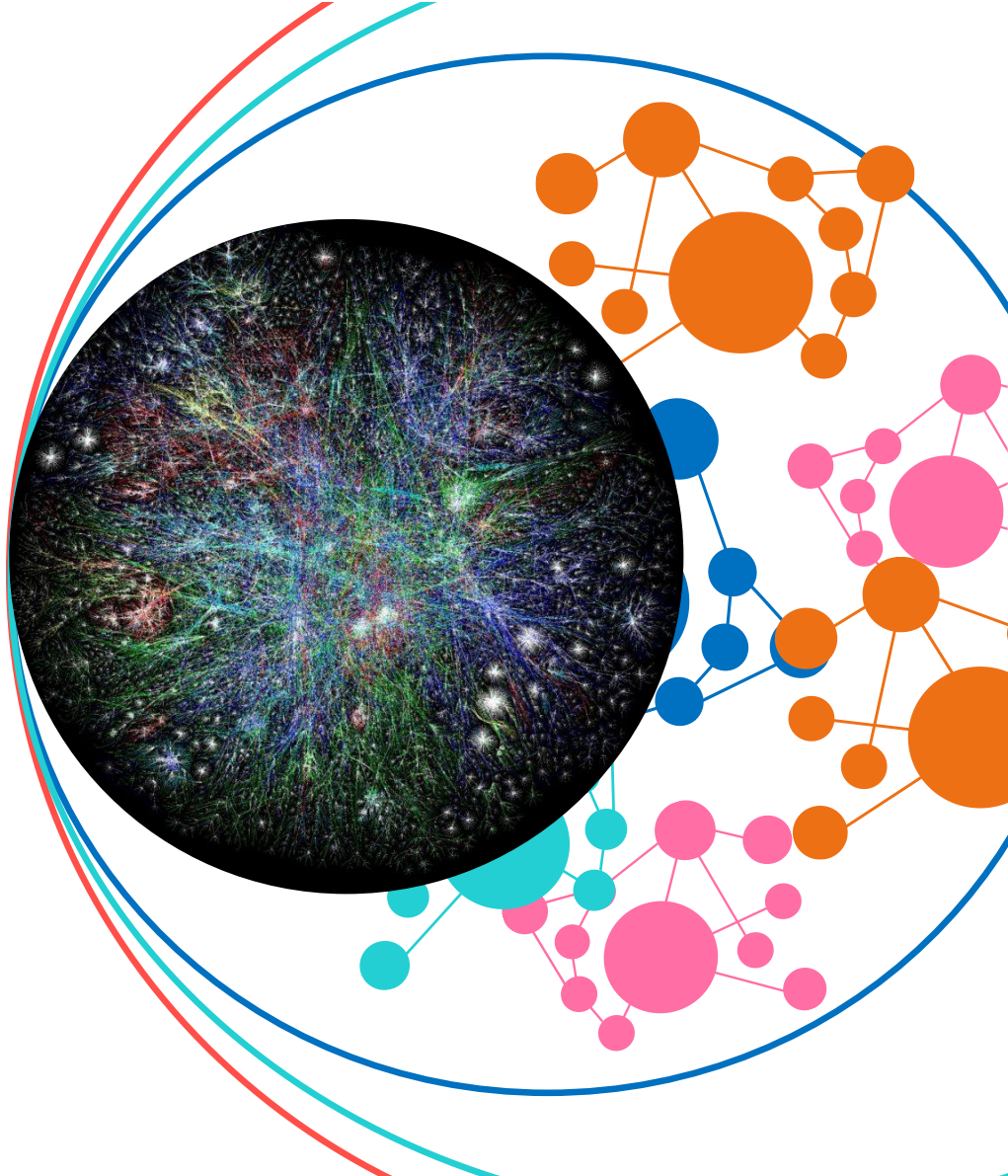
Introduction

Introduction

- The Internet
- Basics
  - Network Edge
  - Network Core
  - Network Performance
    - Loss, Bandwidth, Throughput, Latency, Delay, Jitter
- Protocol Layers and Reference Models
- Network Security: Networks Under Attack
- Brief History of Networking

Introduction:
The Internet



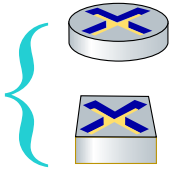
The Internet

- Nuts and Bolts
  - **Hosts**
    - **End systems** running network applications

  - Communication Links
    - Fiber, Copper, Radio, Satellite

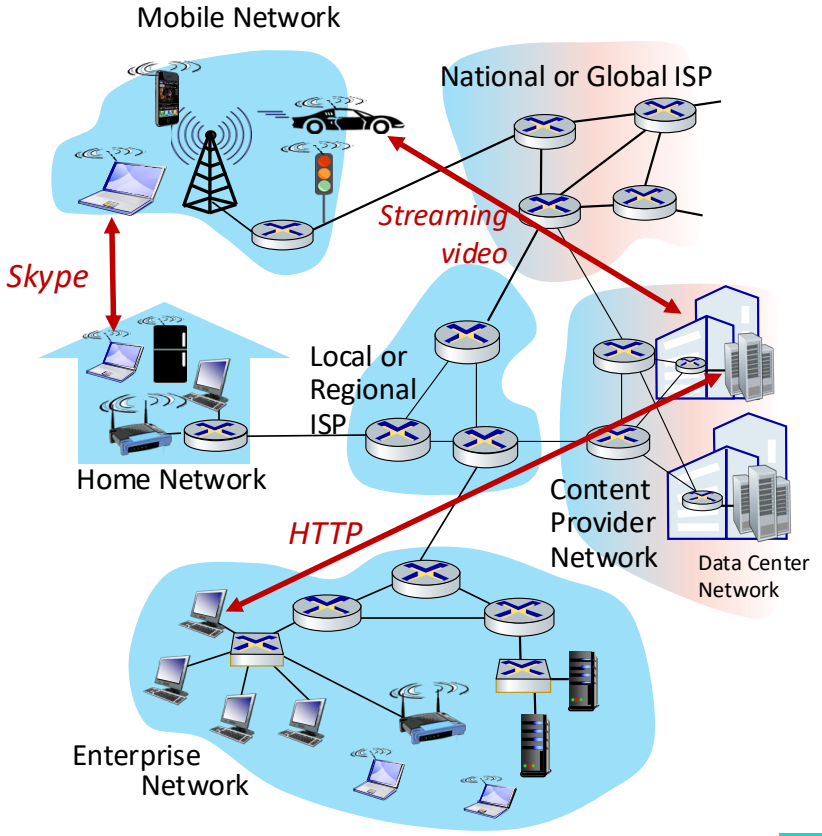
  - Switches and Routers
    - Find other switches and routers along the path to destination and forward data

End to links to the user



The Internet Definition 2

- Network of networks: Interconnected ISPs
  - Infrastructure providing **service** to applications
  - Infrastructure providing **programming interfaces (Socket Interface)** to applications



