CS 305: Computer Networks

Fall 2022

Lecture 2：Introduction

Ming Tang

Department of Computer Science and Engineering
Southern University of Science and Technology (SUSTech)

Chapter 1: roadmap

1.1 what is the Internet?

1.2 network edge

- end systems, access networks, links

1.3 network core

- packet switching, circuit switching, network structure

1.4 delay, loss, throughput in networks

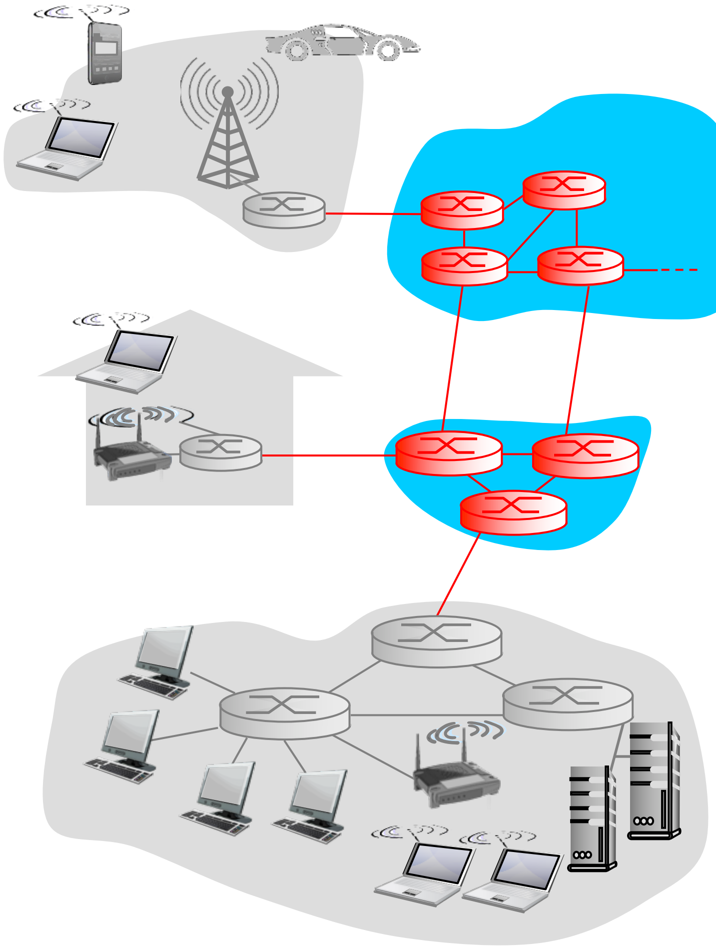
1.5 protocol layers, service models

1.6 networks under attack: security

1.7 history

The network core

The mesh of **packet switches and links** that interconnects the Internet's end systems.



Packet switching:

- Internet
- not reserved (on demand); may wait

Circuit switching:

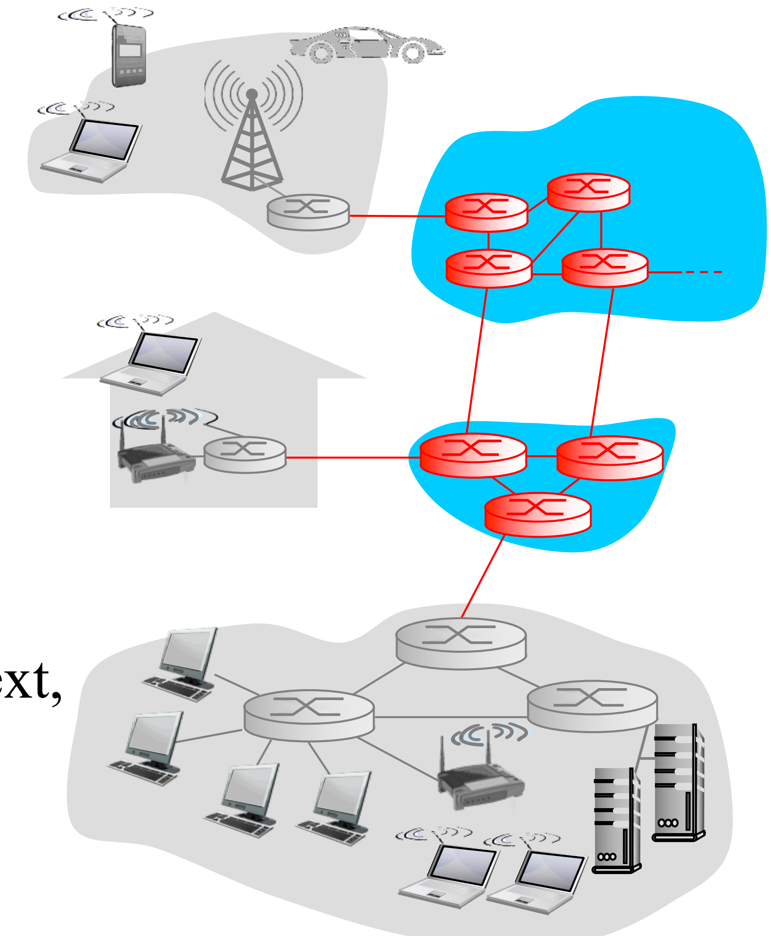
- telephone
- reserved

Packet-switching

Packet switching:

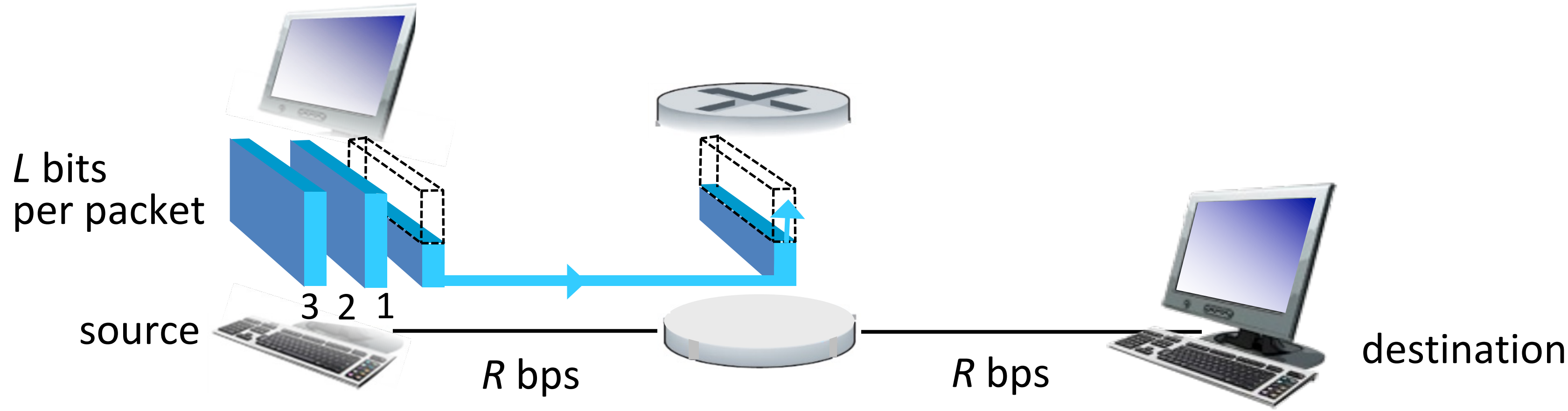
Hosts break long messages into packets; each packet is forwarded independently

- store-and-forward
- each packet is transmitted at full link capacity
- not reserved → queueing delay and packet loss



Forward packets from one router to the next, across links on path from source to destination → routing and forwarding

Packet-switching: store-and-forward



- *Store and forward:* entire packet must arrive at router before it can be transmitted on next link
- Takes L/R seconds to transmit (push out) L-bit packet into link at R bps

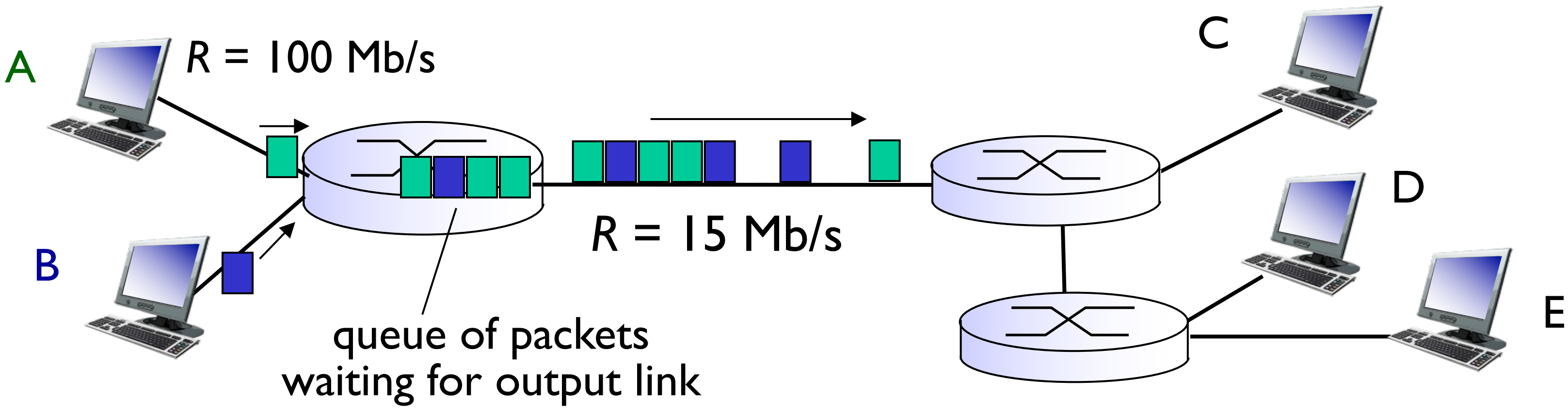
Example:

more on delay shortly …

- $L = 7.5$ Mbits; $R = 1.5$ Mbps
- One-hop transmission delay = $L/R = 5$ sec
- End-end delay = $2L/R$ (assuming zero propagation delay)

How about one packet of length L sending over a path of N links, each of rate R?

Packet switching: delay and packet loss



Output buffer stores packets that the router is about to send into that link.

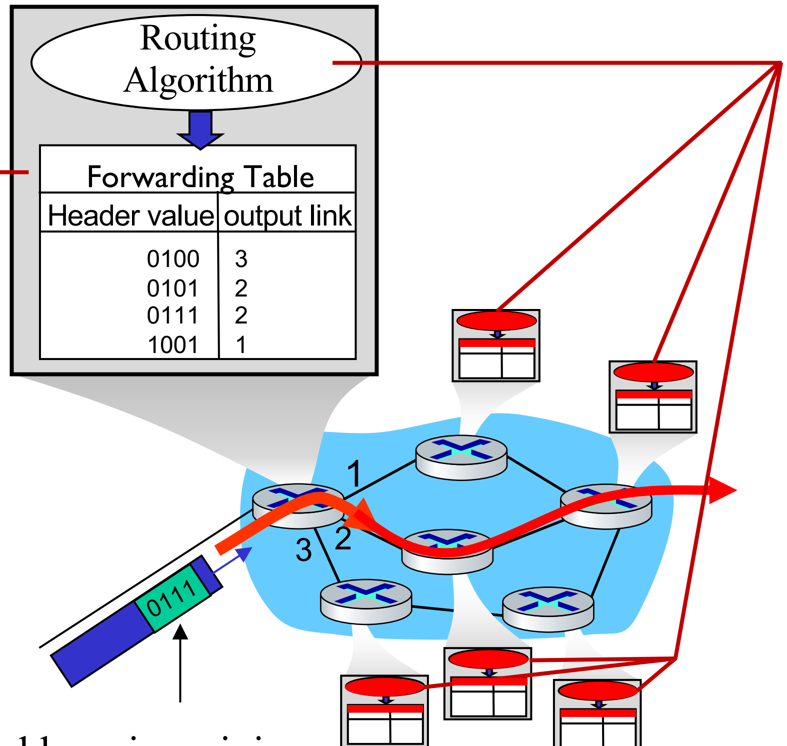
If arrival rate (in bits) to link exceeds transmission rate of link for a period of time:

- packets will queue, wait to be transmitted on link → queuing delay
- packets can be dropped (lost) if memory (buffer) fills up → packet loss

Packet switching: forwarding and routing

Forwarding:

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



destination address in arriving packet's header

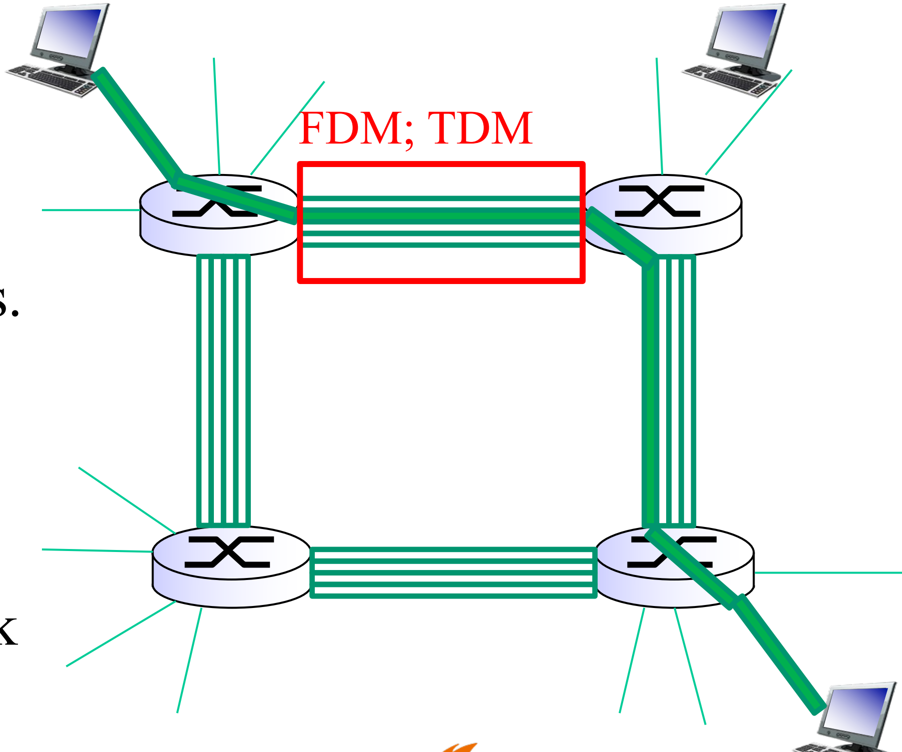
Routing:

- global action: determine source-destination paths taken by packets
- routing algorithms

Circuit switching

Circuit switching: end-to-end resources <u>reserved</u> for “call” between source and destination (traditional telephone networks)

- Reserved (dedicated resources): buffer, link
- A fraction of each link’s capacity
- **Advantage:** Guaranteed constant rate
- **Limitation:** Circuit idle（空闲）if not used by call *(no sharing)*



In diagram, each link has four circuits.

