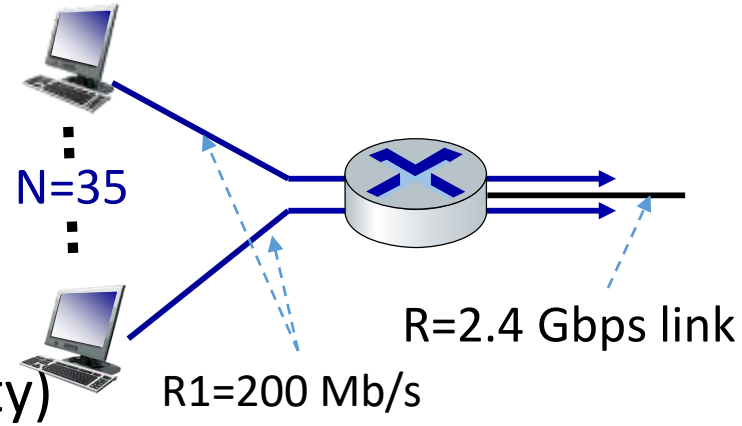
$$end\ to\ end\ delay = \frac{Filesize}{ciruit\ rate\ (bandwith)} + path\ propagation\ delay$$

Multiplexing in Circuit-Switched Networks

- Transmission capacity of a link between two circuit switch divided to n partitions (circuits). It is done by either of following technologies:
  - Time division multiplexing (TDM)
  - Frequency division multiplexing (FDM)
  - Code division multiplexing (CDM)



Packet Switching Versus Circuit Switching



Why is packet switching more efficient? Example:

- **Q:** circuit switch: How many concurrent connections?
- **A:** 2.4/0.2 = 12 (max concurrent connections)

Now suppose: p=0.1 (each computer data-sending probability)

- **Q:** Efficiency of circuit switching? **A:** 0.1 (90% of 2.4Gbps remains idle)
- **Q:** packet switch: How many concurrent connections? **A:** 35
- Q: Efficiency of packet switching? Probability of 12 or more computer sends at same time = 0.00002 (binomial distribution)
  - Efficiency will be 100% when 12 or more computer sending at same time
  - Queuing probability = 0.00002
  - **Q:** what happens if > 35 users? **A:** …

Packet Switching Versus Circuit Switching

- Circuit switching: R1 sending rate is limited to $2.4/n$ Gbps (n : number circuits)
- Packet switching: R1 sending rate is limited to 2.4 Gbps

Pros and prone of packet switched network

- Packet switched networks are great for "Bursty" data (Bursty: sometimes sending a lot packets, but at other times not, Sending a window of packets and wait for ACK, Chapter3)
- Resource sharing
- Simpler, cheaper, no connection (call) setup
- Excessive congestion possible: packet delay and loss due to buffer overflow
  - protocols needed for reliable data transfer, congestion control

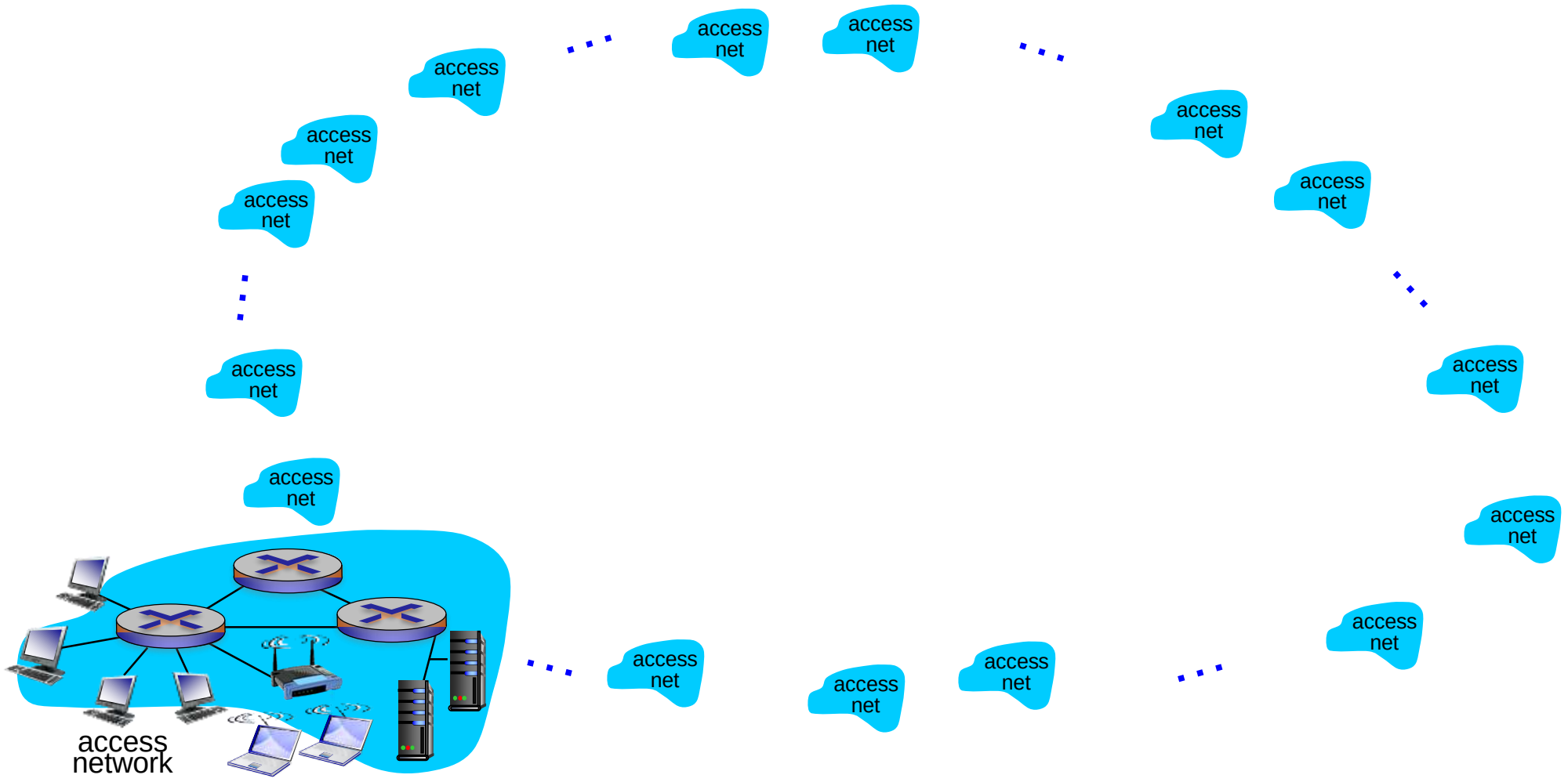
- **Q:** How to provide circuit-like behavior? Virtual Circuit switching
  - bandwidth guarantees

1.3.3 A Network of Networks - NoNs

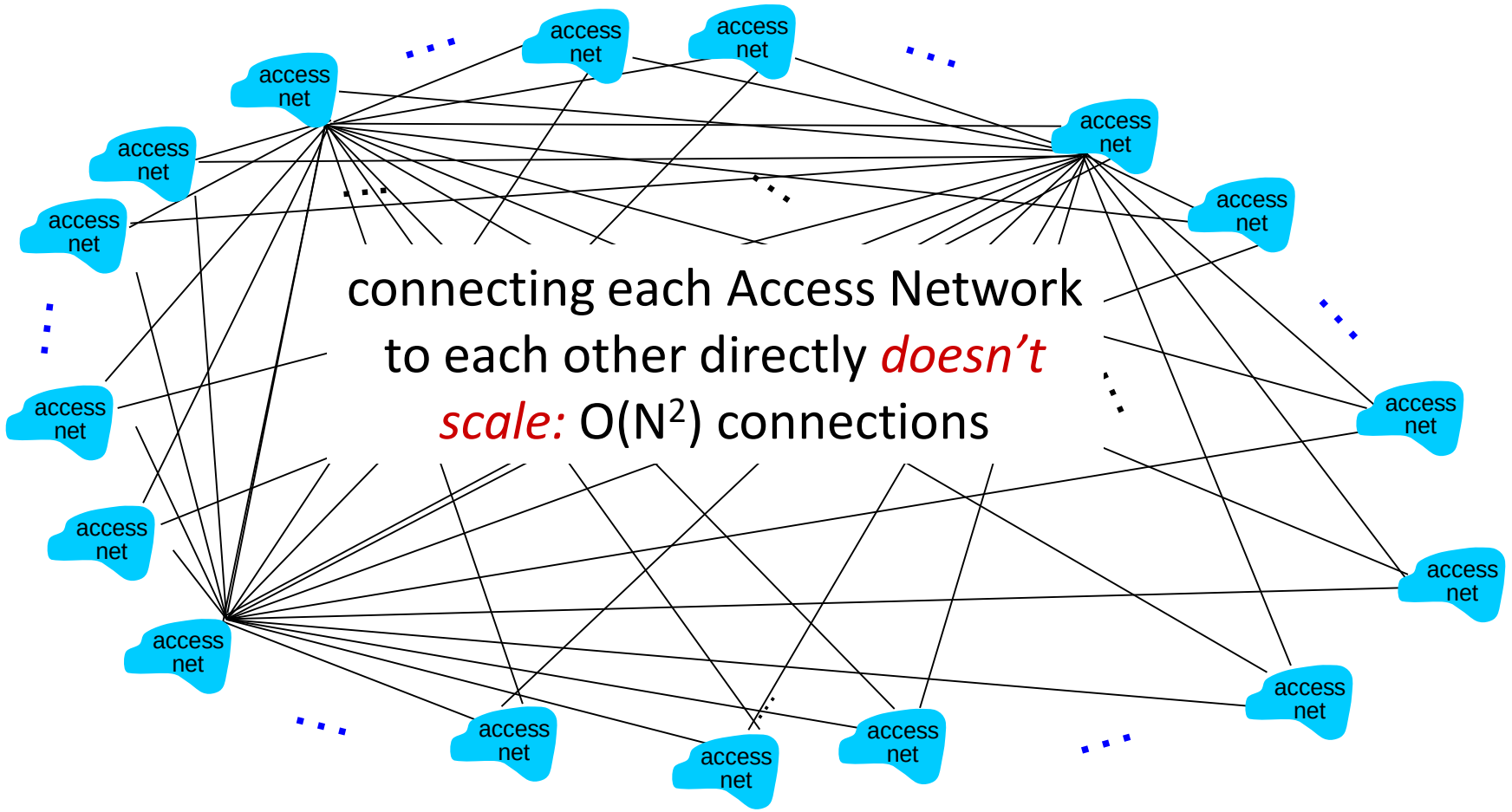
- Hosts are located in Access networks. We call it Access ISP
  - Access ISPs: Residential, enterprise (company, university, commercial), mobile
- Access networks must be interconnected, so any two hosts can send packets to each other
- Resulting is a **network of Access networks (a Network of Networks)**

- Let's take a stepwise approach to describe current Internet structure (NoNs)

NoNs: direct connections - Naive approach



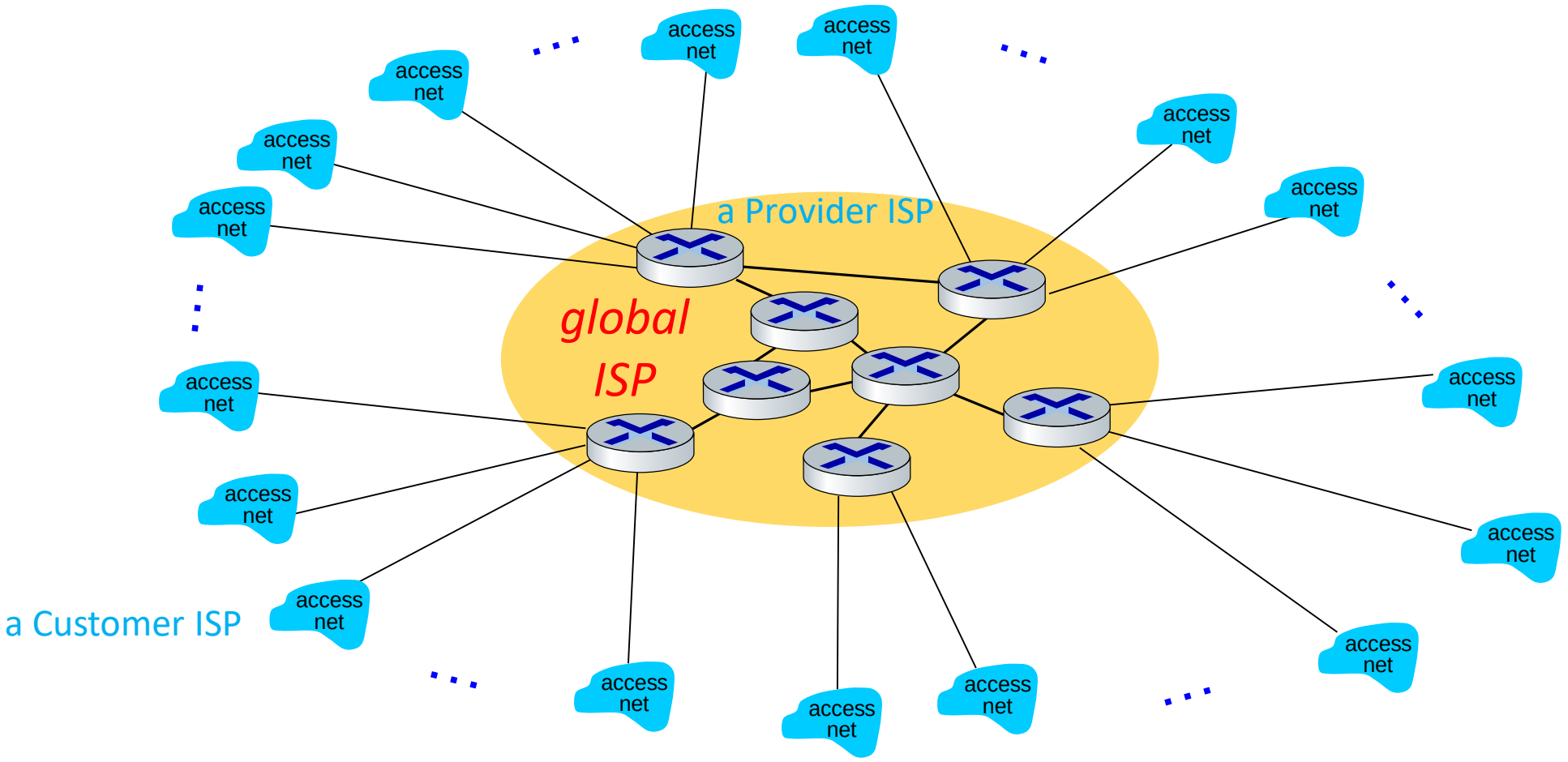
NoNs: direct connections - Naive approach



NoNs: Using (imaginary) global transit ISP

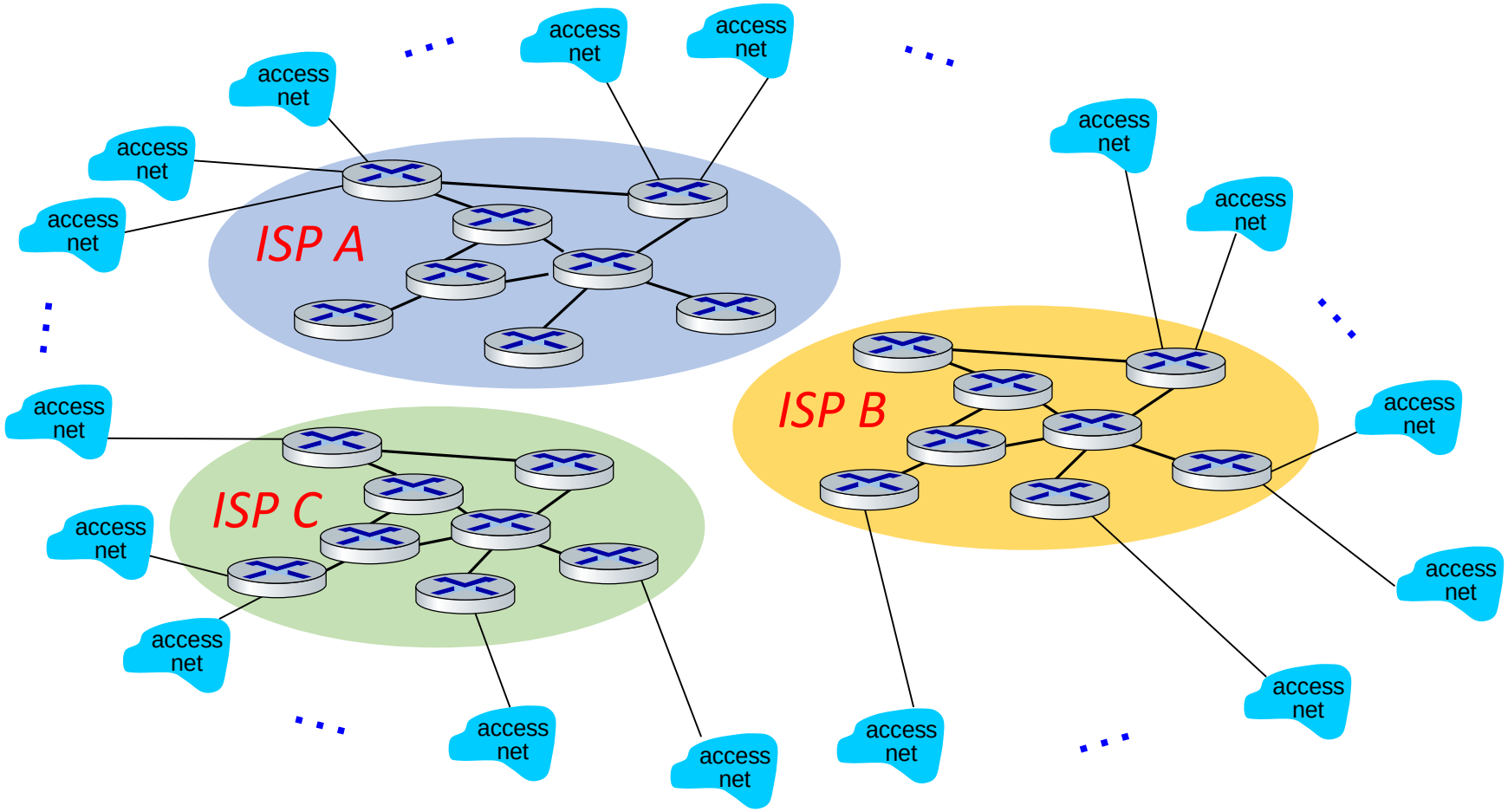
- Connect each Access Network to one global transit ISP?
- Global ISP is a network of routers (network core) and communication links that not only spans globe, but also has at least one router near each of hundreds of thousands of Access networks
- Global transit ISP is a provider ISP
- Access ISPs (network) are customer
- There will be a contract between each customer and provider
- Customers pay to obtain global interconnectivity