Many ways for the services to connect
↳ links
google http protocol

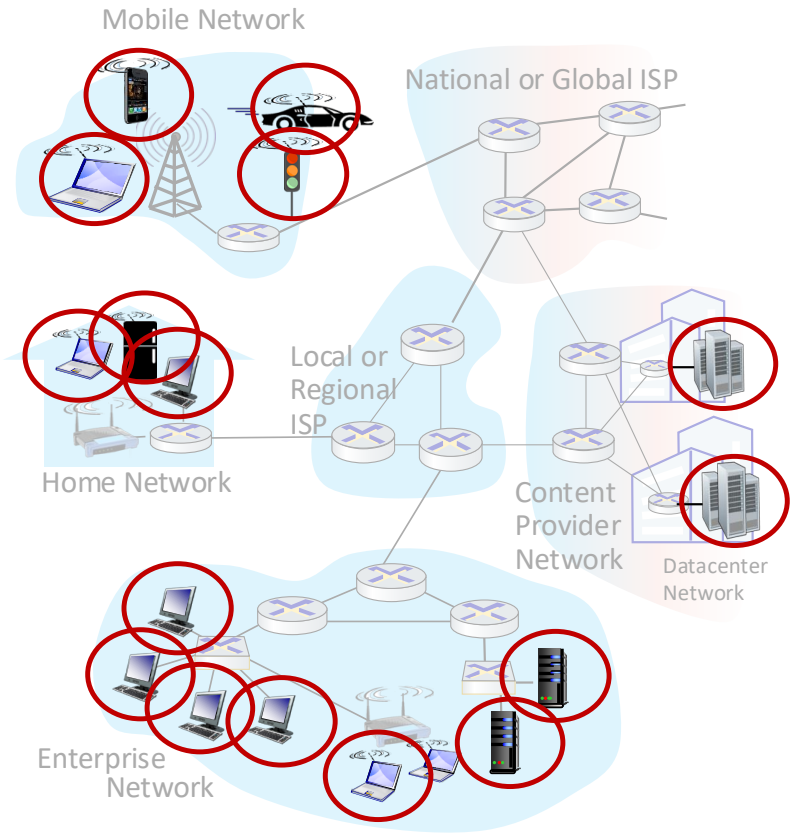
Introduction:
Network Edge

Network Edge

- Network Edge: Includes end systems
- Access Network

user computers, phone connected to access networks

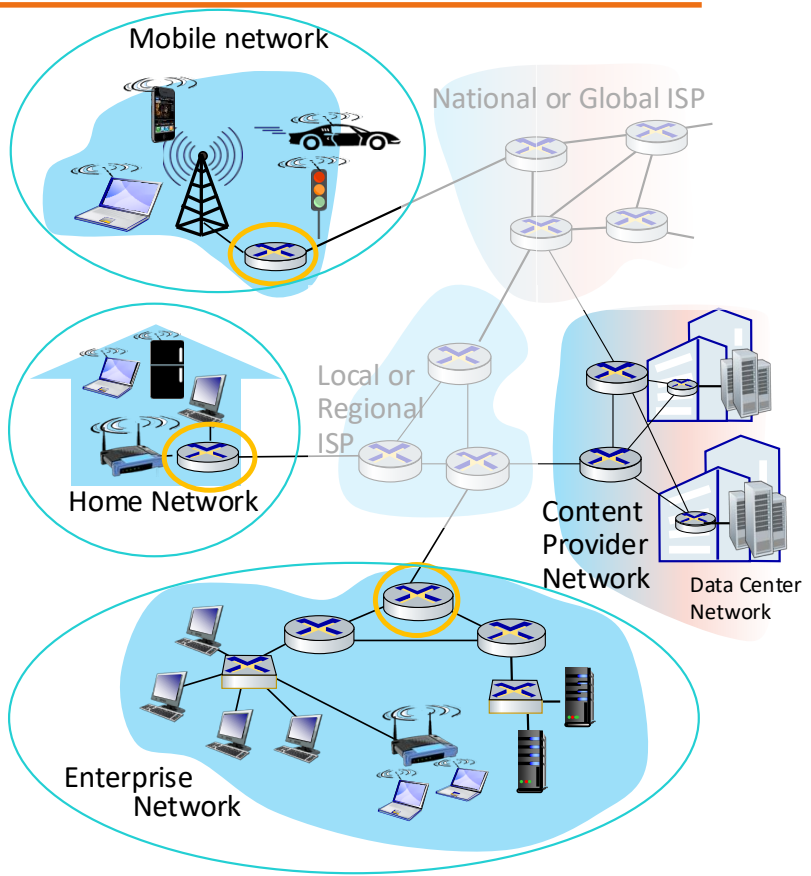
end systems recieve the connectivity service



Network Edge

- Network Edge: Includes end systems

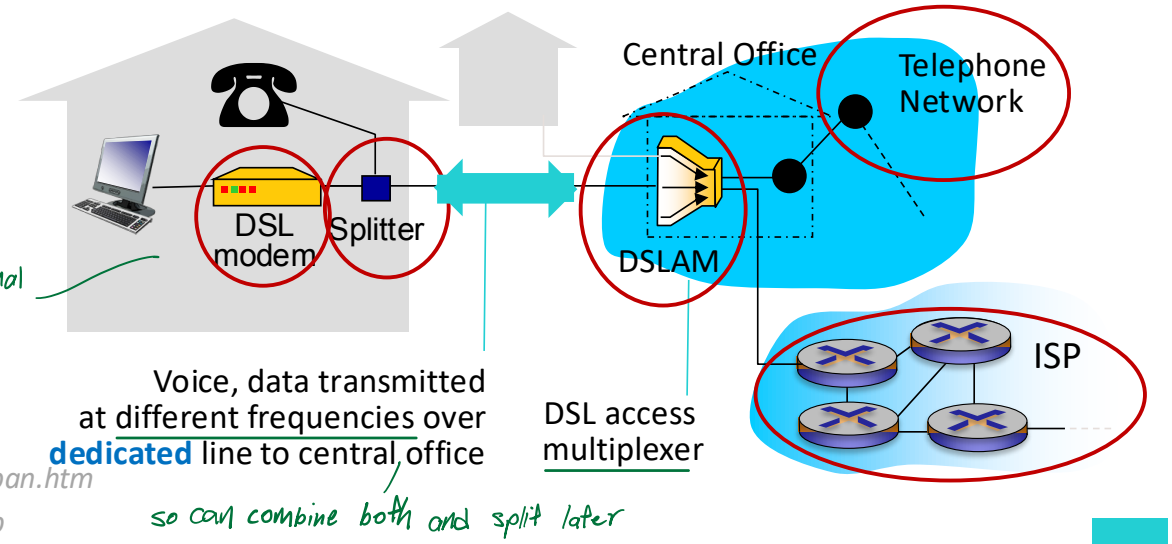
- Access Network
  - Connecting **End systems** to **Edge Routers**
    - Access
      - Digital Subscriber Line (DSL)
      - Cable
      - **Fiber**, Fiber To The Home (FTTH)
      - Dial-up
      - Satellite
    - Home Access
      - WiFi
    - Enterprise Access
      - WiFi
      - Ethernet
    - Mobile & Wide-Area Wireless Access
      - 3G
      - Long Term Evolution (LTE)
      - 5G



Access Networks: DSL

, circuit switch

- Digital Subscriber Line (DSL)
  - no need build infrastructure
  - Uses **pre-existing** telephone line to central office DSLAM
    - Data over DSL phone line goes to Internet
    - Voice over DSL phone line goes to telephone net



