CS 330: Network Applications & Protocols

Introduction to Computer Networks & the Internet

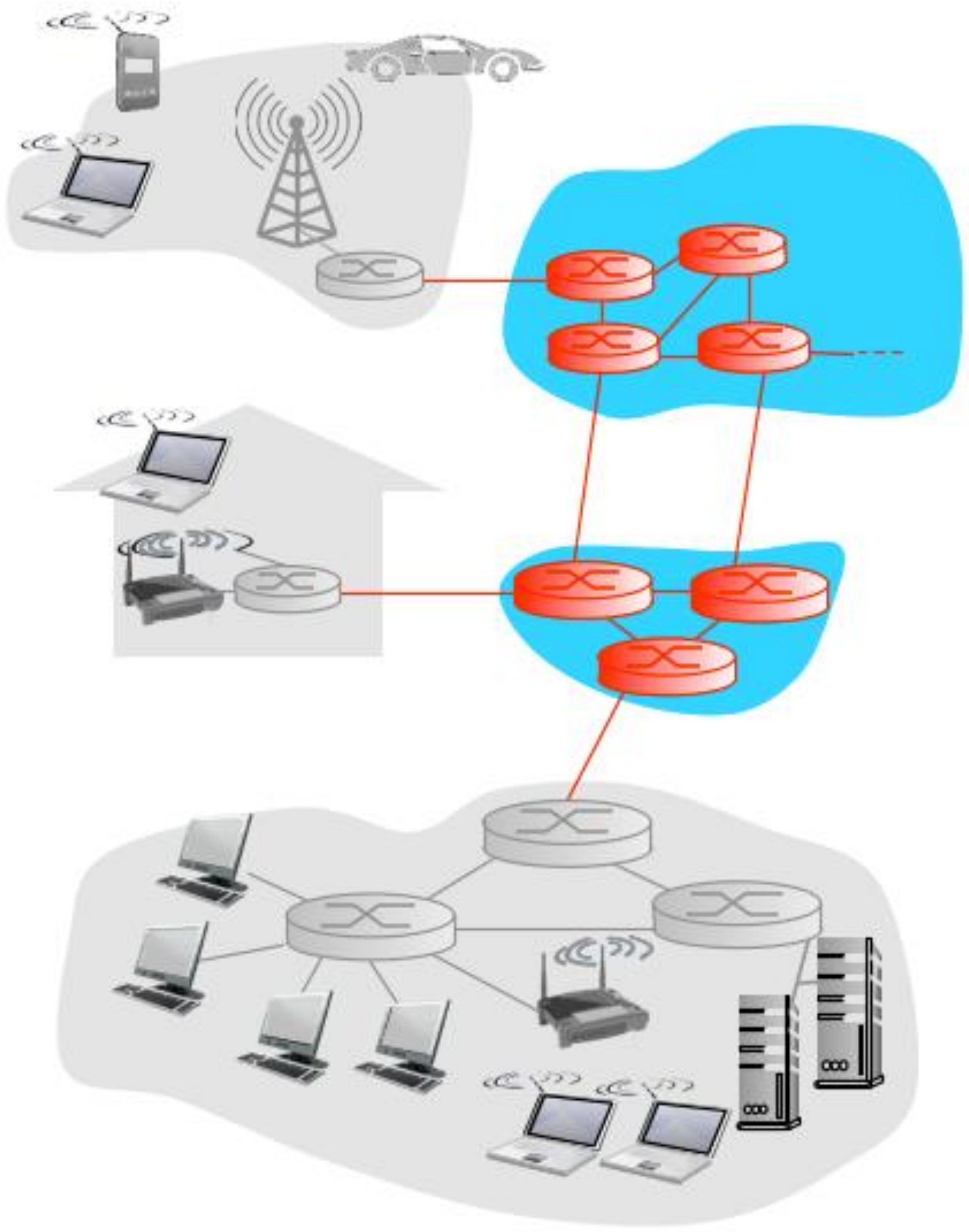
Galin Zhelezov
Department of Physical Sciences
York College of Pennsylvania

Introduction

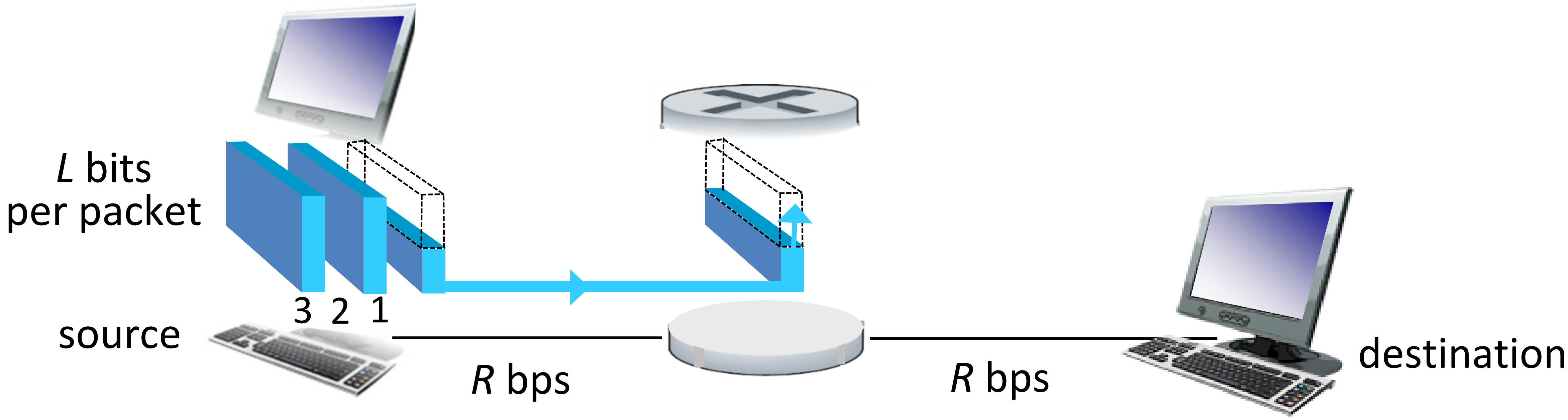
- **What is the Internet?**
- **Network edge**
  - End systems, access networks, links
- **Network core**
  - Packet switching, circuit switching, network structure
- **Delay, loss, throughput in networks**
- **Protocol layers, service models**
- **Networks under attack: security**
- **History**

The Network Core

- **Mesh of interconnected routers**
- **Packet-switching: hosts break application-layer messages into packets**
  - Forward packets from one router to the next, across links on path from source to destination
  - Each packet transmitted at full link capacity



Packet-switching: store-and-forward



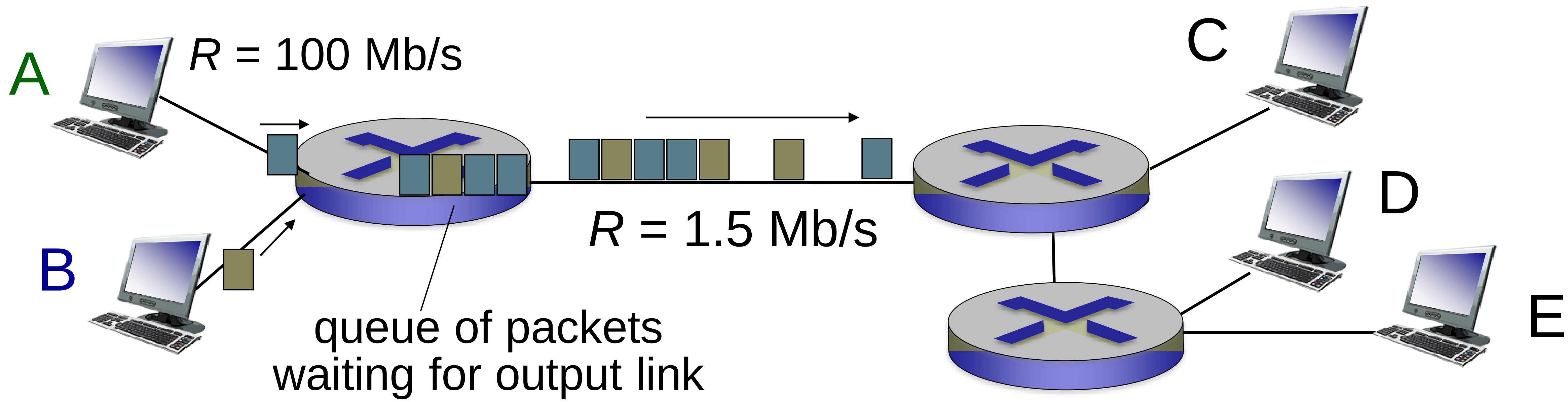
- **takes *L*/*R* seconds to transmit (push out) *L*-bit packet into link at *R* bps**
- ***store and forward:* entire packet must arrive at router before it can be transmitted on next link**
  - end-end delay = 2*L*/*R* (assuming zero propagation delay)

more on delay shortly …

one-hop numerical example:

- ***L* = 7.5 Mbits**
- ***R* = 1.5 Mbps**
- **one-hop transmission delay = 5 sec**

Packet Switching: queueing delay, loss



queuing and loss:

- 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
  - packets can be dropped (lost) if memory (buffer) fills up

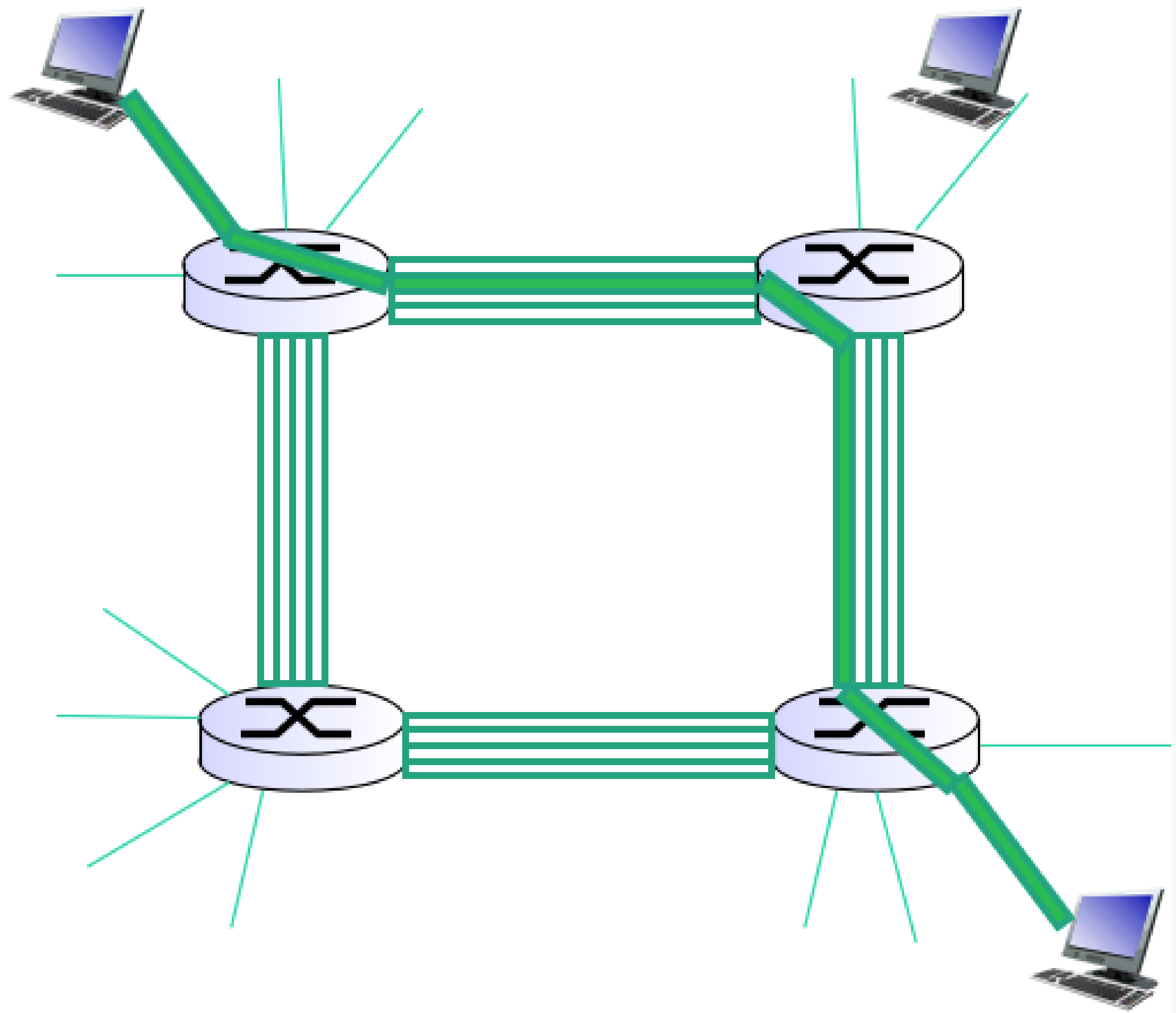
Two Key Functions of the Network-Core

- **Routing**
  - Determines source-destination route taken by packets
  - Utilizes routing algorithms