NoNs: Using (imaginary) global transit ISP



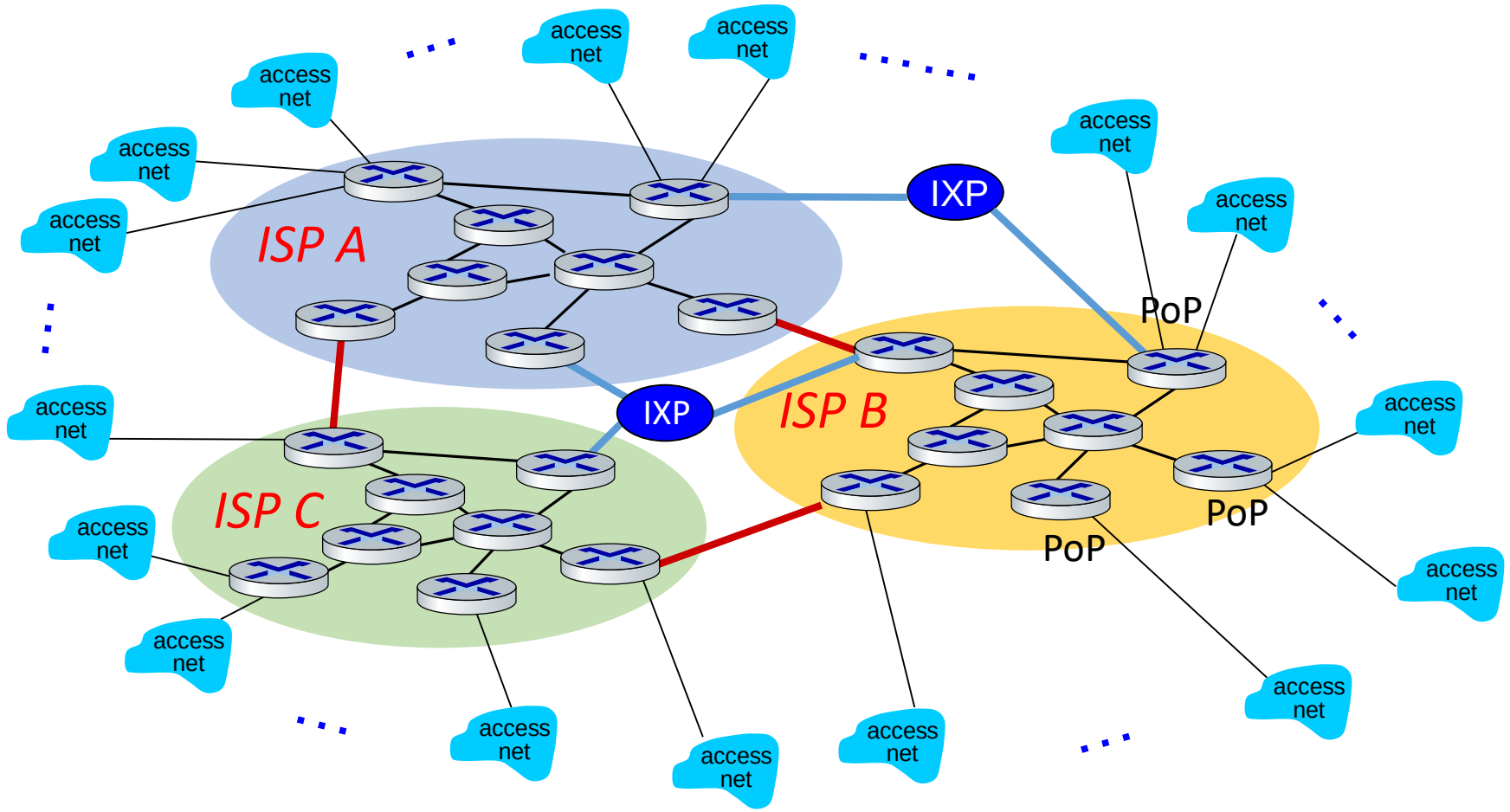
NoNs: Several global transit ISPs

- But, if one global ISP is capable of working successfully, then, there will be competitors ….



Interconnected global transit ISPs

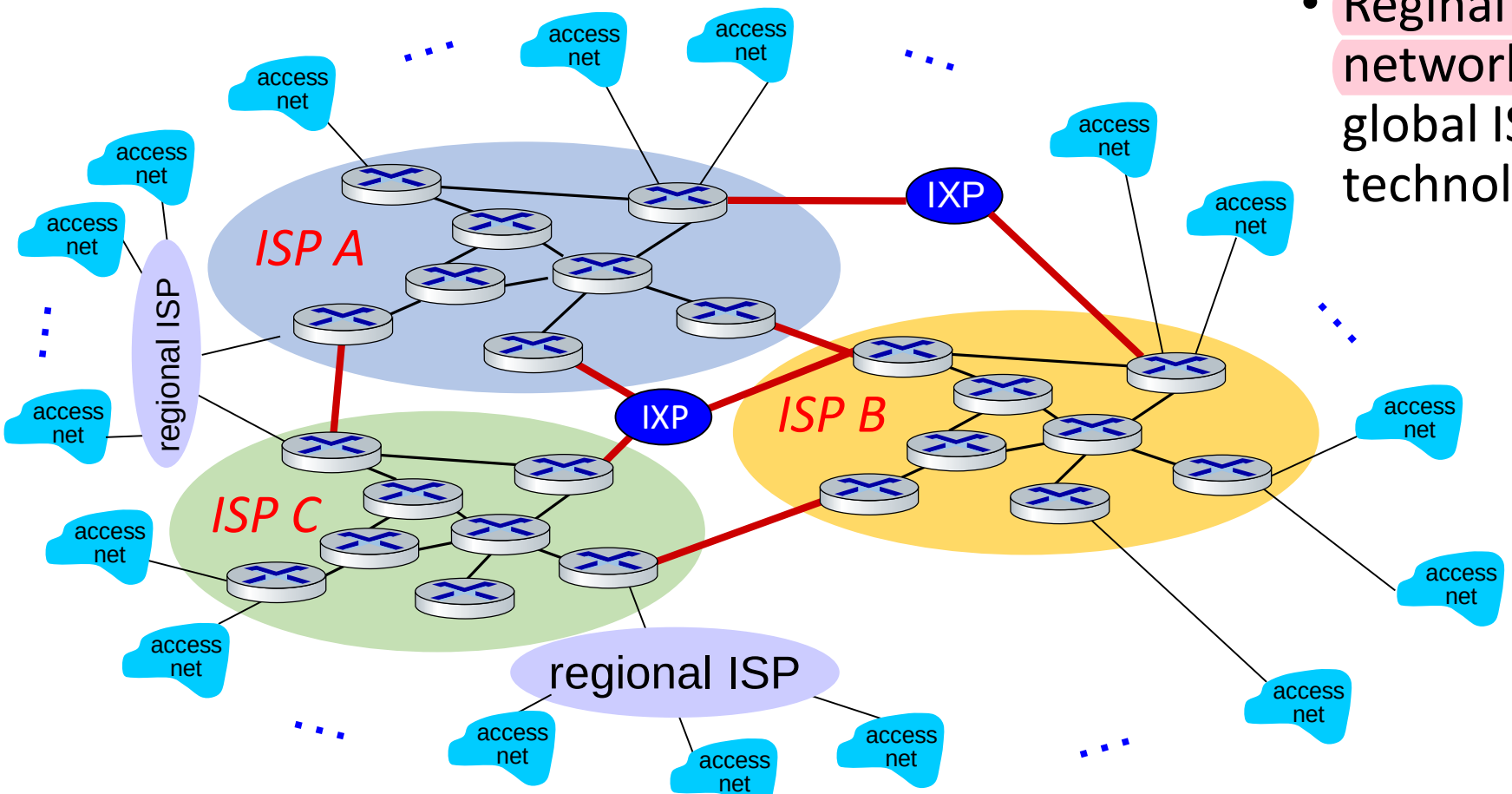
- An Access ISP needs to connect to all global transit ISPs?
- Global ISPs interconnected?



peering link

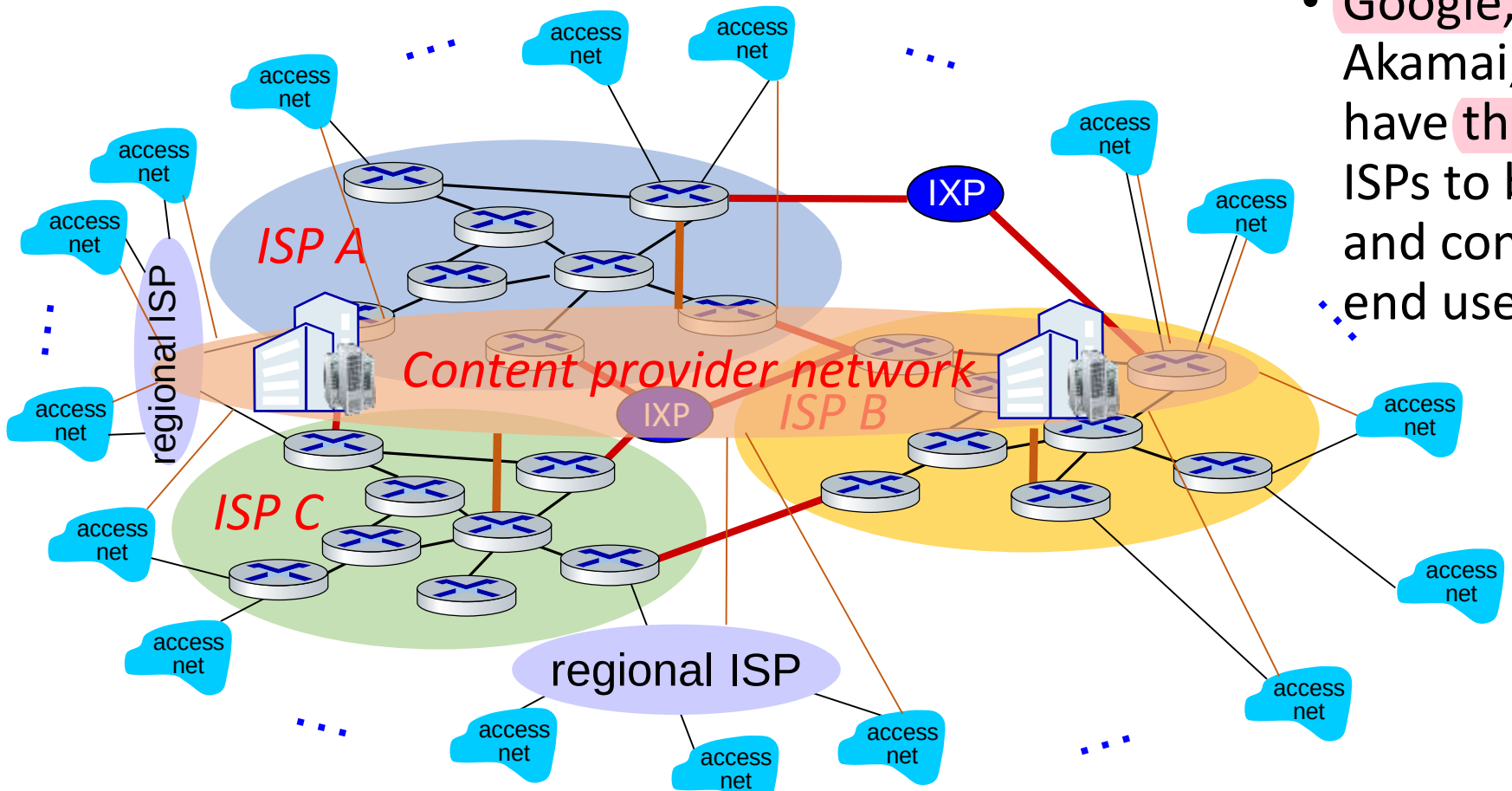
Internet exchange point IXP

Transit ISPs with reginal coverage (regional ISP)



- Reginal ISPs are core networks similar to global ISPs technologies

Private Global ISPs (Content provider networks)

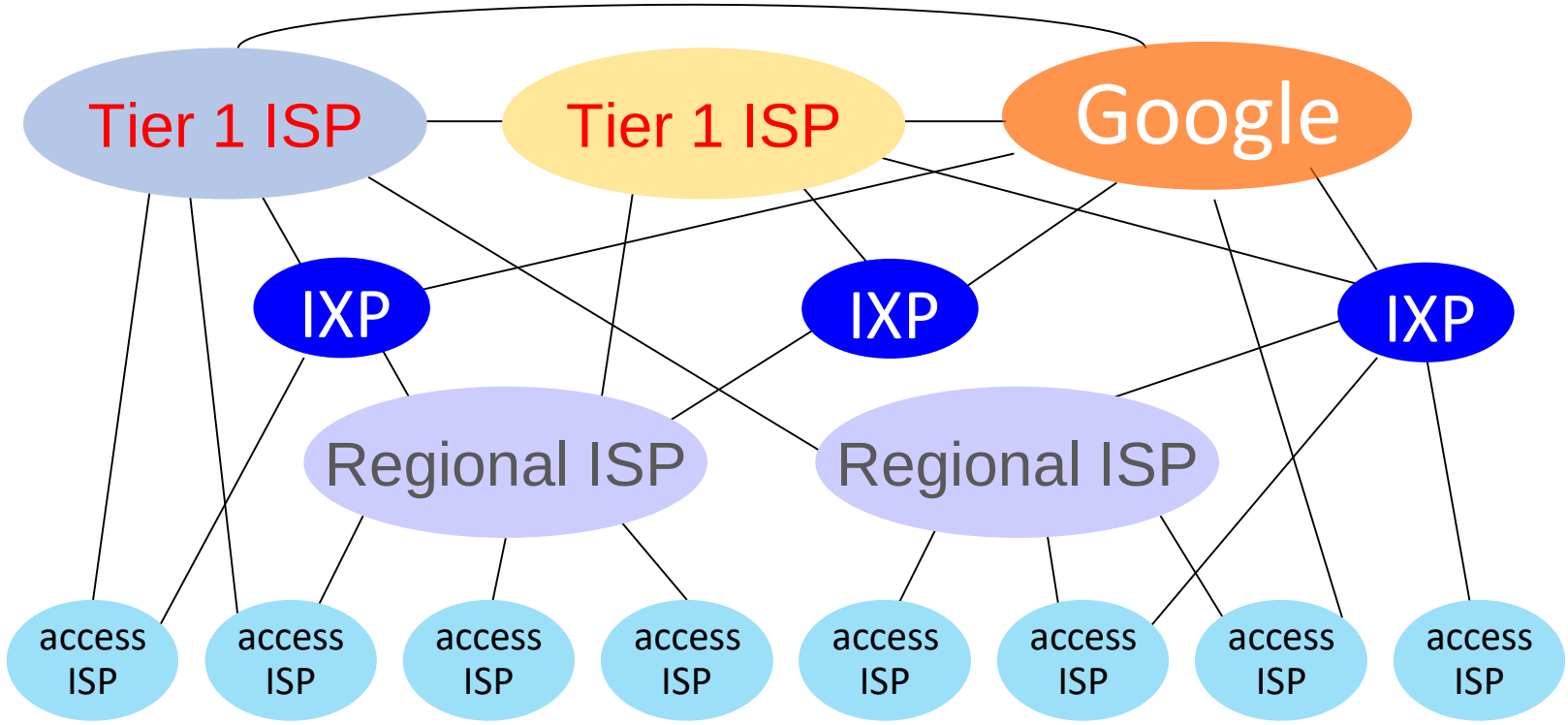


- Google, Microsoft, Akamai, Facebook, …, have their own Global ISPs to bring services and contents close to end users

Internet structure

- **At "top":** small number of well-connected large networks:
  - "tier-1" commercial ISPs (e.g., Level 3, Sprint, AT&T, NTT), national & international coverage

- Content provider networks (e.g., Google, Facebook): private tier-1 network that connects its data centers to Internet



Content-provider networks (CDNs)

- Google has a content-provider network
- Google has nearly 100 data centers distributed across North America, Europe, Asia, South America, and Australia
- Some of these data centers house over 100,000 servers
- Google data centers are all interconnected via Google's private tier1 network, which spans entire globe but is separate from public Internet
- It only carries traffic to/from Google servers
- It "bypasses“ other tier1 ISPs by peering with lower-tier ISPs
- It also connects to tier-1 ISPs, and pays those ISPs for traffic it exchanges with them

Tier-1 ISP Network map: Sprint (2019)

POP: point-of-presence

links to/from Sprint customer networks

Tier-1 ISP: e.g., HE

https://he.net/ip_transit.html?p=&t=&m=p&k=cisco%20core%20router&gclid=CjwKCAiAvonyBRB7EiwAadauqawU2ax8fsYiVzpxNGurHMkKi9cabD_qux-PN_mDBbjXTCT_nYhsOxoCq3sQAvD_BwE

https://he.net/3d-map/

PoP, Multi-homing, Peering

- Points of Presence (PoPs): where customer ISPs can connect into provider ISP (one or more routers in provider's network ready for customer physical connections)
- Multi-homing: Any Access ISP may connect to two or more provider ISPs. Example: an access ISP may multi-home with two regional ISPs, or it may multi home with two regional ISPs and also with a tier-1 ISP
- Peering: a pair of provider ISPs at same level of hierarchy can directly connect their networks together so that all traffic between them passes over direct connection rather than through upper ISP

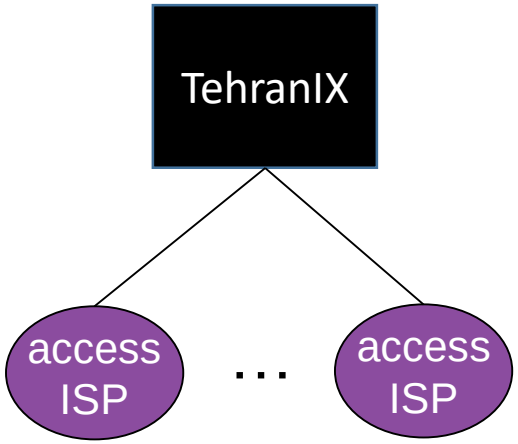
IXP (IX)

- A third-party company can create an Internet Exchange Point (IXP) services, which is a meeting point where multiple ISPs can peer together
- An IXP is typically in a stand-alone building with its own switches

IXs in Iran

- Iran's seven IXs:
  - Tehran (8400Gbps)
  - Mashhad,
  - Tabriz (460Gbps)
  - Shiraz (460Gbps)
  - Qom
  - Ahvaz

https://tehran-ix.ir/



Tehran IX members

Member	Joined	ASNASN
Parsonline	2019-12-18	16322
Pishgaman	2019-12-18	49100
Respina	2019-12-18	42337
Irancell	2019-12-17	44244
Tehran IX	2019-10-28	50722

Member	Joined	ASNASN
Mabna	2020-04-19	51074
Bahar Samaneh	2020-04-02	56466
Iran FCP ertebatat sabet parsian	2020-04-02	44400
Sabanet	2020-04-02	39501
Sharif University	2020-04-02	12660
Tehran University	2020-04-02	29068
Sabaidea	2020-03-18	44932
Greenweb	2020-01-14	61173
Hostiran	2020-01-14	59441
Afranet	2020-01-12	25184
Avabarid	2020-01-12	51431
Fanap	2020-01-12	24631
MCI	2020-01-12	197207
Mobinnet	2020-01-12	50810
Shatel	2020-01-12	31549
TCI	2020-01-12	58224
Rightel	2019-12-19	57218
Asiatech MANRS	2019-12-18	43754
Hiweb	2019-12-18	56402