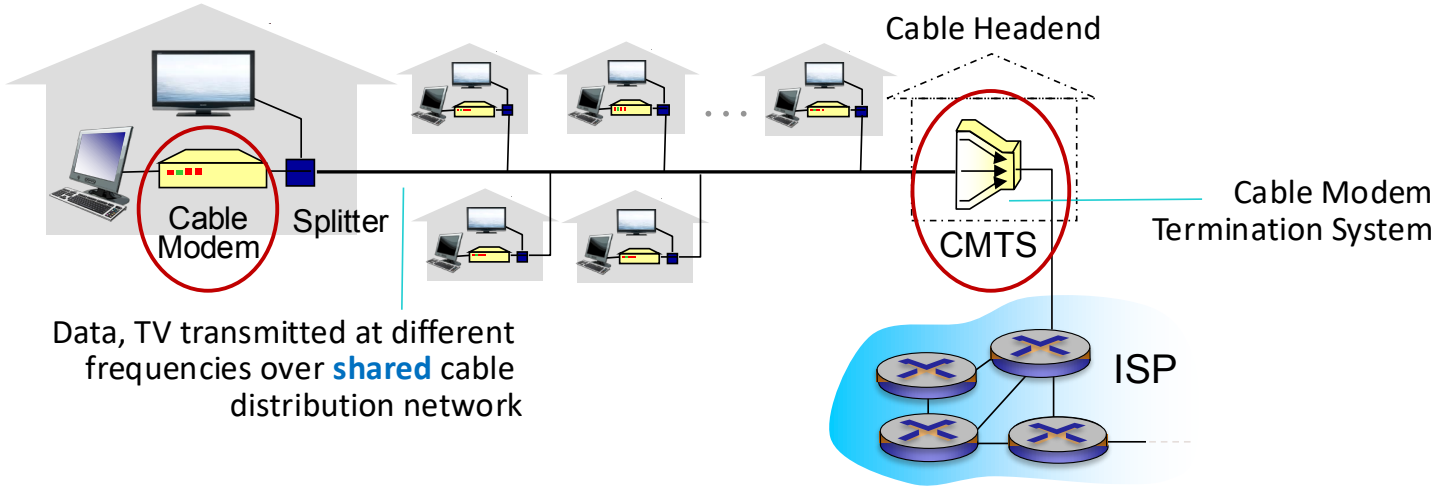
https://crtc.gc.ca/eng/publications/reports/PolicyMonitoring/ban.htm
https://ised-isde.canada.ca/app/scr/sittibc/web/bbmap#!/map

from source - Downlink
to source - Uplink

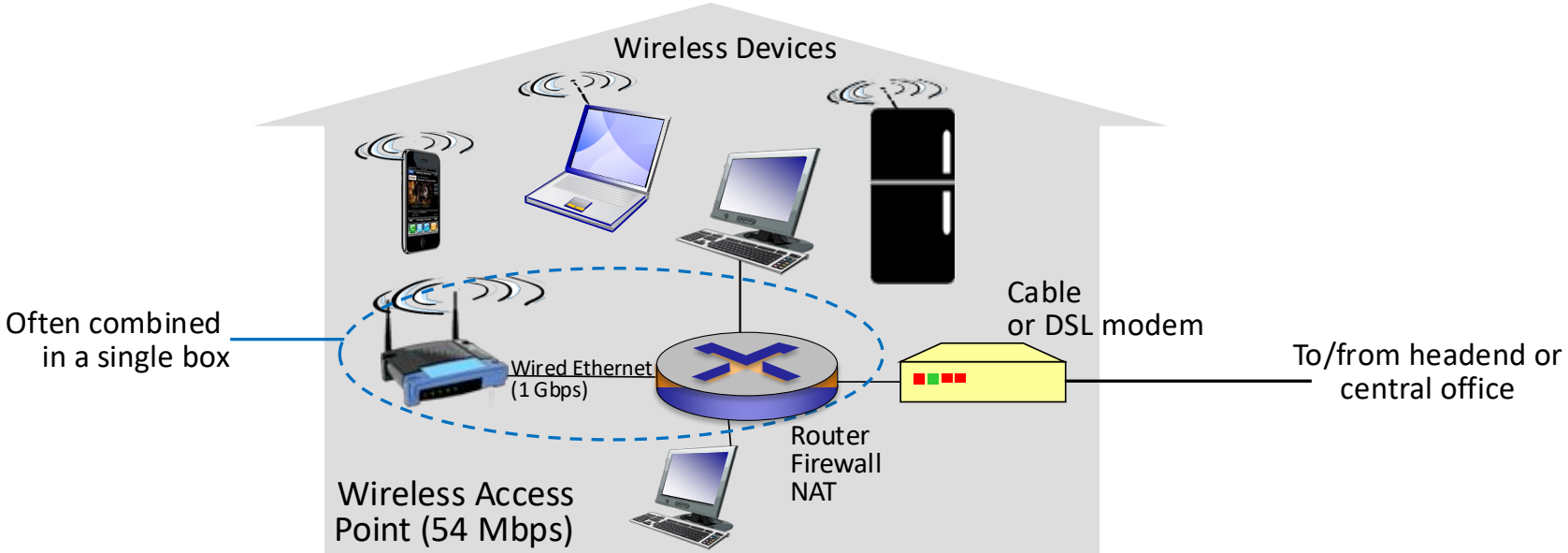
Access Networks: Cable Network

When channel sent, everything sametime

- **HFC (Hybrid Fiber Coax):** Asymmetric
- **FDM (Frequency Division Multiplexing):** Different channels transmitted in different frequency bands
- **Network** of cables and fibers attaches homes to ISP router
  - Homes **share access network** to cable headend
  - Unlike DSL, which has dedicated access to central office

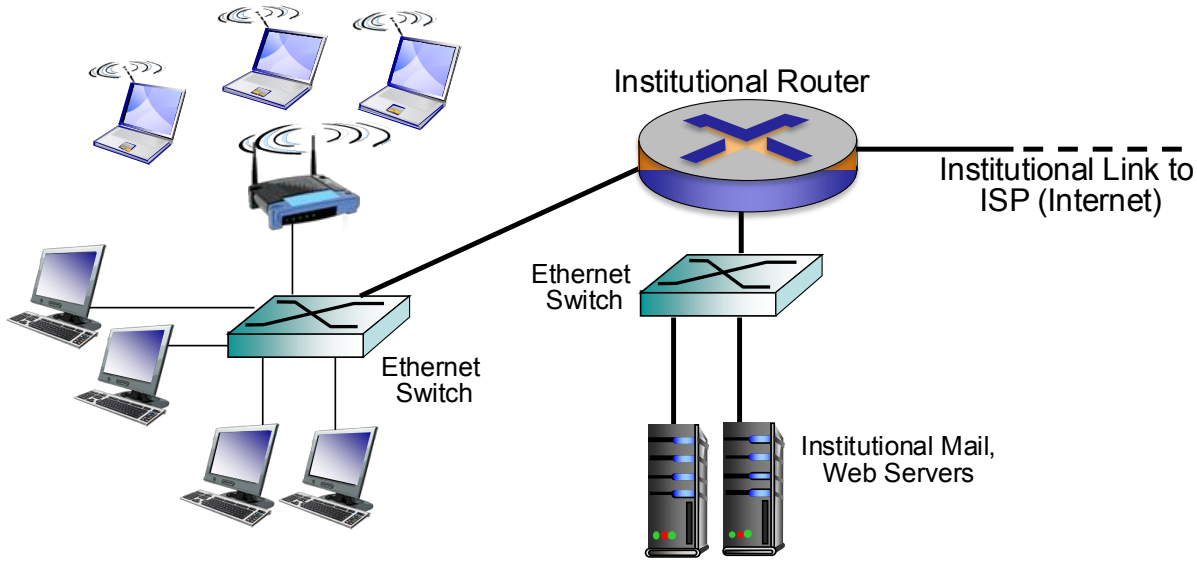


Access Networks: Home Networks



Access Networks: Enterprise Access

- Ethernet
  - Typically used in companies, universities, etc.
  - End systems typically connect into Ethernet switch



Wireless Access Networks

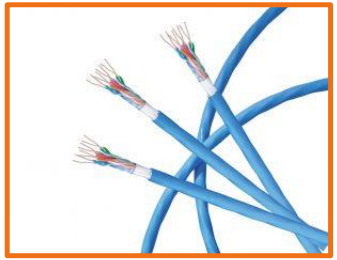
- Wireless Local Access Networks (LANs)
  - Shared **wireless** access network connects end system to router (through access point)
  - Within building
  - 802.11b/g/n/ac/ax (WiFi)