- **Forwarding**
  - Move packets from router’s input to appropriate router output

Alternative Core: Circuit Switching

- **End-to-end resources allocated to, reserved for “call” between source & dest:**
  - In diagram, each link has four circuits
    - Call gets 2nd circuit in top link and 1st circuit in right link
  - Dedicated resources (i.e. no sharing)
    - Circuit-like (guaranteed) performance
  - Circuit segment idle if not used by a call (no sharing)
  - Commonly used in traditional telephone networks

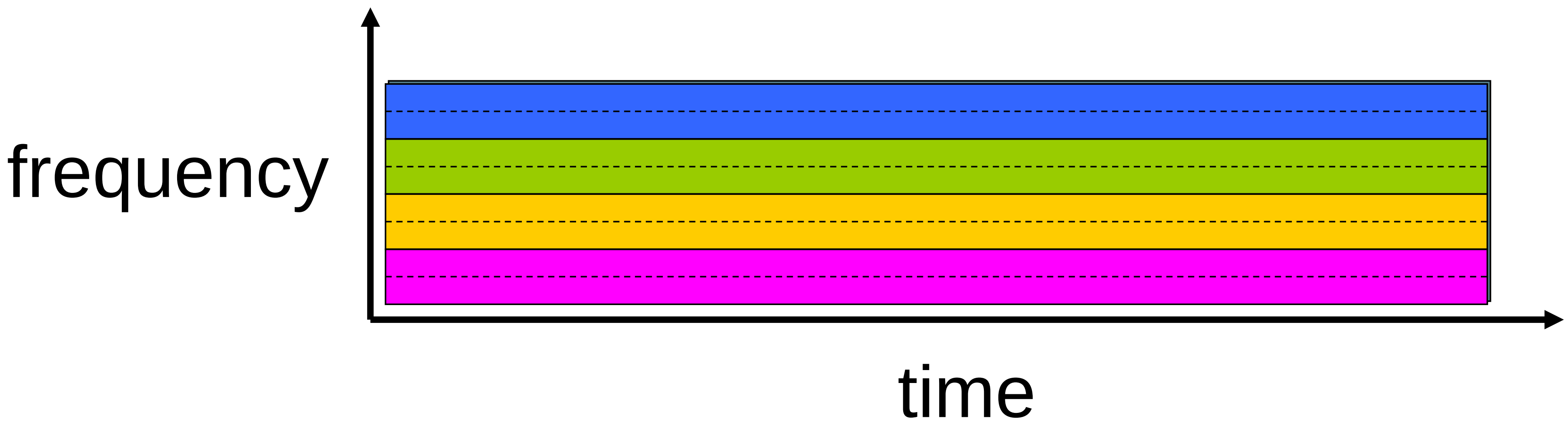


Circuit switching: FDM versus TDM

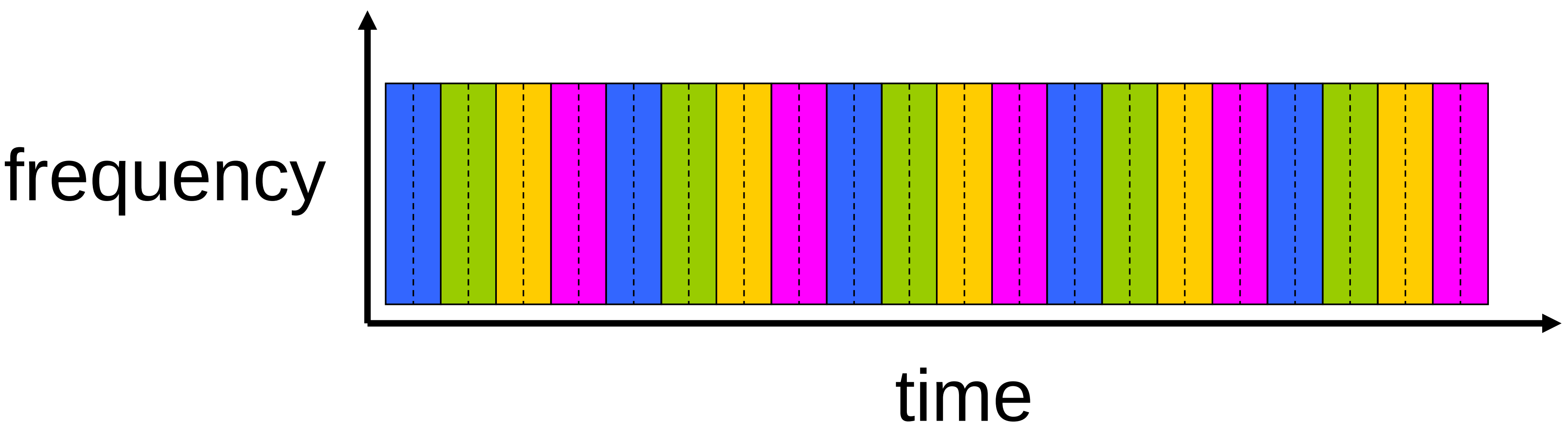
Example:

4 users

FDM



TDM

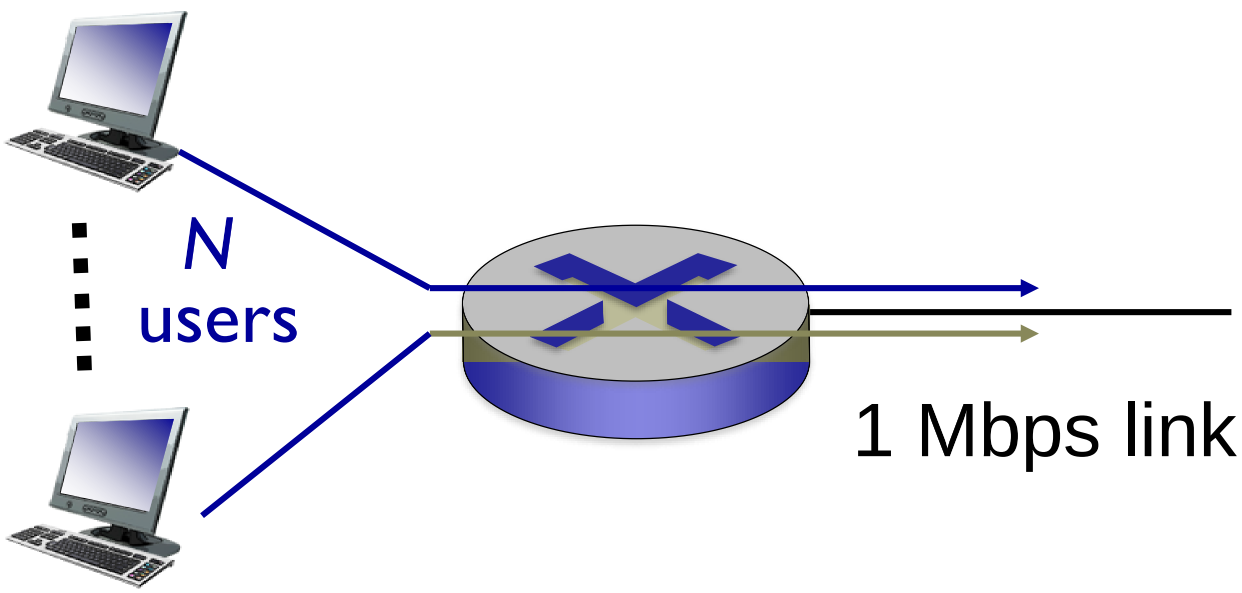


Packet switching versus circuit switching

packet switching allows more users to use network!

example:

- **1 Mb/s link**
- **each user:**
  - 100 kb/s when "active"
  - active 10% of time
- ***circuit-switching:***
  - 10 users
- ***packet switching:***
  - with 35 users, probability > 10 active at same time is less than .0004 *



Q: how did we get value 0.0004?

Q: what happens if > 35 users ?

Packet switching versus circuit switching

is packet switching a "slam dunk winner?"

- **great for bursty data**
  - resource sharing
  - simpler, no call setup
- **excessive congestion possible: packet delay and 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)

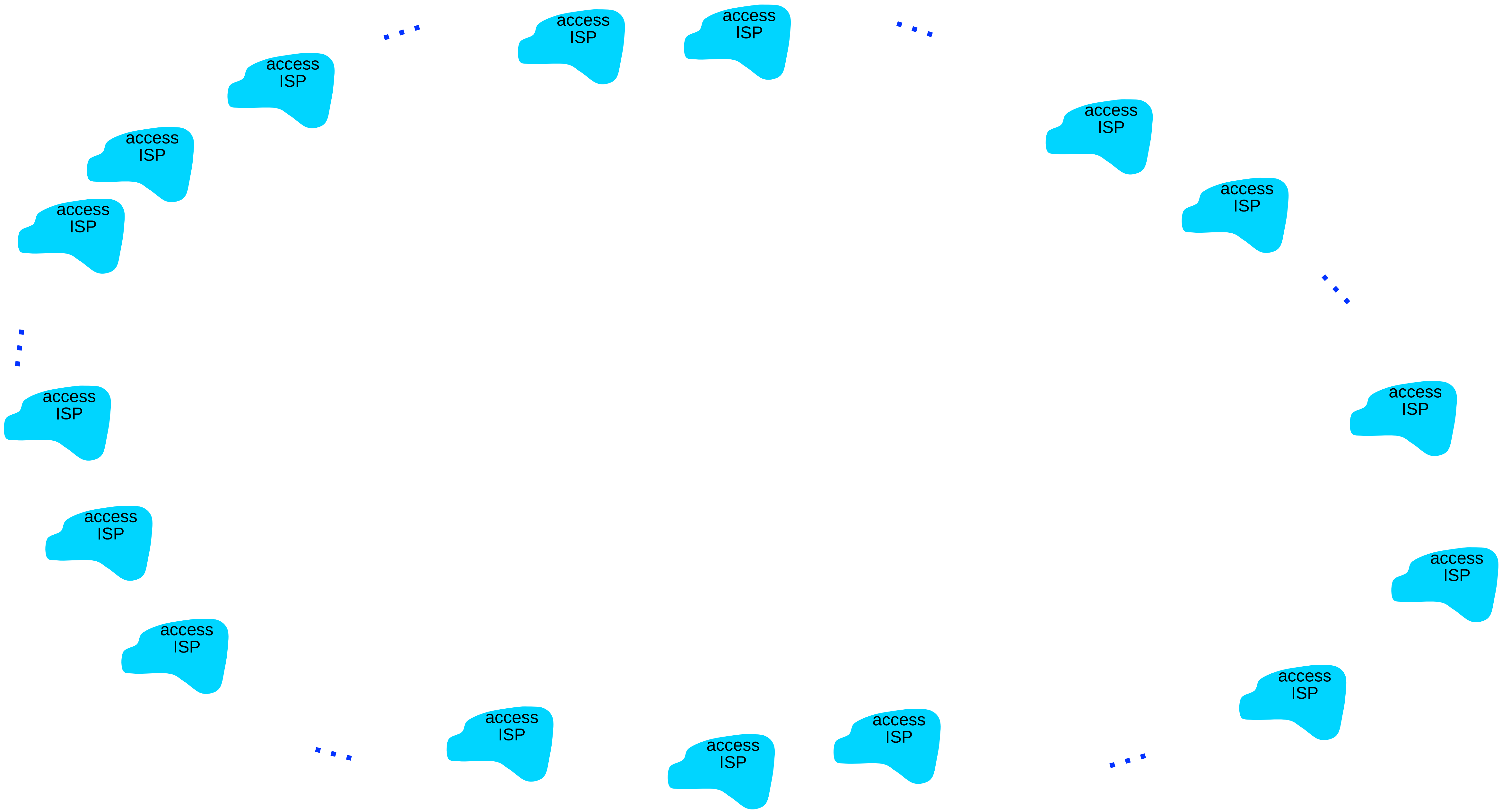
Q: Human analogies of reserved resources (circuit switching) versus on-demand allocation (packet-switching)?