Member	Joined	ASN
Shabdiz MANRS	2021-01-13	50530
Sepanta	2021-01-11	39074
Farhang Azma	2020-12-10	44889
Afrarasa	2020-11-17	202391
Hamta-One	2020-11-09	262143
Hamta-Two	2020-11-09	49832
Amin IDC	2020-09-09	48147
Sima Rayan Sharif	2020-08-20	64422
Bisphone	2020-08-05	34513
IRNIC	2020-07-25	39200
TIC	2020-07-22	12880
Iranet MANRS	2020-06-16	6736
IRIB	2020-05-30	42586
Khalij Online	2020-05-16	48944
Sefroyek	2020-05-16	44285
Sadad	2020-05-09	39571
Fanava	2020-04-19	41881
Farzanegan Pars	2020-04-19	200370
Helma Gostar	2020-04-19	49103
Laser	2020-04-19	34636

Iran University Of Science and Technology

- Autonomous System: AS41620
- AS Name: IUSTCC-AS
- Registry: RIPE
  - **RIPE NCC** is regional Internet registry (RIR) for Europe, West Asia, and former USSR. It is headquartered in Amsterdam, with a branch office in Dubai
- Registration date: 2006-09-26, Registration change: 2018-09-04
- IPv4 prefix: 194.225.224.0/20 (194.225.224.0 - 194.225.239.255), 2^{12}=4096 IP4 addresses
- IPv6 prefix: -
- **276**Hosted Domains

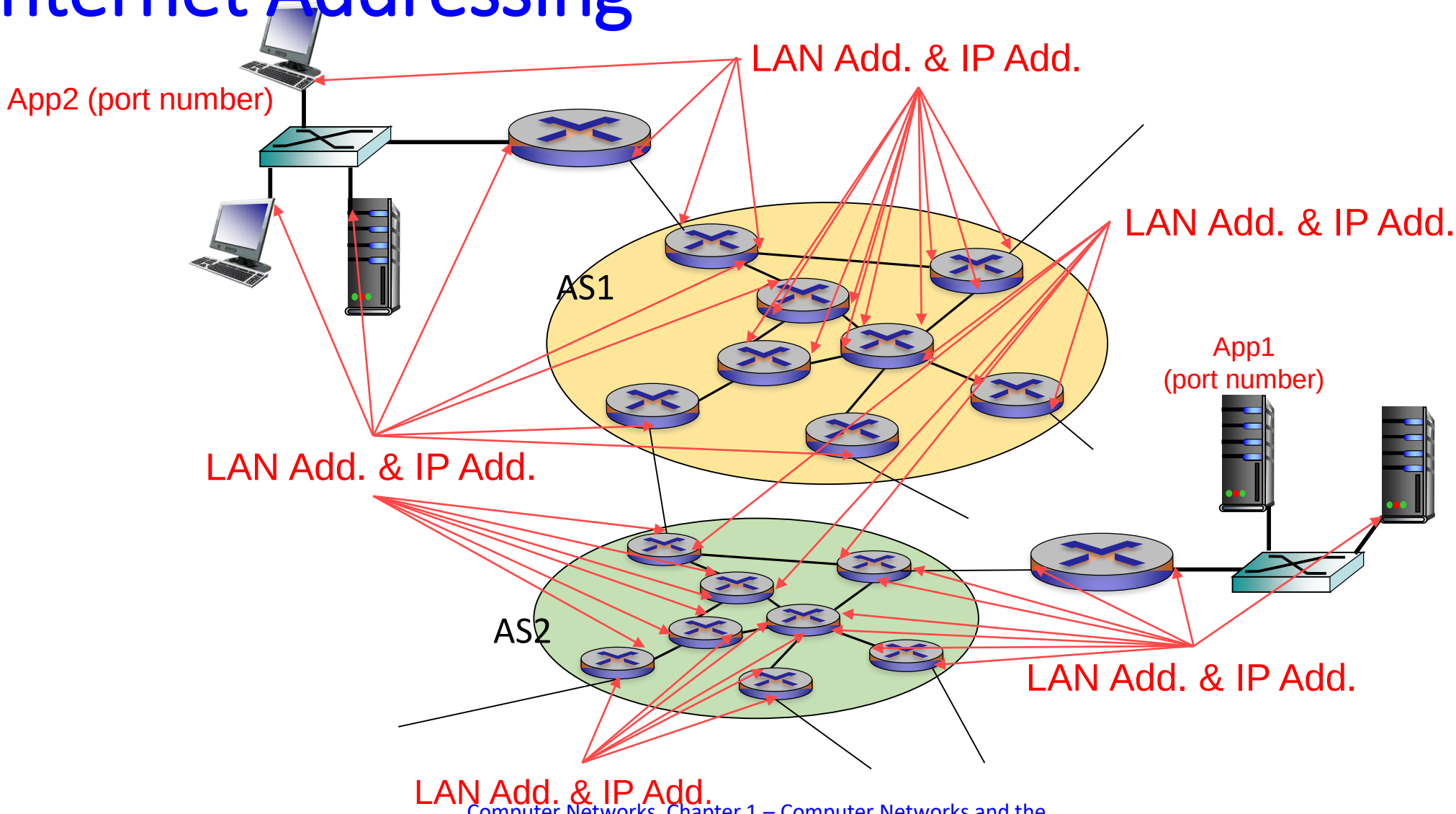
Contents

1.1 What Is the Internet?
1.2 The Network Edge
1.3 The Network Core
1.3’ The Name and Addresses
1.4 Delay, Loss, and Throughput in Packet-Switched Networks
1.5 Protocol Layers and Their Service Models
1.6 Networks Under Attack
1.7 History of Computer Networking and the Internet
1.8 Summary
Appendix

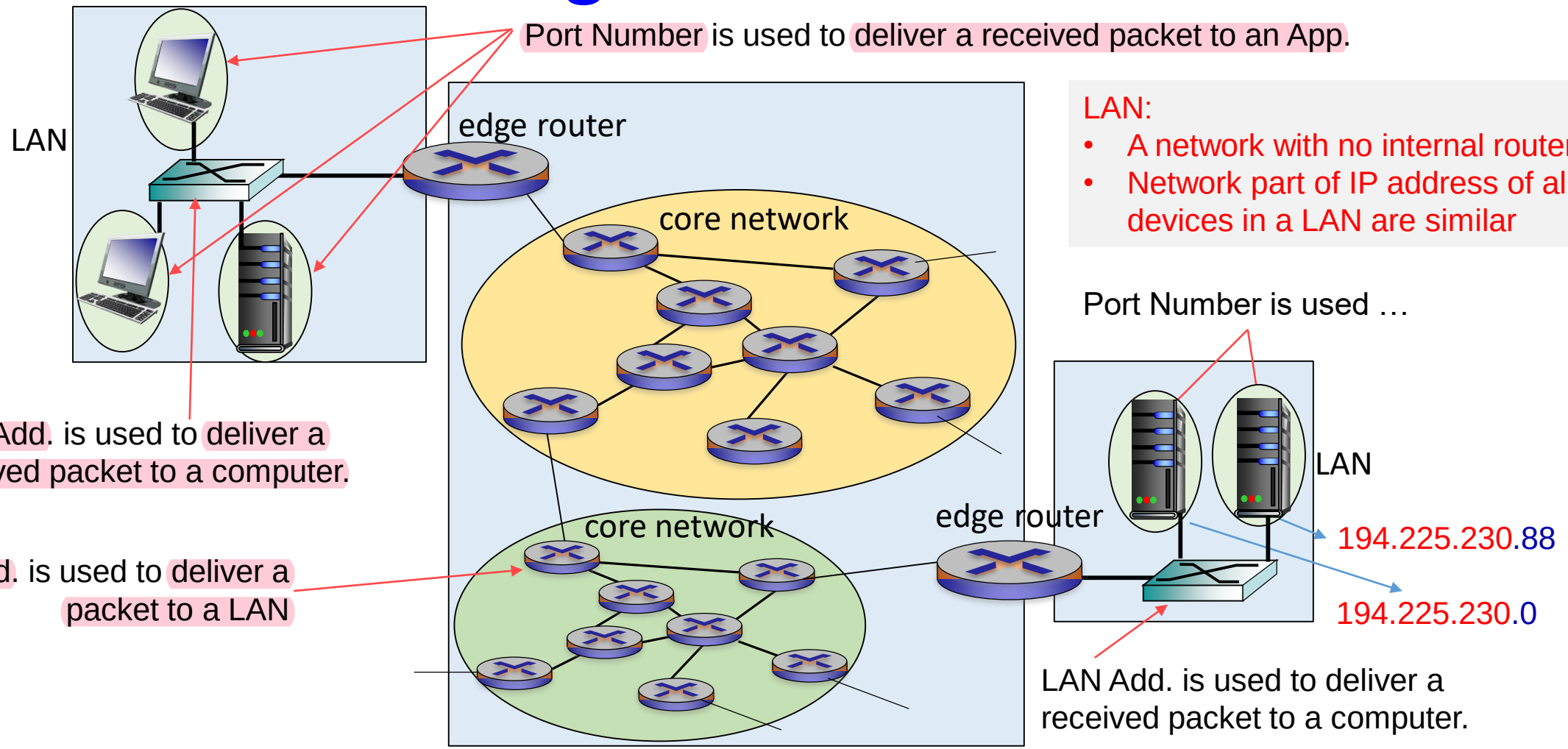
Name and Addresses

- In Internet how to address:
  - A computer, printer, refrigerator, … (end system): 194.225.230.66
  - A server: 194.225.230.66 or mxgate.iust.ac.ir
  - A person (user): azar@iust.ac.ir, http://twitter.com/azar,
  - A file (WWW URL): www.matthewzeiler.com/mattzeiler/hierarchicalconvolutional.pdf
  - A process (running app): 194.225.238.111:80 or www.iust.ac.ir:80
  - A router: Inteface1 62.40.103.54, Interface2 210.14.56.84, …
  - A LAN switch (L2 switch): -
  - A hardware (Physical, MAC, LAN Address): 3A:55:01:EF:49:12

Internet Addressing



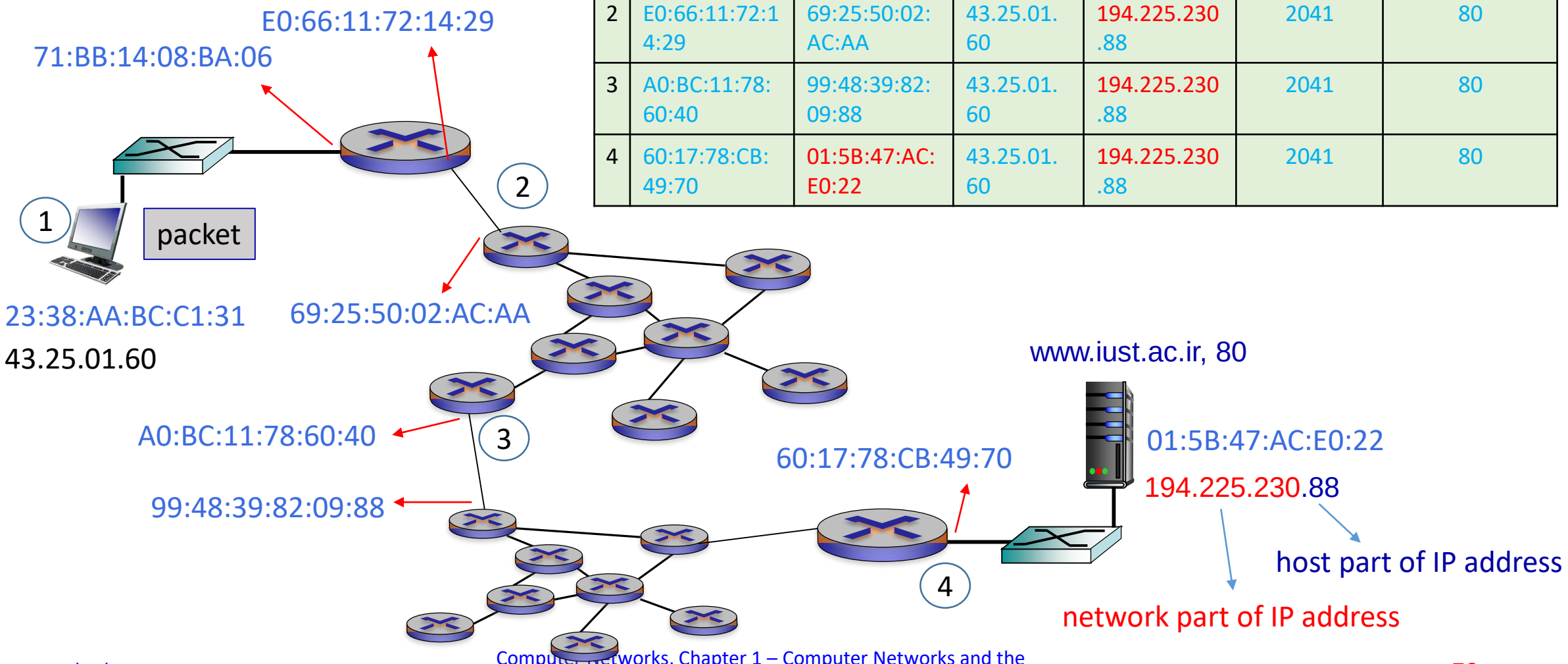
Internet Addressing



Example

Addresses in a packet traveling from source process to destination process

	Source Mac Address	Designation Mac Address	Source IP Address	Destination IP Address	Source Port Number	Destination Port Number
1	23:38:AA:BC:C1:31	71:BB:14:08:BA:06	43.25.01.60	194.225.230.88	2041	80
2	E0:66:11:72:14:29	69:25:50:02:AC:AA	43.25.01.60	194.225.230.88	2041	80
3	A0:BC:11:78:60:40	99:48:39:82:09:88	43.25.01.60	194.225.230.88	2041	80
4	60:17:78:CB:49:70	01:5B:47:AC:E0:22	43.25.01.60	194.225.230.88	2041	80



Contents

1.1 What Is the Internet?
1.2 The Network Edge
1.3 The Network Core
1.3’ The Name and Addresses
1.4 Delay, Loss, and Throughput in Packet-Switched Networks
1.5 Protocol Layers and Their Service Models
1.6 Networks Under Attack
1.7 History of Computer Networking and the Internet
1.8 Summary
Appendix

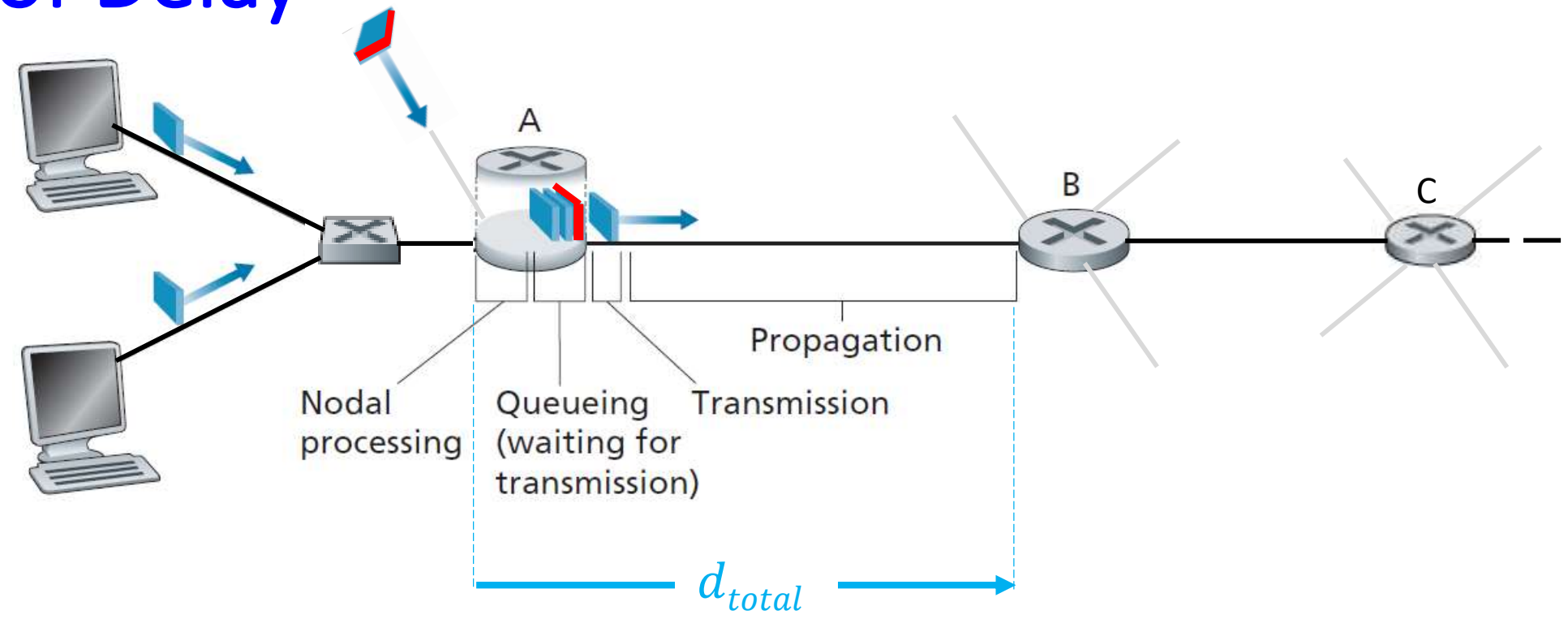
1.4 Delay, Loss, and Throughput in Packet-Switched Networks

- Internet provides services to distributed applications running on end systems
- We like services to be able to move as much data as we want between any two end systems, instantaneously, with no loss of data
- This is unachievable in reality
- Computer networks constrain throughput (amount of data per second that can be delivered) between end systems, introduce delays between end systems, and can actually lose packets
- Here: we discuss quantify delay, loss, and throughput in computer networks

1.4.1 Overview of Delay in Packet-Switched Networks

- A packet starts in a host (source), passes through a series of routers, and ends in another host destination)
- Several types of delays at each node along a path
- Delays: **nodal processing delay**, **queuing delay**, **transmission delay**, **propagation delay**
- These delays accumulate to give a **total nodal delay**
- Performance of network APPS (Web browsing, e-mail, maps, instant messaging, and voice-over-IP) are affected by delays

Types of Delay



A to B (1 hop) → $d_{total} = d_{proc} + d_{queue} + d_{trans} + d_{prop}$

Figure 1.16 The nodal delay for a packet from entering into router A to entering into router B

Processing Delay

d_{proc} : nodal processing

- Check bit errors
- Determine output link (forwarding table lookup)
- Switch packet into buffer of output link
- Typically takes several **Nano to Micro second** timescale

Queuing Delay

d_{queue} : queueing delay

- A packet waits to be transmitted into output link
- Waiting time depends on congestion level at output buffer (number of earlier-arriving packets stored in output buffer)
- If queue is empty, then packet's queuing delay will be zero
- Queuing delays can be on the order of **microseconds to milliseconds in practice**

Transmission Delay

d_{trans}: transmission delay:

- L : packet length (bits)
- R : link transmission rate (bps)
- This is amount of time required to push (transmit) all of packet's bits into link
- d_{trans} = L/R [sec]

Propagation Delay

d_{prop}: propagation delay:

- d : length of physical link
- s : propagation speed (~$2x10^8$ m/sec)
- Once a bit is pushed into link, it needs to propagate to next node
- Time required to propagate from beginning of a link to end of the link is **propagation delay**
- Bit propagates at propagation speed of link. Speed depends on physical medium of link (fiber optics, twisted-pair copper wire, …)
- d_{prop} = d/s [sec]

Comparing Transmission and Propagation Delay

- Cars “propagate” at 100 km/hr
- Toll booth takes 12 sec to service car (bit transmission time=1/R ~12 sec)
- Car ~ bit; caravan ~ packet
- **Q:** How long until caravan is lined up before 2nd toll booth?