    50 m

- Wide-Area Wireless Access (WANs)
  - Provided by telco (cellular) operator
  - 3G, 4G/LTE, 5G

Access Media

- Physical Media
  - **Guided Media** - Wired
    - Twisted-Pair Copper Wire
      Two insulated copper wires
    - Coaxial Cable
      Two concentric copper conductors
      Bidirectional
      Broadband (multiple channels on cable, HFC)
    - Fiber Optics
      Glass fiber carrying light pulses, each pulse a bit
      High-speed point-to-point transmission
      Low error rate: repeaters spaced far apart, immune to electromagnetic noise
    - Ethernet
  - Unguided Media

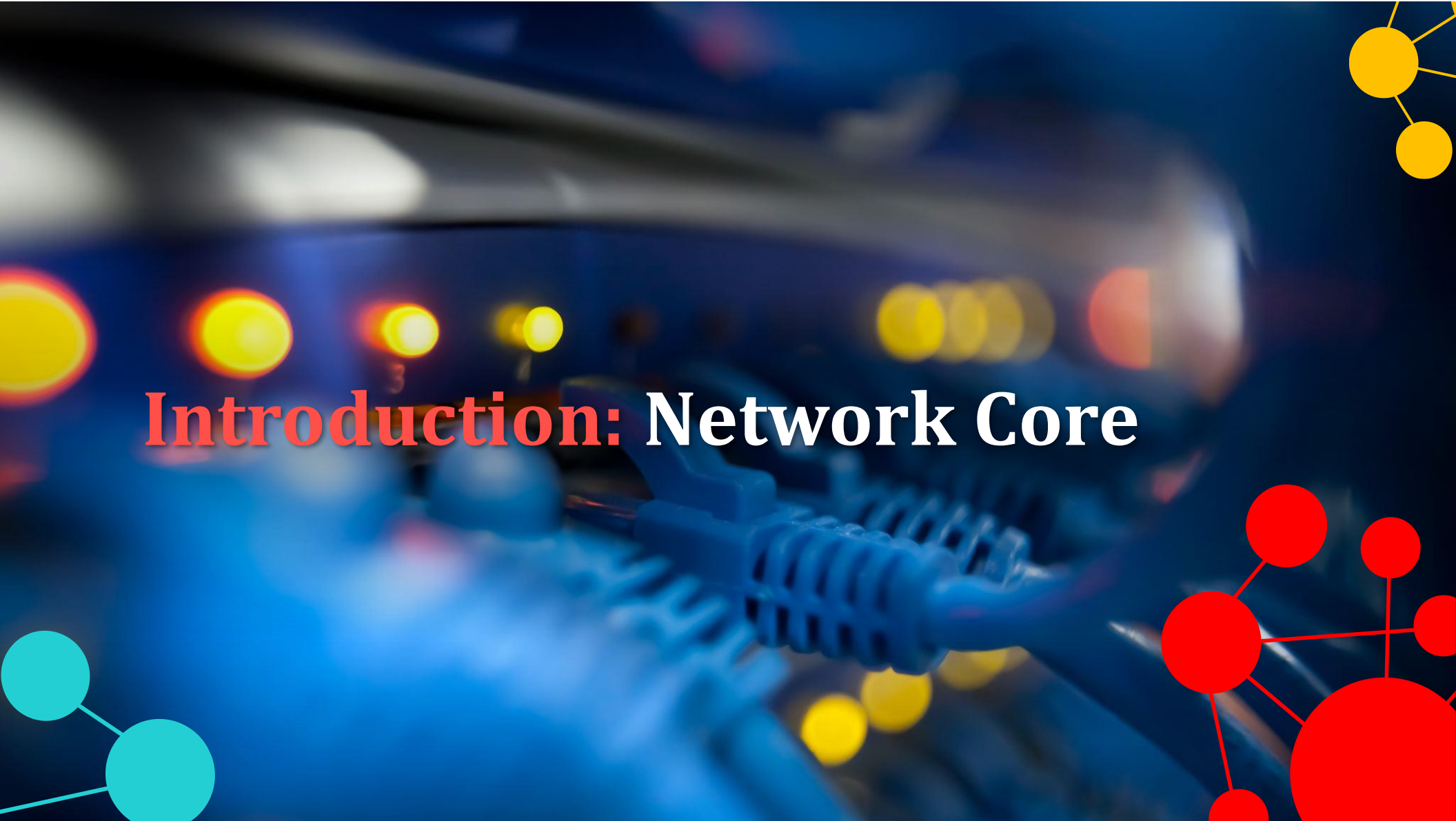


Access Media

- Physical Media
  - Guided Media
  - **Unguided Media**
    - Radio (No wire, Bidirectional)
      - Propagation Environment Effects
        - Reflection - like sound waves
        - Obstruction by objects
        - Interference - talking over eachother if 2 sources send simultaneously
      - Radio Link Types
        - Terrestrial Microwave
          - Personal Area PANs Bluetooth
          - Local Area LAN
          - Wide Area WAN
        - Satellite Radio Channels Starlink
          - Geosynchronous versus low altitude

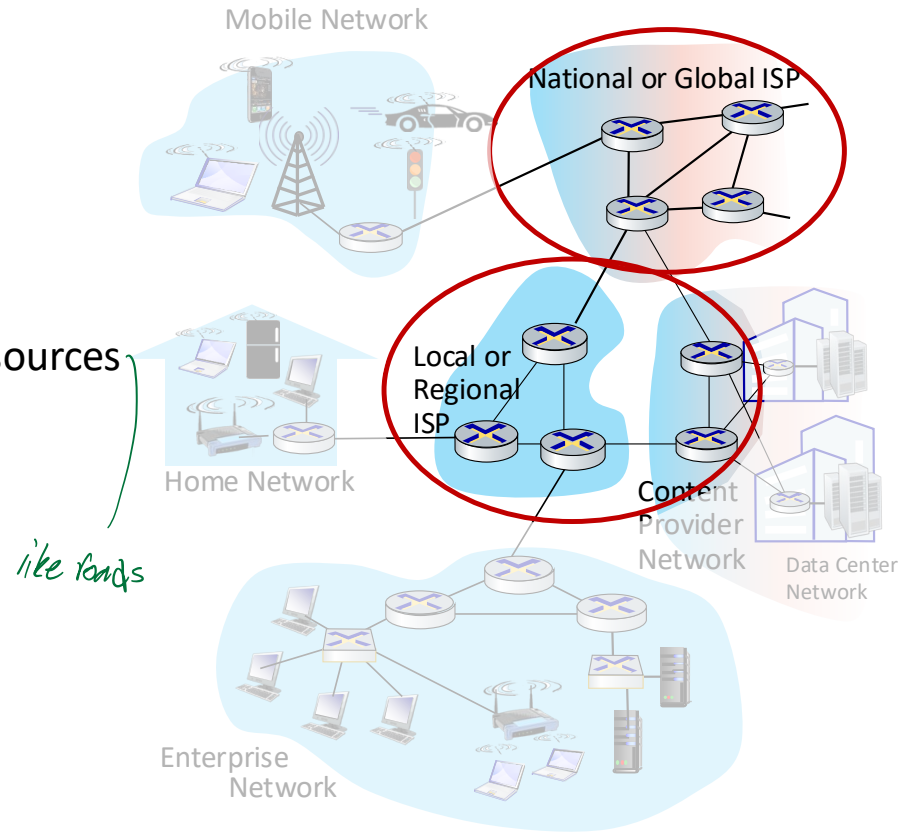
Drone communication

low Earth Orbiting (LEO)


Introduction: Network Core

Network Core

- Connect the access networks
  - Mesh of interconnected routers
  - Providing a path between source and destination through shared network resources
    - **Packet Switching**
    - Circuit Switching