Internet structure: network of networks

- End systems connect to Internet via access ISPs (Internet Service Providers)
  - residential, company and university 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

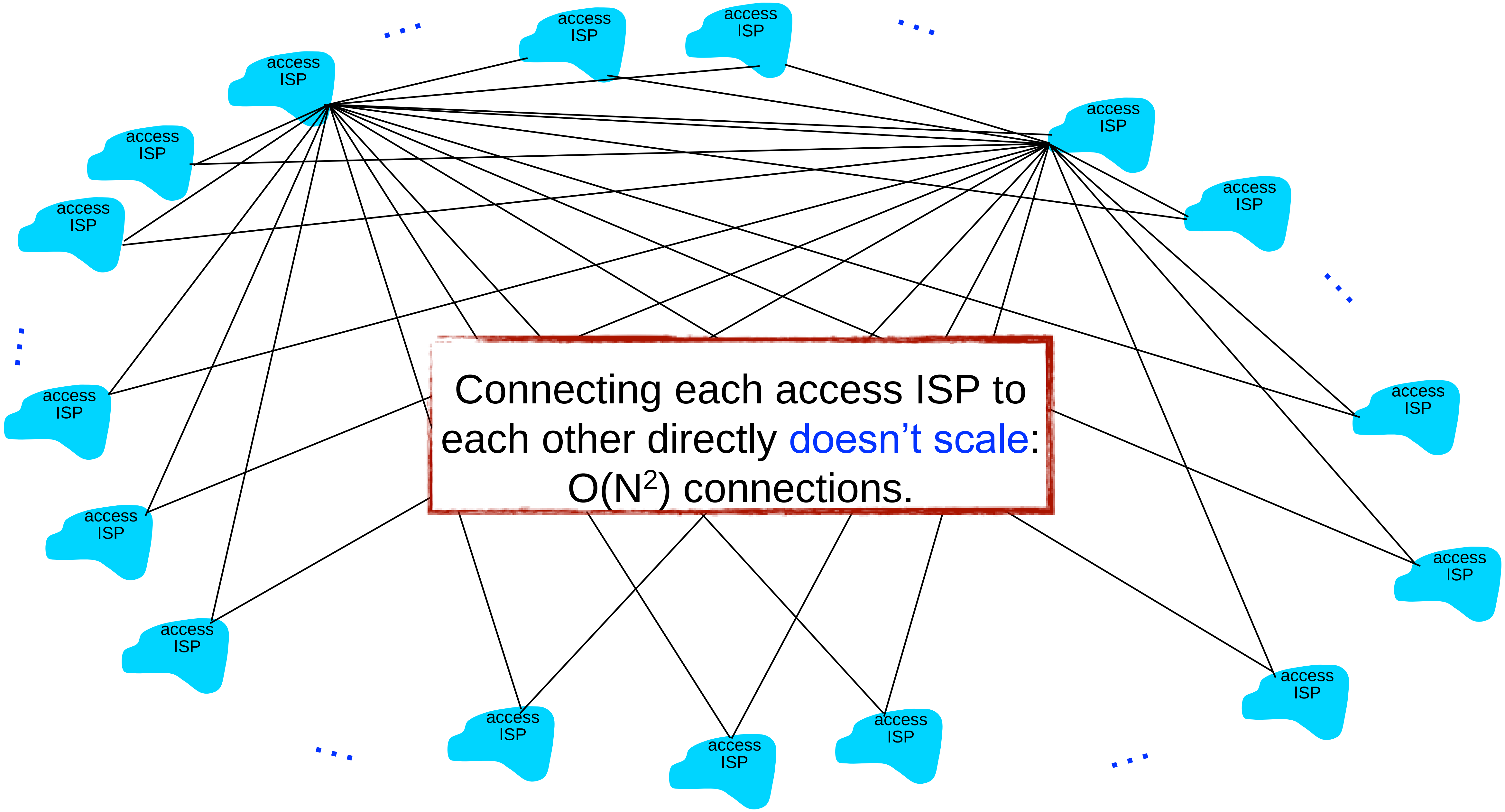
Internet Structure: Network of Networks

- **Question:** **Given millions of access ISPs, how should they be connected?**



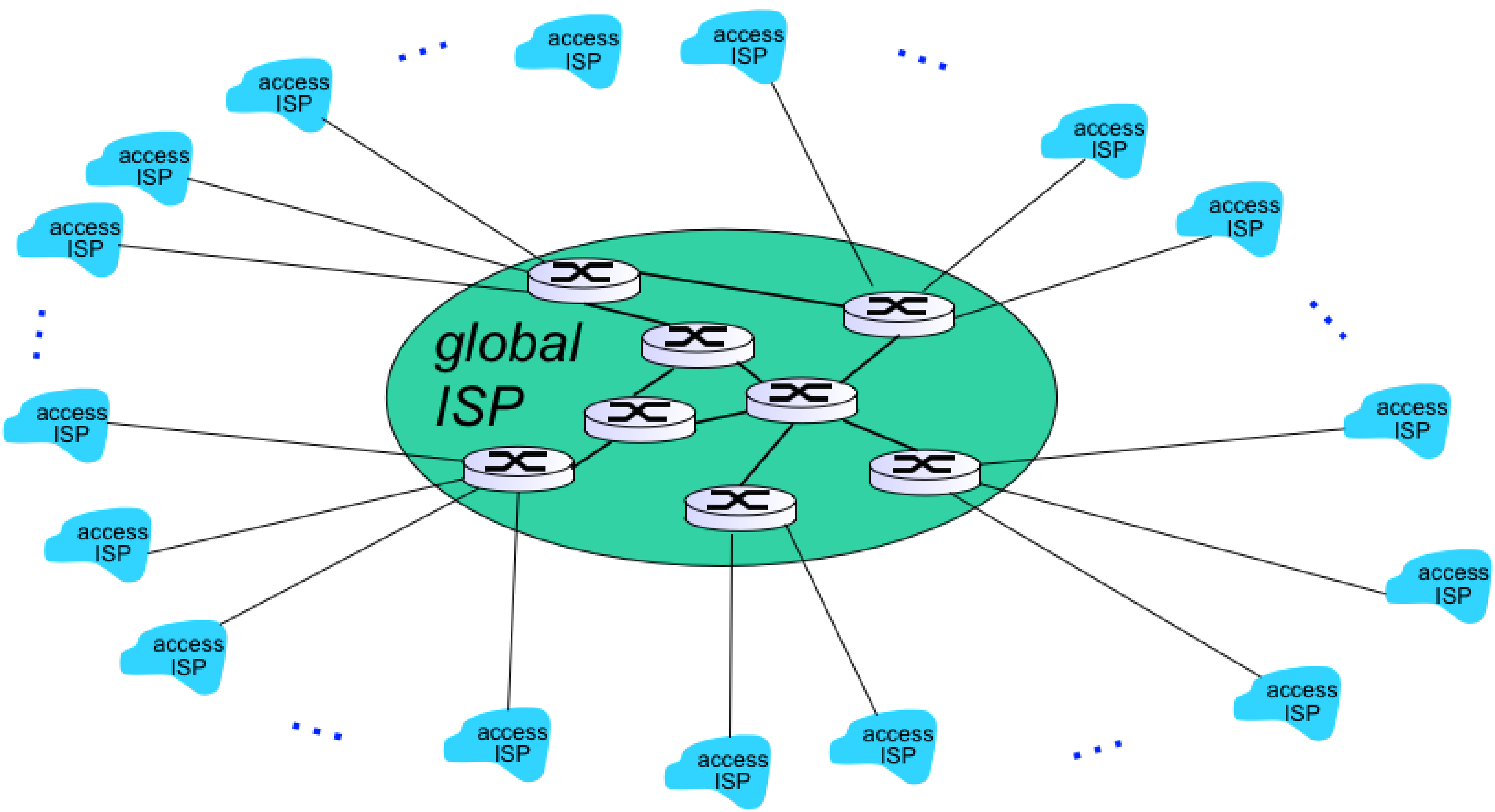
Internet Structure: Network of Networks

- **Question: Given millions of access ISPs, how should they be connected?**



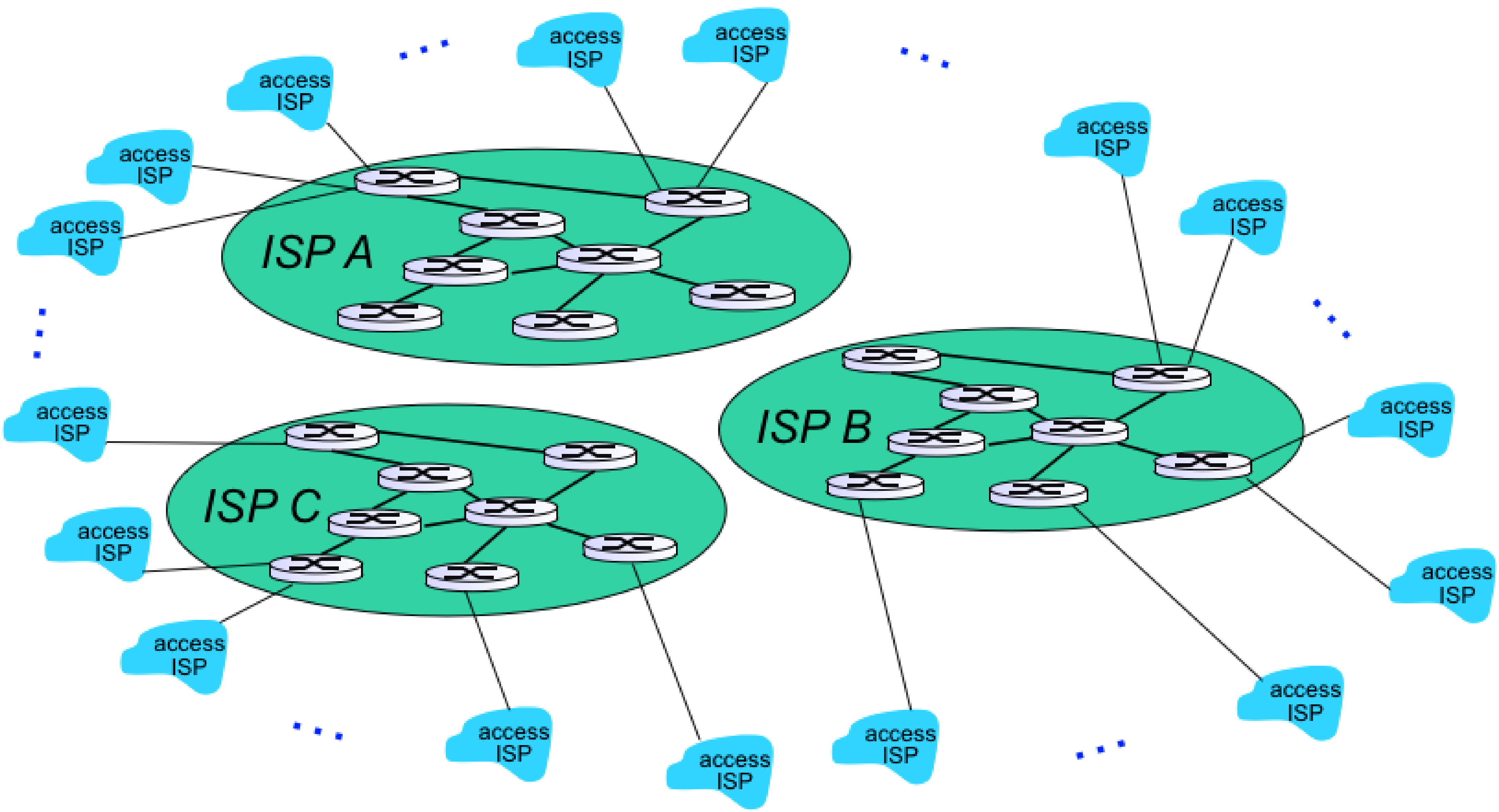
Internet Structure: Network of Networks

- **Option:** **Connect each access ISP to a global transit ISP? Customer and provider ISPs have economic agreement.**