Transmission delay

- Time to “push” entire caravan through toll booth onto highway = 12*10 = 120 sec = 2 minutes
- Time for last car to go from 1st to 2nd toll both: 100km/(100km/hr) = 1 hr= 60 minutes
- **A:** 62 minutes

Propagation delay

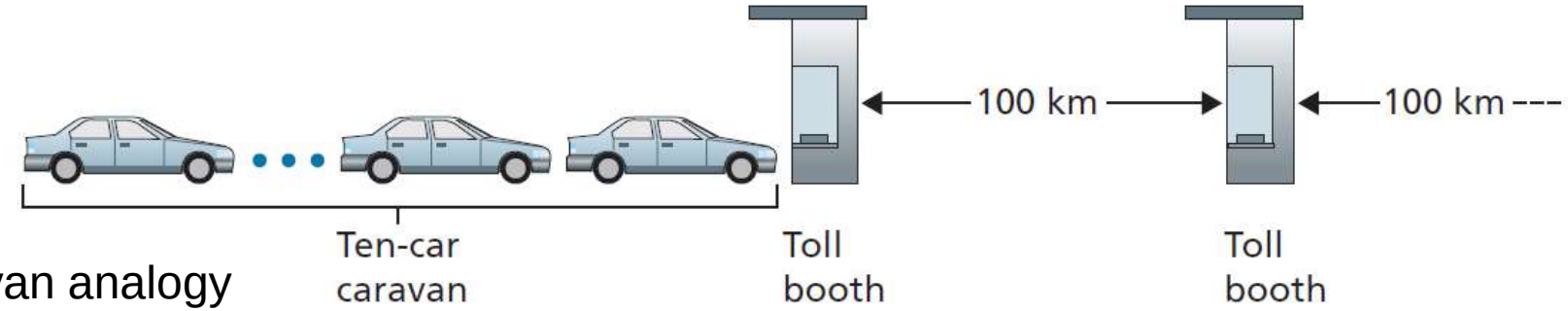


Figure 1.17 Caravan analogy

Comparing Transmission and Propagation Delay

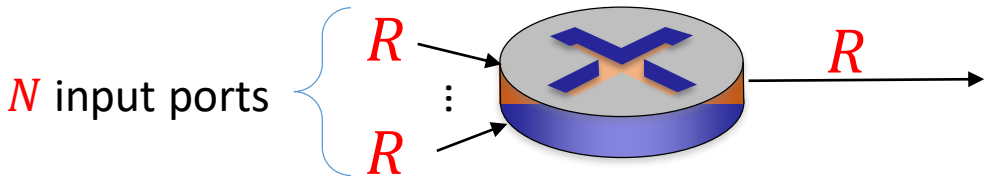
Suppose:

- 1- Cars “propagate” at 1000 km/hr (a car needs 6 min to travel 100Km)
- 2- Toll booth takes one min to service a car
- **Q:** Will cars arrive to 2nd booth before all cars serviced at first booth?
- **A:** Yes. After **1+6=7** min, first car arrives at second booth; **three** cars still at first booth
- This situation also arises in packet-switched networks
  - First bits in a packet can arrive at a router while many of remaining bits in packet are still waiting to be transmitted by preceding router

1.4.2 Queuing Delay and Packet Loss

Why queuing?

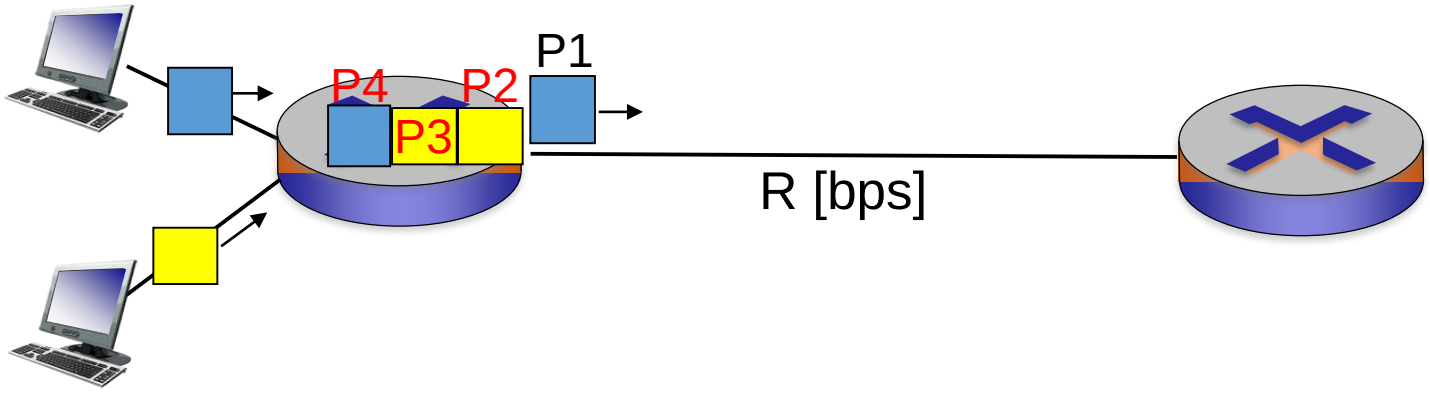
- Packet entered into N input ports may send to same output port:
- **Packet arrival rate temporarily exceeds output link capacity**



- **Packets queue in buffer of output port**
- Suppose N packets arrive at buffer of output port at same time, first packet will suffer no queuing delay, while last packet will suffer a relatively large queuing delay

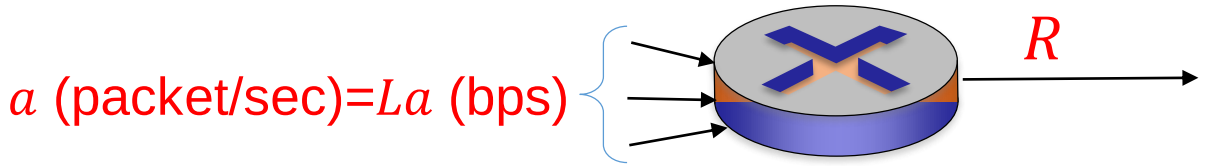
Average Packet Queuing Delay

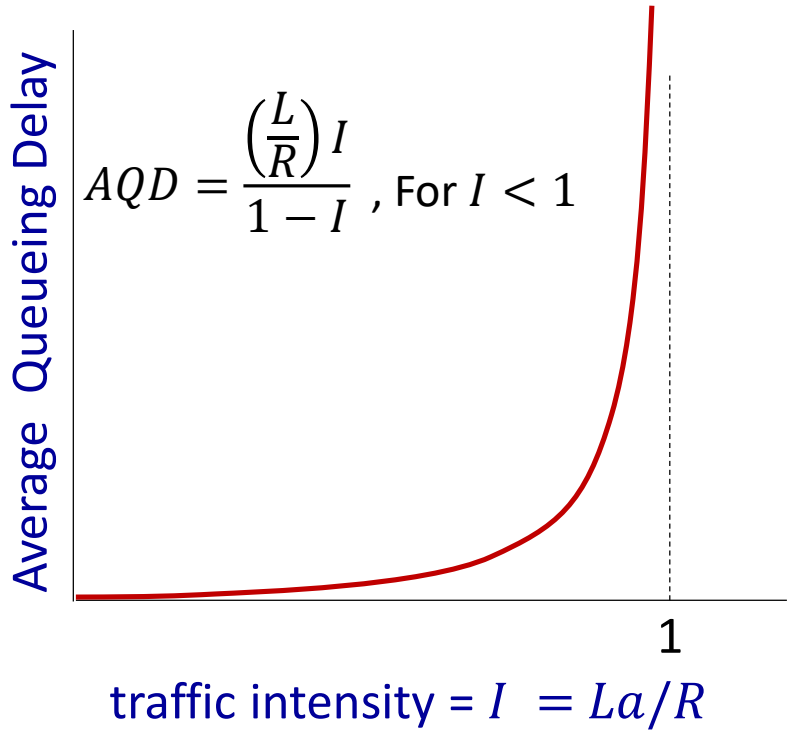
- P1 Queuing Delay = 0 sec
- P2 Queuing Delay = P1/R
- P3 Queuing Delay = P1/R + P2/R
- P4 Queuing Delay = P1/R + P2/R + P3/R
- Average Queuing Delay = (3P1/R + 2P2/R + P3/R)/4



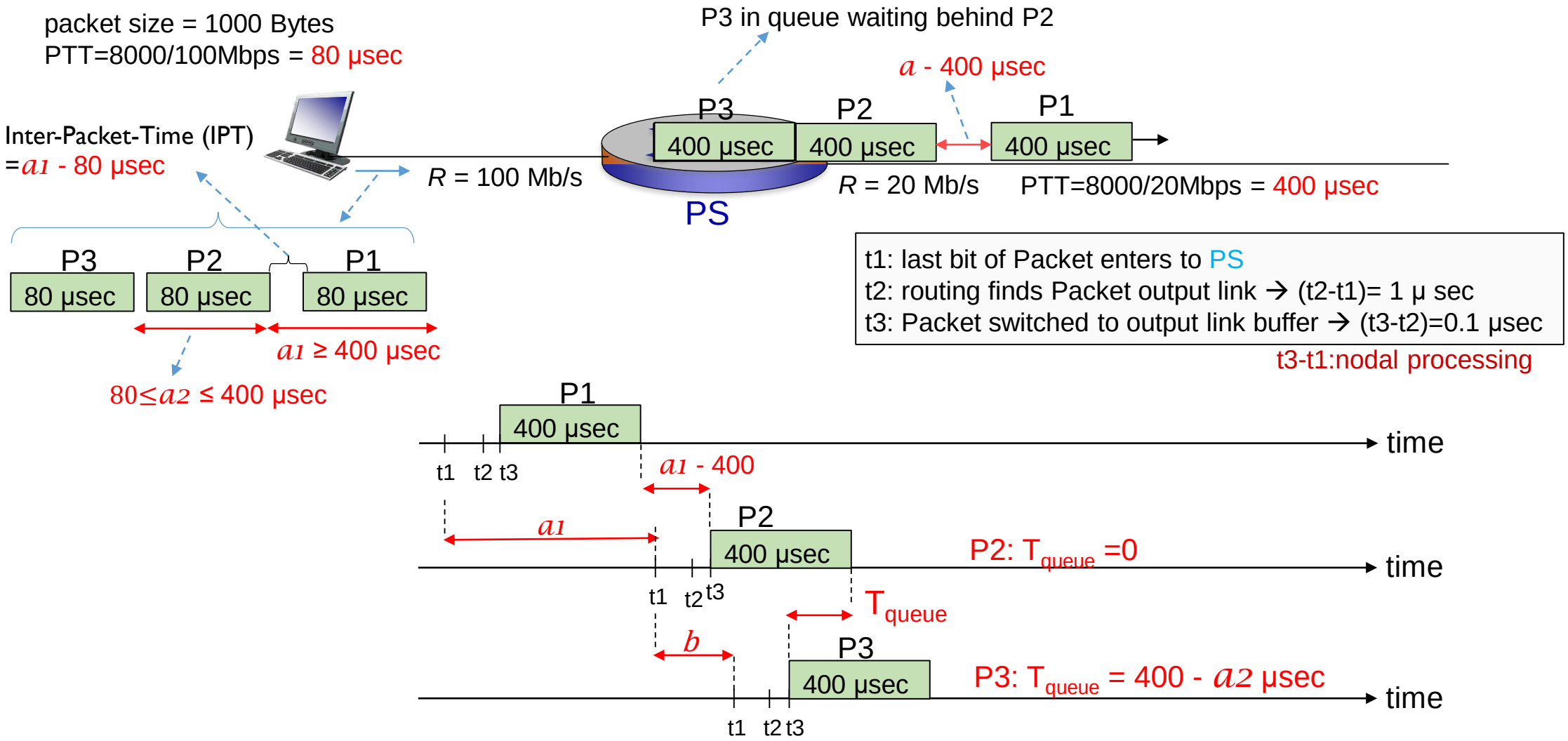
Average Packet Queueing Delay

- R: link bandwidth (bps)
- L: packet length (bits)
- a: average packet arrival rate
- $\frac{La}{R} \cong 0$: Very small average queueing delay
- $\frac{La}{R} \cong 1$: Very large average queueing delay
- $\frac{La}{R} > 1$: more "packet" arriving is more than can be send



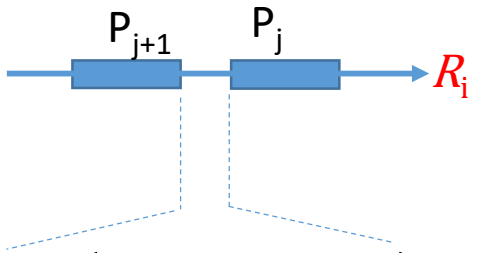


Temporal Packet Queuing Delay at a router

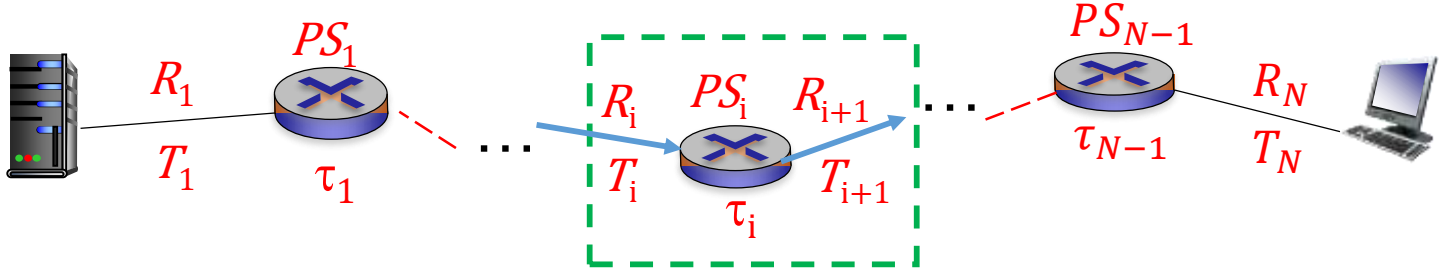


Temporal Packet Queuing Delay at a router

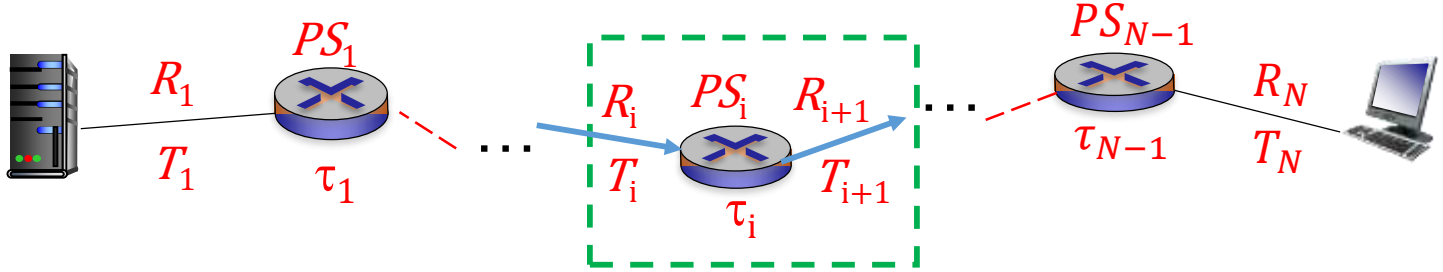
- Connection: $N-1$ packet switches and N link between Source and Destination
- L [bits]: file size
- m: packet numbers, packet size= L/m bits
- R_i [bps] : bandwidth
- QD_{P1} [sec]: queuing delay of 1_{th} packet
- QD_{P2} [sec]: queuing delay of 2_{nd} packet
- …
- QD_{Pk} [sec]: queuing delay of k_{th} packet



$IPT\ (P_{j+1};\ P_j:\ Link_i)$: Inter-Packet-Time
(time between packets j+1 and j on Link i



Temporal Packet Queuing Delay at a router



If $R_{i+1} < R_i$:

$QD_{p1} = 0$

If $R_{i+1} > R_i$: *No queuing delay*

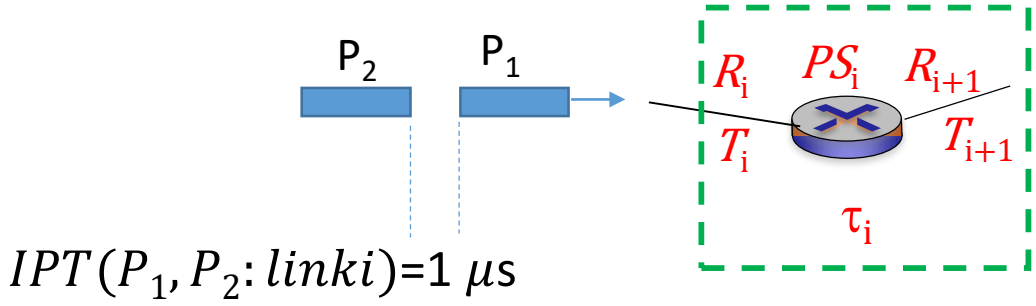
$$QD_{p2} = \frac{PS}{R_{i+1}} - \frac{PS}{R_i} - IPT(P_2;\ P_1:\ Link_i)$$

$$QD_{p3} = 2\left(\frac{PS}{R_{i+1}} - \frac{PS}{R_i}\right) - [IPT(P_2;\ P_1:\ Link_i) + IPT(P_3;\ P_2:\ Link_i)]$$

$$QD_{pk} = (k-1)\left(\frac{PS}{R_{i+1}} - \frac{PS}{R_i}\right) - \sum_{j=1}^{k-1} IPT(P_{j+1};\ P_j:\ Link_i)$$

Example

- R_i = 2Gbps,R_{i+1}= 1Gbps, PS=8000bits, PS/R_i=4μs, PS/R_{i+1}=8μs



$$QD_{P2} = 8 - 4 - 1 = 3\mu s$$

Packet Loss

- Queue capacity is finite: a packet can arrive to find a full queue, router will **drop** that packet; that is, packet will be **lost**
- Fraction of lost packets increases as traffic intensity increases
- Performance at a router:
  1. delay ($d_{proc} + d_{queue} + d_{trans}$)
  2. probability of packet loss
- A lost packet may be retransmitted by packet source to ensure that it eventually will be delivered to destination