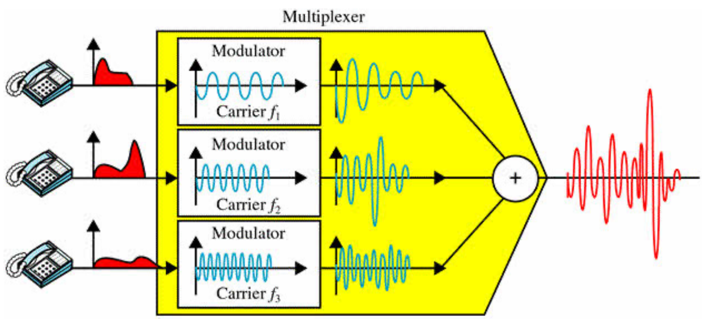
- The call gets 2nd circuit in top link and 1st circuit in right link.

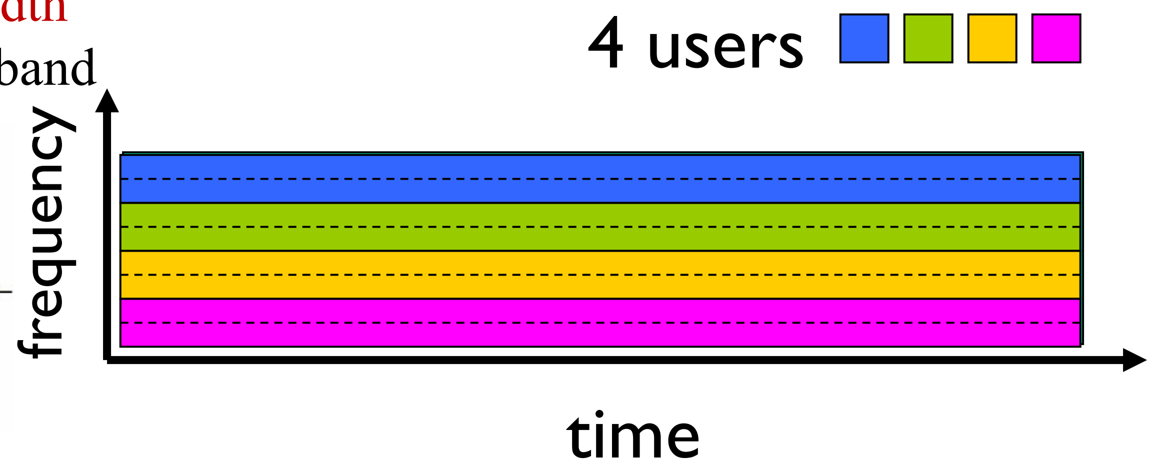
How could we split each physical link to multiple virtual links?

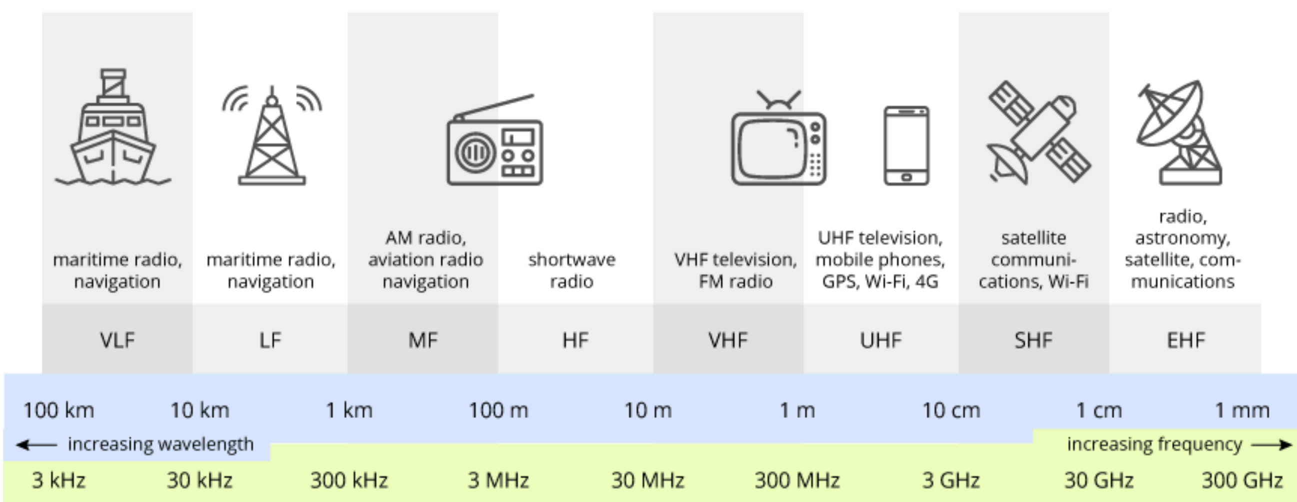
Circuit switching: FDM versus TDM

Frequency Division Multiplexing (多路复用) (FDM)

- Optical, electromagnetic frequencies divided into (narrow) frequency bands → bandwidth
- At max rate of that narrow band



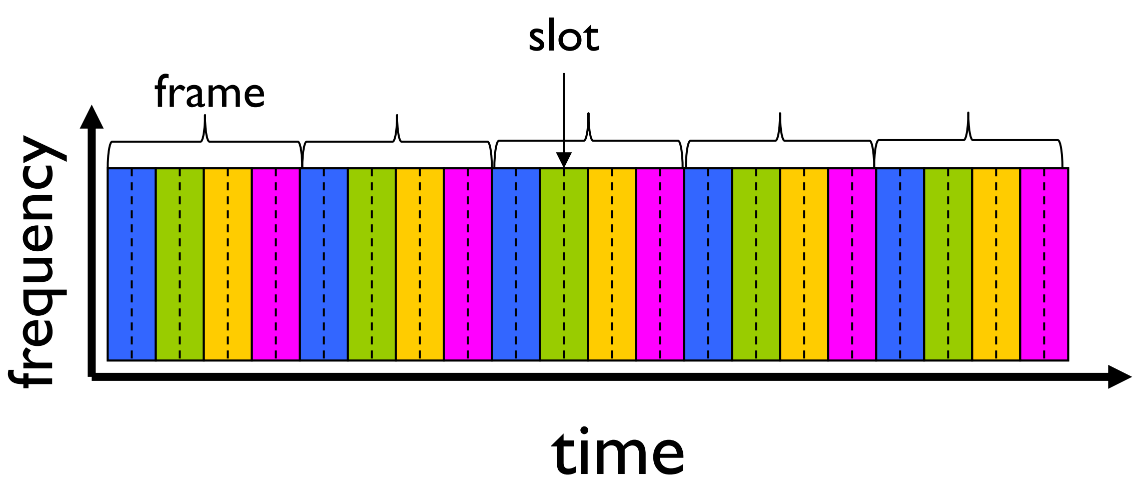




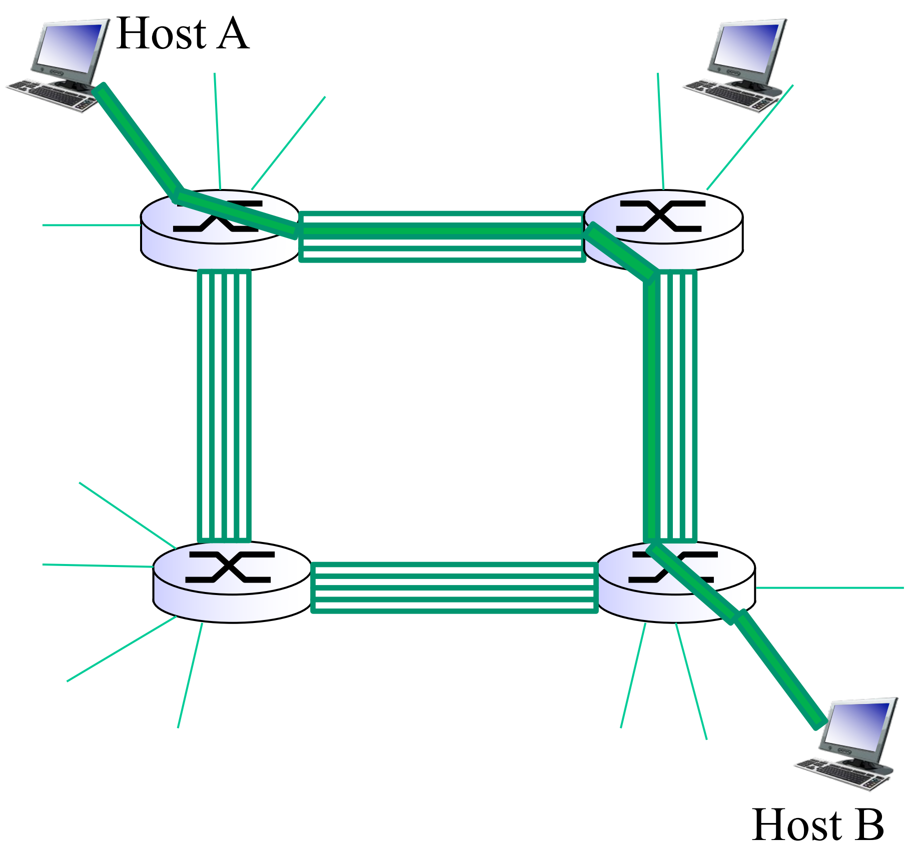
Circuit switching: FDM versus TDM

Time Division Multiplexing (TDM)

- Time divided into slots
- At maximum rate of (wider) frequency band, but only during its time slot (s)
- *T* slots in each frame: *T* virtual links



Circuit Switching: Delay



- A file of 640,000 bits
- All links in the network
  - use TDM with 24 slots per frame
  - a total bit rate of 1.536 Mbps
- It takes 500 msec to establish an end-to-end link.

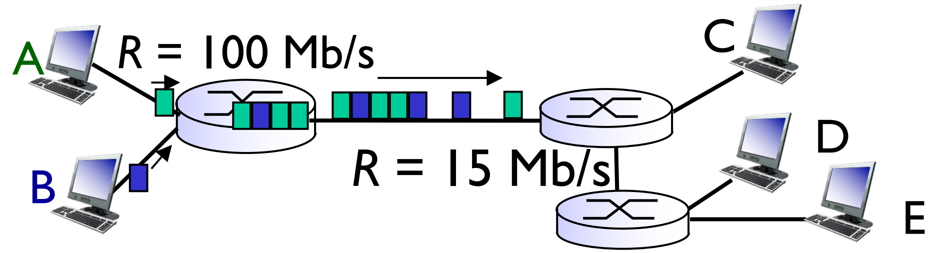
How long does it take to send the file?

- Each circuit has a transmission rate of 1.536 Mbps/24 = 64 kbps
- It takes 640,000 bits / 64 kbps = 10 sec
- Total: 10 sec + 500 msec = 10.5 sec

Packet Switching vs Circuit Switching

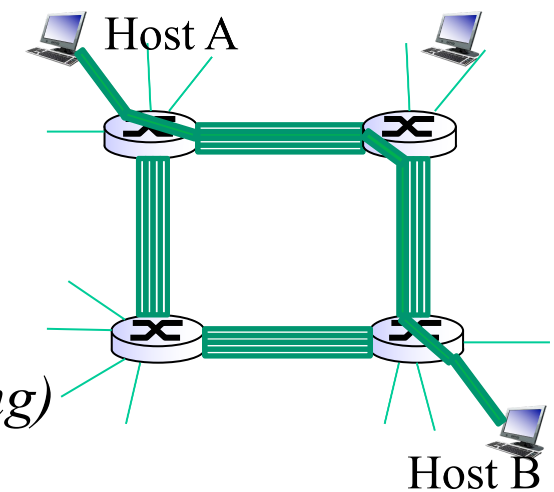
Packet switching:

- not reserved
- packet forwarded independently
  - store-and-forward
  - routing and forwarding
- at full link capacity
- packet loss, queuing delay



Circuit switching:

- reserved (dedicated resources): buffer, link, transmission rate
- establish an end-to-end link
- a fraction of each link's capacity
- guaranteed constant rate
- circuit segment idle if not used by call *(no sharing)*



Which is better?

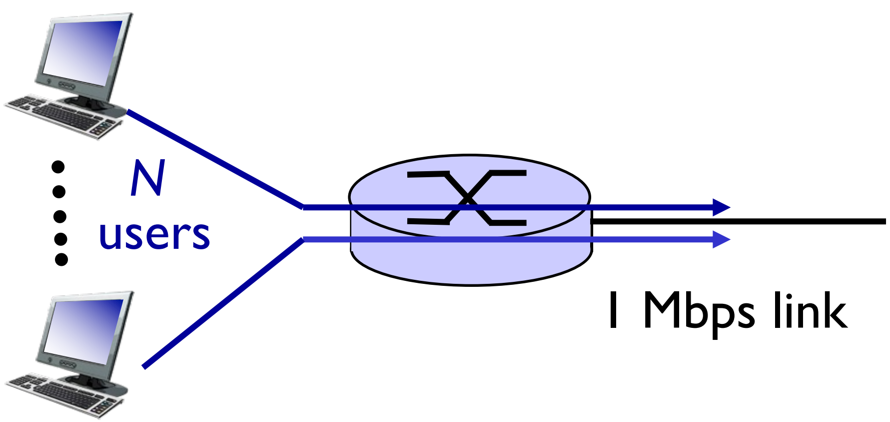
Packet switching versus circuit switching

Packet switching allows more users to use network!