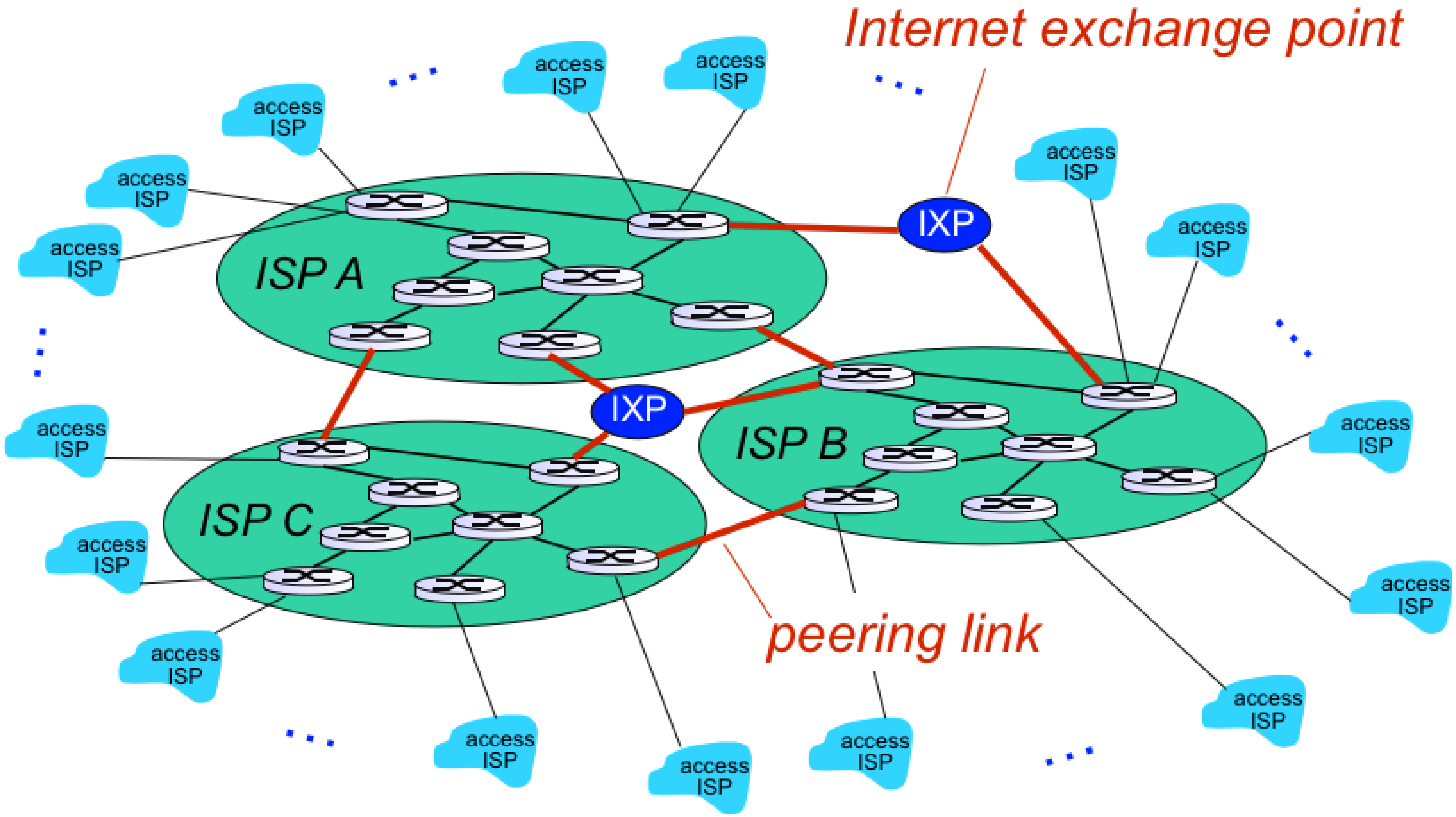
Internet Structure: Network of Networks

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



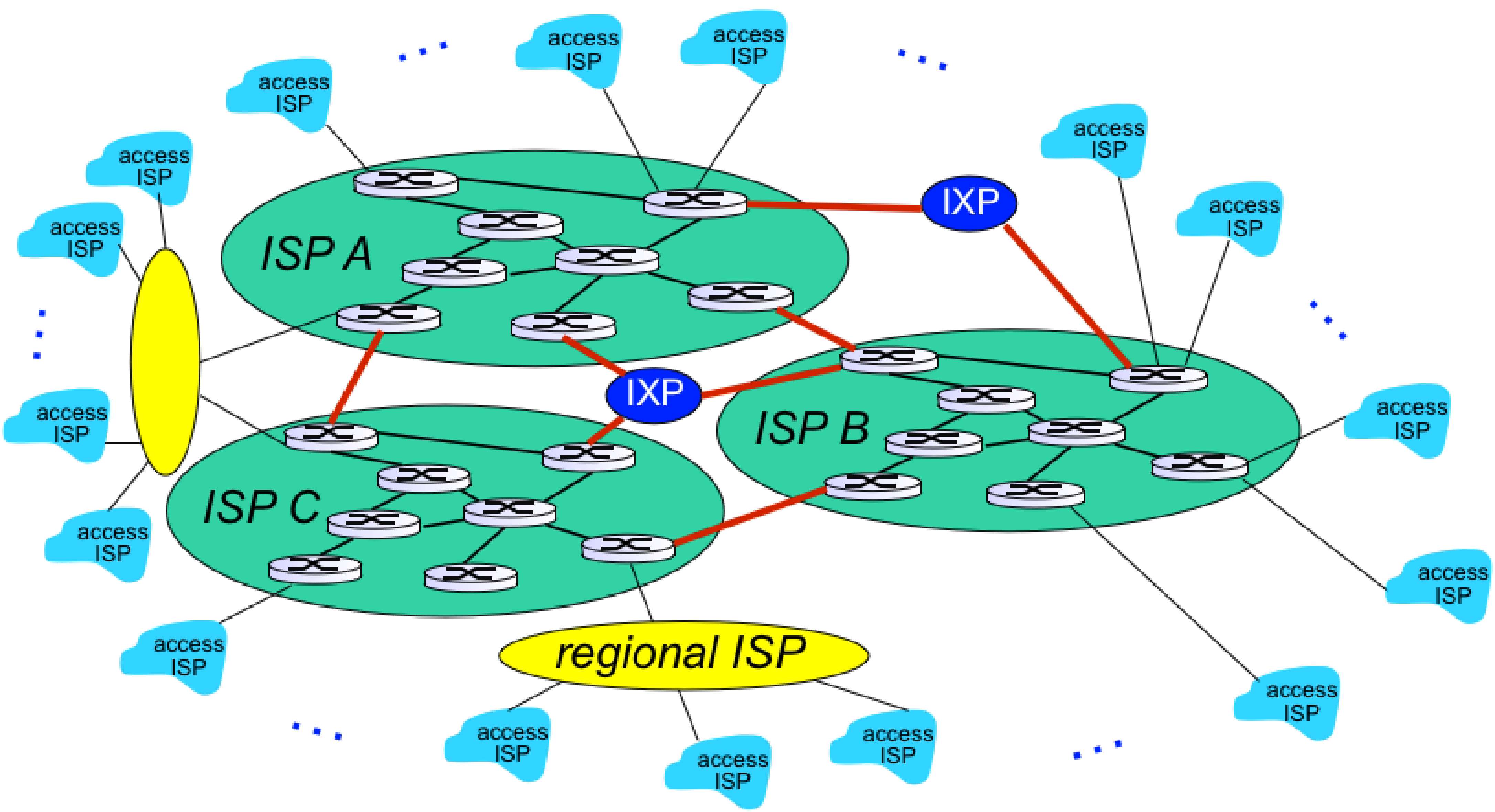
Internet Structure: Network of Networks

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



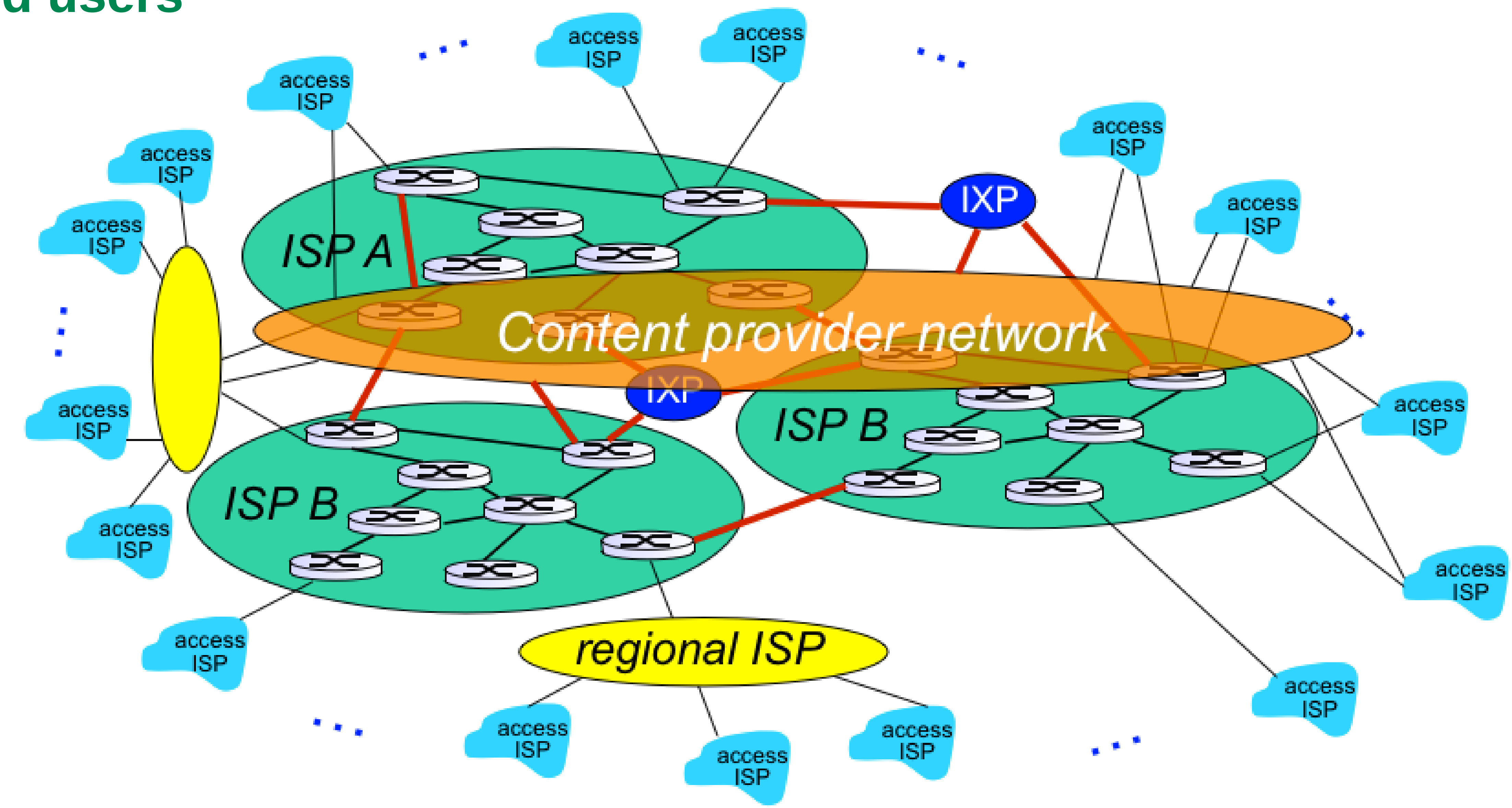
Internet Structure: Network of Networks

- … and regional networks may arise to connect access ISPs to ISPs

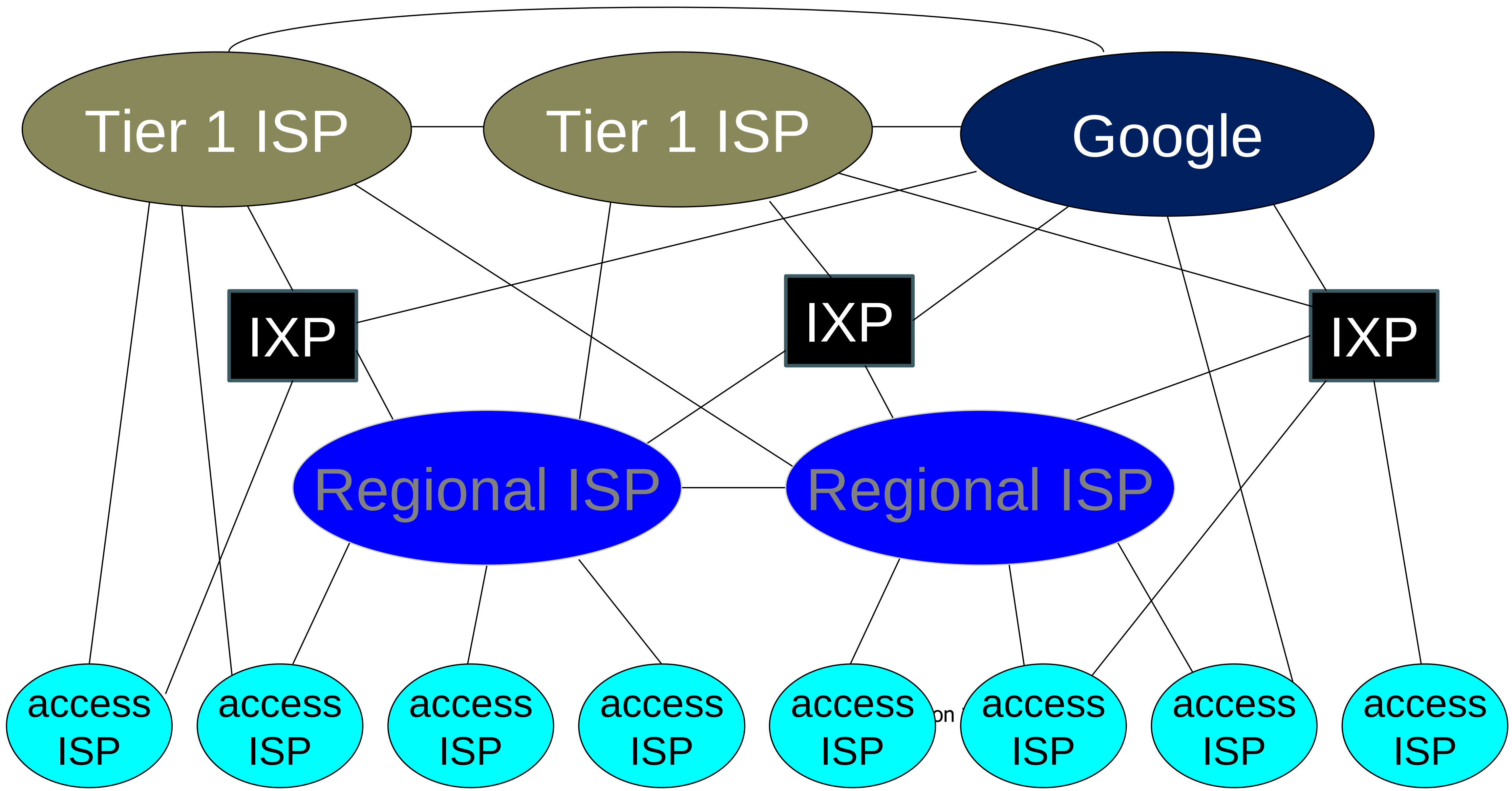


Internet Structure: Network of Networks

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



Internet structure: network of networks



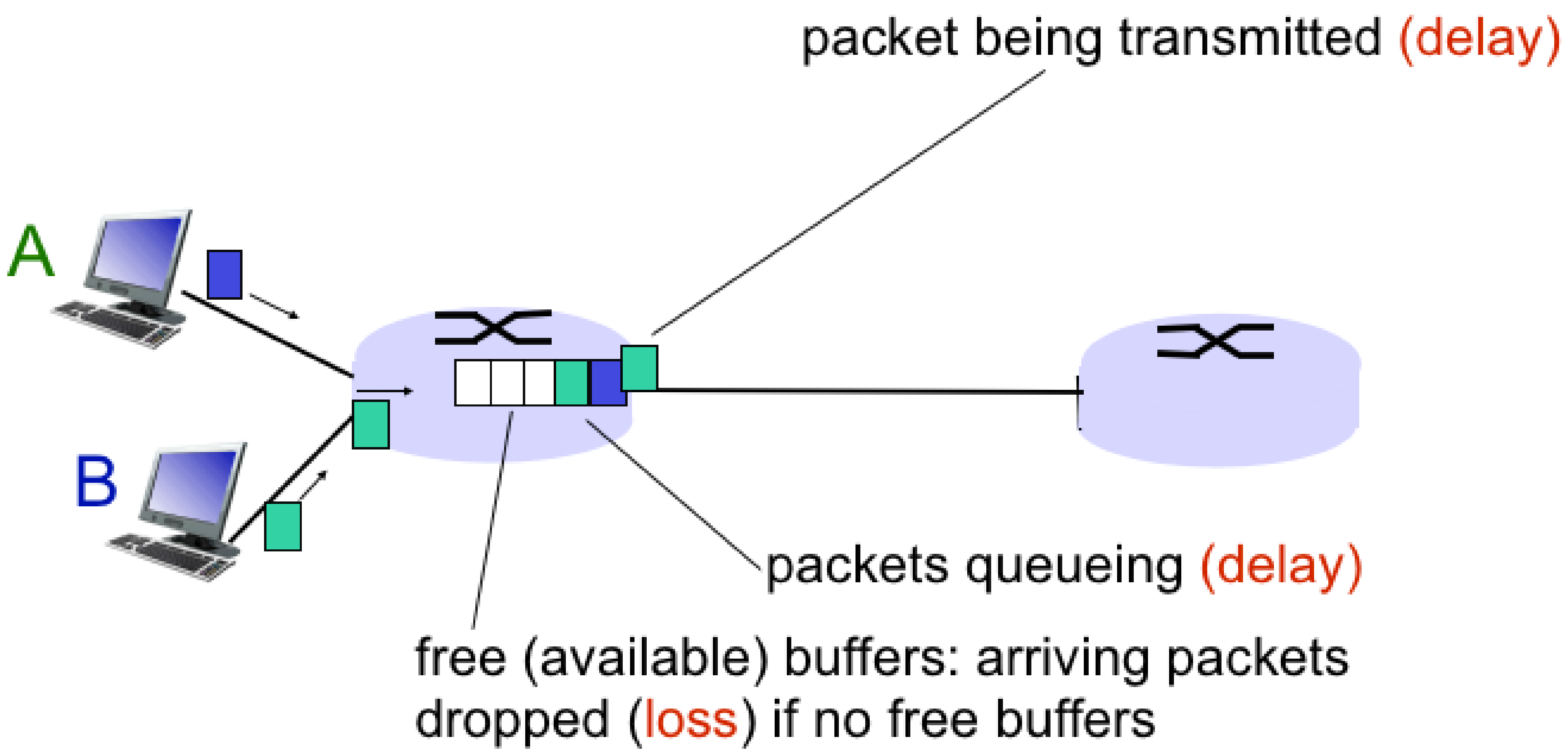
- **Center: small # of well-connected large networks**
  - "tier-1" commercial ISPs (e.g., Level 3, Sprint, AT&T, NTT), national & international coverage
  - content provider network (e.g., Google): private network that connects it data centers to Internet, often bypassing tier-1, regional ISPs

Introduction

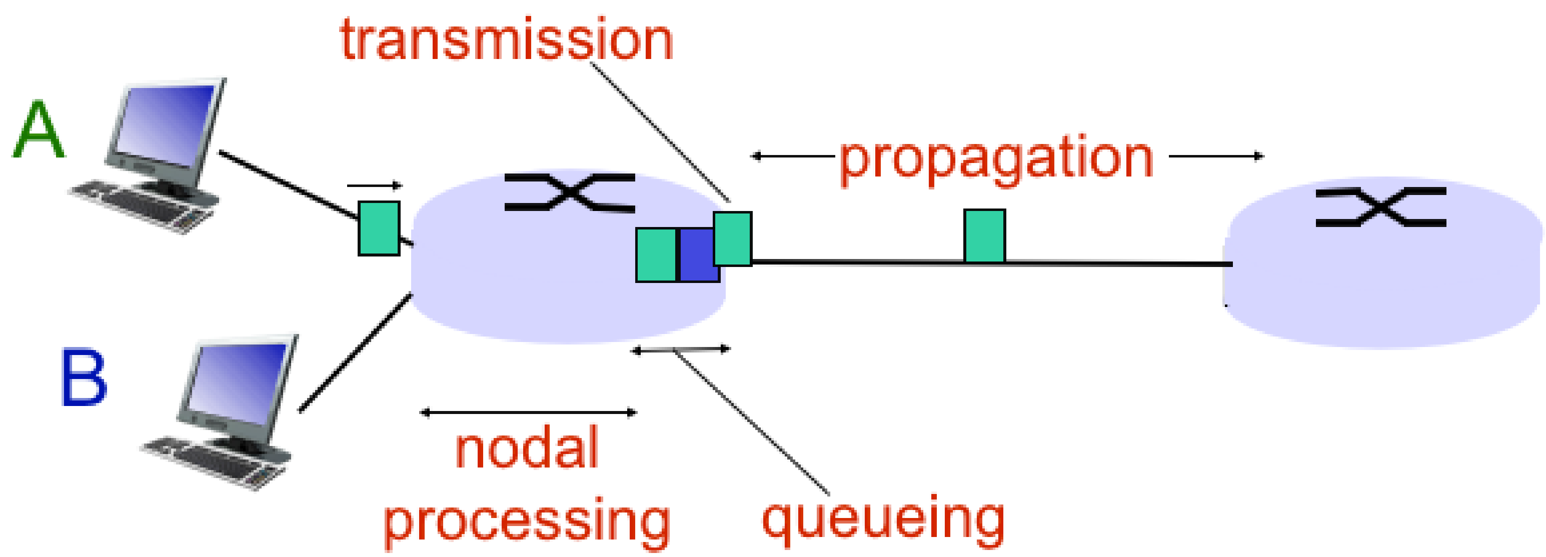
- **What is the Internet?**
- **Network edge**
  - End systems, access networks, links
- **Network core**
  - Packet switching, circuit switching, network structure
- **Delay, loss, throughput in networks**
- **Protocol layers, service models**
- **Networks under attack: security**
- **History**

How Does Packet Delay Occur?

- **Packets queue in router buffers**
  - Packet arrival rate to link (temporarily) exceeds output link capacity