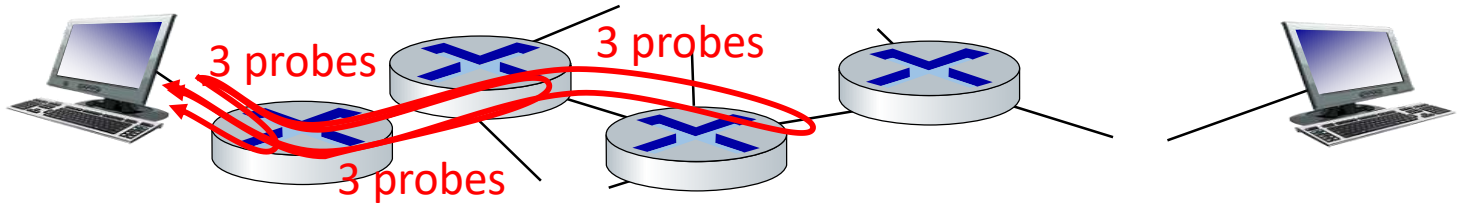
1.4.3 End-to-End Delay

- End-to-End Delay: total delay from source to destination
- Suppose: $N-1$ routers between source host and destination host

$$e-e-d = (d_{trans} + d_{prop})_{source\ host} + \sum_{m=1}^{N-1} (d_{proc} + d_{queue} + d_{trans} + d_{prop})_m$$

Traceroute (RFC 1393)

- Traceroute is a simple program that can run in any Internet host
- Traceroute program sends sends multiple, special packets (IP-ICMP option packet, Chapter 5) toward a given destination
- When a router receives one of these special packets, it sends back to source a short message that contains name and address of router
- Traceroute program: provides round trip time (RTT) measurement from source to routers along end-end path towards destination



Traceroute Example

name of router (IP Add. of router) RTT1 RTT2 RTT3

- Source Host: gaia.cs.umass.edu (at University of Massachusetts), Dest. Host: 134.157.254.10 (a host in computer science department at University of Sorbonne in Paris)

1. **gw-vlan-2451.cs.umass.edu (128.119.245.1) 1.899 ms 3.266 ms 3.280 ms**
2. j-cs-gw-int-10-240.cs.umass.edu (10.119.240.254) 1.296 ms 1.276 ms 1.245 ms
3. n5-rt-1-1-xe-2-1-0.gw.umass.edu (128.119.3.33) 2.237 ms 2.217 ms 2.187 ms
4. core1-rt-et-5-2-0.gw.umass.edu (128.119.0.9) 0.351 ms 0.392 ms 0.380 ms
5. border1-rt-et-5-0-0.gw.umass.edu (192.80.83.102) 0.345 ms 0.345 ms 0.344 ms
6. nox300gw1-umass-re.nox.org (192.5.89.101) 3.260 ms 0.416 ms 3.127 ms
7. nox300gw1-umass-re.nox.org (192.5.89.101) 3.165 ms 7.326 ms 7.311 ms
8. 198.71.45.237 (198.71.45.237) 77.826 ms 77.246 ms 77.744 ms
9. renater-lb1-gw.mx1.par.fr.geant.net (62.40.124.70) 79.357 ms 77.729 79.152 ms
10. 193.51.180.109 (193.51.180.109) 78.379 ms 79.936 80.042 ms
11. * 193.51.180.109 (193.51.180.109) 80.640 ms *
12. * 195.221.127.182 (195.221.127.182) 78.408 ms *
13. 195.221.127.182 (195.221.127.182) 80.686 ms 80.796 ms 78.434 ms
14. r-upmc1.reseau.jussieu.fr (134.157.254.10) 78.399 ms * 81.353 ms

transatlantic fiber-optic link between routers 7 and 8

no response due to packet loss in network

router dose not have a name

Varying queuing delay

- Queuing delay is varying with time
  - Round-trip delay of packet n sent to a router n can sometimes be longer than round-trip delay of packet $n + 1$ sent to router $n + 1$
  - Delay to Router 12 is smaller than the delay to Router 11
- Also note the big increase in the round-trip delay when going from router 7 to router 8.

- Graphical interface software for Traceroute: http://www.pingplotter.com

End System, Application, and Other Delays

- In addition to processing, transmission, and propagation delays, there can be additional significant delays in end systems
- Example: a packet in a shared medium (e.g., as in a WiFi or cable modem scenario) may purposefully delayed as part of shared medium protocol (Chapter 6)
- Another important delay is media packetization delay, which is present in Voiceover-IP (VoIP) applications
  - In VoIP, sending side must first fill a packet with encoded digitized speech before passing packet to Internet
  - This time to fill a packet, called the packetization delay, can be significant and can impact the user perceived quality of a VoIP call

1.4.4 Throughput in Computer Networks

- In addition to delay and packet loss, another critical **performance measure** in computer networks is end-to-end network throughput
- Network throughput: consider transferring a large file from Host A to Host B across a computer network
- Instantaneous network throughput: at any instant of time is rate (in bits/sec) at which Host B is receiving the file
- Average network throughput: If Host B takes T sec to receive an F bits file, then average throughput is F/T bits/sec
- For some applications, such as Internet telephony, it is desirable to have a low delay and an instantaneous throughput consistently above some threshold (24 kbps for audio connection or 256 kbps for real-time video connection)

Simple Ideal scenario

- **Q:** a. What is average end-end throughput? **A:** $X \leq \min(R_s, R_c)$
- **Q:** b. What is average end-end throughput?
- **A:** $X \leq \min(R_1, R_2, \ldots, R_n)$
- bottleneck link:
  $\min(R_1, R_2, \ldots, R_n)$

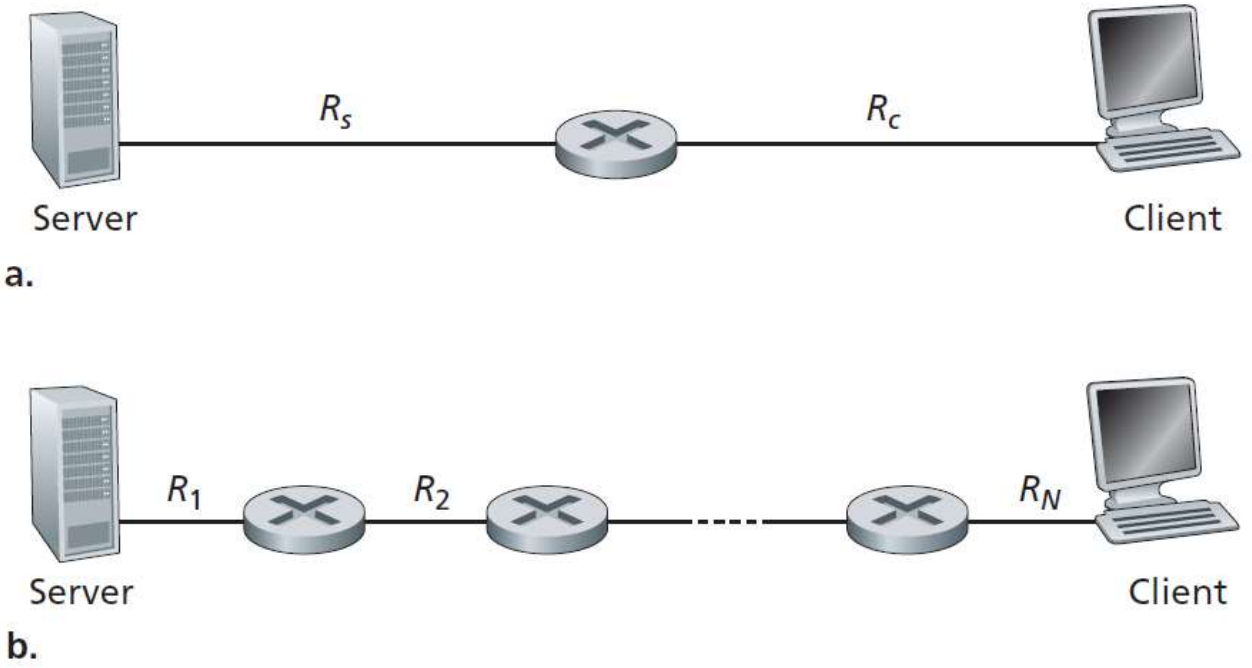


Figure 1.19 Throughput for a file transfer from server to client

Simple Ideal scenario

- **Q:** a. What is average end-end throughput? **A:** $X \leq \min(R_s, R_c)$
  - Constraining factor for throughput in today's Internet is typically **access network**
- Suppose R_s=200Mbps, R_c=100Mbps, R=500Mbps
- **Q:** b. What is average end-end throughput? **A:** $X \leq \min\left(R_s, R_c, \frac{R}{10}\right) = 50Mbps$

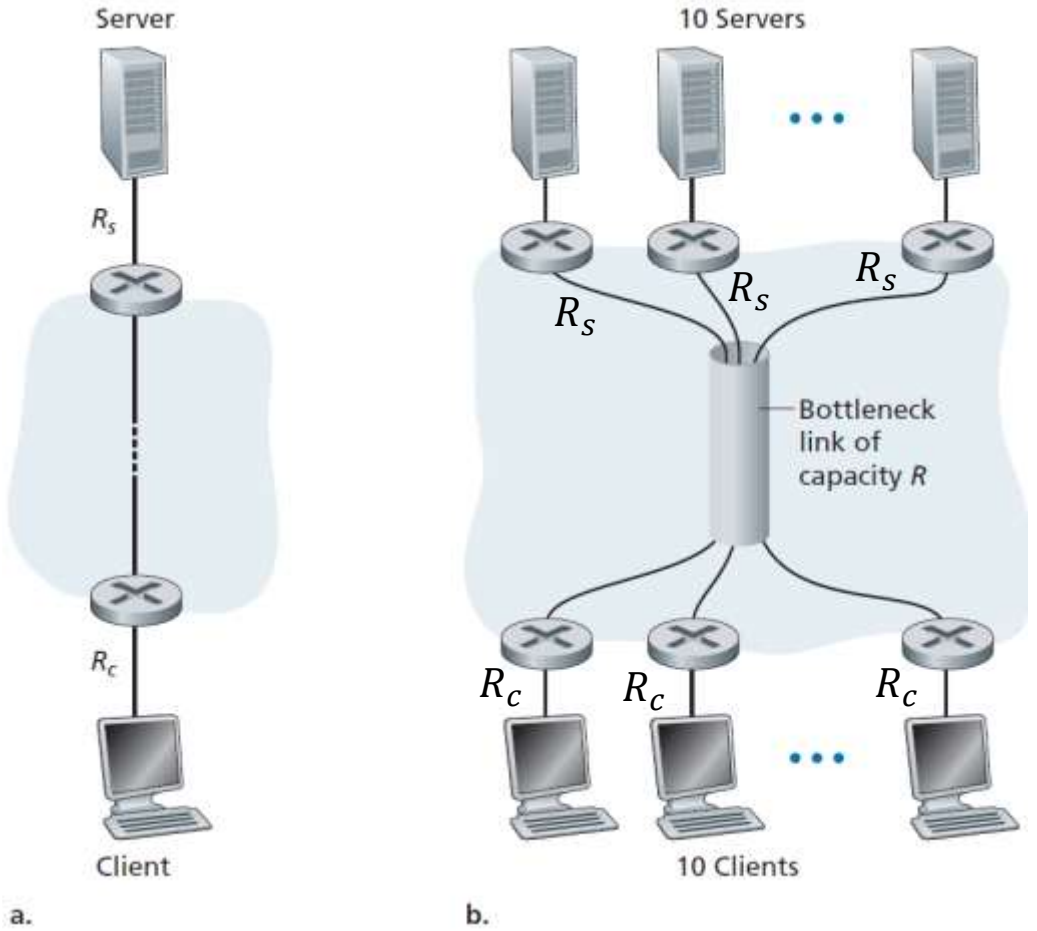
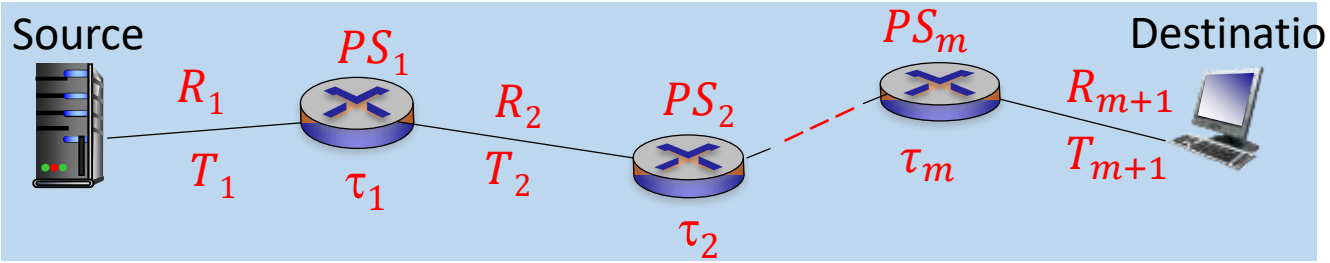


Figure 1.20 End-to-end throughput:
(a) Client downloads a file from server
(b) (b) 10 clients downloading with 10 servers

Real scenario

- Connection: m packet switches and $m+1$ link between Source and Destination
- L [bits]: file size
- n: packet numbers, packet size= L/n bits
- R_i [bps] : bandwidth
- T_i [sec]: link propagation delay
- τ_I [sec]: nodal processing and switching time, queuing delay



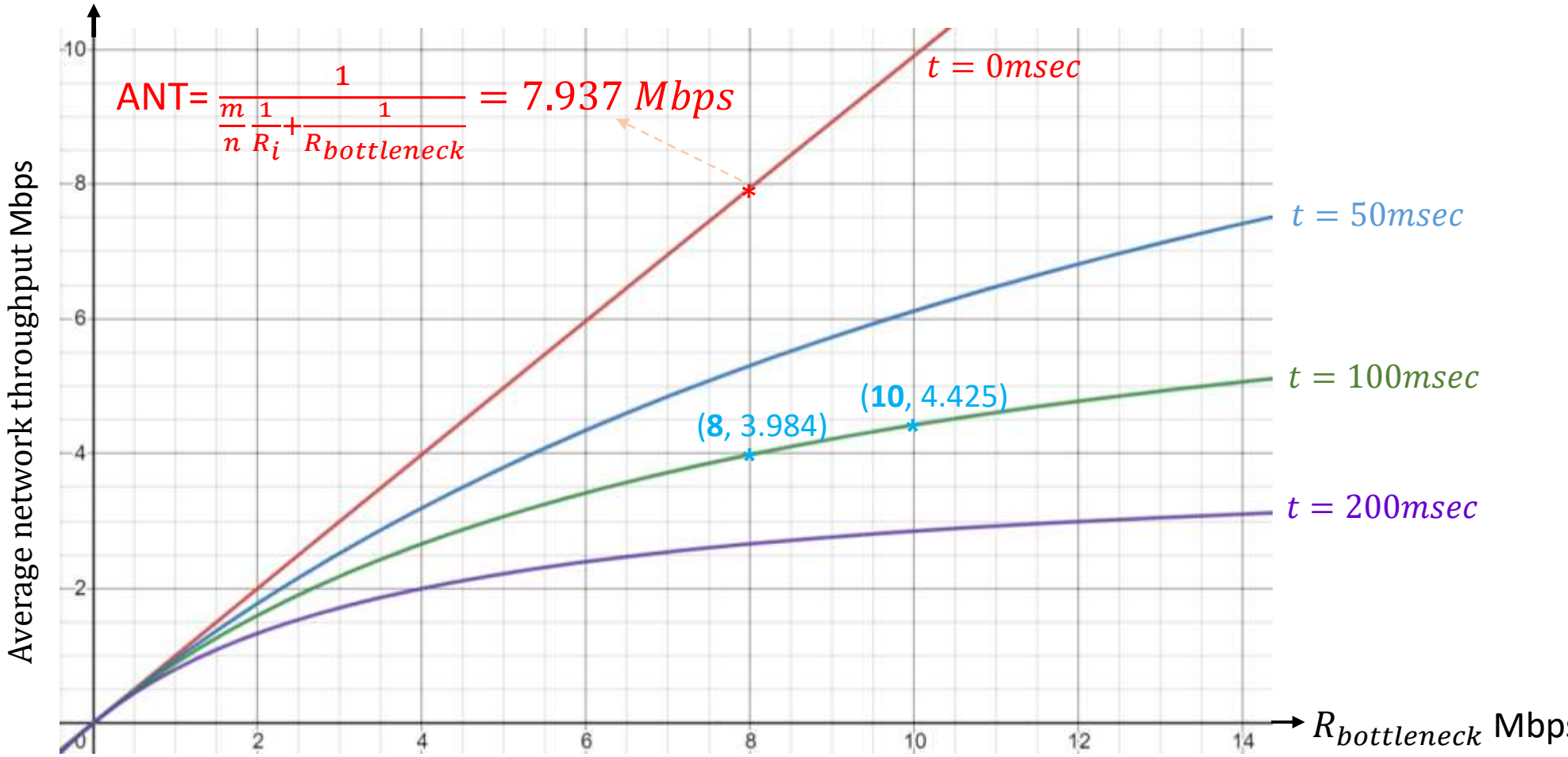
$$\text{End to End delay=e-e-d=} \sum_{i=1}^{m+1} T_i + \sum_{i=1}^{m} \tau_i + \sum_{R_i \neq bottleneck}^{m+1} \frac{L/n}{R_i} + \frac{L}{R_{bottleneck}}$$

$$\text{ANT=Average Network Throughput} = \text{Effective Bandwidth} = \frac{L}{e-e-d}[bps] < R_{bottleneck}$$

$$R_{bottleneck} = \min(R_1, R_2, \dots, R_{m+1})$$

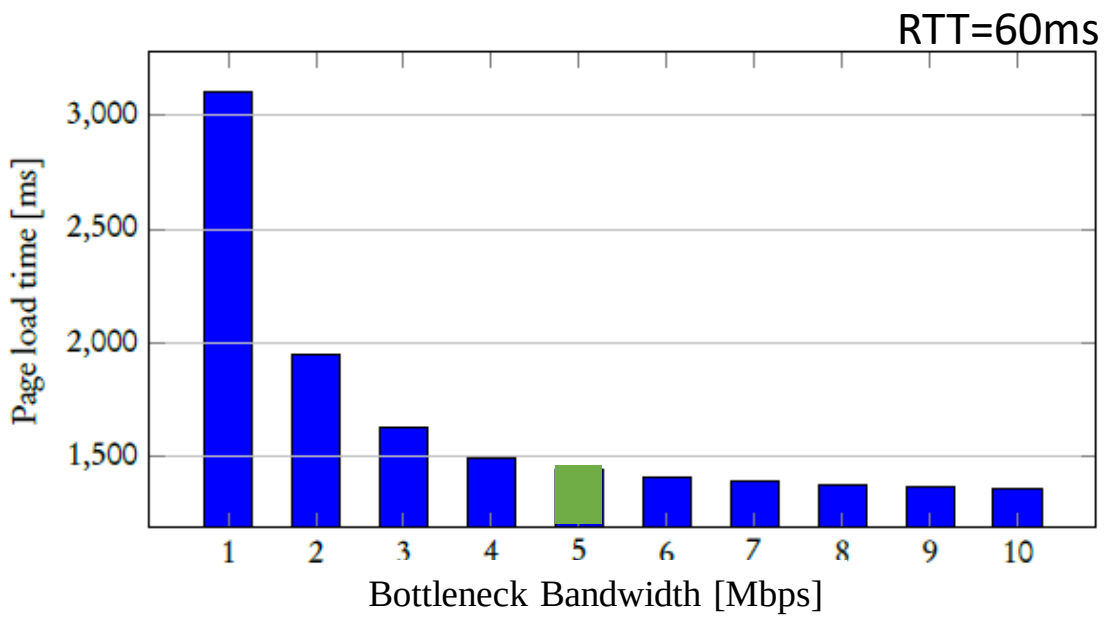
Example

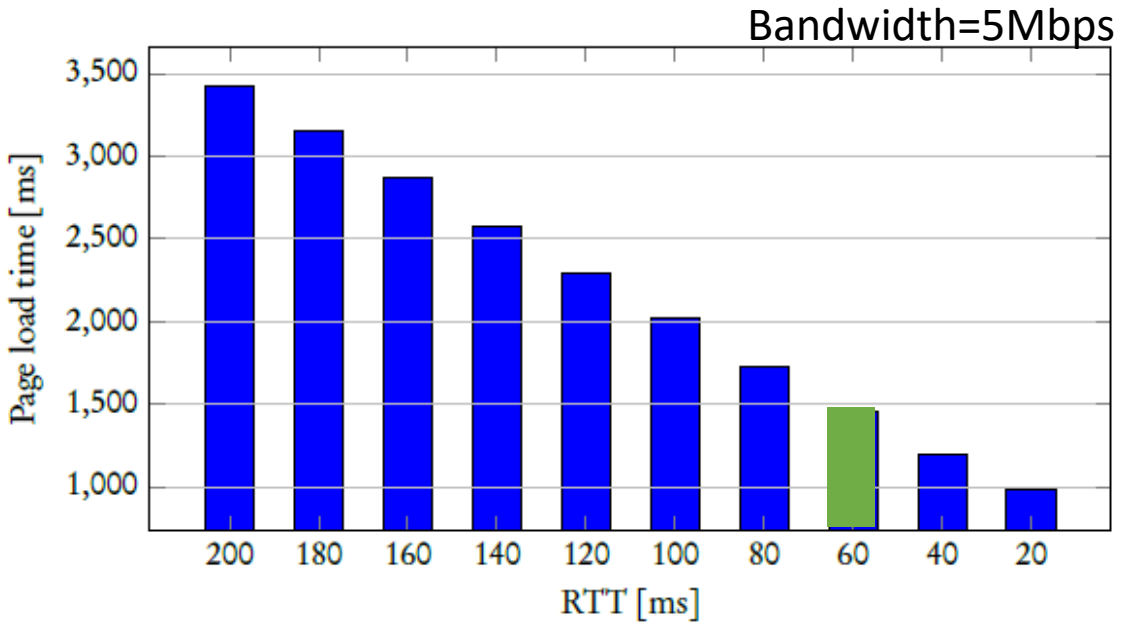
$L = 100{,}000$ Bytes, $n = 100$ packets, $m = 10$ routers, $t = \sum_{i=1}^{m+1} T_i + \sum_{i=1}^{m} \tau_i$ and $R_i = 100Mbps$