- *Users may be active with a small probability*
- **Circuit switching** pre-allocates use of the transmission link regardless of demand, with allocated but unneeded link time going unused.
- **Packet switching** on the other hand allocates link use *on demand.*

Example:

- 1 Mb/s link
- Each user:
  - At each time, active with a probability of 10%
  - 100 kbps when "active"



Circuit switching:

10 users

Packet switching:

With 35 users, that probability that more than 10 users are active at same time is less than 0.0004.

$$\sum_{n=11}^{35} 0.1^{n} \times 0.9^{35-n} \binom{35}{n}$$

SUSTech Southern University of Science and Technology

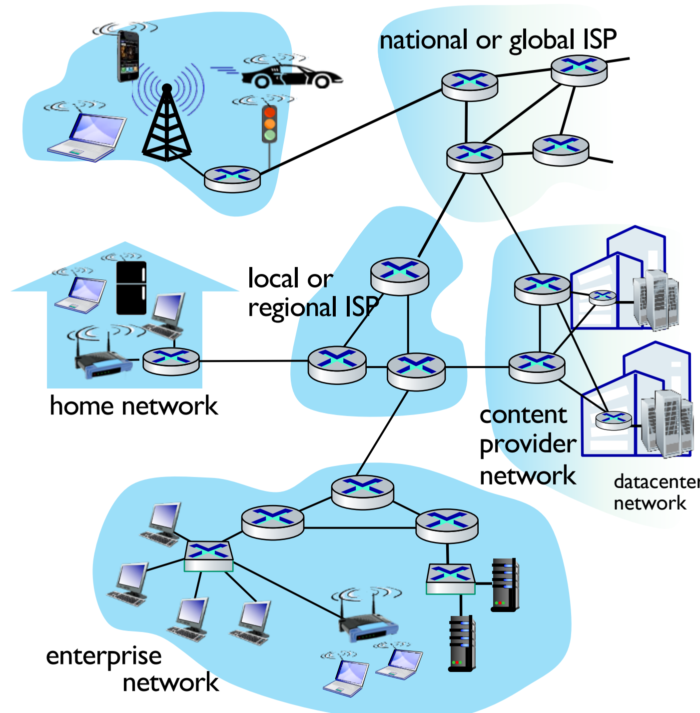
Packet switching versus circuit switching

Packet switching:

- great for bursty data
  - resource sharing
  - simpler, no call setup
- excessive congestion possible: queuing delay and packet loss
  - protocols needed for reliable data transfer, congestion control
- *Q*: How to provide circuit-like behavior?
  - bandwidth guarantees needed for audio/video apps
  - still an unsolved problem (chapter 7)

Internet structure: network of networks

- End systems connect to Internet via access ISPs (Internet Service Providers)
  - Residential ISPs
  - University and company ISPs
- Access ISPs in turn must be interconnected.
  - So that any two hosts can send packets to each other
- Resulting network of networks is very complex
  - Evolution was driven by economics and national policies

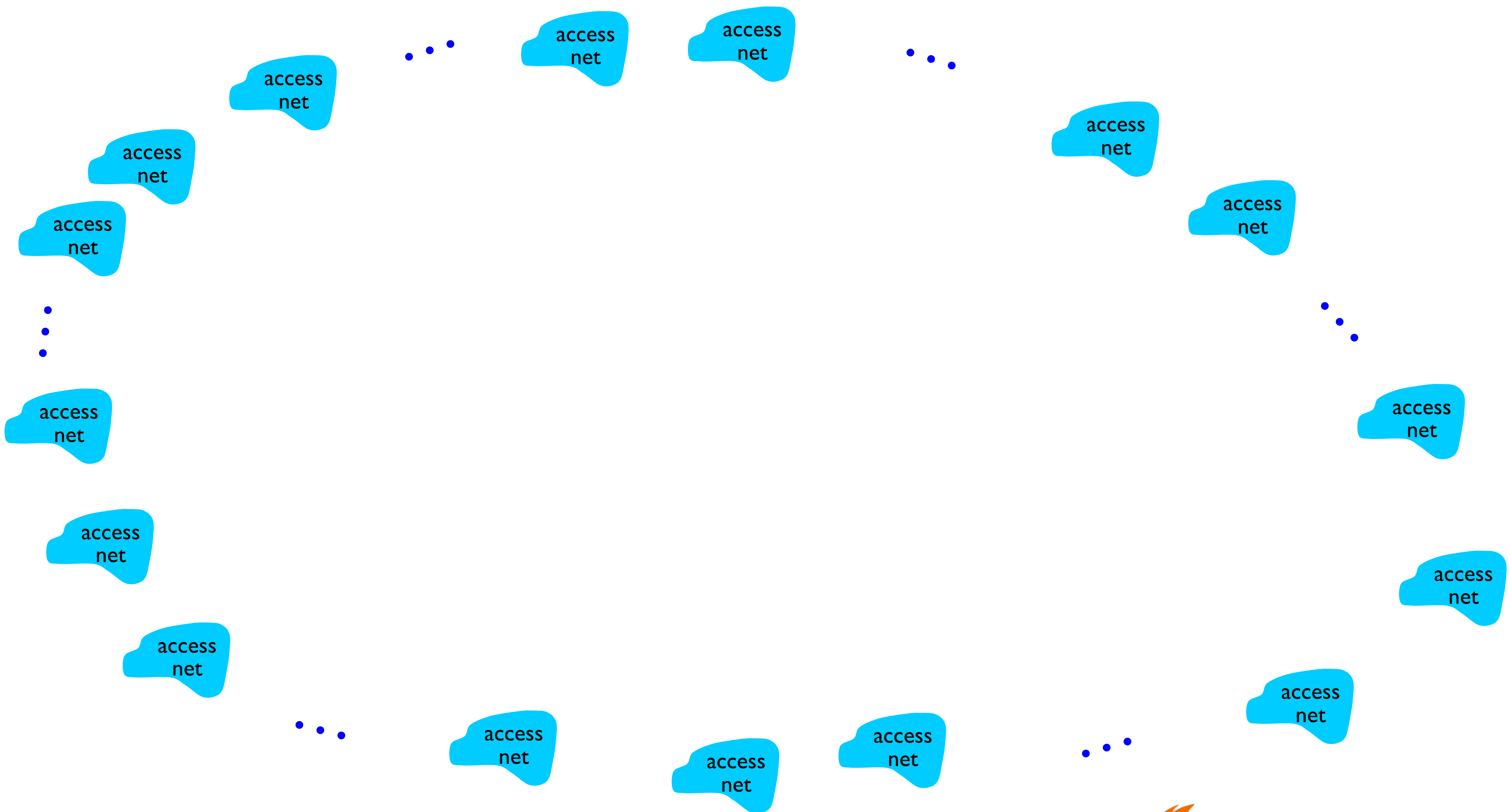


Let's take a stepwise approach to describe current Internet structure.

SUSTech Southern University of Science and Technology

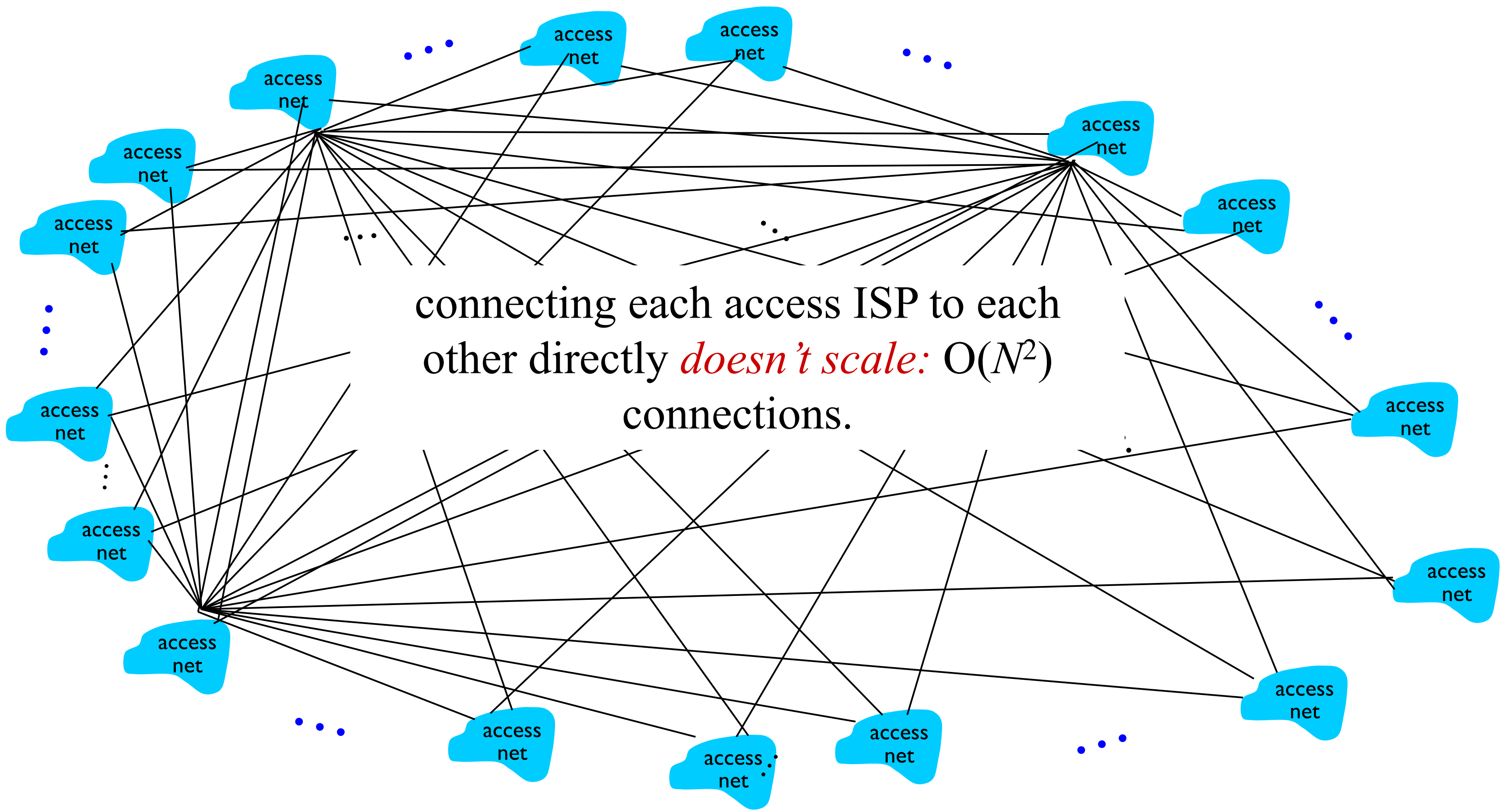
Internet structure: network of networks

Question: given *millions* of access ISPs, how to connect them together?



Internet structure: network of networks

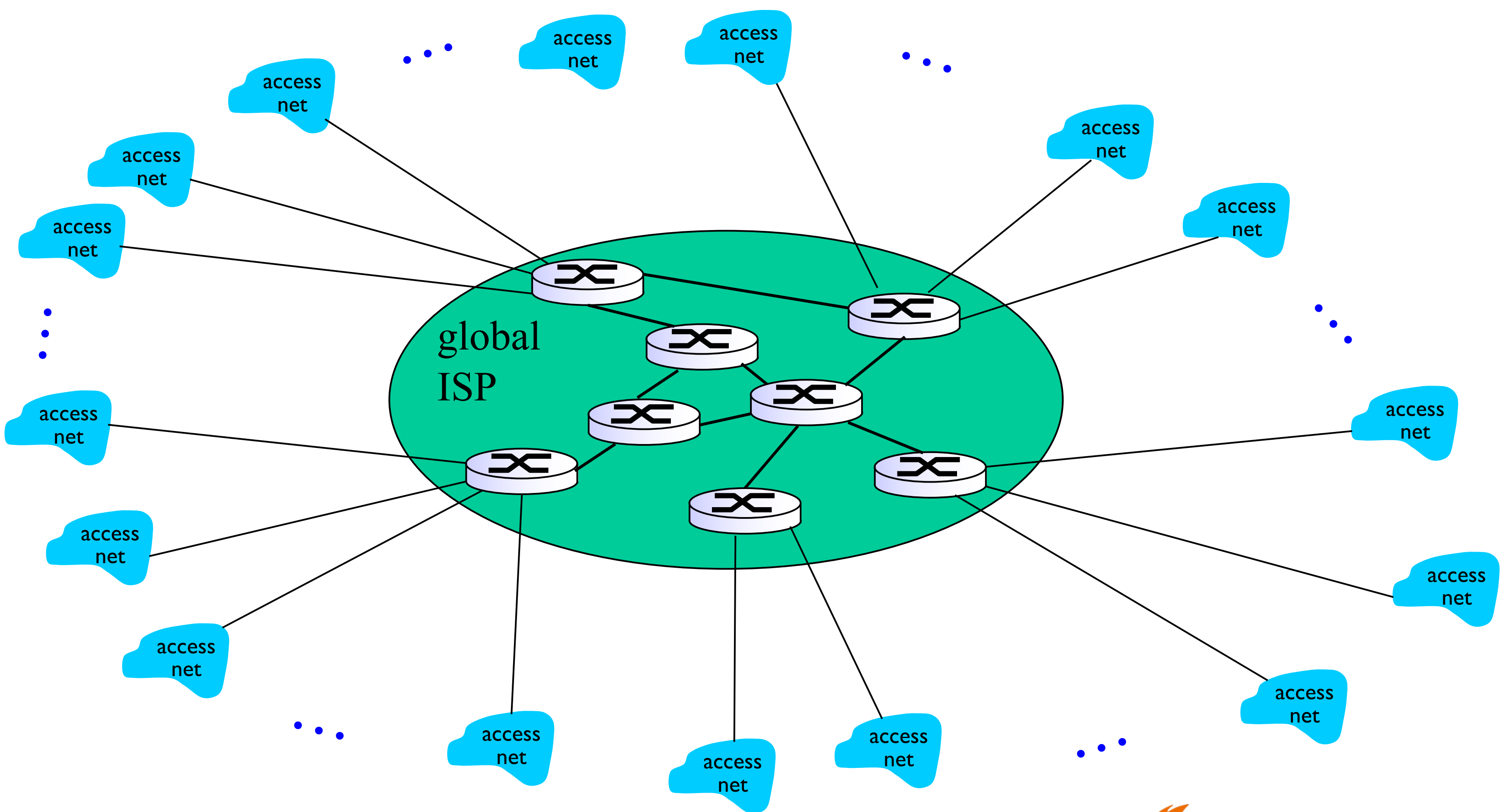
Option 1: connect each access ISP to every other access ISP?



connecting each access ISP to each other directly *doesn't scale:* $O(N^2)$ connections.

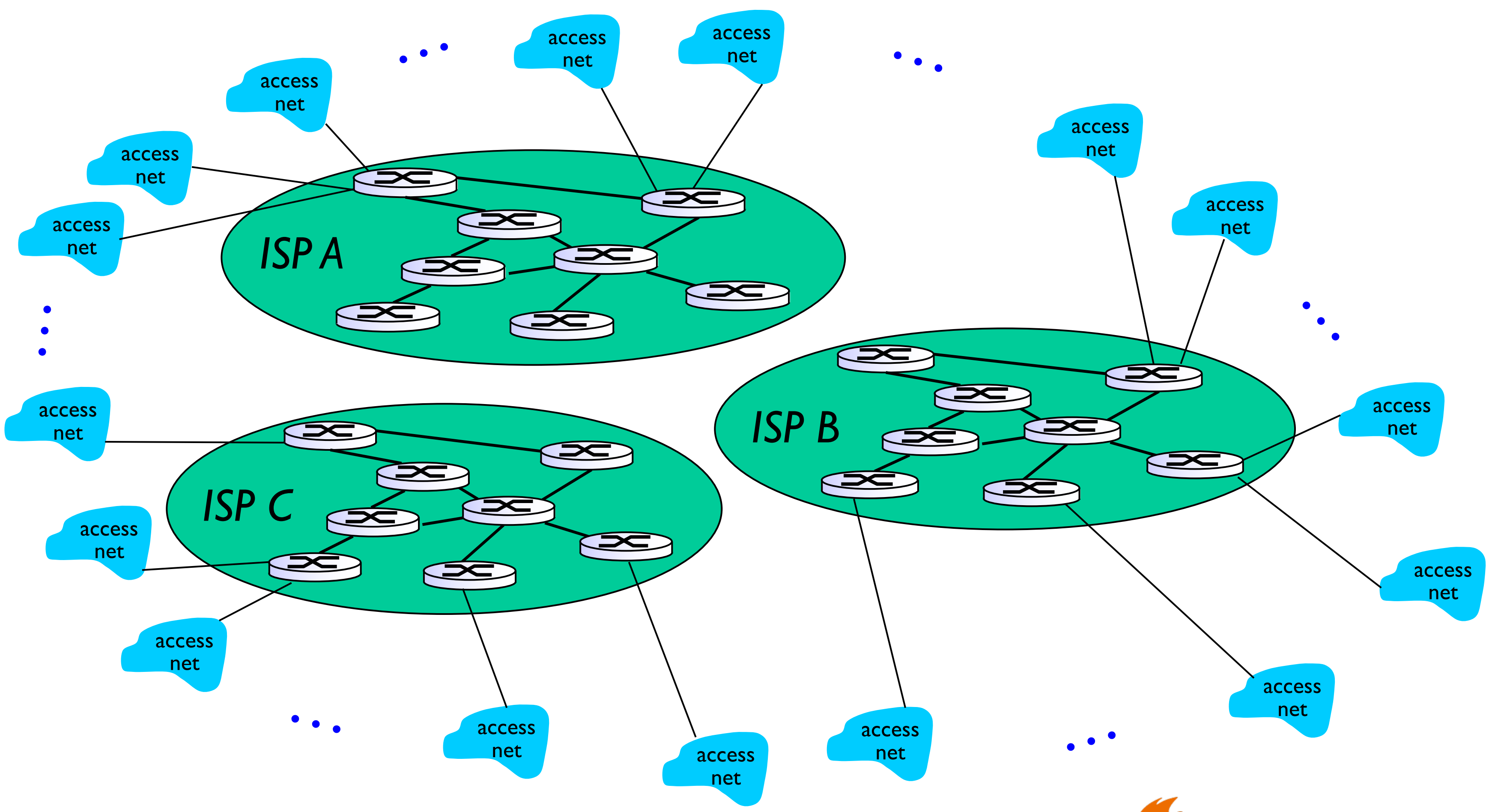
Internet structure: network of networks

Option 2: connect each access ISP to a global transit ISP? Customer and provider ISPs have economic agreement.