  - Packets queue, wait for turn (i.e. they are delayed)



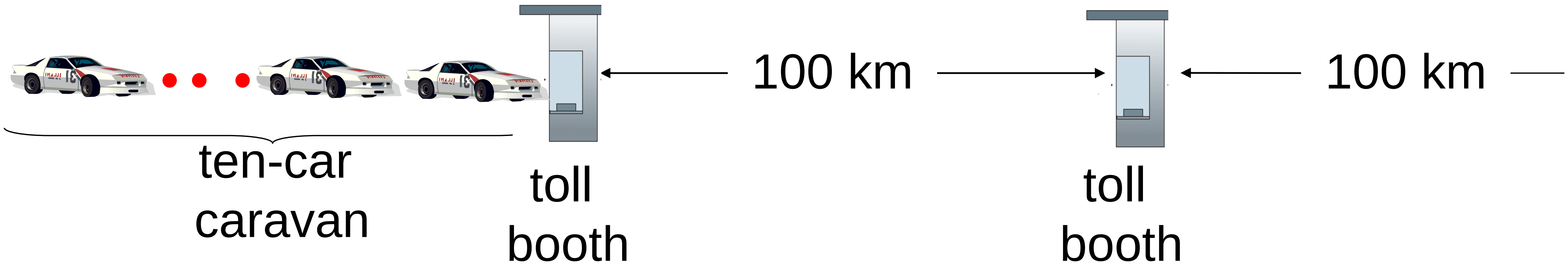
Four Sources of Packet Delay



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

- d_{proc}: **Nodal processing delay** - **Check for bit errors, determine output link**
- d_{queue}: **Queueing delay** - **Time waiting for output link**
- d_{trans}: **Transmission delay** - **(packet length) / (link bandwidth)**
- d_{prop}: **Propagation delay** - **(length of physical link) / (propagation speed)**

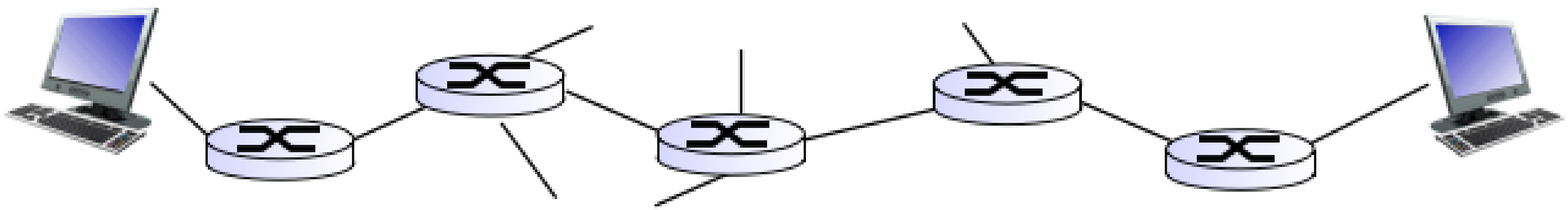
Caravan analogy



- **cars "propagate" at 100 km/hr**
- **toll booth takes 12 sec to service car (bit transmission time)**
- **car ~ bit; caravan ~ packet**
- ***Q:* How long until caravan is lined up before 2nd toll booth?**
- **time to "push" entire caravan through toll booth onto highway = 12 * 10 = 120 sec**
- **time for last car to propagate from 1st to 2nd toll both: 100km / (100km/hr) = 1 hr**
- ***A:* 62 minutes**

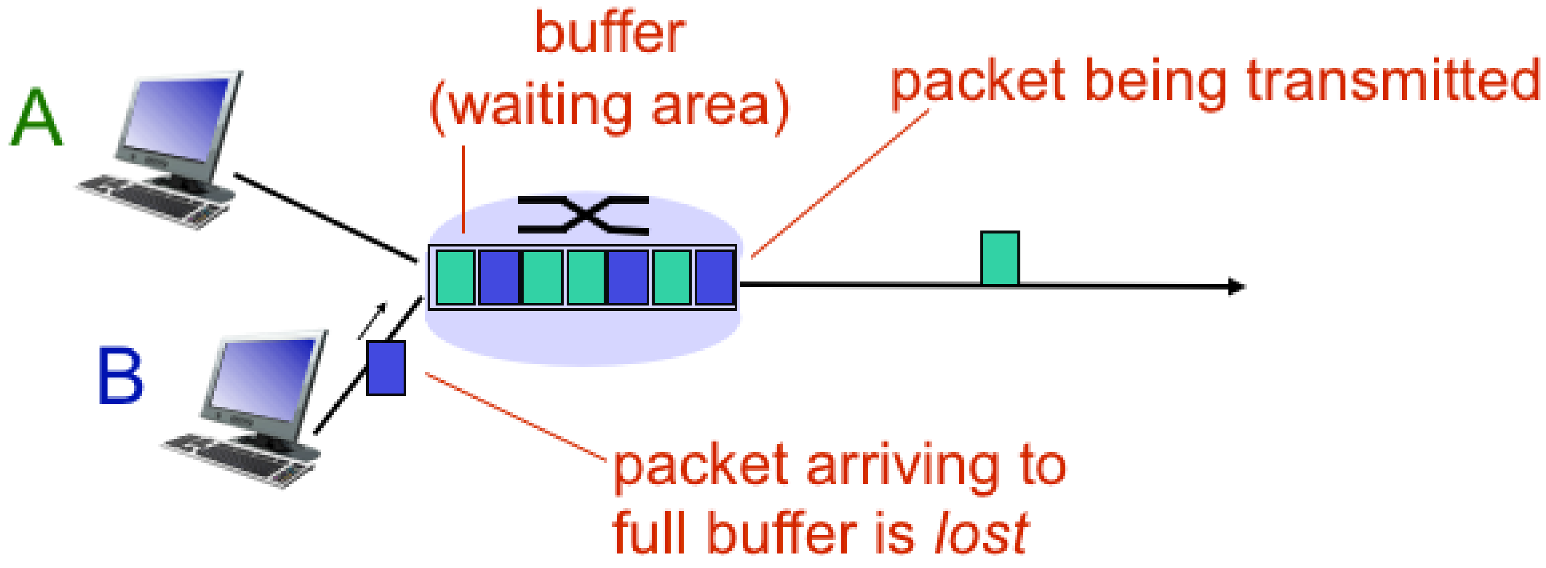
“Real” Internet Delays and Routes

- **What do “real” Internet delay & loss look like?**
- **Use `traceroute` program**
  - Provides delay measurement from source to router along end-to-end Internet path towards destination
  - For all `i`:
    - Sends 3 packets that will reach router `i` on path towards destination
    - Router `i` will return packets to sender
    - Sender times interval between transmission and reply