Latency (RTT) vs Bandwidth

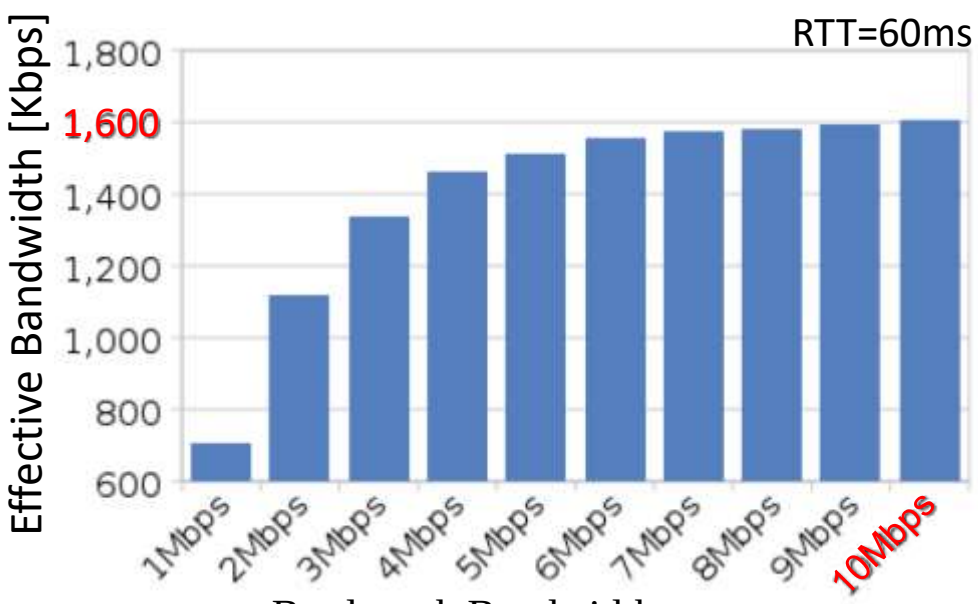
- **Bandwidth dose not matter (much)**
- Latency is bounded by speed of light
- Improving bandwidth is much easier than latency

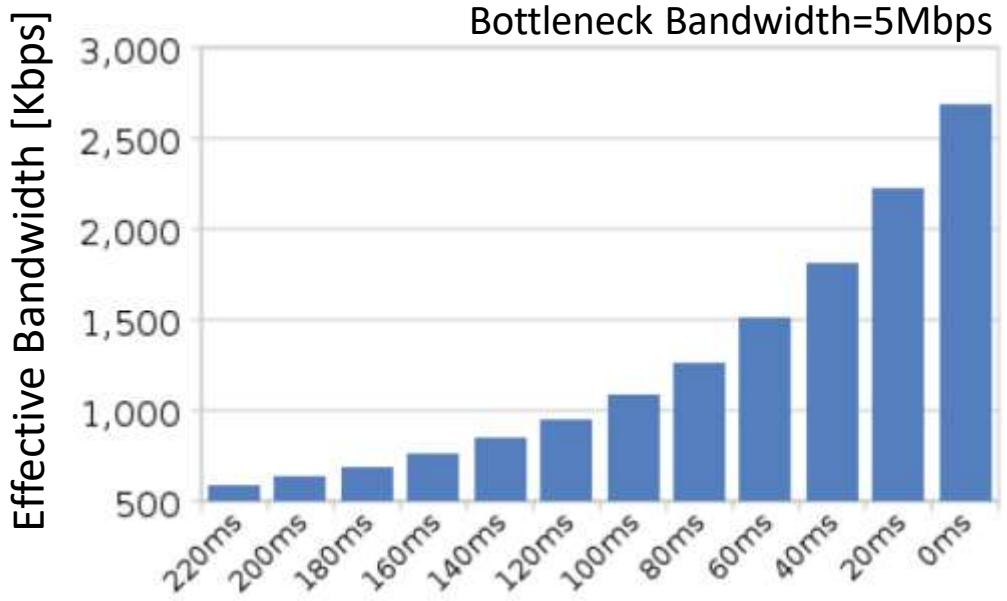




Latency (RTT) vs Bandwidth

- At 10Mbps, web pages can only be downloaded at 1,600Kbps/10Mbps = 0.16% of bottleneck bandwidth
- With RTT=220ms, effective bandwidth=550Kbps, which is a little more than 10% of 5Mbps
- With low RTT, web page downloads still only achieve ~54% of bottleneck bandwidth. This is due to other factors of how web pages are loaded





Average Effective Bandwidth

Akamai's state of the Internet, Q1 2017 Report

- Average Effective Bandwidth (IPv4):
  - Global: 7.2Mbps
  - Iran: 4.7Mbps
  - South Korea (Top) 28.6Mbps
  - USA: 18.7Mbps

Contents

1.1 What Is the Internet?
1.2 The Network Edge
1.3 The Network Core
1.3’ The Name and Addresses
1.4 Delay, Loss, and Throughput in Packet-Switched Networks
1.5 Protocol Layers and Their Service Models
1.6 Networks Under Attack
1.7 History of Computer Networking and the Internet
1.8 Summary
Appendix

1.5 Protocol Layers and Their Service Models

- Internet is an extremely complicated system
- There are many pieces to Internet:
  - numerous APPs and protocols
  - various types of end systems and hosts
  - packet switches
  - various types of link-level media

A key question:

- How to organize a complex network architecture?

1.5.1 Layered Architecture

- A human analogy - airline system
- Airline system has **ticketing agents**, **baggage checkers**, **gate personnel**, **pilots**, **airplanes**, **air traffic control**, and a worldwide system for **routing** airplanes

Series of actions:

- Traveler (user): purchases ticket, checks bags, goes to gate, and reaches to plane seat
- Plane: takes off and is routed to its destination, and landed
- Traveler (user): deplanes at gate and claim bags
- If trip was bad, traveler complains about flight to ticket agent

Figure 1.21 Taking an airplane trip: actions

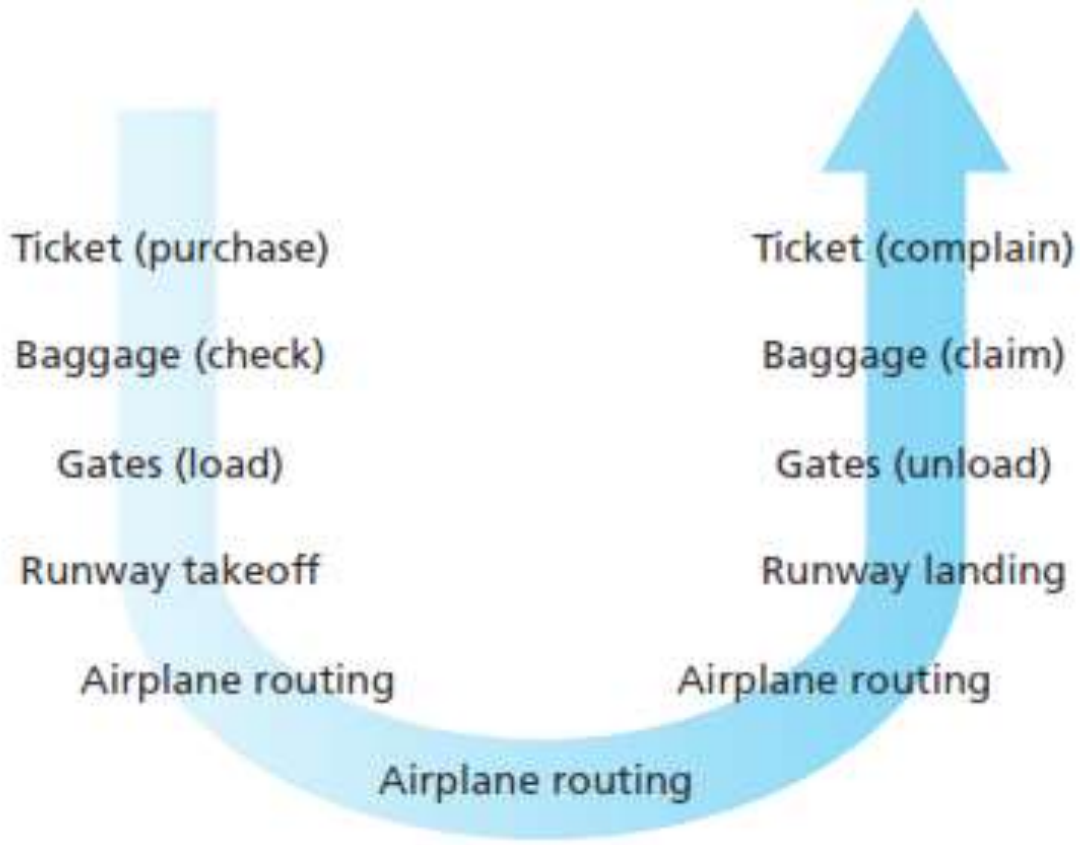
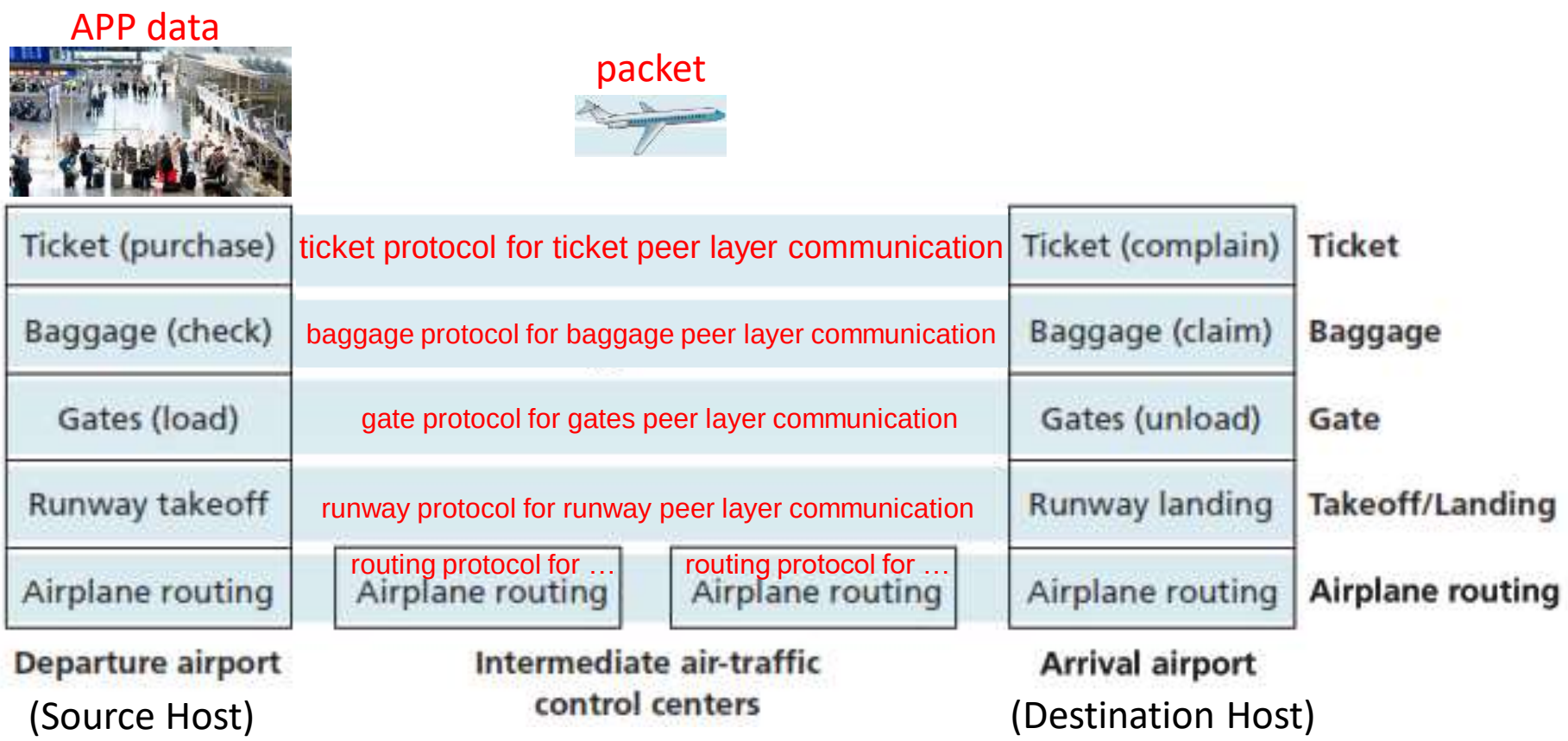


Figure 1.22 Horizontal layering of airline functionality

- Each layer implements a service

Example:

- Takeoff/Landing Layer provides: "runway-to-runway passenger transfer service"



Layers provide service(s)

- Each layer provides its service by

1. Performing certain actions within that layer (for example, at gate layer, loading and unloading people from an airplane)
2. Using services of layer directly below it (for example, in gate layer, using "runway-to-runway passenger transfer service" of takeoff/landing layer)

- Each layer, combined with layers below it, implements some functionality, some service

Why layering

- Layered architecture (explicit structure) allows identification, relationship of complex system's pieces
- Layering (modularization) eases **maintenance** and **updating** **of system**
- change in layer's service implementation is transparent to rest of system
  - As long as layer provides same service to layer above it, and uses same services from layer below it, remainder of system remains unchanged when a layer's implementation is changed
  - For large and complex systems that are constantly being updated, ability to change implementation of a service without affecting other components of system is an important advantage
  - e.g., change in ticket procedure doesn't affect rest of system
- Layered reference model for discussion

Protocol Layering

- Network designers organize protocols, and network hardware and software that implement protocols, in **layers**
- Each protocol belongs to one of layers, just as each function in the airline architecture in Figure 1.22 belonged to a layer
- We are interested in **services** that a layer offers to layer above (so-called **service model** of a layer)

Each layer provides its service by

1. Performing certain actions within that layer
2. Using services of layer directly below it

- For example, services provided by layer n may include reliable delivery of messages from one host to other host. This might be implemented by using an unreliable host-to-host message delivery service of layer $n-1$, and adding layer n functionality to detect and retransmit lost messages

Layering drawbacks

- Layer may duplicate lower-layer functionality. For example, many protocol stacks provide error recovery on both a per-link basis and an end-to-end basis
- A second potential drawback is that functionality at one layer may need information (for example, a timestamp value) that is present only in another layer; this violates the goal of separation of layers

Internet Layered Architecture: Protocol stack

- **Protocol stack:** Taking together protocol layers and their services
- Internet protocol stack consists of five layers:
  - application layers
  - transport
  - network
  - link
  - physical