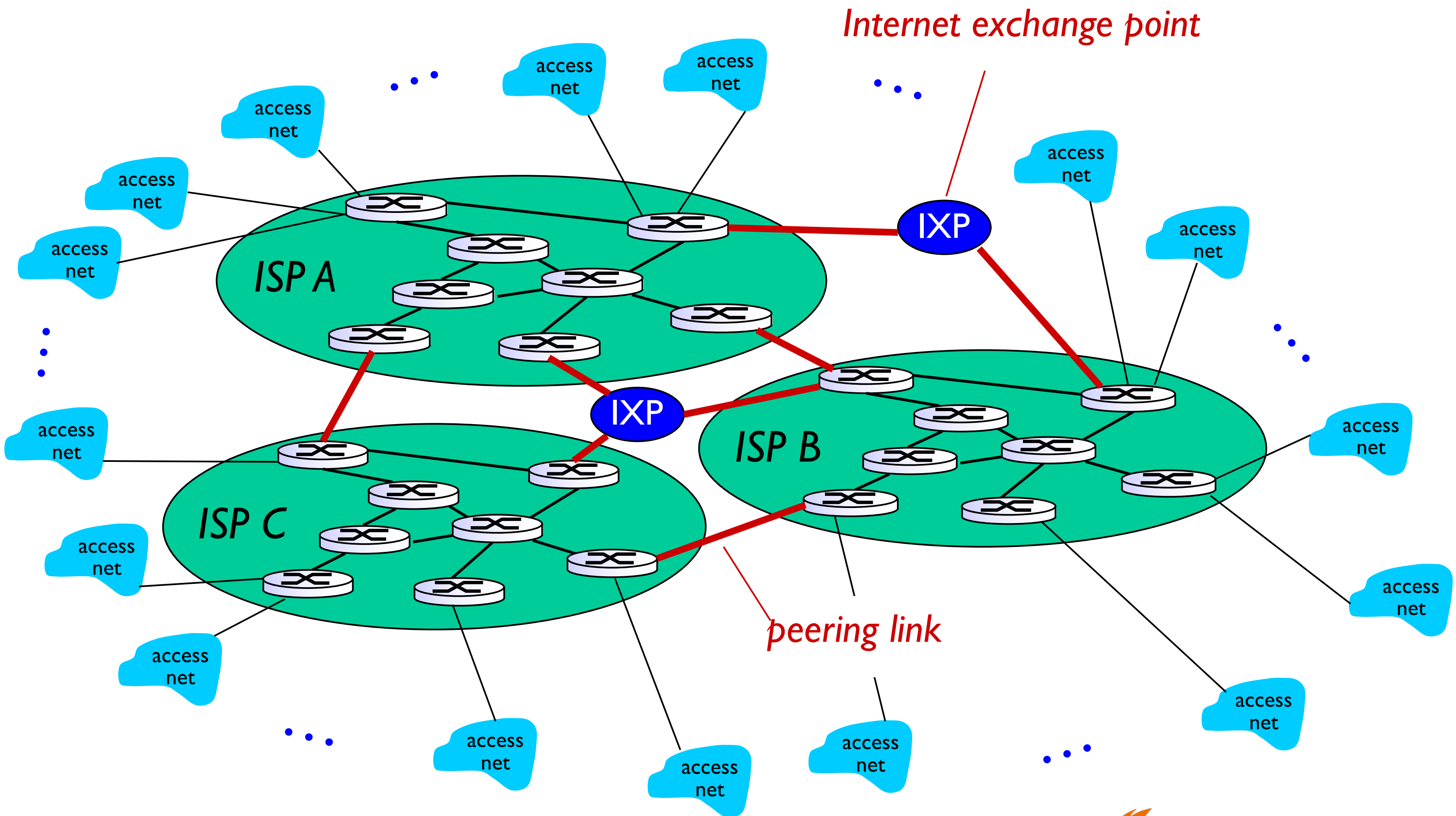
Internet structure: network of networks

But if one global ISP is viable business, there will be competitors ….



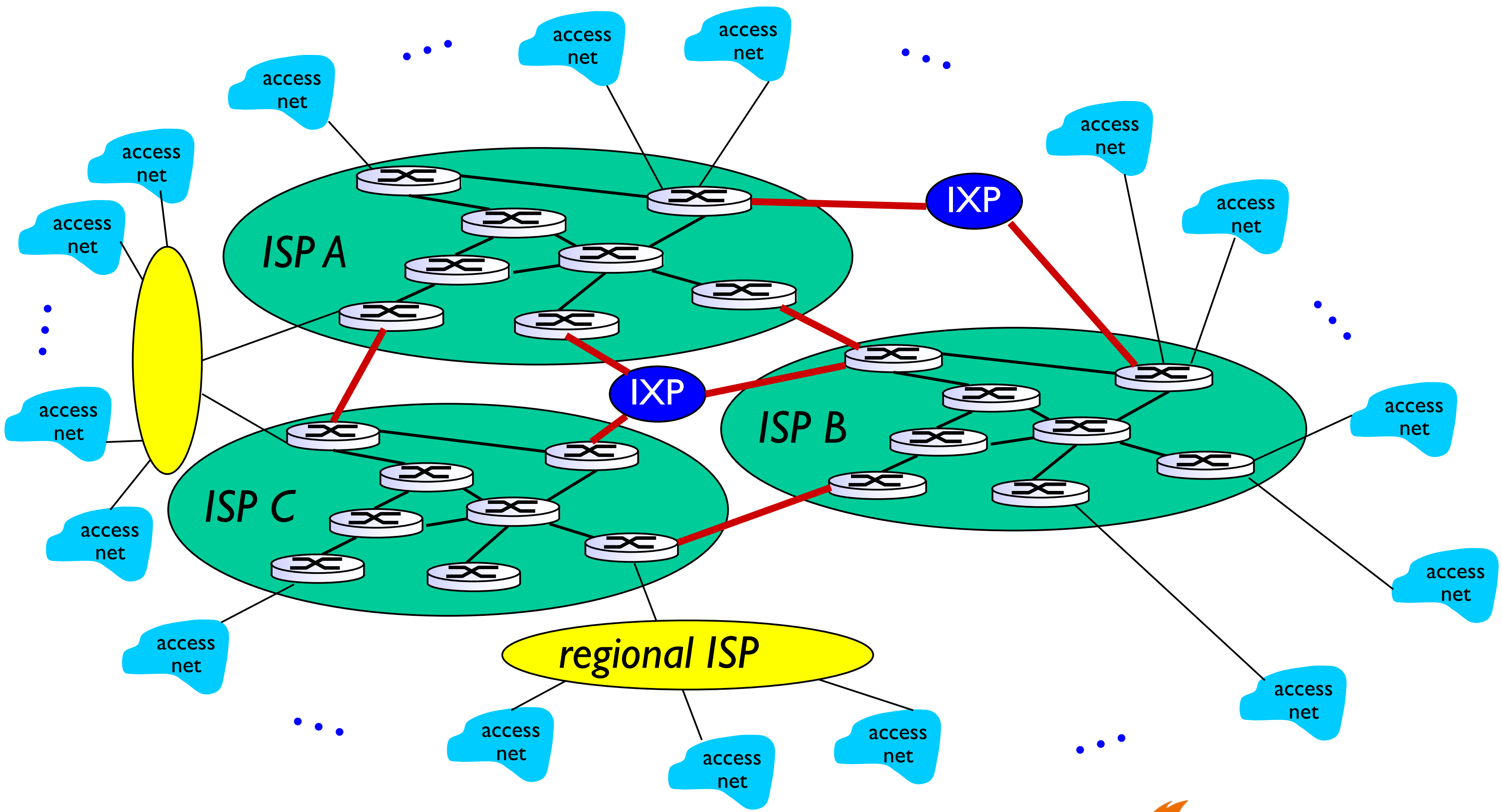
Internet structure: network of networks

But if one global ISP is viable business, there will be competitors …. which must be interconnected