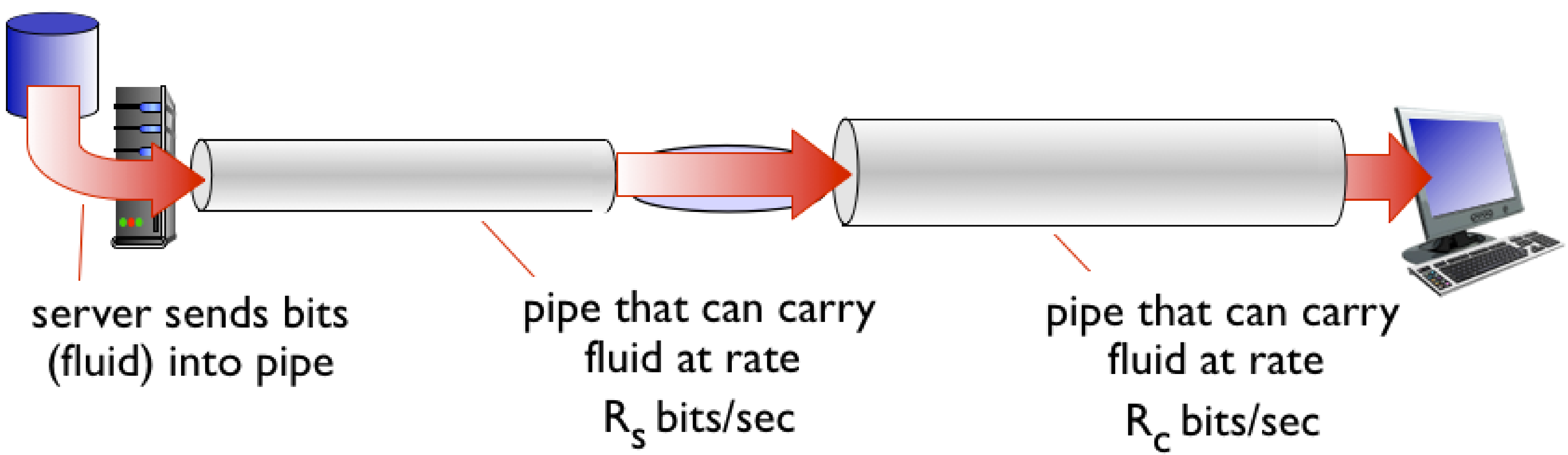
How Does Packet Loss Occur?

- **Packet queue preceding link has finite capacity**
- **Packets arriving to a full queue are dropped**
- **Lost packets may be retransmitted by previous node, by original source, or not at all**



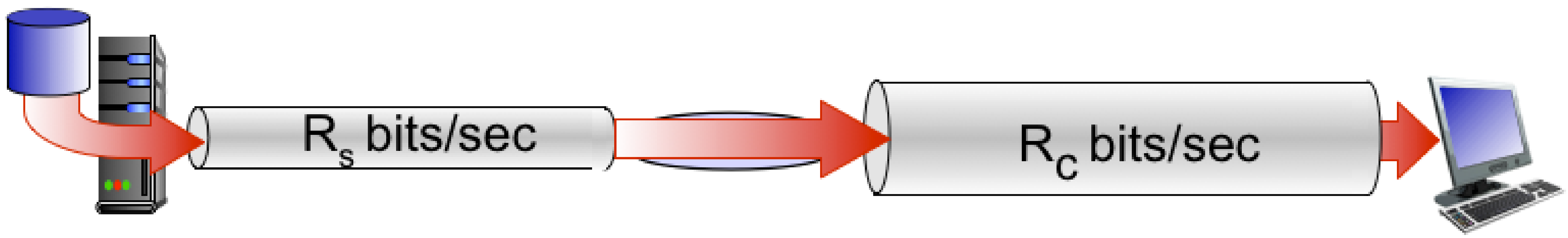
Throughput

- **Throughput: the rate (bits/time unit) at which bits can be transferred between sender/receiver**
  - Instantaneous throughput: rate at given point in time
  - Average throughput: rate over longer period of time

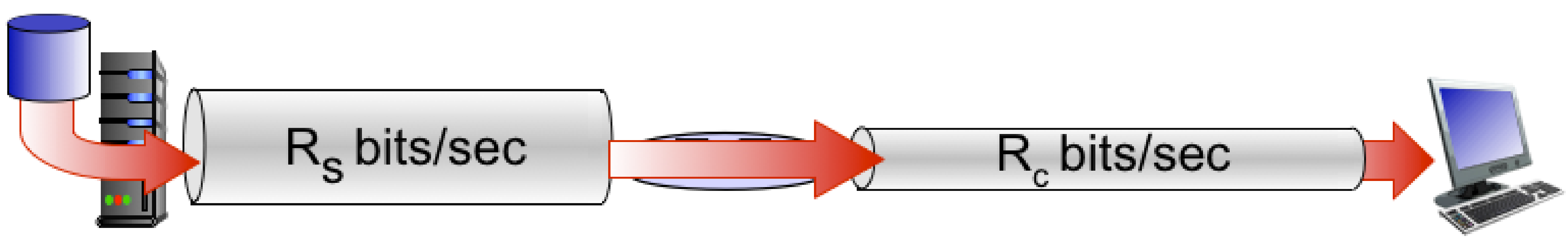


Throughput (Cont.)

- **$R_s < R_c$: What is average end-to-end throughput?**



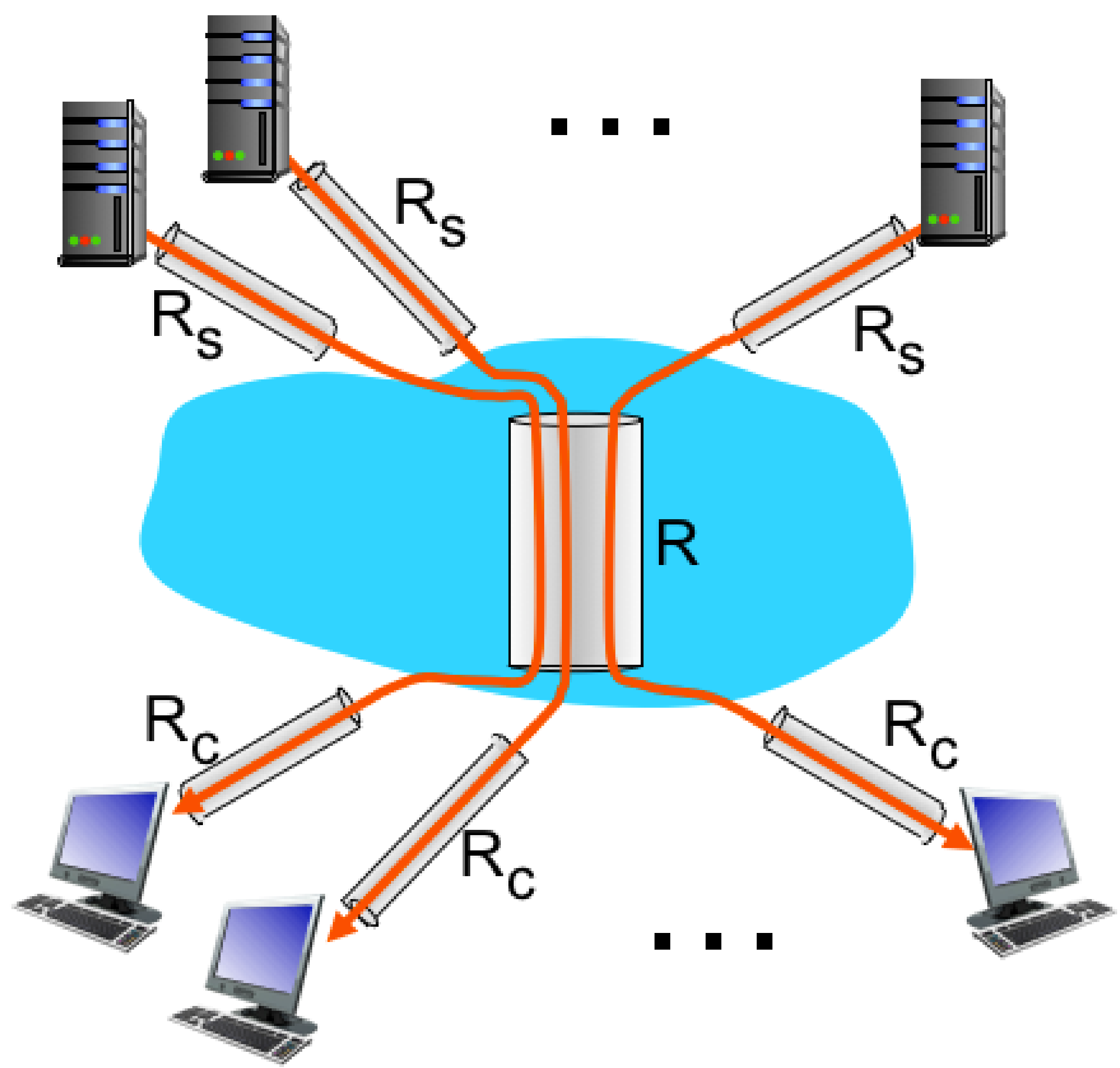
- **$R_s > R_c$: What is average end-to-end throughput?**



Bottleneck Link: A link on an end-to-end path that constrains the end-end throughput

Throughput: Internet Scenario

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



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

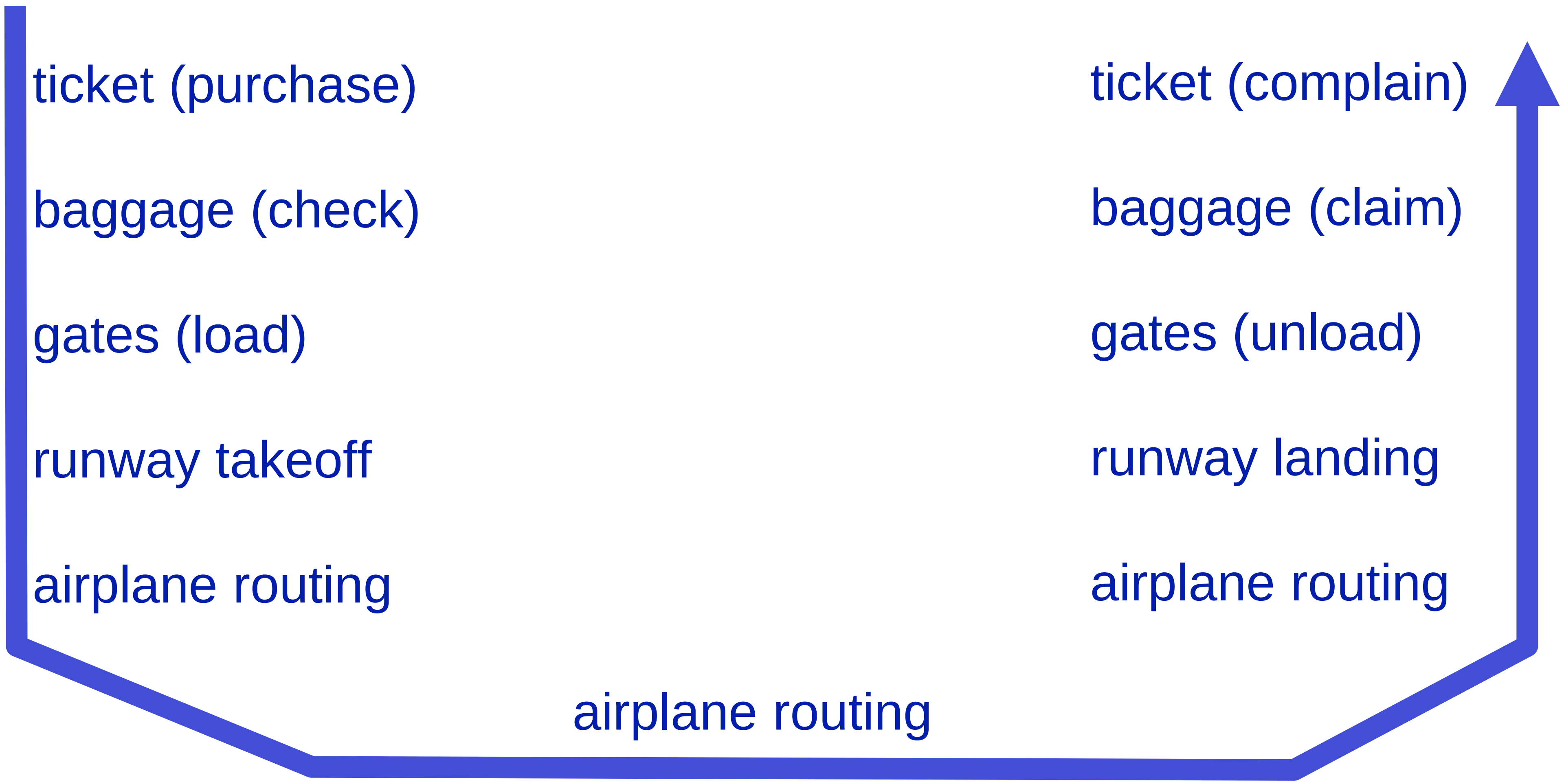
Introduction

- **What is the Internet?**
- **Network edge**
  - End systems, access networks, links
- **Network core**
  - Packet switching, circuit switching, network structure
- **Delay, loss, throughput in networks**
- **Protocol layers, service models**
- **Networks under attack: security**
- **History**