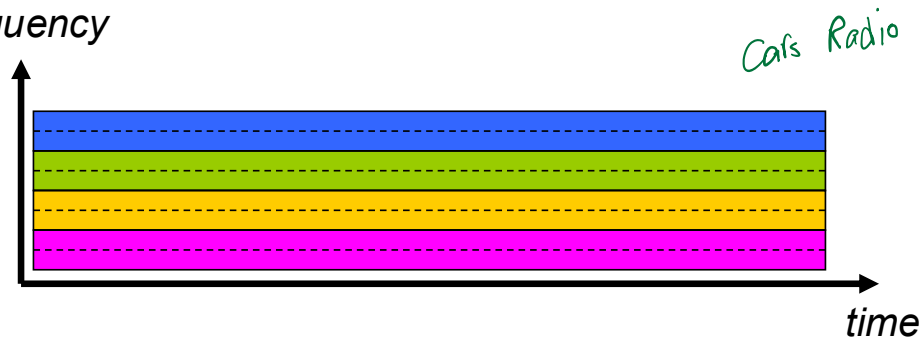
Circuit Switching

- Reserve all resources from 1 end to another

- End to end resources reserved and allocated between source and destination
  - **FDM**: Frequency-Division Multiplexing
    - Frequency spectrum of a link is divided among the connections
  - **TDM**: Time-Division Multiplexing
    - Users time share by receiving a dedicated fix-length time slot in a time frame

Circuit Switching: FDM & TDM

Figure (1): **FDM**

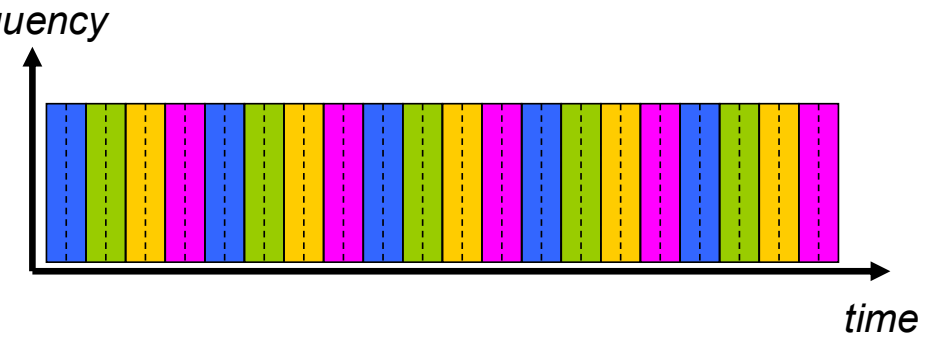


Cars Radio

Example:

Four users 1 2 3 4

Figure (2): **TDM**



10

so only certain users

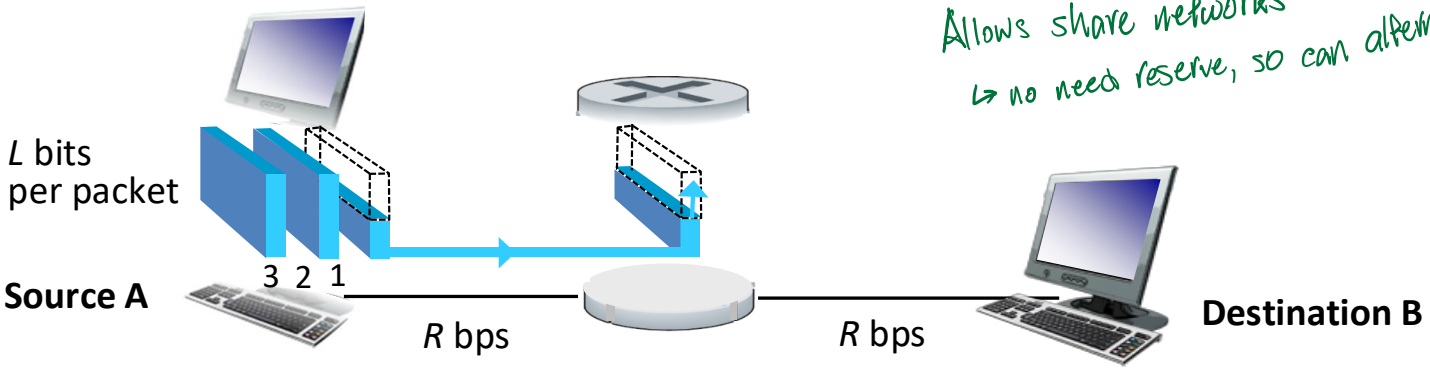
Combine fdm + TDM + codes

Doesn't reserve anything (BANDWIDTH)

Packet Switching: Store & Forward

- **Packet Switching:** Store-and-forward a chunk of data called packet

Allows share networks
↳ no need reserve, so can alternate between the users



Entire packet must arrive at router before it can be transmitted on the next link

Packet switching allows more users to use network!

Packet Switching

- Hosts break messages into packets
- Packets go **from one switch to the next** across links on path from source to destination
- Each packet transmitted at full link capacity
- Two key network-core functions
  - **Routing:** Determines source-destination route taken by packets
  - **Forwarding:** Moves packets from router's input to appropriate router output

predetermined unless issue

ex- google maps

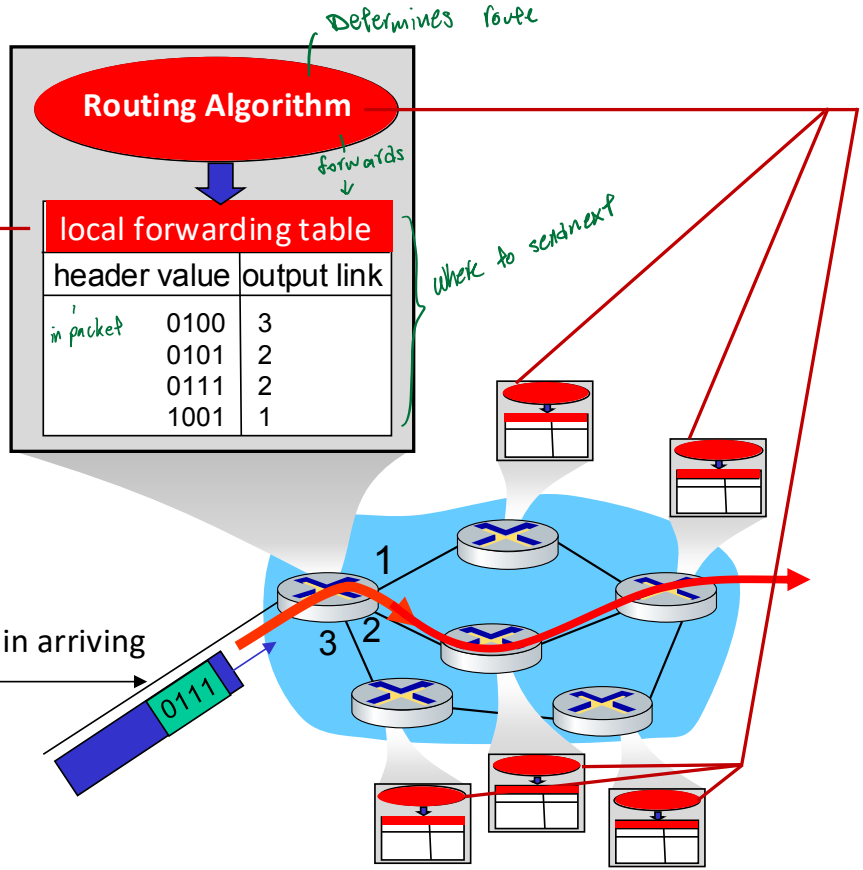
decision making where to place packet

ex- I decide which direction given route(s)