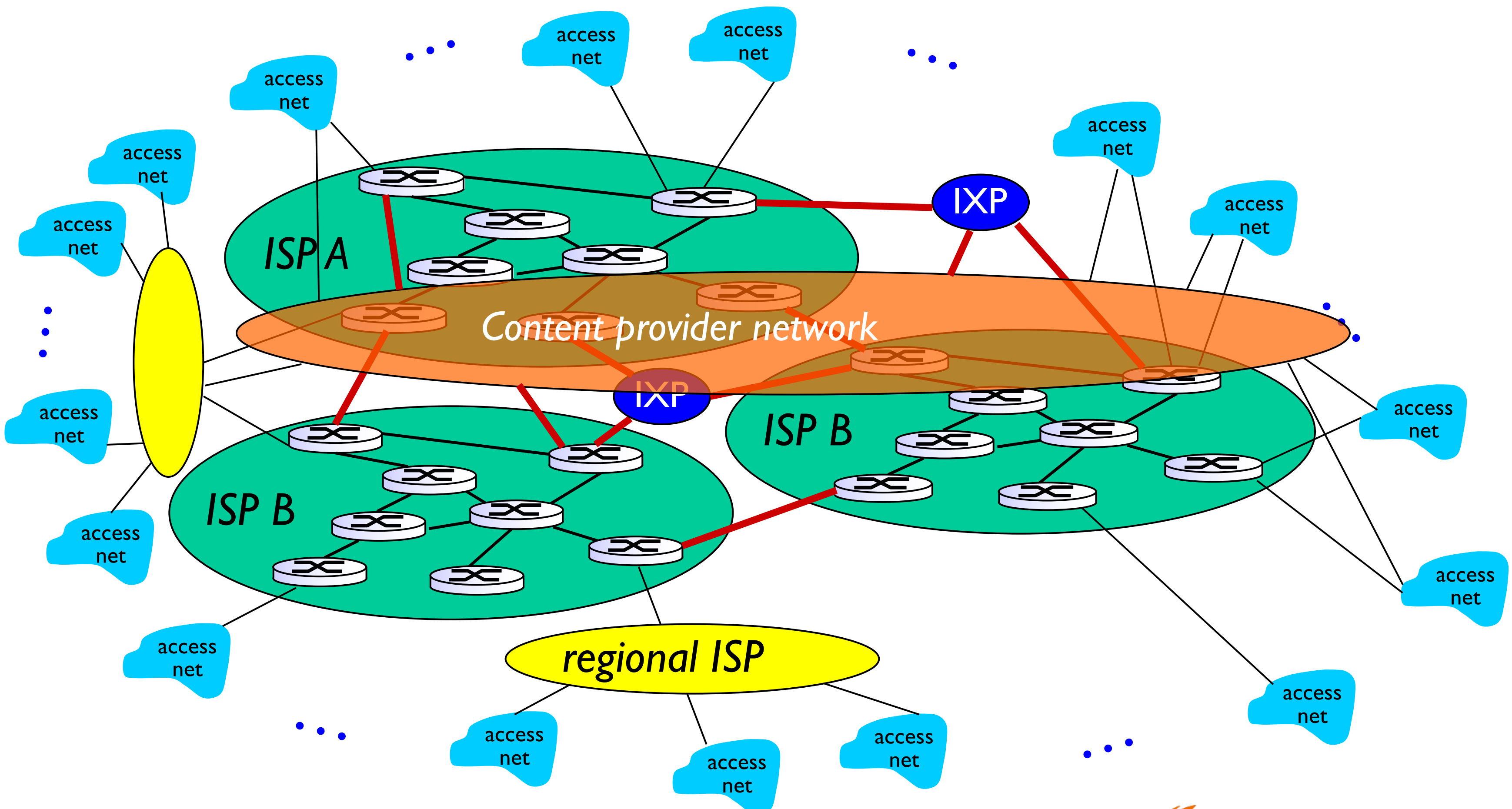
Internet structure: network of networks

… and regional networks may arise to connect access nets to ISPS

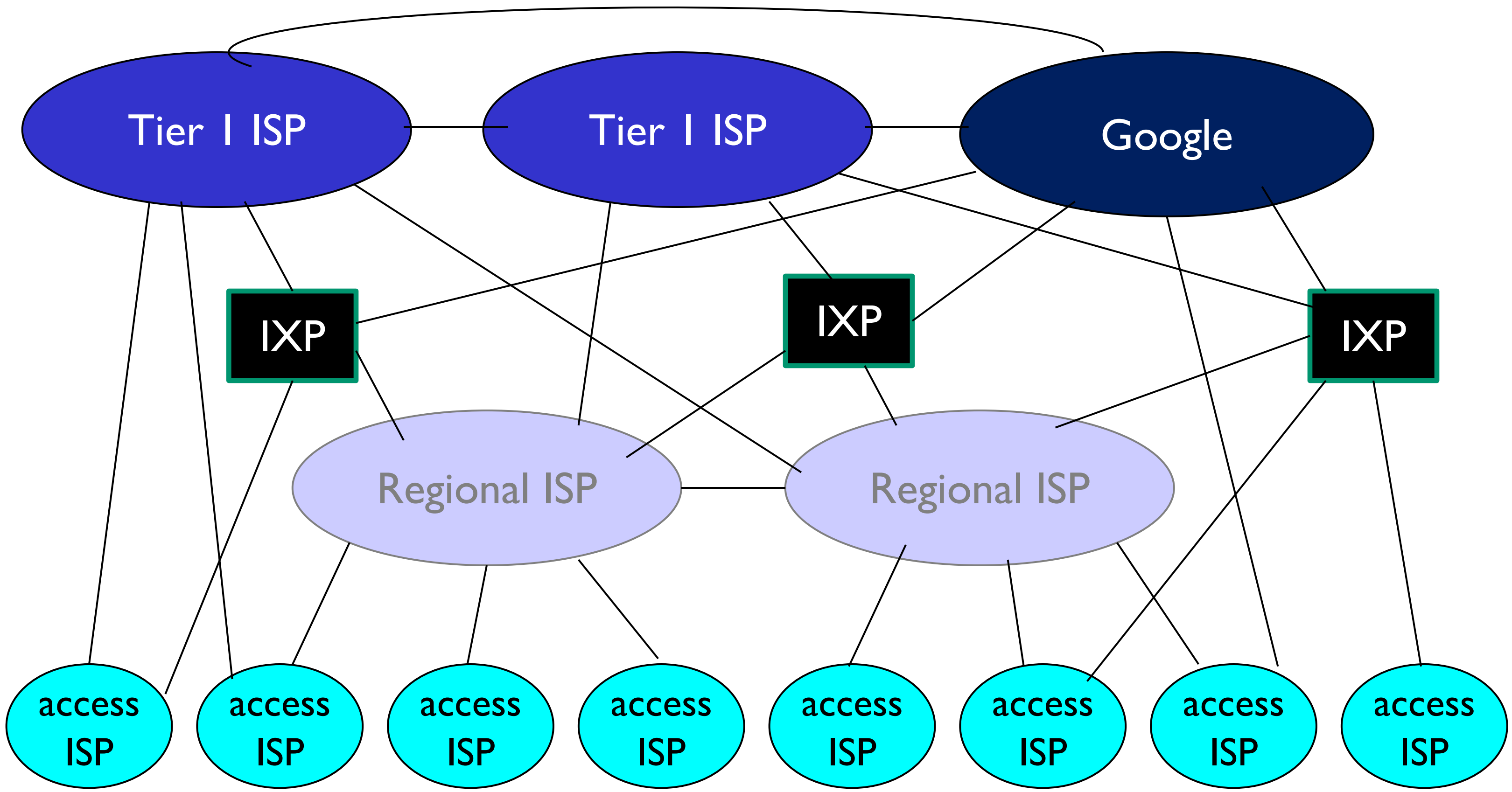


Internet structure: network of networks

… and content provider networks (e.g., Google, Microsoft) may run their own network, to bring services, content close to end users



Internet structure: network of networks



- "Tier-1" commercial ISPs (e.g., Level 3, Sprint, AT&T, NTT), national & international coverage
- Multi-homing: to connect to two or more provider ISPs
- Peer: a pair of nearby ISPs at the same level
- Content provider network (e.g, Google): private network that connects it data centers to Internet, often bypassing tier-1, regional ISPs

Chapter 1: roadmap

1.1 what is the Internet?

1.2 network edge

- end systems, access networks, links

1.3 network core

- packet switching, circuit switching, network structure

1.4 delay, loss, throughput in networks

1.5 protocol layers, service models

1.6 networks under attack: security

1.7 history

Performance Metric（性能指标）

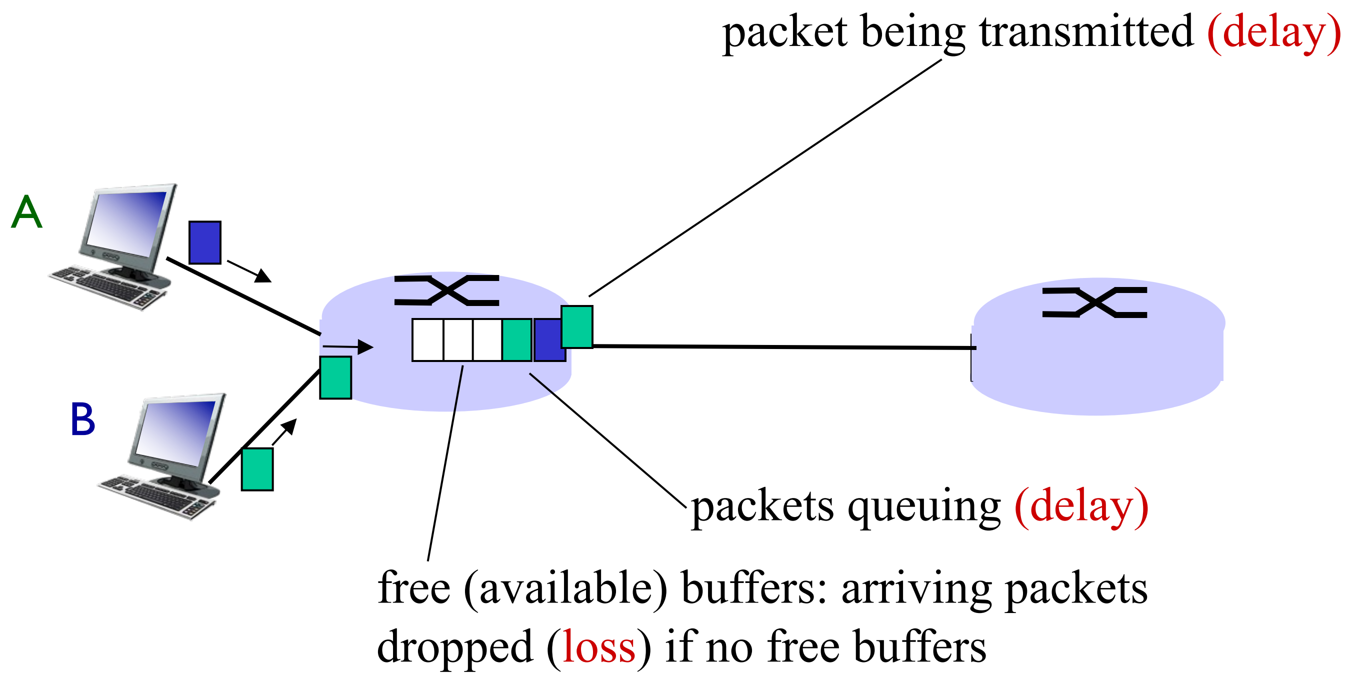
We would like Internet services to be able to move as much data as we want between any two end systems, instantaneously, without any loss of data.

- Delay（时延）
  - Nodal delay; end-to-end delay
- Packet loss（丢包）
- Throughput（吞吐量）: the amount of data per second that can be transferred between end systems

How do loss and delay occur?

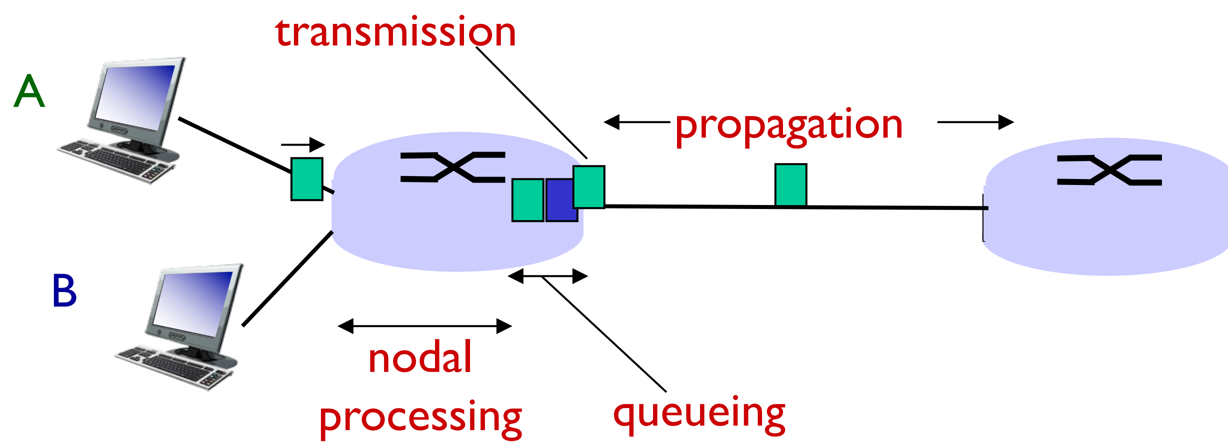
Packets queue in router buffers

- packet arrival rate to link (temporarily) exceeds output link capacity
- packets queue, wait for turn

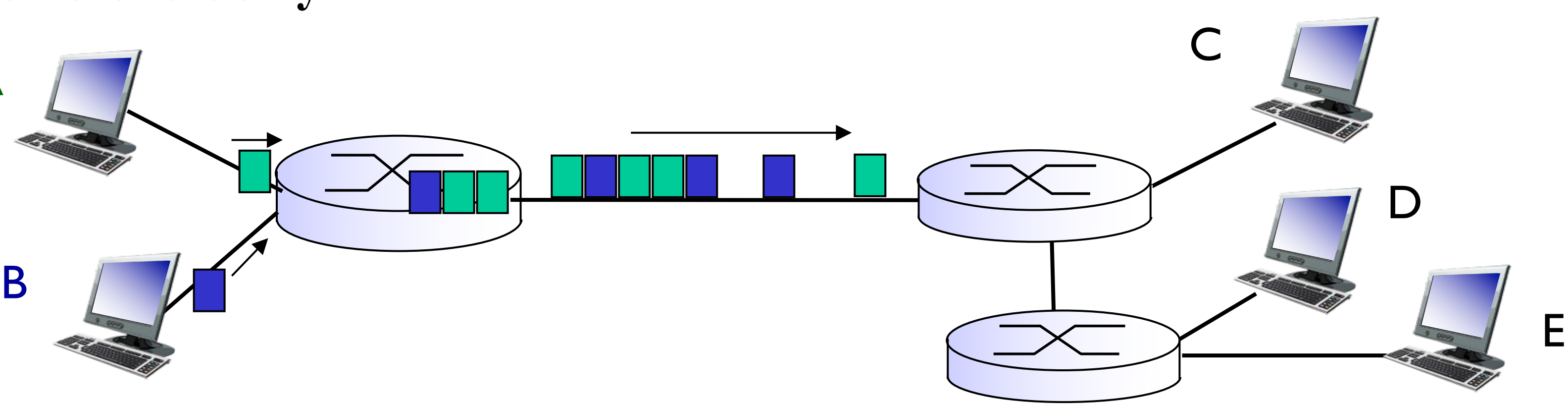


Nodal vs End-to-End Delay

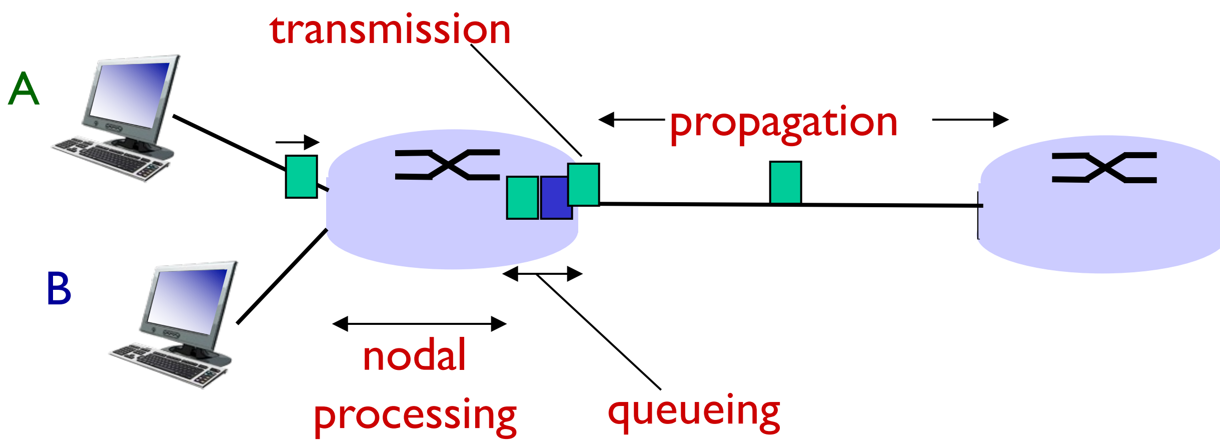
Nodal (节点) delay



End-to-end delay



Four sources of nodal delay



Nodal (节点) delay

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

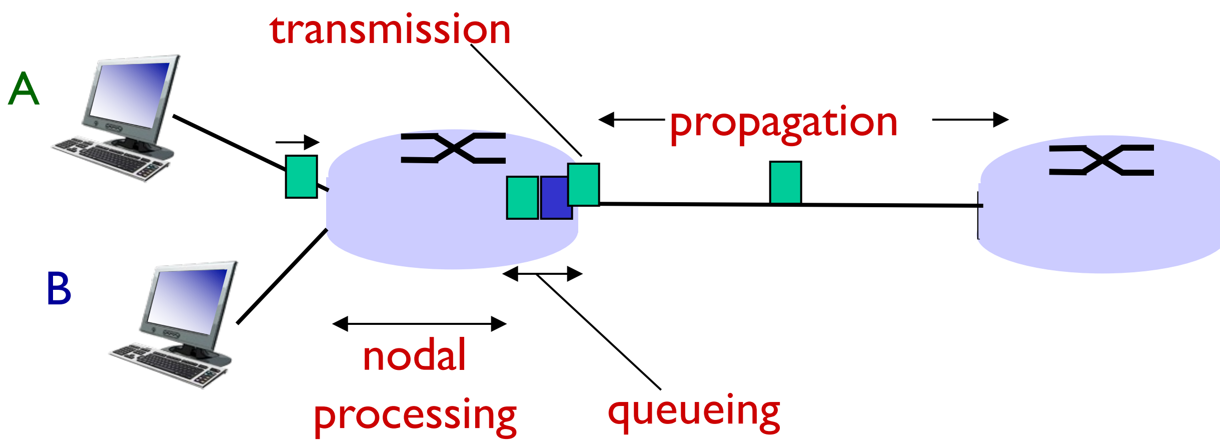
d_{proc}: nodal processing

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

d_{queue}: queueing delay

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

Four sources of nodal delay



Nodal (节点) delay

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

d_{trans}: transmission delay (传输时延):

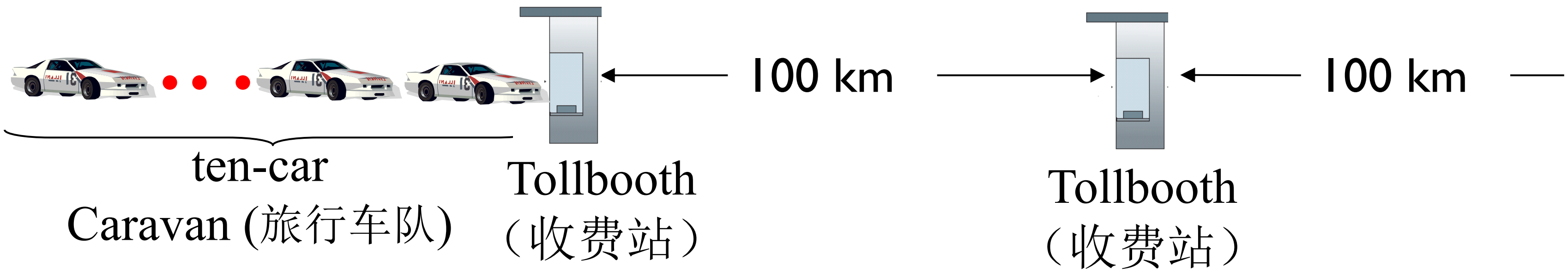
- L: packet length (bits)
- R: transmission rate (bps)
- $d_{trans} = L/R$

d_{prop}: propagation delay (传播时延):