Protocol “Layers”

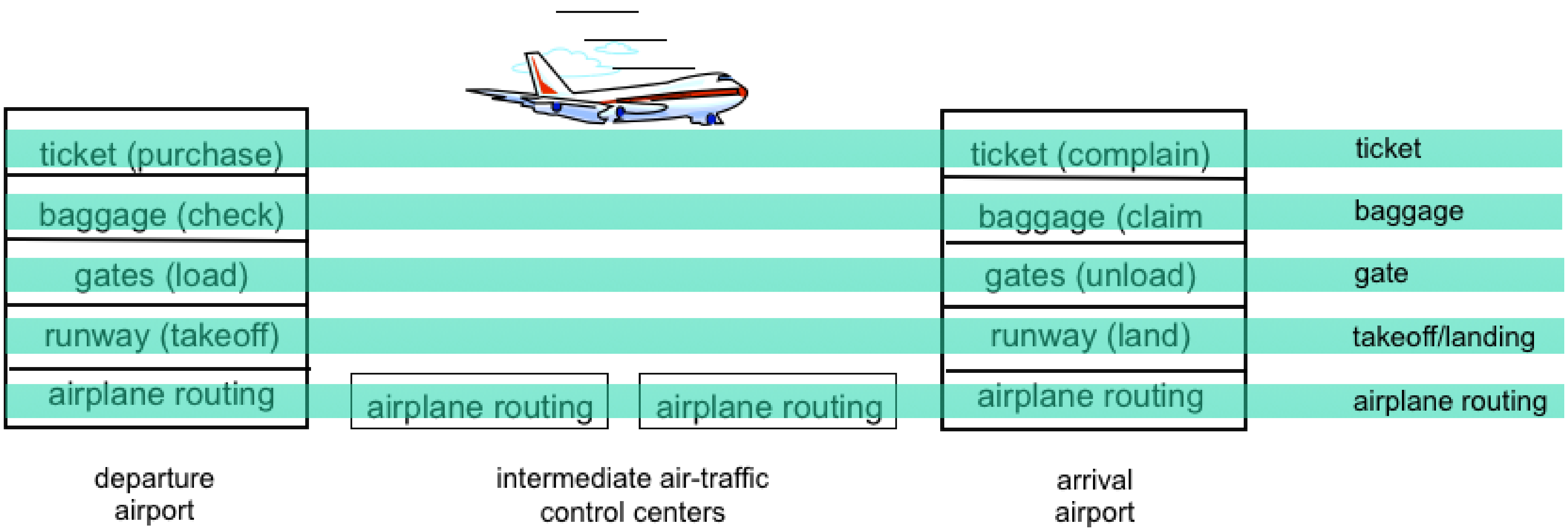
- **Networks are complex, with many components:**
  - Hosts
  - Routers
  - Links of various media
  - Applications
  - Protocols
  - Hardware, software
- **Need some way to organize everything**

Example: Organization of Air Travel



- **Each layer implements a service**
  - Uses its own internal-layer actions
  - Relies on services provided by layer below

Example: Organization of Air Travel



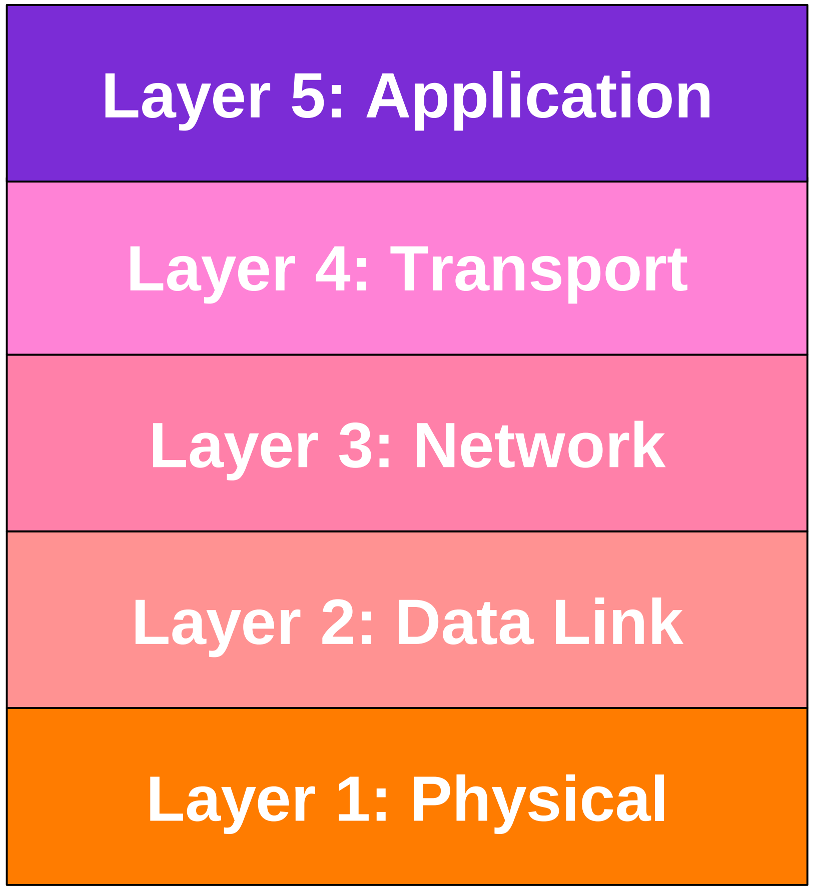
- **Each layer implements a service**
  - Uses its own internal-layer actions
  - Relies on services provided by layer below

Why Layering?

- **Systems can be complex**
- **Explicit structure allows clear identification and relationship of complex system components**
  - Layered reference model for discussion
- **Modularization eases maintenance, updating of system**
- **Changing implementation of a single layer's service is transparent to rest of system**
  - e.g. change in gate procedure doesn't affect rest of system

Five-Layer Internet Protocol Stack (TCP/IP Model)

- **Application: supporting network applications**
  - FTP, SMTP, HTTP
- **Transport: process-to-process data transfer**
  - TCP, UDP
- **Network: routing of datagrams from source to destination**
  - IP, routing protocols
- **Link: data transfer between neighboring network elements**
  - Ethernet, 802.11 (WiFi), PPP
- **Physical: bits on the wire**

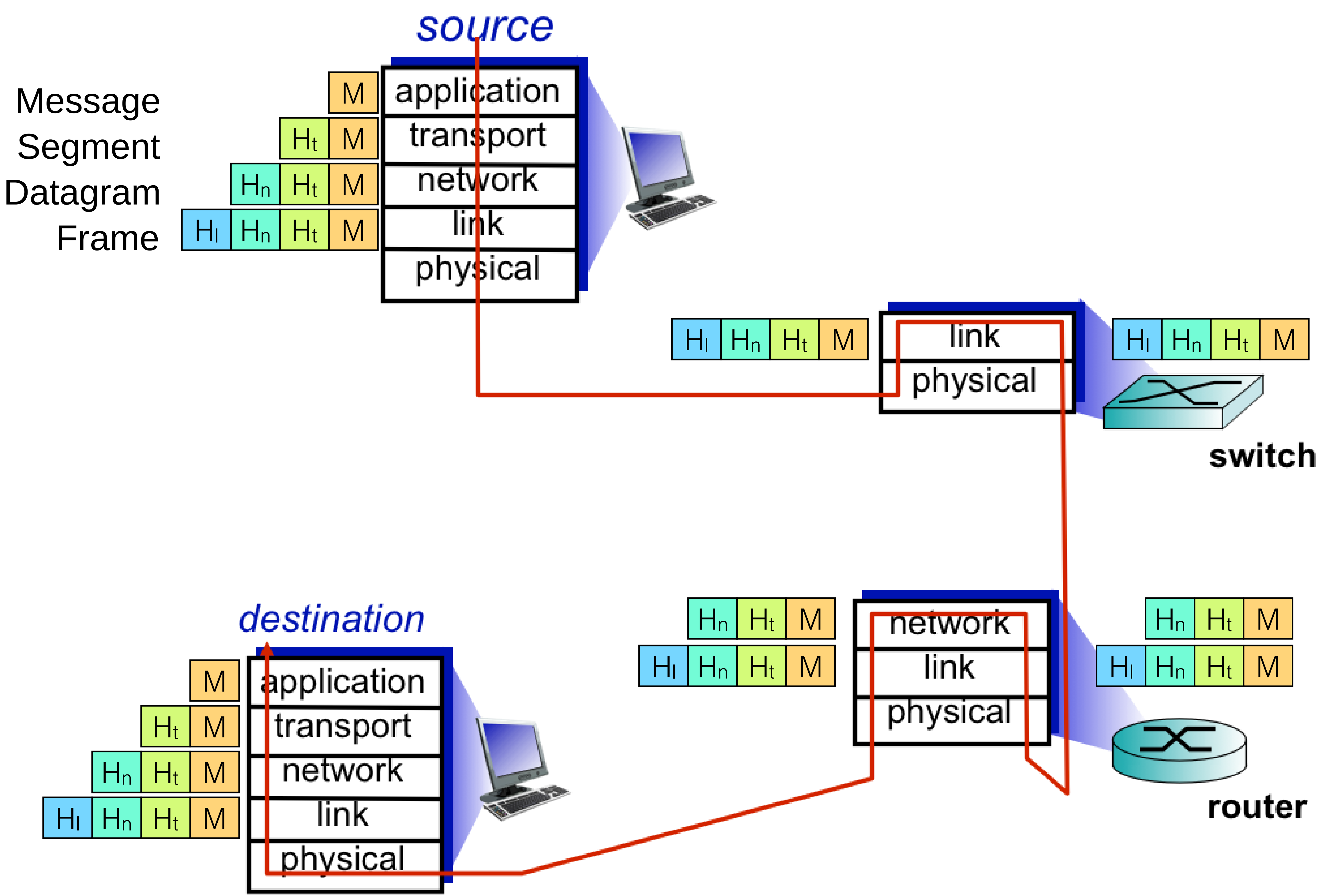


Seven-Layer ISO/OSI Network Model

- **Presentation: allows applications to interpret meaning of data**
  - e.g. encryption, compression, machine-specific conventions
- **Session: synchronization, checkpointing, recovery of data exchange**
- **Internet stack is missing these layers**
- **These services, if needed, must be implemented in the Application Layer**

Layer 7: Application
Layer 6: Presentation
Layer 5: Session
Layer 4: Transport
Layer 3: Network
Layer 2: Data Link
Layer 1: Physical

Encapsulation



Introduction

- What is the Internet?
- Network edge
  - End systems, access networks, links
- Network core
  - Packet switching, circuit switching, network structure
- Delay, loss, throughput in networks
- Protocol layers, service models
- **Networks under attack: security**
- **History**

Network security

- **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!