Always need paths first

Routers queue if same destinations
↳ traffic jams → different route

Packet Switching: Routing & Forwarding



Forwarding:

- Local action: Move arriving packets from router's input link to appropriate router output link

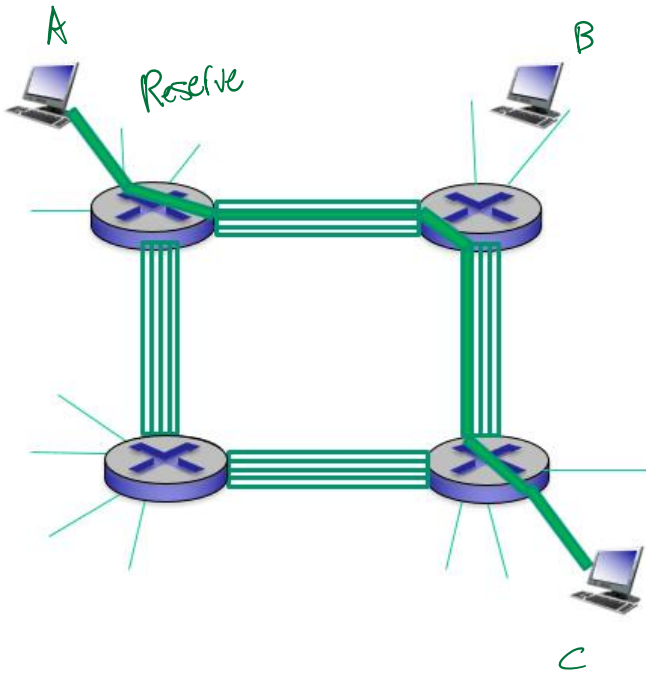
Routing:

- Global action: Determine source-destination paths taken by packets
- Routing algorithms

fifo Dijkstra

Packet Switching & Circuit Switching

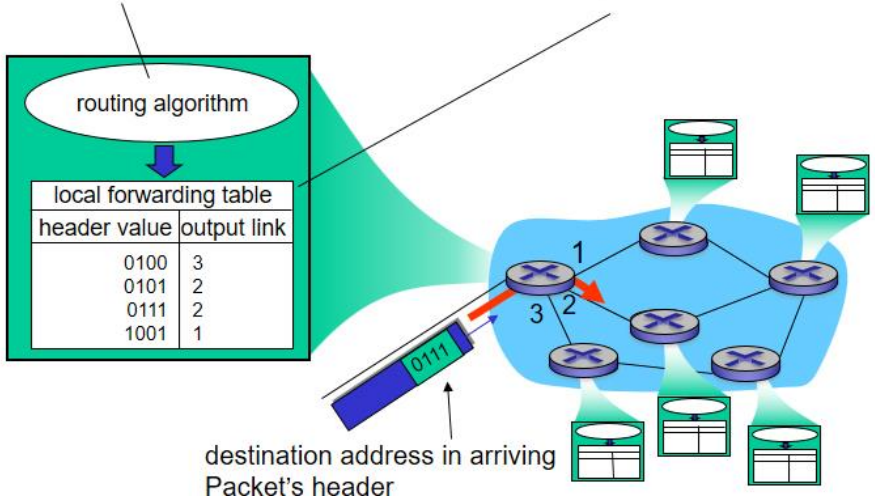
Circuit Switching



Packet Switching - just send packet

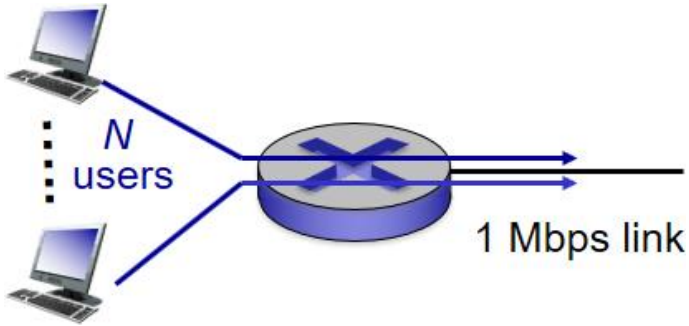
routing: determines source-destination route taken by packets
- routing algorithms

forwarding: move packets from router's input to appropriate router output



Example

- Each user **100 kbps** when active, and **active 10% of time**



- **Circuit-switching:** 10 users (1Mbps / 100 Kbps = 10)

link / usage

Active % unused because reserved

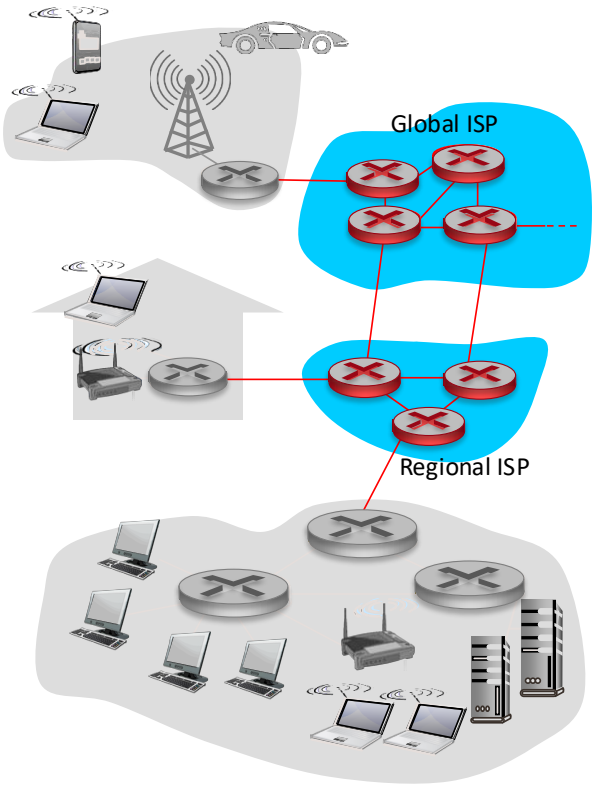
↳ doesn't matter 90 % no usage

- **Packet switching:**
  - Probability of single user active at any given time: 0.1 probability incoming packet
  - With 35 users probability that at any given time exactly 10 users are transmitting simultaneously: $\binom{35}{10} \times 0.1^{10} \times (1 - 0.1)^{35-10}$ inactive probability
  - Probability that when we have 35 users, more than 10 users are transmitting simultaneously?

Even if overload, buffer fine

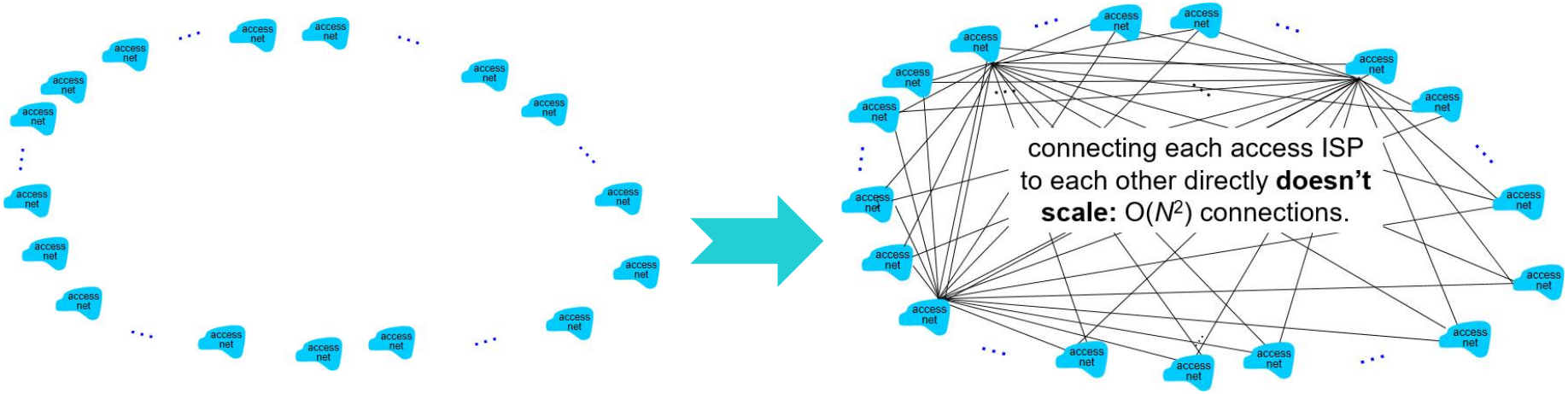
Network Core: Packet Switching

- Connect the access networks
  - Mesh of interconnected routers
  - Providing a path between source and destination through shared network resources
    - **Packet Switching** share core capacity