- d: length of physical link
- s: propagation speed in medium (~2×10^8 m/sec)
- $d_{prop} = d/s$

d_{trans} and d_{prop}
very different

Caravan analogy

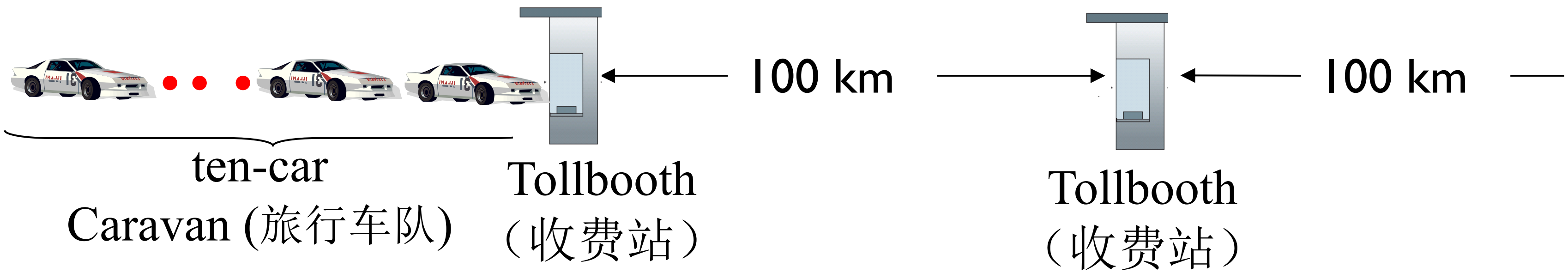


- Car ~ bit; caravan ~ packet
- Tollbooth takes 12 sec to service one car ~ transmission time
- Cars "propagate" at 100 km/hr ~ propagation speed

Q: How long does it take the last car to arrive at the 2nd tollbooth?

- Transmission delay : time to "push" entire caravan through tollbooth onto highway = 12*10 = 120 sec
- Propagation delay : time for last car to propagate from 1st to 2nd toll both: 100km/(100km/hr)= 1 hr
- *A:* 62 minutes

Caravan analogy (more)



- Suppose cars now "propagate" at 1000 km/hr
- and suppose tollbooth now takes one min to service a car
- *Q:* How long does it take the last car to arrive at the 2nd tollbooth?
  - Transmission delay: 1 min * 10 = 10 min
  - Propagation delay: 100 km / 1000 km/hr = 6 min

After 7 min, 1st car arrives at second booth; three cars still at 1st booth.

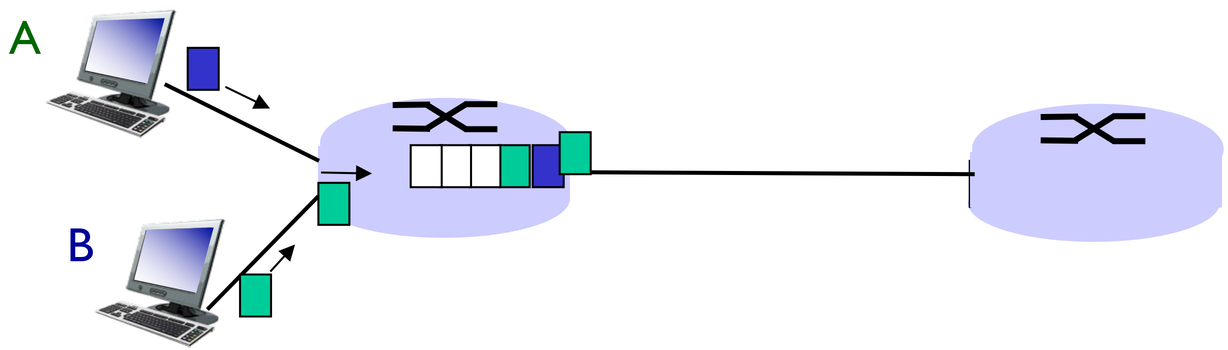
[Transmission versus Propagation Delay Interactive Animation (unicam.it)]

Queueing delay

Nodal (节点) delay

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

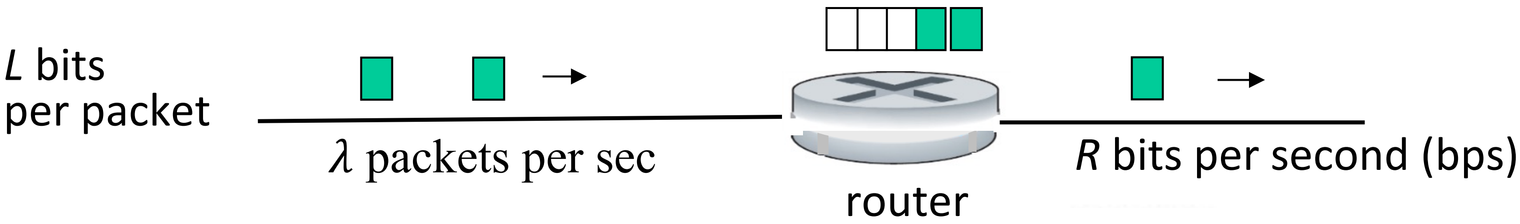
Unlike the other three delays (namely, d_{proc} , d_{trans} , d_{prop}), the queuing delay can vary from packet to packet.



When characterizing queuing delay, statistical measures:

- average queuing delay
- variance of queuing delay
- the probability that the queuing delay exceeds some specified value

Queueing delay at a router



- L: packet length (bits)
- λ : average packet arrival rate (packets per sec)
- R: transmission rate (bps)

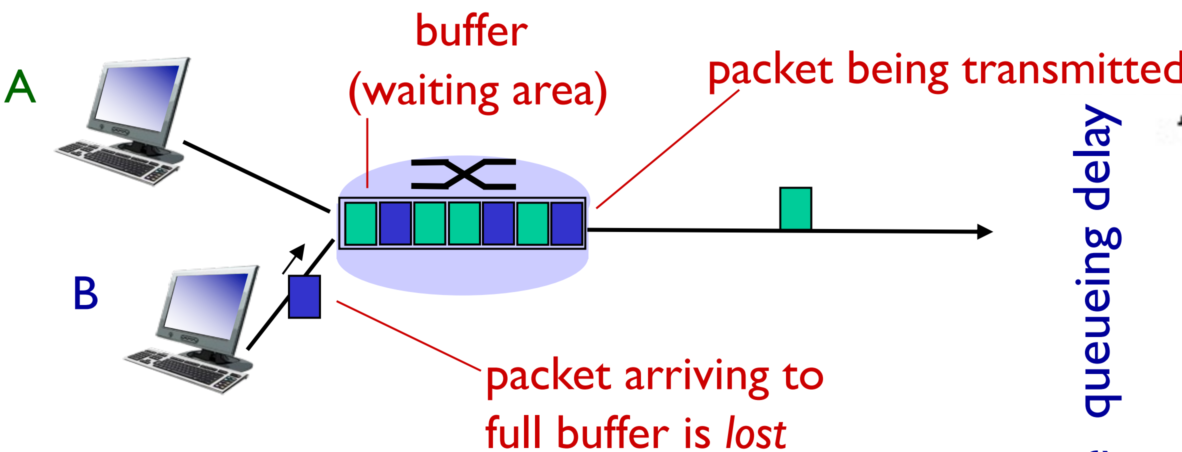
Traffic intensity = $L\lambda/R$

- $L\lambda/R \rightarrow 0$: avg. queueing delay small
- $L\lambda/R \rightarrow 1$: avg. queueing delay $\rightarrow \infty$ (queuing theory)
- $L\lambda/R > 1$: more "work" arriving than can be serviced, average delay infinite!

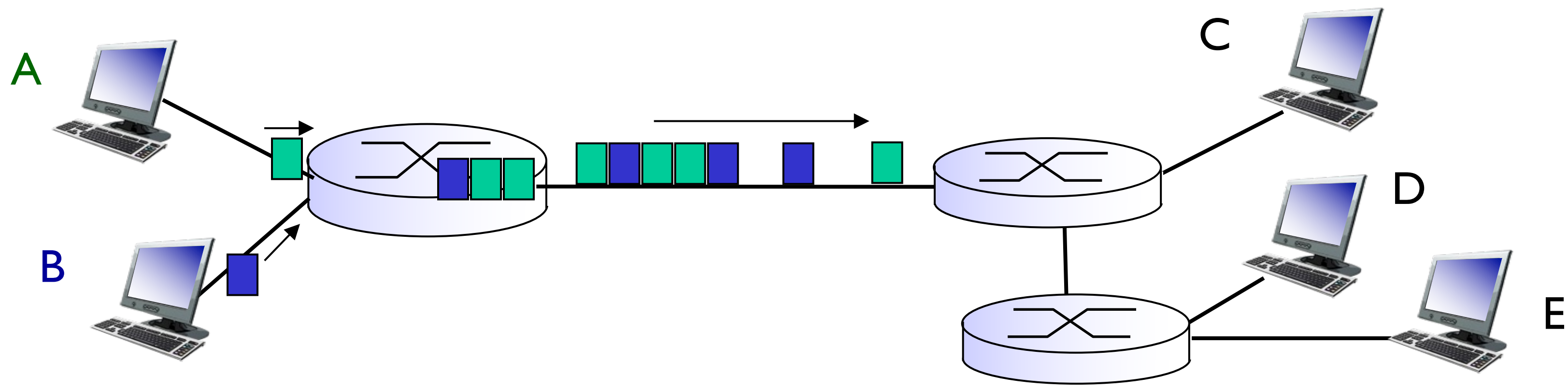
[Queuing and Loss Interactive Animation (unicam.it)]

Packet loss

- Queue (aka buffer) preceding link in buffer has finite capacity
- Packet arriving to full queue will be dropped (aka lost)
- Lost packet may be retransmitted by previous node, by source end system, or not at all



End-to-End Delay



Suppose d_{queue} is negligible; *N-1* routers between two hosts

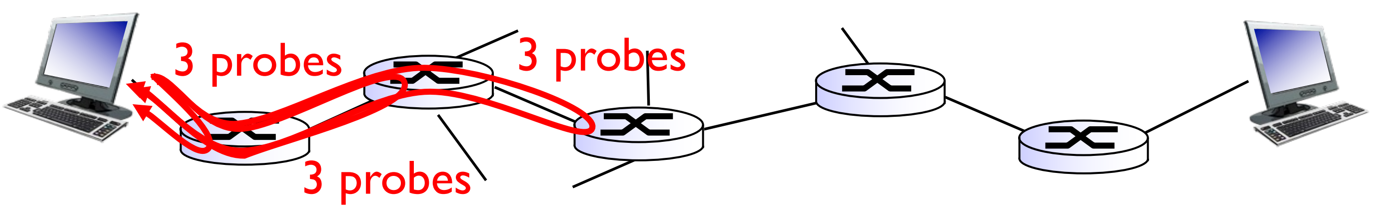
End-to-End delay

$$d_{end\text{-}end} = N\,(d_{proc} + d_{trans} + d_{prop})$$

However, queuing delay can be significant and time-varying

"Real" Internet delays and routes

- What do "real" Internet delay & loss look like?
- Traceroute program: provides delay measurement from source to router along end-end Internet path towards destination. For all *i:*
  - sends three packets with a time-to-live (TTL) of *i*; will reach router *i* on path towards destination
  - router *i* will return packets to sender
  - sender times interval between transmission and reply.



N-1 routers, send *3N* packets

"Real" Internet delays, routes

traceroute: gaia.cs.umass.edu to www.eurecom.fr

3 round-trip delay measurements from gaia.cs.umass.edu to cs-gw.cs.umass.edu

```
1  cs-gw (128.119.240.254)  1 ms  1 ms  2 ms
2  border1-rt-fa5-1-0.gw.umass.edu (128.119.3.145)  1 ms  1 ms  2 ms
3  cht-vbns.gw.umass.edu (128.119.3.130)  6 ms 5 ms 5 ms
4  jn1-at1-0-0-19.wor.vbns.net (204.147.132.129)  16 ms 11 ms 13 ms
5  jn1-so7-0-0-0.wae.vbns.net (204.147.136.136)  21 ms 18 ms 18 ms
6  abilene-vbns.abilene.ucaid.edu (198.32.11.9)  22 ms  18 ms  22 ms
7  nycm-wash.abilene.ucaid.edu (198.32.8.46)  22 ms  22 ms  22 ms
8  62.40.103.253 (62.40.103.253)  104 ms 109 ms 106 ms
9  de2-1.de1.de.geant.net (62.40.96.129)  109 ms 102 ms 104 ms
10  de.fr1.fr.geant.net (62.40.96.50)  113 ms 121 ms 114 ms
11  renater-gw.fr1.fr.geant.net (62.40.103.54)  112 ms  114 ms  112 ms
12  nio-n2.cssi.renater.fr (193.51.206.13)  111 ms  114 ms  116 ms
13  nice.cssi.renater.fr (195.220.98.102)  123 ms  125 ms  124 ms
14  r3t2-nice.cssi.renater.fr (195.220.98.110)  126 ms  126 ms  124 ms
15  eurecom-valbonne.r3t2.ft.net (193.48.50.54)  135 ms  128 ms  133 ms
16  194.214.211.25 (194.214.211.25)  126 ms  128 ms  126 ms
17  * * *
18  * * *
19  fantasia.eurecom.fr (193.55.113.142)  132 ms  128 ms  136 ms
```

trans-oceanic link

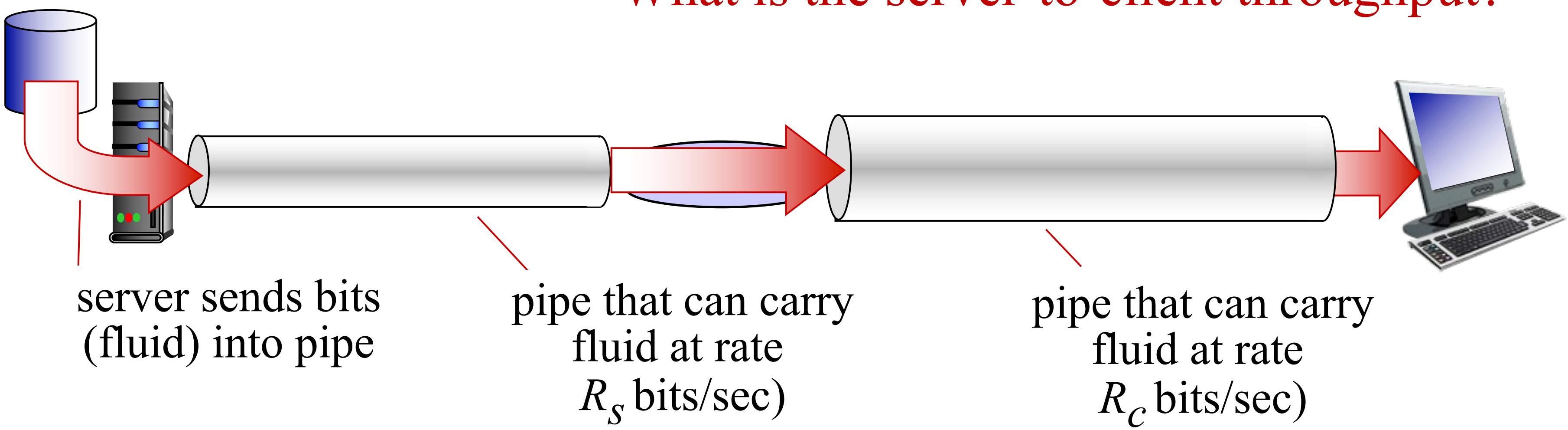
looks like delays *decrease*! Why?