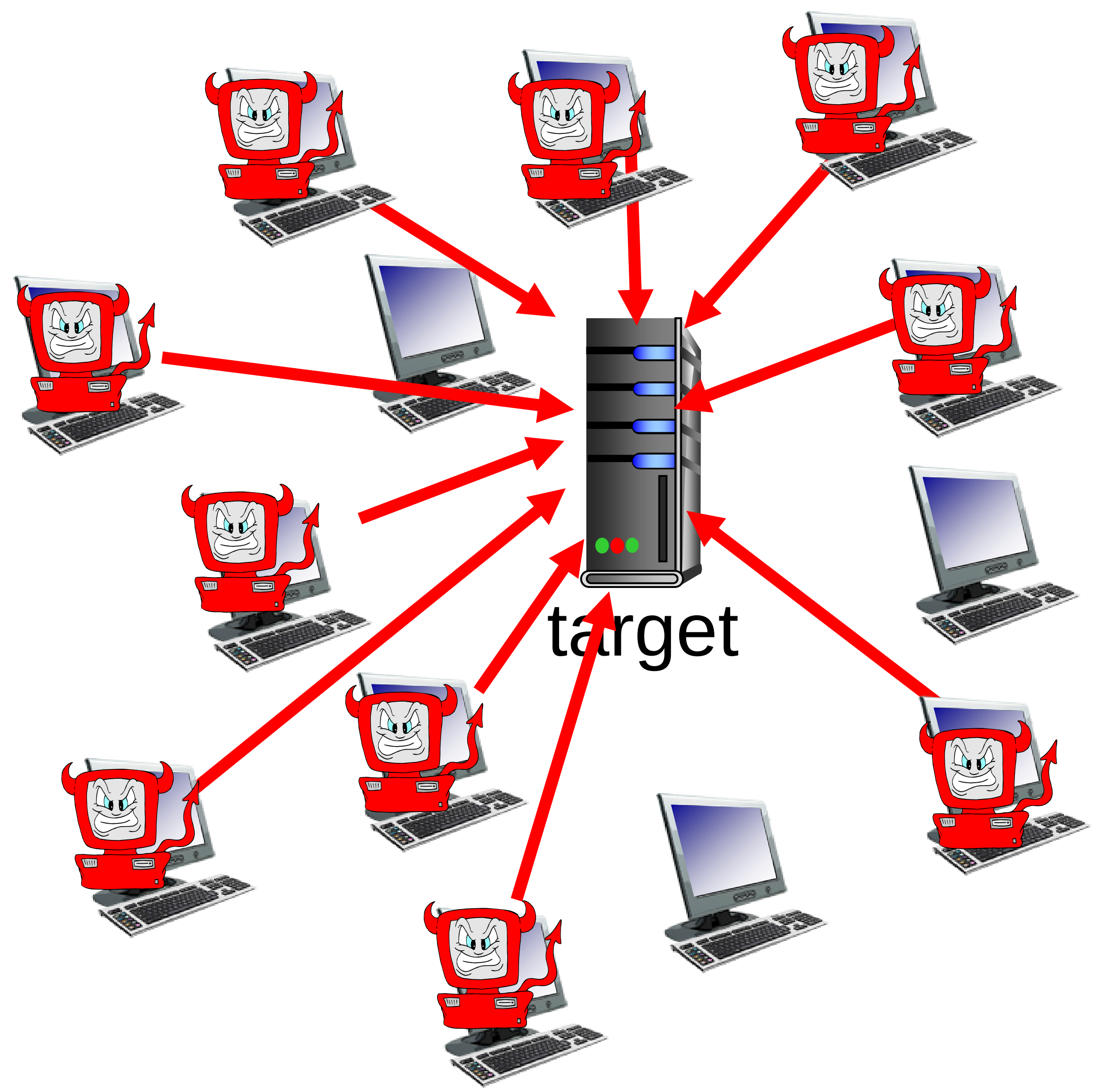
Bad guys: put malware into hosts via Internet

- **malware can get in host from:**
  - *virus:* self-replicating infection by receiving/executing object (e.g., e-mail attachment)
  - *worm:* self-replicating infection by passively receiving object that gets itself executed
- **spyware malware can record keystrokes, web sites visited, upload info to collection site**
- **infected host can be enrolled in botnet, used for spam. DDoS attacks**

Bad guys: attack server, network infrastructure

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

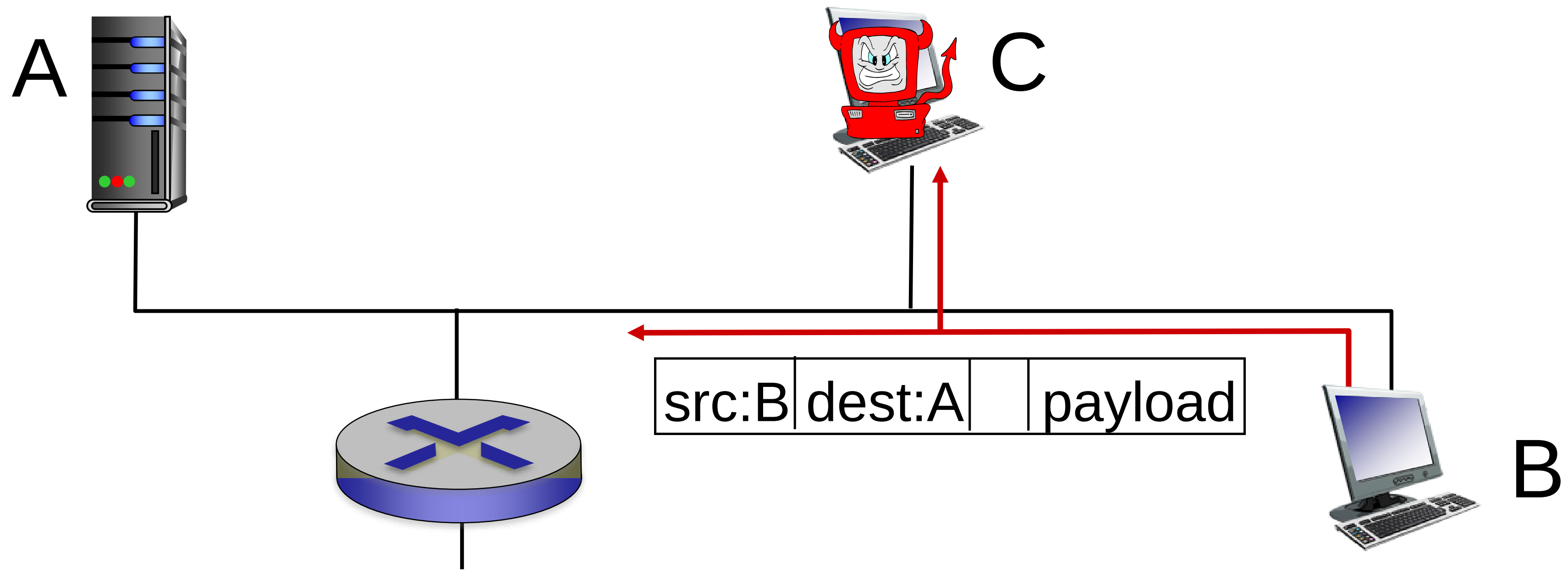
1. select target
2. break into hosts around the network (see botnet)
3. send packets to target from compromised hosts



Bad guys can sniff packets

packet "sniffing":

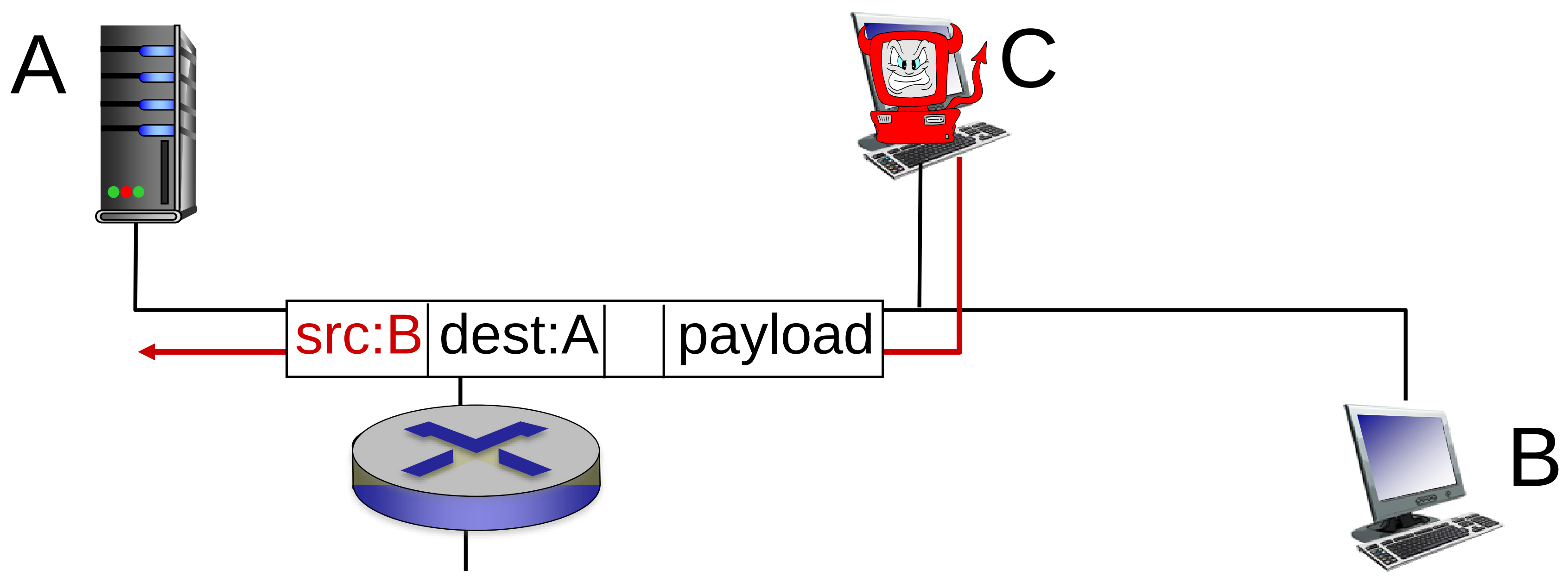
- broadcast media (shared Ethernet, wireless)
- promiscuous network interface reads/records all packets (e.g., including passwords!) passing by



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**



… lots more on security (throughout, Chapter 8)

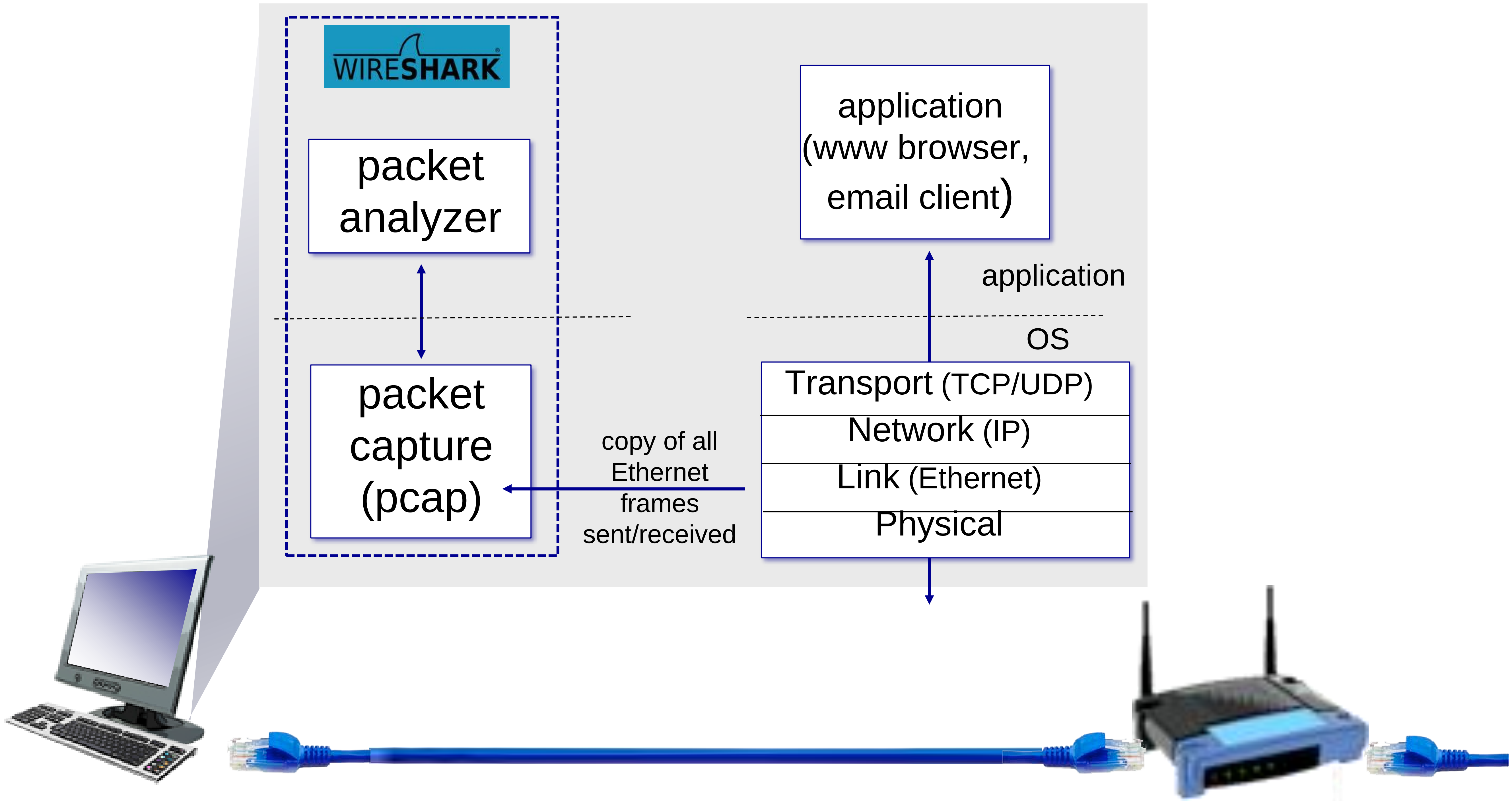
Introduction: summary

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!***


WIRESHARK
packet analyzer
packet capture (pcap)
copy of all Ethernet frames sent/received
application (www browser, email client)
application
OS
Transport (TCP/UDP)
Network (IP)
Link (Ethernet)
Physical