Internet Structure

- How to connect ISPs together? Everybody using different ISPs

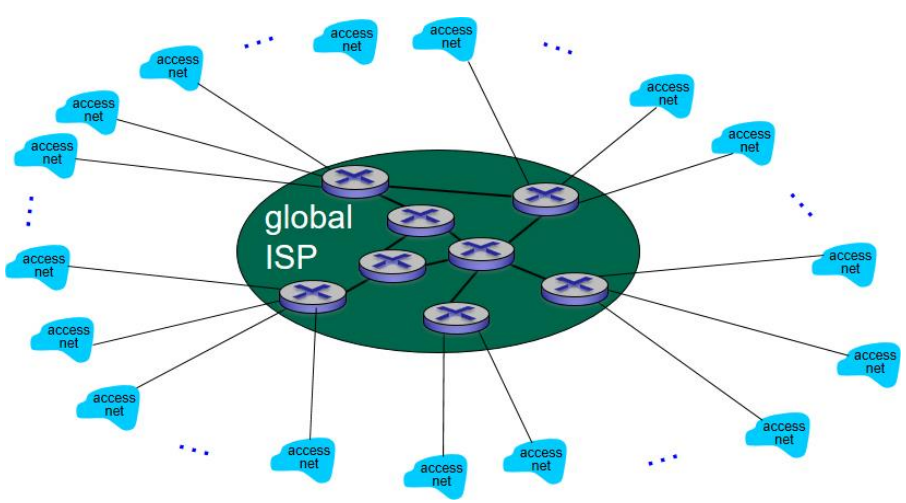


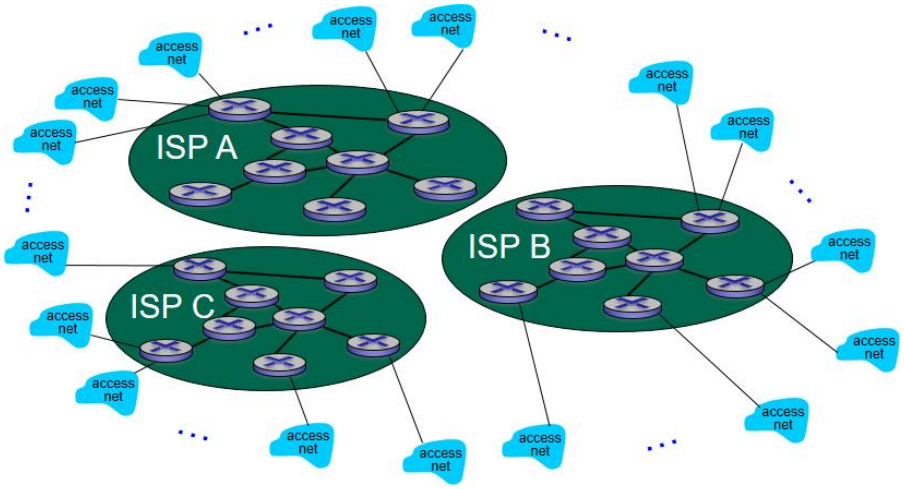
- Connect each access ISP to every other access ISP?

Internet Structure

Connect each access ISP to one global transit ISP?
Customer & provider ISPs have economic agreement.

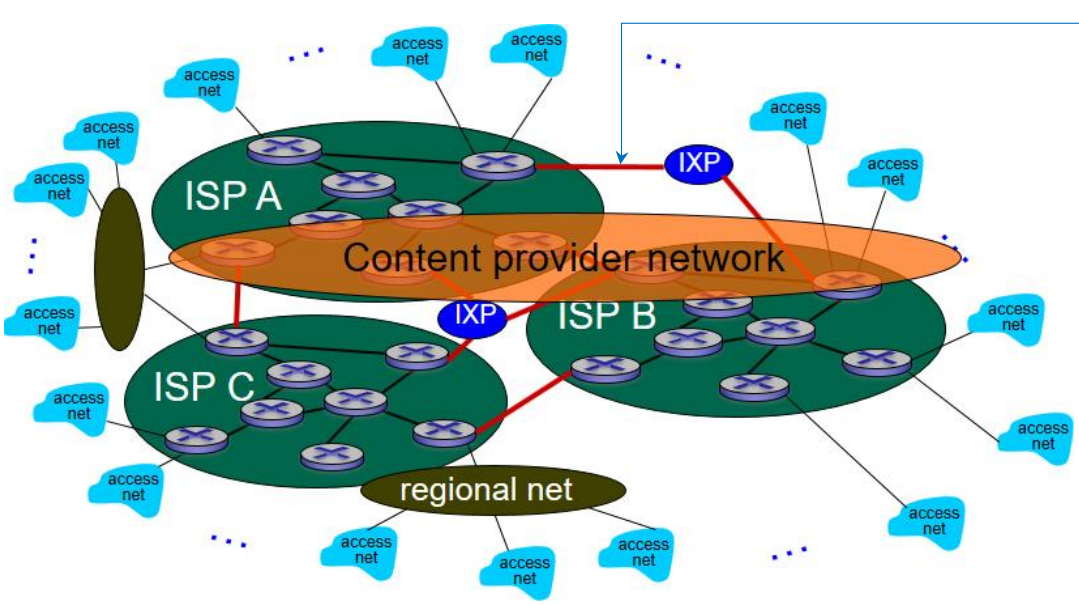
If one global ISP is a viable business, there will be competitors!





Internet Structure

- **POP (Point of Presence):** A provider ISP's location with a group of one or more routers where customer ISPs can connect into the provider ISP
- **Regional networks** may arise to connect access nets to ISPs
- **Content provider networks** (e.g., Google, Microsoft, Akamai) may run their own network, to bring services and content close to end users