* means no response (probe lost, router not replying)

* Do some traceroutes from exotic countries at www.traceroute.org

Throughput

What is the server-to-client throughput?



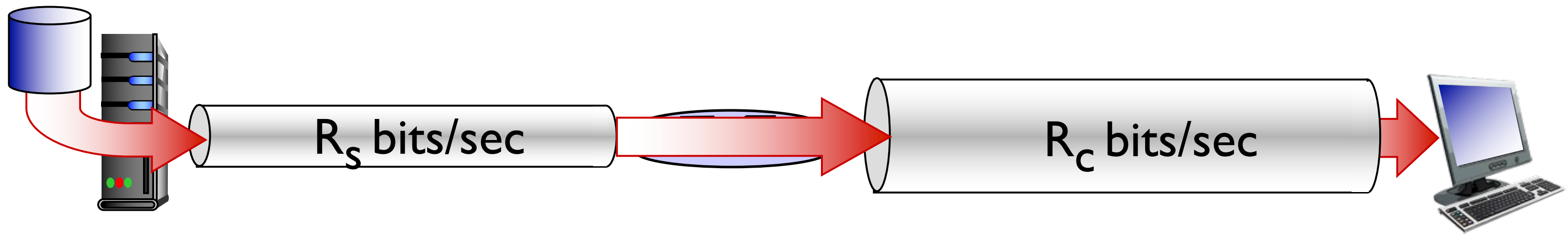
Throughput: rate (bits/time unit) at which bits transferred between sender/receiver

- *instantaneous:* rate at given point in time
- *average:* rate over longer period of time
  - A file of *F* bits; it takes *T* seconds for the receiver to receive the file → Average throughput of *F/T* bits/sec.

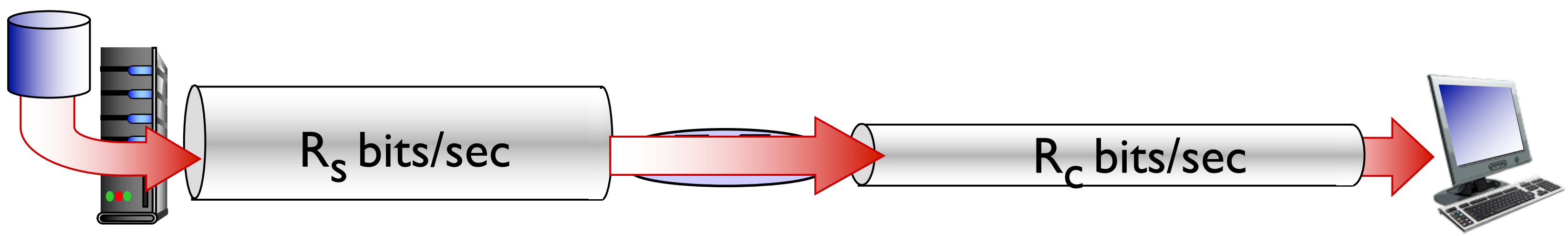
SUSTech Southern University of Science and Technology

Throughput (more)

- $R_s < R_c$ What is average end-end throughput?



- $R_s > R_c$ What is average end-end throughput?

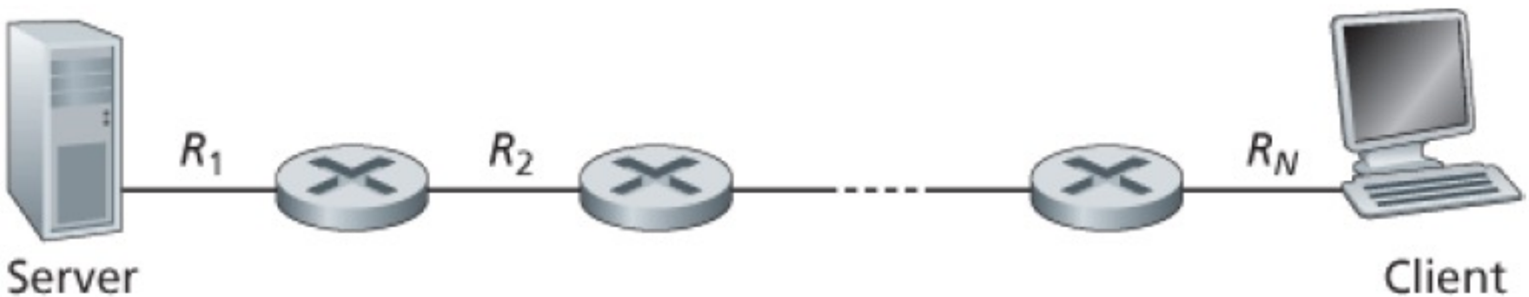


$\min\{R_s, R_c\}$

bottleneck (瓶颈) link

link on end-end path that constrains end-end throughput

Throughput (more)



What is average end-end throughput?

$\min\{R_1, R_2, \ldots, R_N\}$

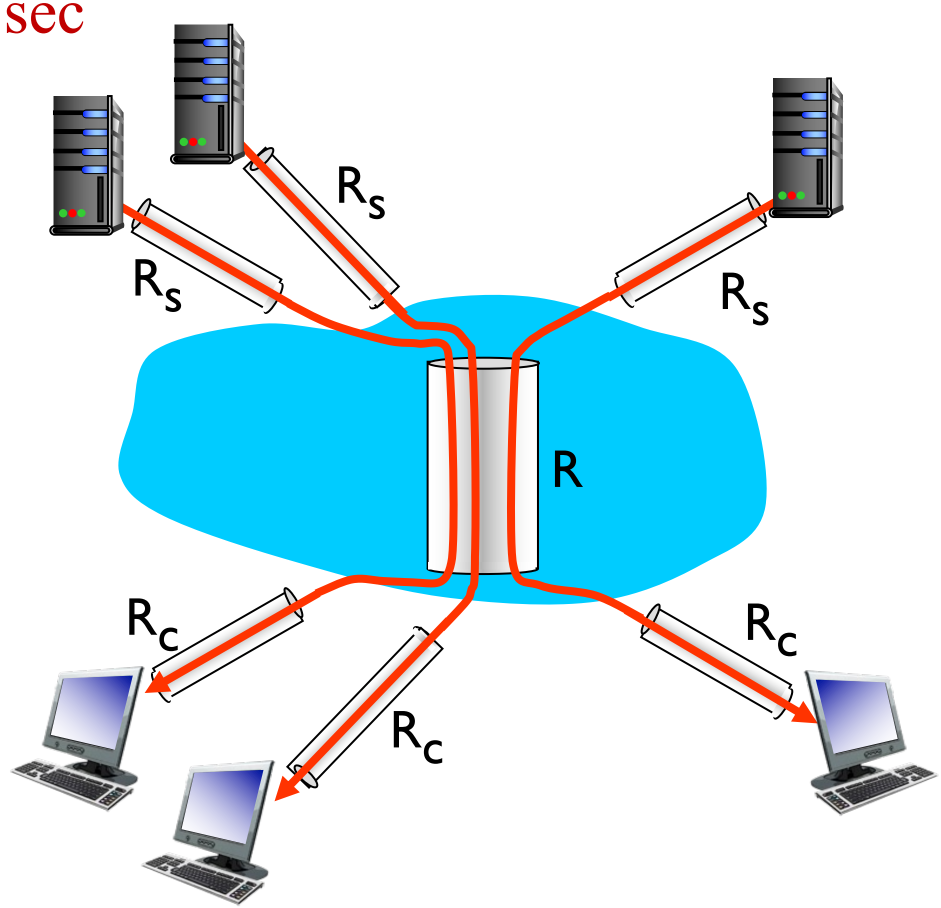
Throughput: Internet scenario

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

- Per-connection end-end throughput:

  $$\min\{R_c, R_s, R/10\}$$

- In practice: R_c or R_s is often bottleneck



Chapter 1: roadmap

1.1 what is the Internet?

1.2 network edge

- end systems, access networks, links

1.3 network core

- packet switching, circuit switching, network structure

1.4 delay, loss, throughput in networks

1.5 protocol layers, service models

1.6 networks under attack: security

1.7 history

Protocol "layers"

Networks are complex, with many "pieces":

- hosts
- routers
- links of various media
- applications
- protocols
- hardware, software

Question:

is there any hope of *organizing* structure of network?

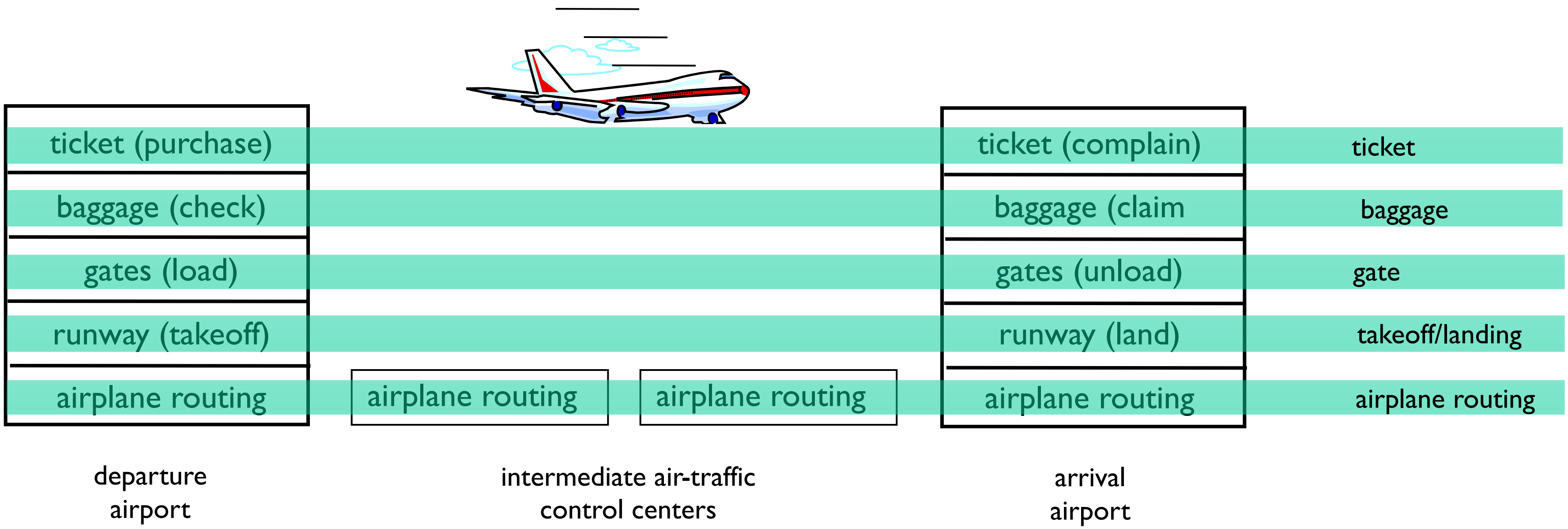
…. or at least our discussion of networks?

Organization of air travel

ticket (purchase)	ticket (complain)
baggage (check)	baggage (claim)
gates (load)	gates (unload)
runway takeoff	runway landing
airplane routing	airplane routing

airplane routing

Layering of airline functionality



Layers: Each layer provide services to the layer above

- **via** its own internal-layer actions
- **relying on** services provided by layer below

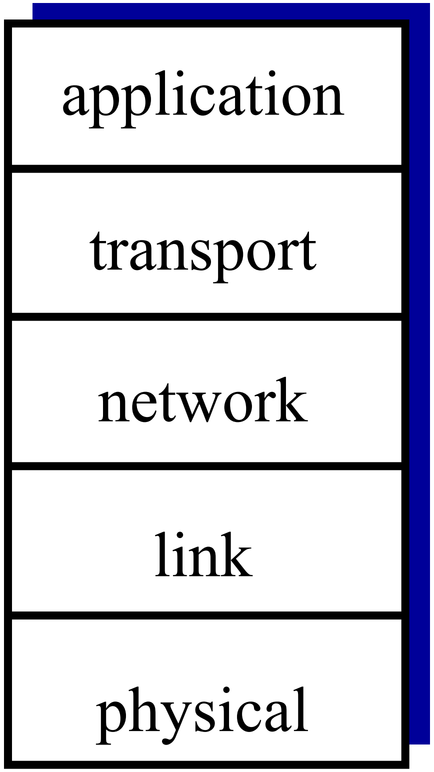
Why layering?