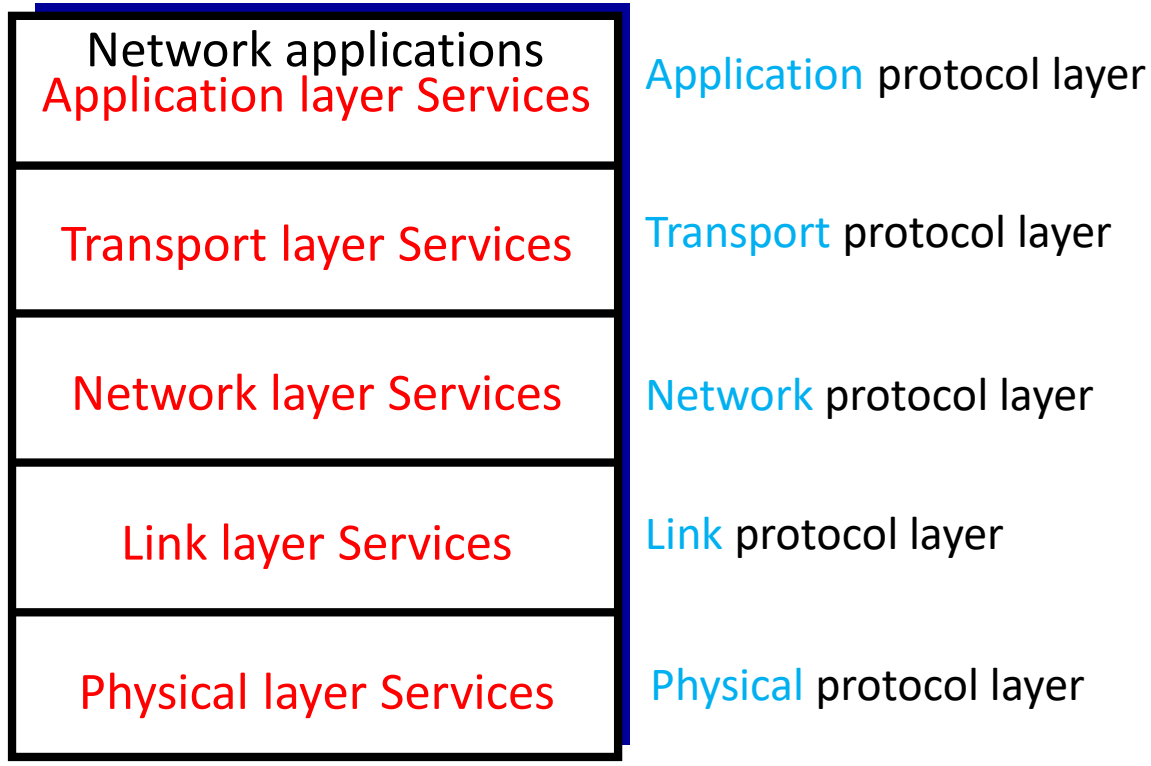
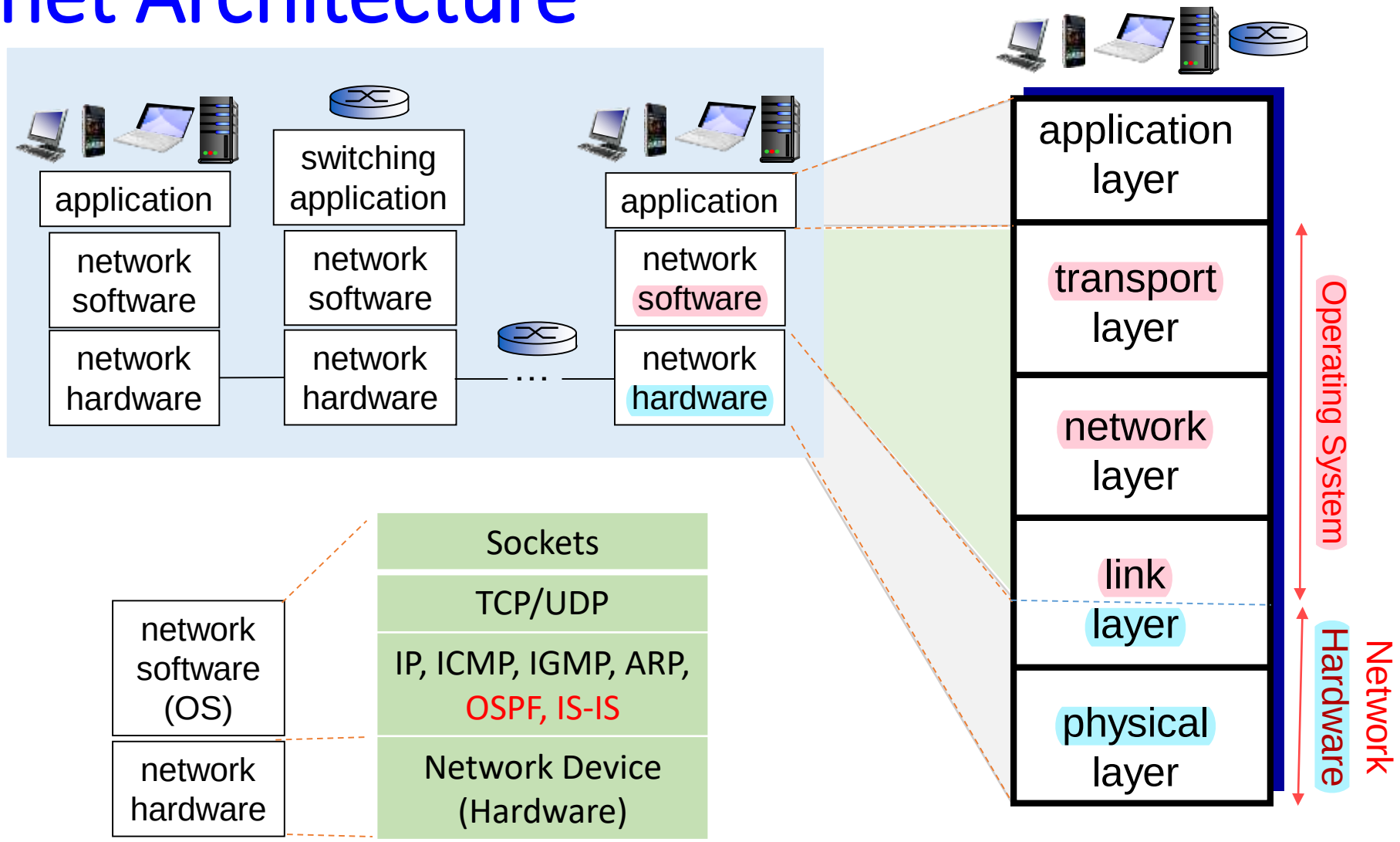
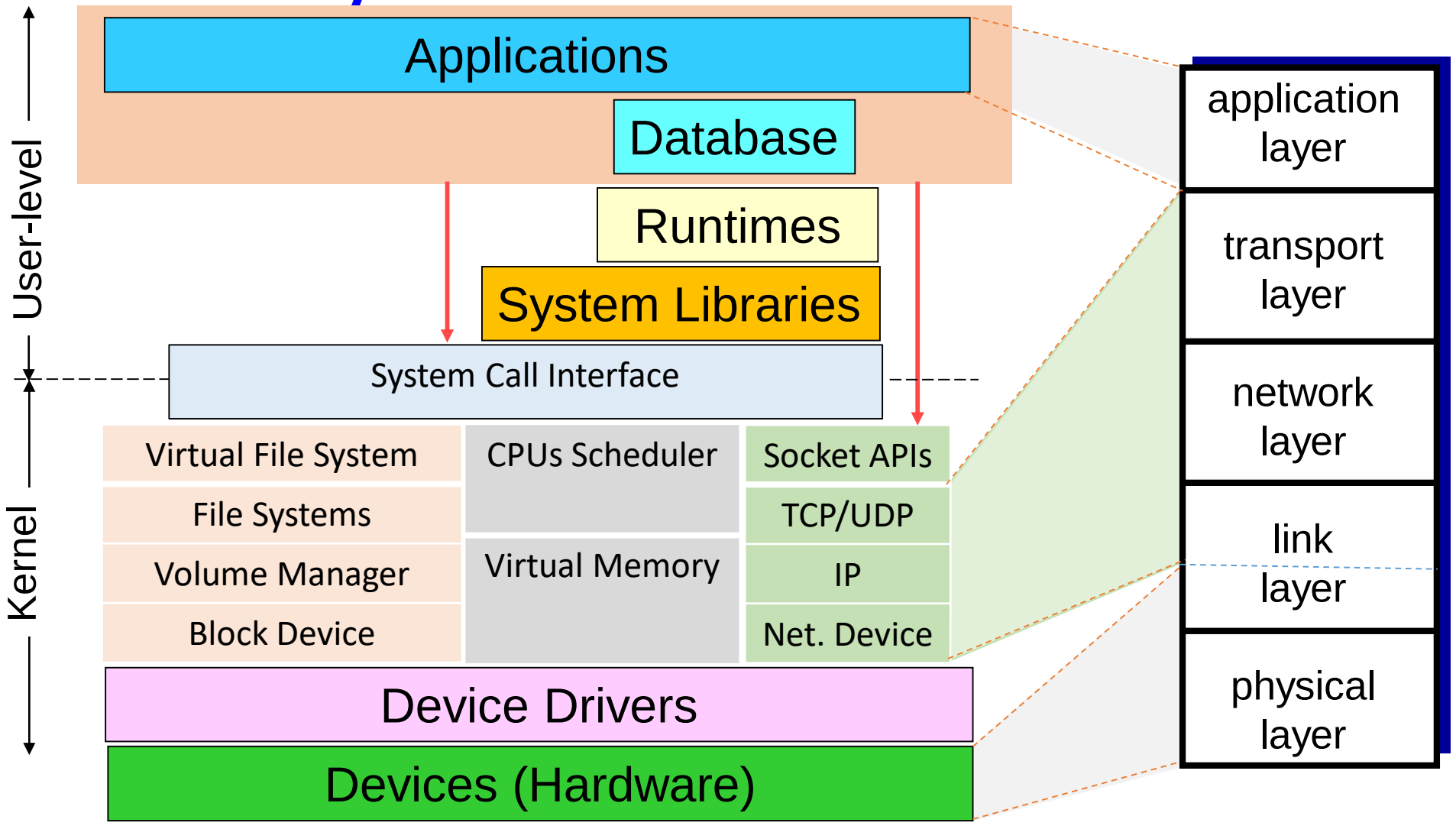


Figure 1.23 The Internet protocol stack

Internet Architecture



Generic System Software Stack on a Server



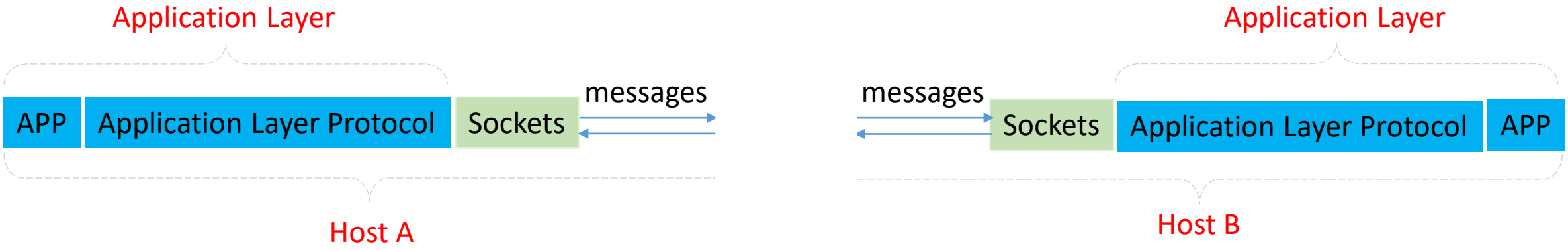
Protocol layer implementation

- A protocol layer can be implemented in software, in hardware, or in a combination of the two
- Application-layer protocols, such as HTTP and SMTP, are almost always implemented in software in end systems; so are transport-layer protocols
- Network layer is often a mixed implementation of hardware and software
- Because **physical layer** and **data link layers** are responsible for handling communication over a specific link, they are typically implemented in a **network interface card** (for example, Ethernet or WiFi interface cards) associated with a given link
- A layer protocol distributed among **end systems**, **packet switches**, and **other components**
- There's often a piece of a layer protocol in each of these network components

Application Layer

- Application layer is where network APPs and their application-layer protocols reside
- Internet's application-layer protocols: HTTP protocol (which provides for Web document request and transfer), SMTP (which provides for the transfer of e-mail messages), and FTP (which provides for the transfer of files between two end systems), DNS (name address translation), QUIC, …
- An application-layer protocol is **distributed over multiple end systems**, with application in one host using protocol to exchange packets of information with application in another host
- We'll refer to this packet of information at application layer as a **message**

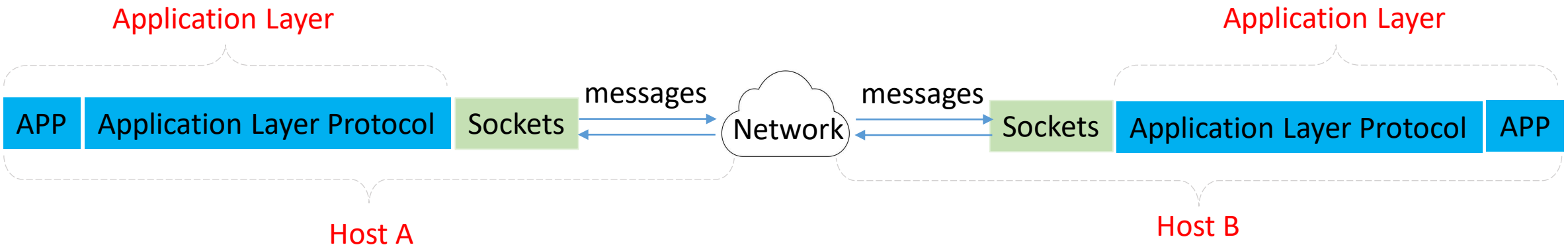
What APP sees



Transport Layer

- Internet's transport layer transports application-layer messages between application endpoints
- Internet's transport protocols: TCP and UDP
- TCP provides a **connection-oriented service** to its applications
  - TCP service includes: **guaranteed delivery of application-layer messages to destination**, flow control (that is, sender/receiver speed matching) and **congestion-control mechanism**, so that a source throttles its transmission rate when network is congested
- UDP protocol provides a connectionless service to its applications
  - This is a no-frills service that provides no reliability, no flow control, and no congestion control.
- We'll refer to a transport-layer packet as a **segment**
- TCP and UDP breaks long messages into shorter segments

What Transport Layer sees



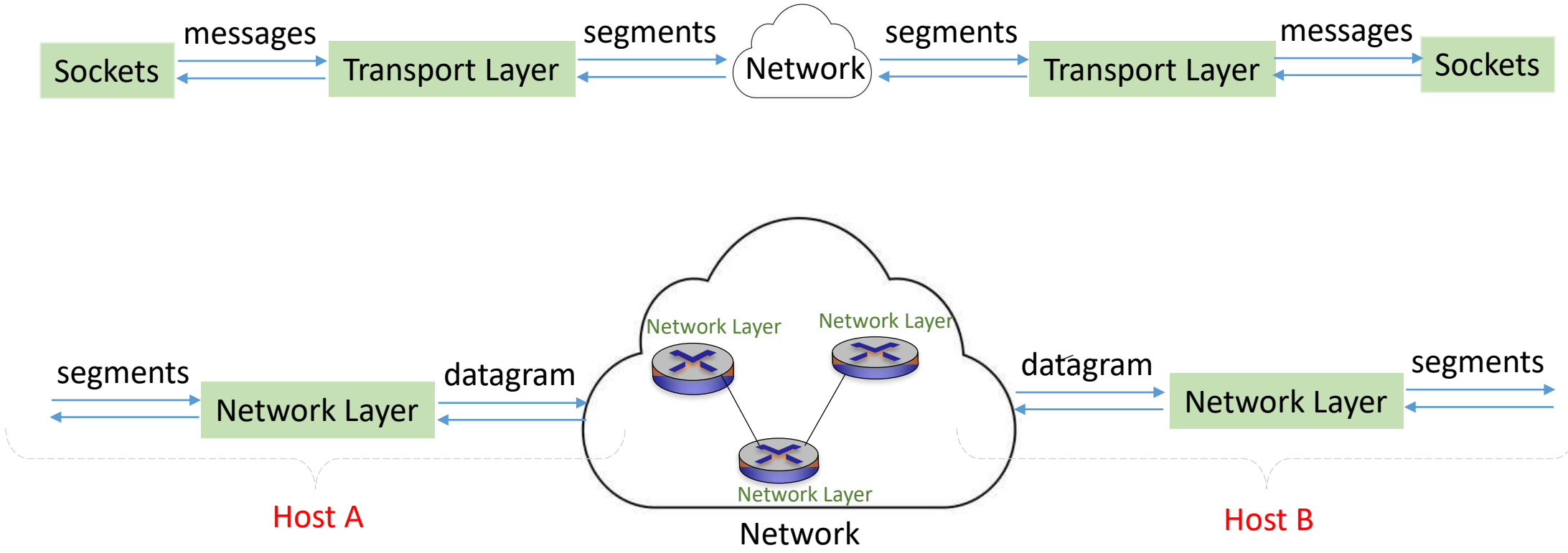
Network Layer

- Transport-layer protocol (TCP or UDP) in a source host passes a **segment** and **a destination address** to network layer in the source host, just as you would give postal service a letter with a destination address
- Network layer put its own protocol information in a header to segment (network-layer packet=datagram)
- Internet's network layer is responsible for delivering network-layer packets known as **datagrams from source host to first router**, **from first router to next router**, …, **from last router to destination host**
- Network layer provides service of delivering segment to transport layer in destination host

Network Layer: *Routing* Protocol and *Routing* Algorithm

- Network layer includes **IP protocol**, which defines header fields in datagram as well as how end systems and routers act on these fields
- **There is only one IP protocol**, and all Internet components that have a network layer must run IP protocol
- Network layer also contains **routing protocols** that determine routes that datagrams take between sources and destinations
- There are many routing protocols
- Although network layer contains both IP protocol and numerous routing protocols, it is often simply referred to as **IP layer**

What Network Layer sees



Link Layer

- Network layer routes a datagram through a series of routers between source and destination
- To move a packet from one node (host or router) to next node in route, network layer relies on services of link layer
- At each node, network layer passes datagram to link layer, which delivers datagram to next node along route
- At this next node, link layer passes datagram up to network layer
- **Services provided by link layer** depend on specific link-layer protocol that is employed over link
- For example, some link-layer protocols provide **reliable delivery**, from transmitting node, over one link, to receiving node

Link layer

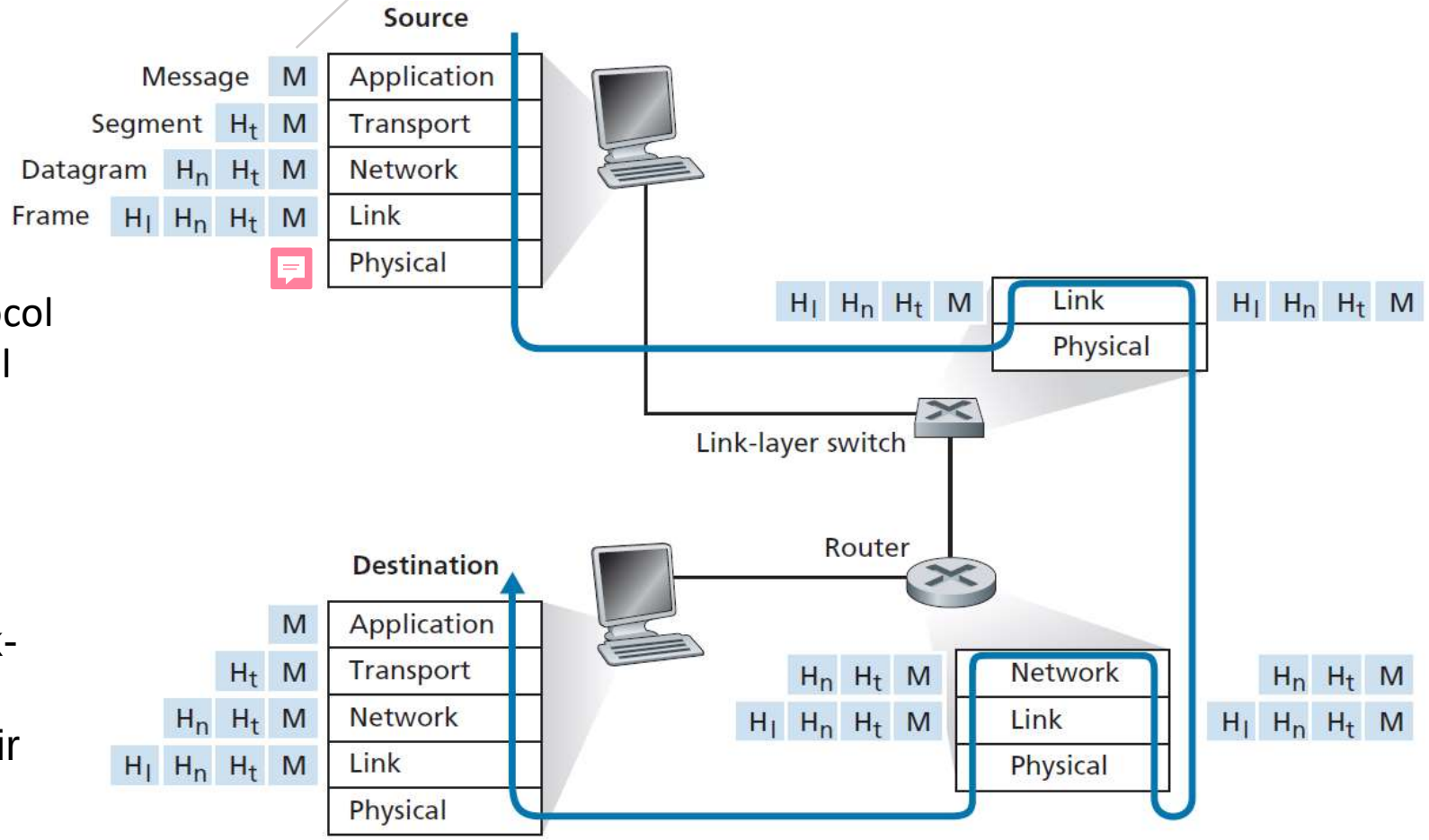
- Note that TCP provides reliable delivery from one end system to another
- Examples of link-layer protocols include **Ethernet**, **WiFi**, and **cable access network's DOCSIS protocol**
- As datagrams typically need to traverse several links to travel from source to destination, a datagram may be handled by different link-layer protocols at different links along its route
- For example, a datagram may be handled by Ethernet on one link and by PPP on the next link
- We refer to the link-layer packets as **frames**