Dealing with complex systems:

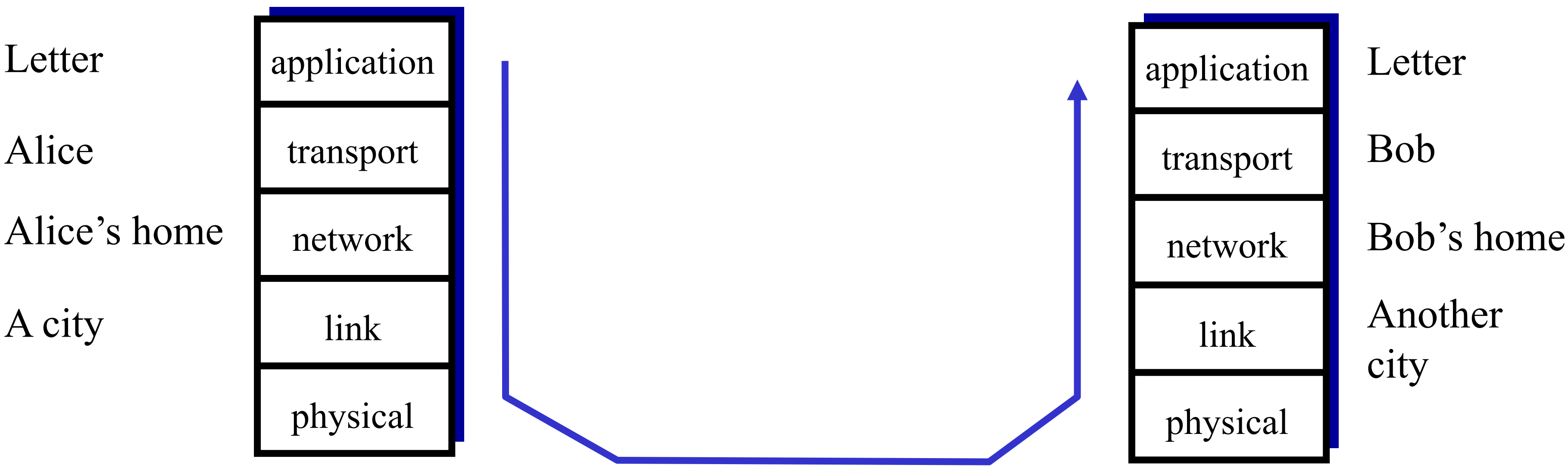
- Explicit structure allows identification, relationship of complex system's pieces
  - layered reference model for discussion
- modularization eases maintenance, updating of system
  - change of implementation of layer's service transparent to rest of system
  - e.g., change in gate procedure doesn't affect rest of system
- drawback?
  - One layer may duplicate lower layer functionality
  - Functionality at one layer may need information that is present only in another layer

Internet protocol stack

- **Application:** supporting network applications
  - FTP, SMTP, HTTP
- **Transport:** process（进程）-process data transfer
  - TCP, UDP
- **Network:** routing of datagrams from source host to destination host
  - IP, routing protocols
- **Link:** data transfer between neighboring network elements
  - Ethernet, 802.111 (WiFi), PPP
- **Physical:** bits "on the wire"



Layering of functionality



How the mail is delivered (e.g., through highway or rail) between cities

Layers: Each layer provide services to the layer above

- **via** its own internal-layer actions
- **relying on** services provided by layer below

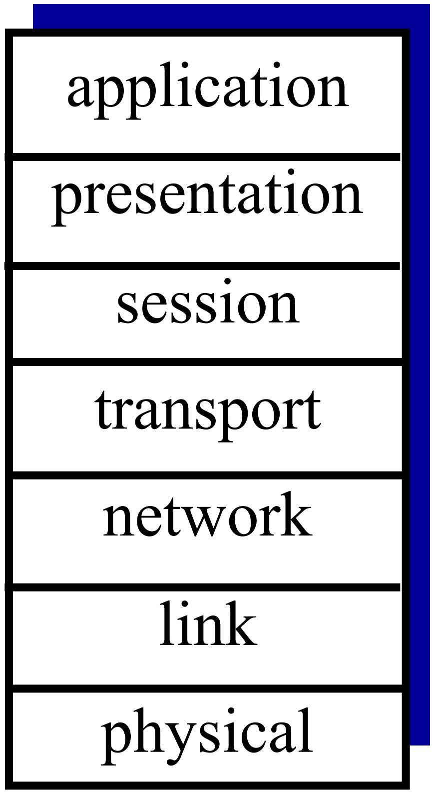
For example, reliable delivery of messages at layer *n*:

- Adding layer *n* functionality to detect and retransmit lost messages
- Using an unreliable message delivery service at layer *n-1*

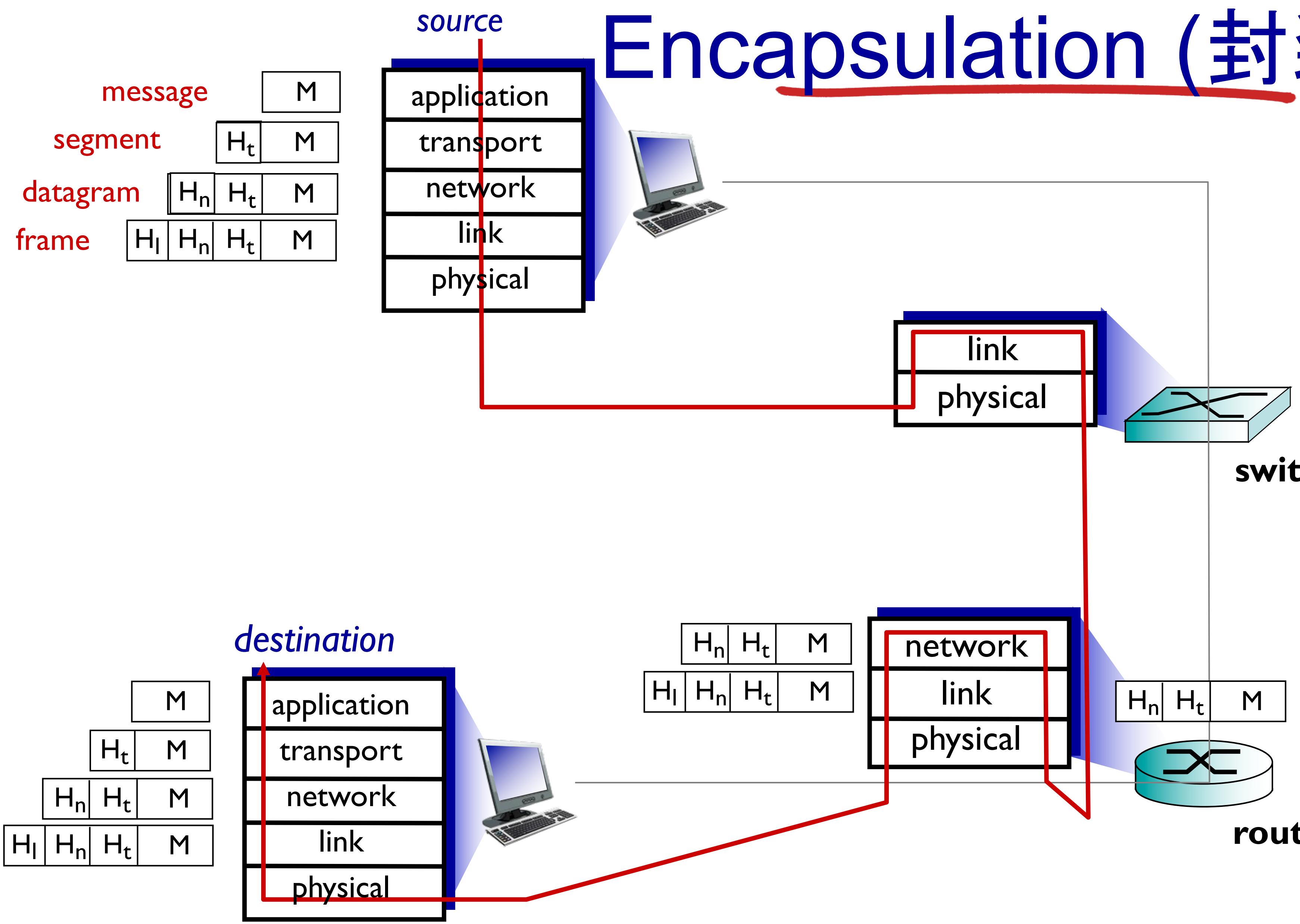
Internet protocol stack

ISO/OSI reference model

- Presentation: allow applications to interpret meaning of data, e.g., encryption, compression, machine-specific conventions
- Session: synchronization, checkpointing, recovery of data exchange
- Internet stack "missing" these layers!
  - these services, *if needed,* must be implemented in application
  - needed?



Encapsulation (封装)



SUSTech Southern University of Science and Technology

Chapter 1: roadmap

1.1 what is the Internet?

1.2 network edge

- end systems, access networks, links

1.3 network core

- packet switching, circuit switching, network structure

1.4 delay, loss, throughput in networks

1.5 protocol layers, service models

1.6 networks under attack: security

1.7 history

Network security

We attach devices to the Internet because we want to receive/send data from/to the Internet

- Field of network security:
  - how bad guys can attack computer networks
  - how we can defend networks against attacks
  - how to design architectures that are immune to attacks
- Internet not originally designed with (much) security in mind
  - *original vision:* "a group of mutually trusting users attached to a transparent network" ☺
  - Internet protocol designers playing "catch-up"
  - security considerations in all layers!

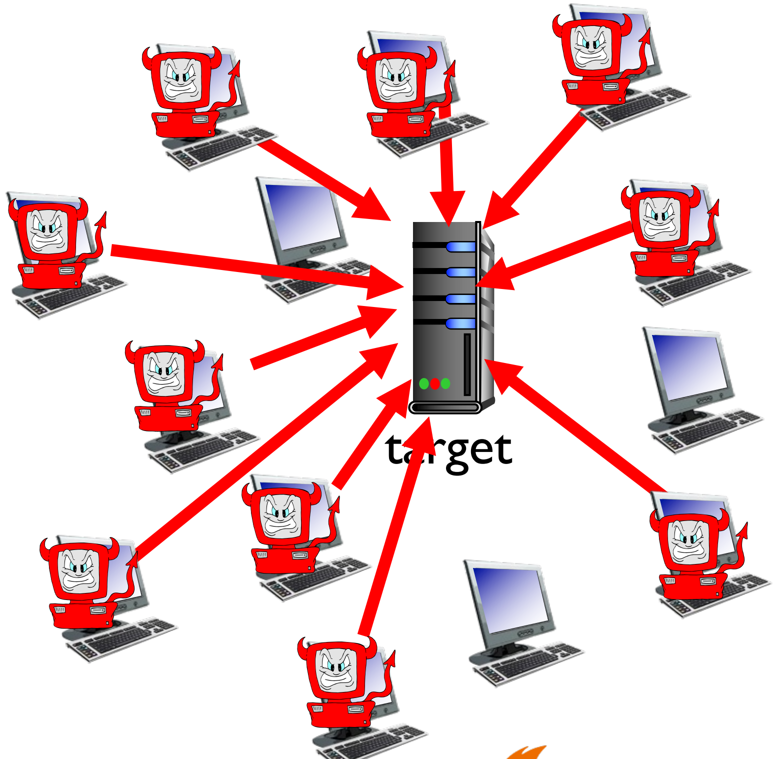
Bad guys: put malware (恶意软件) into hosts via Internet

- malware can get in host from:
  - *virus:* self-replicating infection by receiving/executing object (e.g., e-mail attachment) – with user interaction
  - *worm:* self-replicating infection by passively receiving object that gets itself executed – without user interaction
- spyware malware can record keystrokes, web sites visited, upload info to collection site
- infected host can be enrolled in botnet (僵尸网络), used for spam. DDoS attacks

Bad guys: attack server, network infrastructure

Denial of Service (DoS): attackers make resources (server, bandwidth) unavailable to legitimate（合法的） traffic by overwhelming resource with bogus（伪造的） traffic

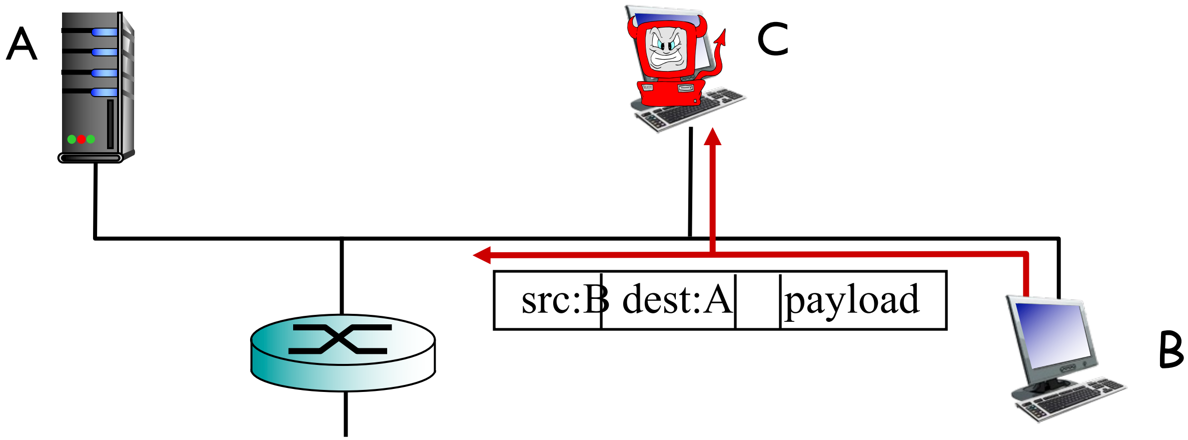
1. select target
2. break into hosts around the network (see botnet)
3. send packets to target from compromised hosts



Bad guys can sniff packets

Packet "sniffing":

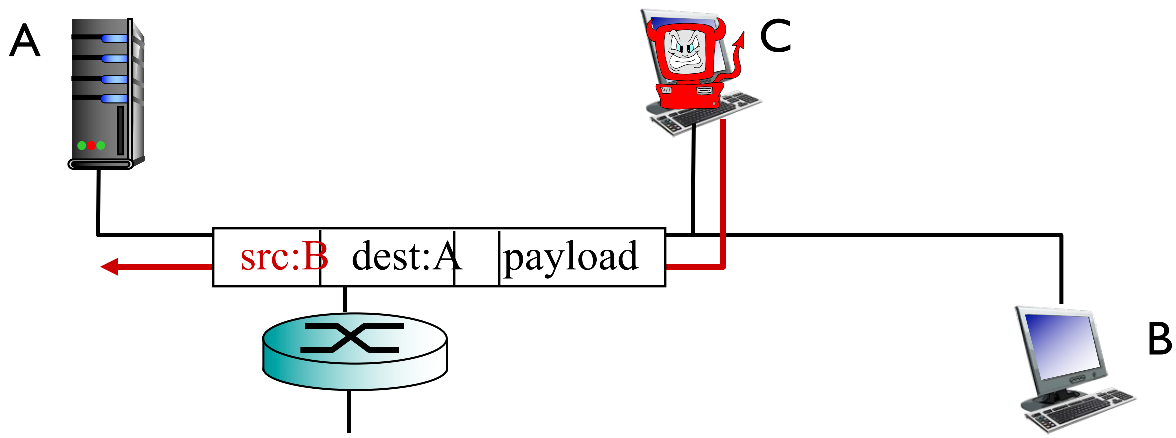
- Broadcast media (shared ethernet, wireless)
- Reads/records all packets (e.g., including passwords!) passing by
- They are difficult to detect



- wireshark software used for end-of-chapter labs is a (free) packet-sniffer

Bad guys can use fake addresses

IP spoofing: send packet with false source address



Lines of defense:

- authentication: proving you are who you say you are
  - cellular networks provides hardware identity via SIM card; no such hardware assist in traditional Internet
- confidentiality: via encryption
- integrity checks: digital signatures prevent/detect tampering
- access restrictions: password-protected VPNs
- firewalls: specialized "middleboxes" in access and core networks:
  - off-by-default: filter incoming packets to restrict senders, receivers, applications
  - detecting/reacting to DOS attacks

... lots more on security (throughout, Chapter 8)

Network under Attack

Chapter 1: roadmap

1.1 what is the Internet?

1.2 network edge

- end systems, access networks, links

1.3 network core

- packet switching, circuit switching, network structure

1.4 delay, loss, throughput in networks

1.5 protocol layers, service models

1.6 networks under attack: security

1.7 history

SUSTech Southern University of Science and Technology

Internet history

Introduction: summary

Covered a "ton" of material!

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

you now have:

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