Physical Layer

- Link layer moves entire frames from one network element to an adjacent network element
- Physical layer moves individual bits within frame from one node to the next
- Protocols in this layer are again link dependent and further depend on actual transmission medium of link (twisted-pair copper wire, single-mode fiber optics)
- **Ethernet has many physical-layer protocols**: one for twisted-pair copper wire, another for coaxial cable, another for fiber, and so on
  - In each case, a bit is moved across the link in a different way

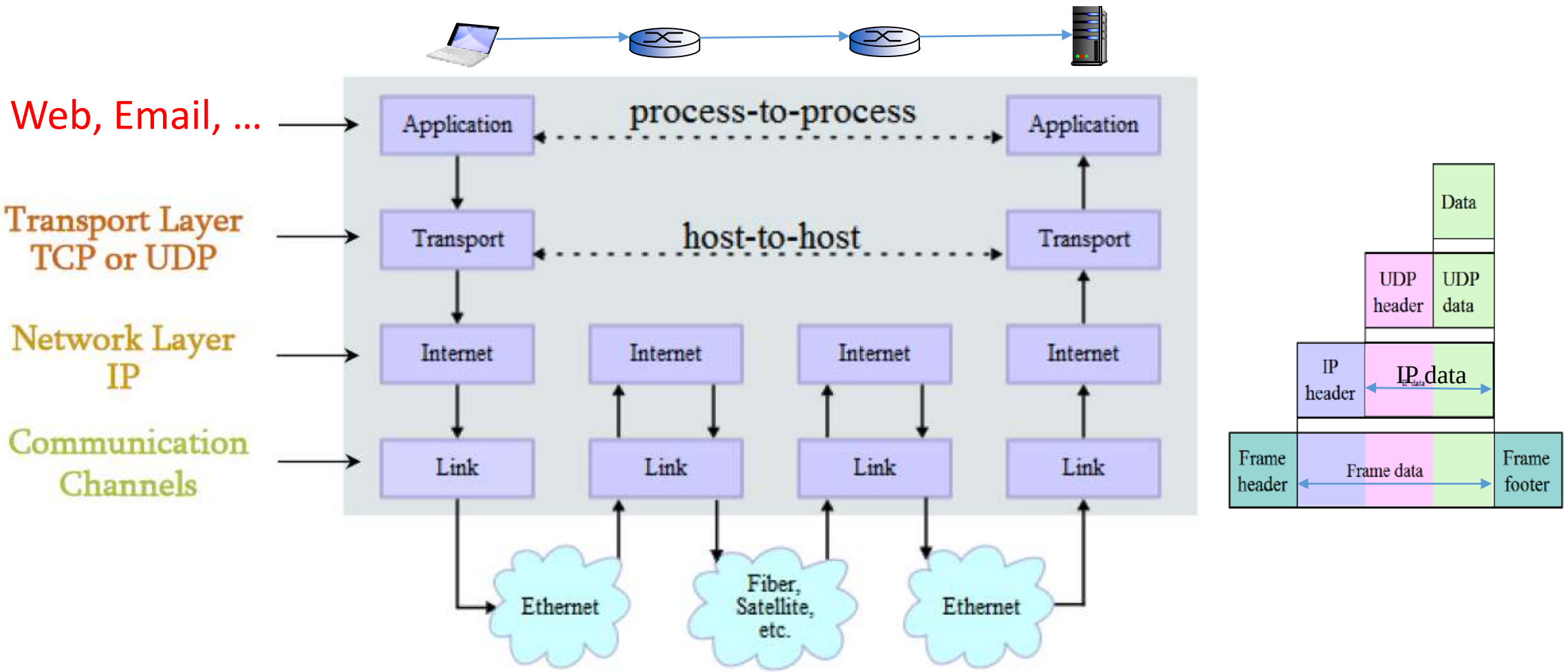
1.5.2 Encapsulation

M: Ha Data or File or Object or nothing



Ha = Header of application layer protocol
H_t = Header of transport layer protocol
H_n = Header of network layer protocol
H_l = Header of link layer protocol

Figure 1.24 Hosts, routers, and link-layer switches; each contains a different set of layers, reflecting their differences in functionality


Web, Email, …
Transport Layer
TCP or UDP
Network Layer
IP
Communication
Channels
Application
process-to-process
Application
Transport
host-to-host
Transport
Internet
Internet
Internet
Internet
Link
Link
Link
Link
Ethernet
Fiber,
Satellite,
etc.
Ethernet
Data
UDP
header
UDP
data
IP
header
IP data
Frame
header
Frame data
Frame
footer

Other protocol stacks

- Netware (Novell Corporation)
- AppleTalk (Apple Computer Inc.)
- DECNET (Digital Equipment Inc.)
- SNA, SNA/IP (IBM)
- TCP/IP (The Internet)

Contents

1.1 What Is the Internet?
1.2 The Network Edge
1.3 The Network Core
1.3’ The Name and Addresses
1.4 Delay, Loss, and Throughput in Packet-Switched Networks
1.5 Protocol Layers and Their Service Models
1.6 Networks Under Attack
1.7 History of Computer Networking and the Internet
1.8 Summary
Appendix

1.6 Networks Under Attack

- Field of network security is about how bad guys can attack computers and computer networks and about how we can defend against those attacks, or better yet, **design new architectures** that are **immune** to such attacks in first place
- **Network security is a central topic in field of computer networking**
- We begin here by surveying some of today's more prevalent security related problems

Malware into Your Host Via Internet

- Malicious softwares (known as **malware)** that can also enter and infect our APPs, data, devices, …
- **Malware** can delete our files, install spyware that collects our private information, passwords, and keystrokes, and then sends this (over Internet, of course) back to bad guys
- Compromised host may also be enrolled in a network of thousands of similarly compromised devices, collectively known as a **botnet**, which bad guys control and leverage for **spam e-mail distribution** or **distributed denial-of-service attacks** against targeted hosts
- Much of **malwares** are **self-replicating**: once it infects one host, from that host it seeks entry into other hosts over Internet, and can spread exponentially fast

Attacking Servers and Network Infrastructure

- A class of security threats are known as **denial-of-service (DoS) attacks**
- A **DoS** attack makes a network, host, or other piece of infrastructure unusable by legitimate users
- Web servers, e-mail servers, DNS servers, and institutional networks can all be subject to DoS attacks
- Most Internet DoS attacks fall into one of three categories: Vulnerability attack, Connection flooding and Bandwidth flooding

Three DoS attacks

- Vulnerability attack: Sending a few **well-crafted messages to a vulnerable application** or **operating system** running on a targeted host. If right sequence of packets is sent to a vulnerable application or operating system, **service can stop** or, worse, **host can crash**
- Connection flooding: Attacker establishes a large number of TCP connections at target host. Host can become so occupied with these bogus connections that it stops accepting legitimate connections
- Bandwidth flooding: Attacker sends a host too many packets to targeted host so that target's access link becomes clogged, preventing legitimate packets from reaching server (see next slide)

Bandwidth-flooding attack

- Recalling our delay and loss analysis discussion in Section 1.4.2, it's evident that if server has an bandwidth of R bps, then attacker will need to send traffic at a rate of approximately R bps to cause damage
- If R is very large, a single attack source may not be able to generate enough traffic to harm server

- Detecting and blocking DoS attack:
  - Upstream router may be able to **detect attack and block** all traffic, block all traffic emanates from a single source, before traffic gets near server

Distributed DoS (DDoS) attack

- In a **DDoS** attack attacker controls multiple sources (botnet) and has each source blast traffic at target
- Aggregate traffic rate across all controlled sources needs to be approximately R to cripple the service
- DDoS attacks with **thousands of comprised hosts** are a common occurrence today
- DDoS attacks are **much harder to detect and defend** against than a DoS attack from a single host

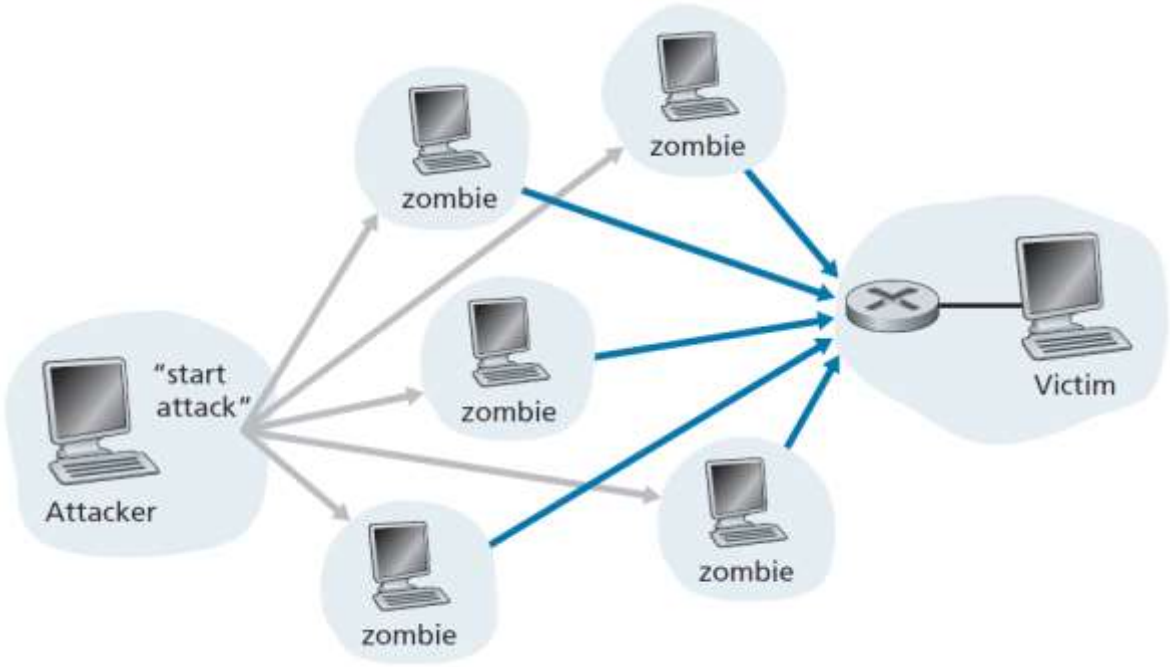


Figure 1.25 A distributed denial-of-service attack

Packet sniffing

- Many users today access Internet via **wireless devices**, such as WiFi-connected laptops or handheld devices with cellular Internet connections
- It creates a **major security vulnerability** by placing a **passive receiver** in vicinity of wireless transmitter, that receiver can obtain a copy of every packet that is transmitted
- These packets can contain all kinds of **sensitive information**, including passwords, social security numbers, trade secrets, and private personal messages
- A passive receiver that records a copy of every packet that flies by is called a **packet sniffer**

Packet sniffing

- Sniffers can be deployed in wired environments as well
- In wired broadcast environments, as in many Ethernet LANs, a packet sniffer can obtain copies of broadcast packets sent over the LAN
- Cable access technologies (section 1.2) also broadcast packets and are thus vulnerable to sniffing
- Bad guy who gains access to an institution's access router or access link to Internet may be able to plant a sniffer that makes a copy of every packet going to/from organization
- Sniffed packets can then be analyzed offline for sensitive information

IP spoofing

- It is surprisingly easy to create a packet with an arbitrary source address, and packet content and then transmit into Internet
- Imagine a node (say an Internet router) who receives such a packet, takes source address as being **truthful**, and then performs some **command embedded in packet's contents** (say modifies its forwarding table)
- Ability to inject packets into Internet with a false source address is known as **IP spoofing**, and is one of many ways in which one user/node can pretend as another user/node

Contents

1.1 What Is the Internet?
1.2 The Network Edge
1.3 The Network Core
1.3’ The Name and Addresses
1.4 Delay, Loss, and Throughput in Packet-Switched Networks
1.5 Protocol Layers and Their Service Models
1.6 Networks Under Attack
1.7 History of Computer Networking and the Internet
1.8 Summary
Appendix

1.7 History of Computer Networking and the Internet: 1.7.1 1961-1972: Early packet-switching principles

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

1.7.2 Proprietary Networks and Internetworking: 1972–1980

- 1970: ALOHAnet wireless 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 routing
- decentralized control

define today's Internet architecture

1.7.3 A Proliferation of Networks: 1980–1990

- 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

1.7.4 The Internet Explosion: The 1990s, 2000s

- Early 1990s: 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 1990s: commercialization of the Web
- Late 1990s – 2000s:
- more killer apps: instant messaging, **P2P file sharing** (pioneered by Napster)
- network security
- 50 million host, 100 million+ users
- backbone links running at Gbps

1.7.5 The New Millennium

- In first two decades of 21st century, perhaps no other technology has **transformed society** more than **Internet** along with Internet-connected **smartphones**
- Advances are being made on all fronts, including deployments of **faster routers** and **higher transmission speeds** in both access networks and in network backbones
- But following developments merit special attention:
  - Deployment of **broadband Internet access to homes**, not only cable modems and DSL but also **fiber to the home**, and now **5G fixed wireless**
  - High-speed Internet access has promoted **video applications**, including distribution of user-generated video (**YouTube**, **Aparat**, …), on-demand streaming of movies and television shows (**Netflix**, **Filimo**, **Namava** …), and **multiperson video conference** (**Skype**, **Facetime**, and Google Hangouts)

1.7.5 The New Millennium

- High-speed wireless Internet access is enabling new location-specific applications such as **Balad**, **Yelp**, **Tinder**, and **Waze**
  - Number of wireless devices connecting to Internet surpassed the number of wired devices in 2011
- Online social networks, such as **Facebook**, **Instagram**, **Twitter**, and **WeChat**, have created massive people networks on top of Internet
- **Mobile payments**
- Service providers, such as Google and Microsoft, have deployed their own **private backbone**, and as a result, they provide **search results** and **e-mail access** almost instantaneously, as if their data centers were running within one's own computer

1.7.5 The New Millennium

- Many **Internet commerce companies** are now running their applications in “cloud”, such as in Amazon’s EC2, in Microsoft’s Azure, or in Alibaba Cloud
- Many companies and universities have also migrated their Internet applications (e.g., e-mail and Web hosting) to cloud
- Cloud companies not only provide **APPs scalable computing and storage environments**, but also provide **APPs implicit access** to their high-performance private networks

Contents

1.1 What Is the Internet?
1.2 The Network Edge
1.3 The Network Core
1.3’ The Name and Addresses
1.4 Delay, Loss, and Throughput in Packet-Switched Networks
1.5 Protocol Layers and Their Service Models
1.6 Networks Under Attack
1.7 History of Computer Networking and the Internet
1.8 Summary
Appendix

1.8 Summary

- We started at edge of network, looking at end systems and applications, and at **transport service provided to applications running on end systems**
- We also looked at link-layer technologies and physical media typically found in the access network
- We then looked at network core, identifying packet switching and circuit switching as two basic approaches for **transporting data through a telecommunication network**, and we examined the strengths and weaknesses of each approach
- We also examined structure of global Internet, learning that Internet is a network of networks

1.8 Summary

- We saw that Internet's hierarchical structure, consisting of higher- and lower-tier ISPs, has allowed it to scale to include thousands of networks
- We introduced causes of delay, throughput and packet loss in a packet-switched network
- We developed simple quantitative models for **transmission**, **propagation**, and **queuing** delays as well as for **throughput**
- Next we examined protocol layering and service models
- We also surveyed some of more prevalent **security attacks** in Internet
- We finished Chapter 1 with a brief history of computer networking

1.8 Summary

- Chapter 1 in itself constitutes a mini-course in computer networking
- We have covered a tremendous amount of ground in this first chapter
- If you're a bit overwhelmed, don't worry. In following chapters, we'll revisit all of these ideas, covering them in much more detail
- At this point, we hope you leave this chapter with a still-developing intuition for pieces that make up a network

Contents

1.1 What Is the Internet?
1.2 The Network Edge
1.3 The Network Core
1.3’ The Name and Addresses
1.4 Delay, Loss, and Throughput in Packet-Switched Networks
1.5 Protocol Layers and Their Service Models
1.6 Networks Under Attack
1.7 History of Computer Networking and the Internet
1.8 Summary
Appendix

DATA CENTERS AND CLOUD COMPUTING

- Data centers are engines behind Internet applications that we use on a daily basis
- Internet companies such as **Google**, **Microsoft**, **Amazon**, and **Alibaba** have built massive data centers
- Each data center housing tens to **hundreds of thousands of server hosts**
- Server hosts in data centers, called blades, are generally commodity hosts that include CPU, memory, and disk storage
- Server hosts are stacked in racks, with each rack typically having 20 to 40 blades
- Racks are then interconnected using sophisticated and evolving data center network designs (Chapter6)
- Hosts serve **content (e.g., Web pages and videos)**, **store e-mails and documents**, and collectively **perform massively distributed computations**

CLOUD COMPUTING

- A major trend in computing is for companies to use a cloud provider such as Amazon to handle essentially all of their IT needs
- For example, Airbnb and many other Internet-based companies do not own and manage their own data centers but instead run their entire Web-based services in Amazon cloud, called Amazon Web Services (AWS)
- Worker bees in a data center are server hosts. They serve content (e.g., Web pages and videos), store e-mails and documents, and collectively perform massively distributed computations
- As an example, Amazon's data centers serve three purpose:
    1. They serve Amazon e-commerce pages to describe products and purchase information to users (customers)
    2. They serve as massively parallel computing infrastructures for Amazon-specific data processing tasks
    3. They provide cloud computing to other companies and individual users