Connecting ISPs

Internet Exchange Point
Vancouver - Vanix

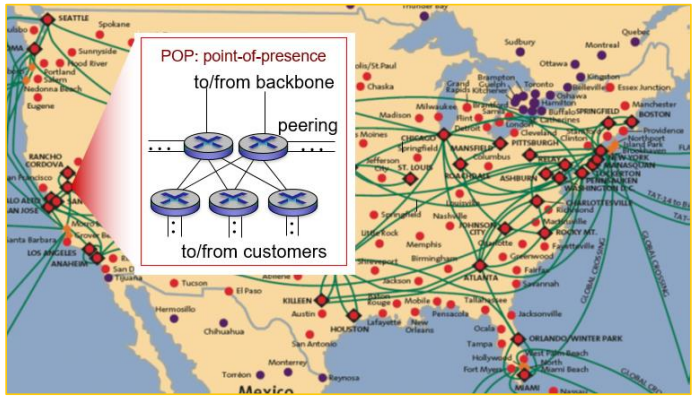
Internet Structure

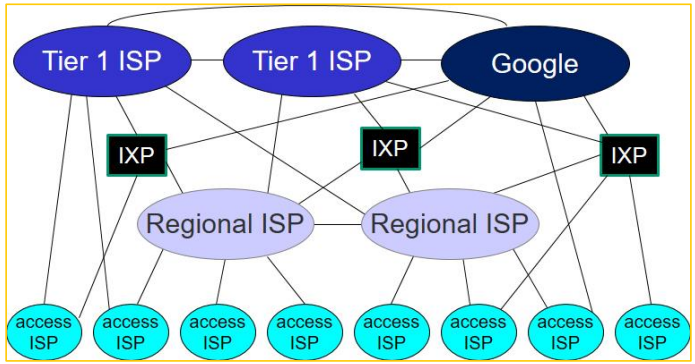
- Center: Small number of well-connected large networks
  - **Tier-1 commercial ISPs** - Worldwide reach through peers
    - Example: Sprint, AT&T
    - National and international coverage

so Rogers etc needs pay T1

  - **Content provider network**
    - Example: Google
    - Private network that connects its data centers to Internet often bypassing Tier-1 and regional ISPs

↳ Built close to IXPs so closer users





Each tier above - customer of

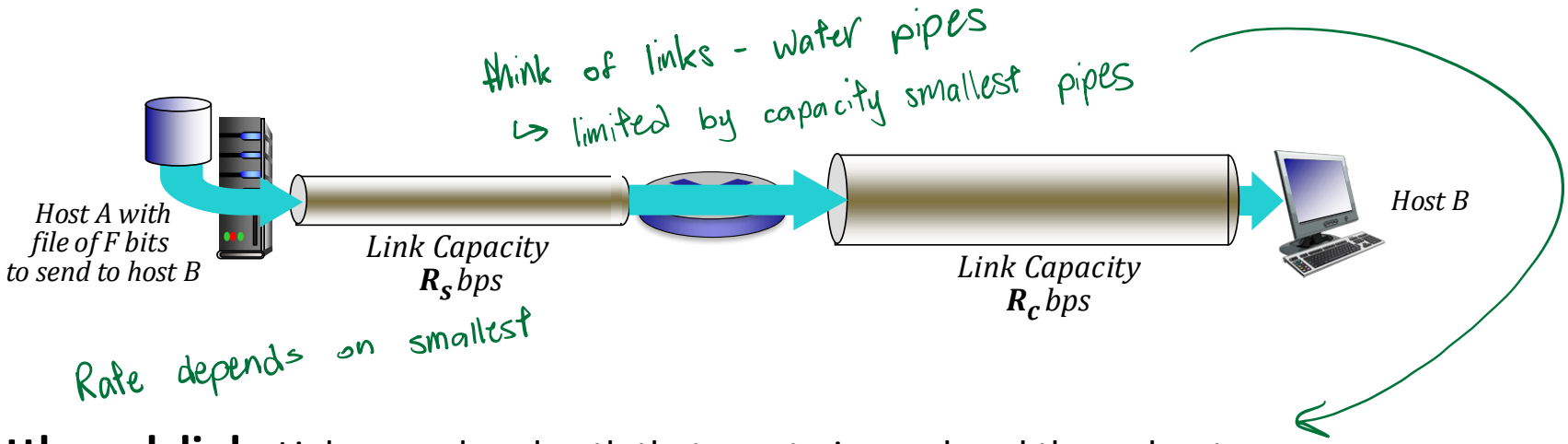
Introduction:
Network Performance

Network Performance

- Network Performance Metrics
  - Throughput/(Bandwidth) / capacity
  - Delay (Latency)
  - Loss
  - Jitter
  - Delay x Bandwidth Product

Throughput (Bandwidth)

- **Throughput:** Rate (bits/time unit) at which bits transferred between sender and receiver
  - **Instantaneous:** Rate at given point in time
  - **Average:** Rate over longer period of time



- **Bottleneck link:** Link on end-end path that constrains end-end throughput

Throughput (Bandwidth)

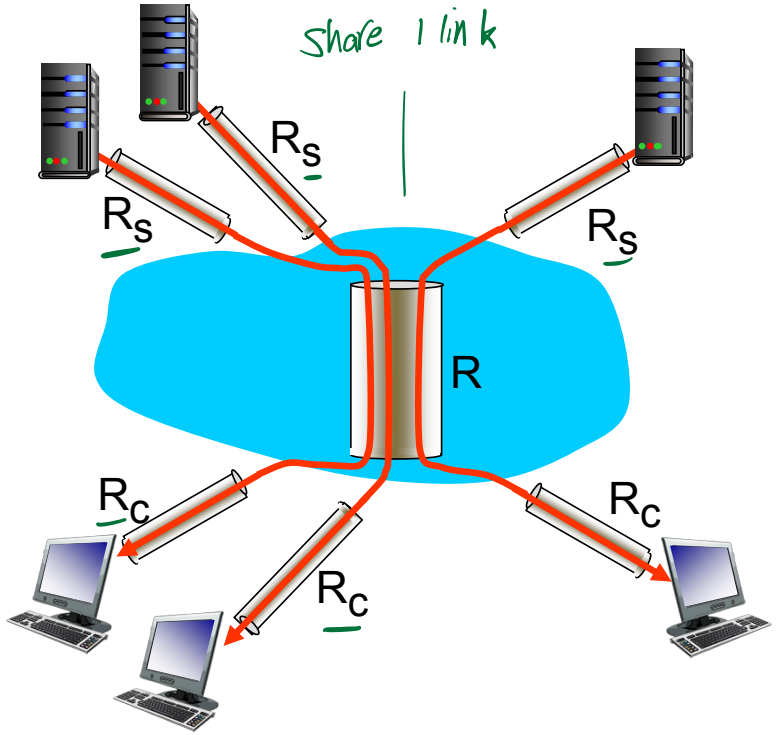
- N connections fairly share backbone bottleneck link *R* bps
- Per-connection end-end throughput: **min(R_c,R_s,R/N)**
- In practice R_c or R_s is often the bottleneck usually

what ISPs guarantee

55

Even though highest capacity - Bottleneck cause share

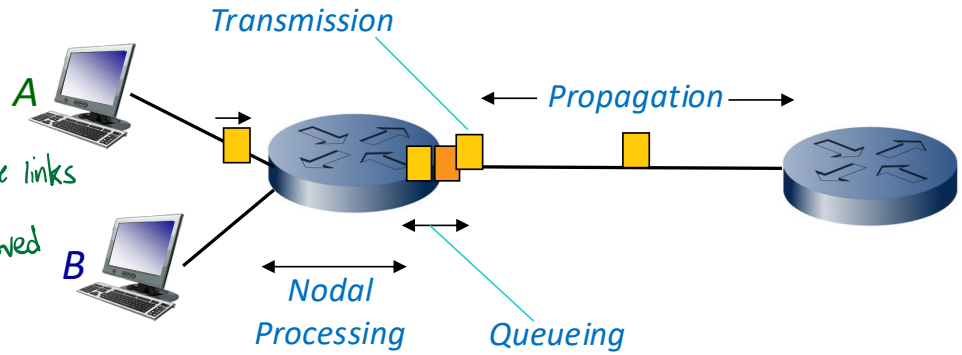
share 1 link



Delay (Latency) IMPORTANT

- Packet Switching
  - Four sources of delay routing table quick on each packet
    1. Nodal Processing Delay ✓
    2. Queueing Delay
    3. Transmission Delay ✓ how long through the links
    4. Propagation Delay - time sent → recieved

physical



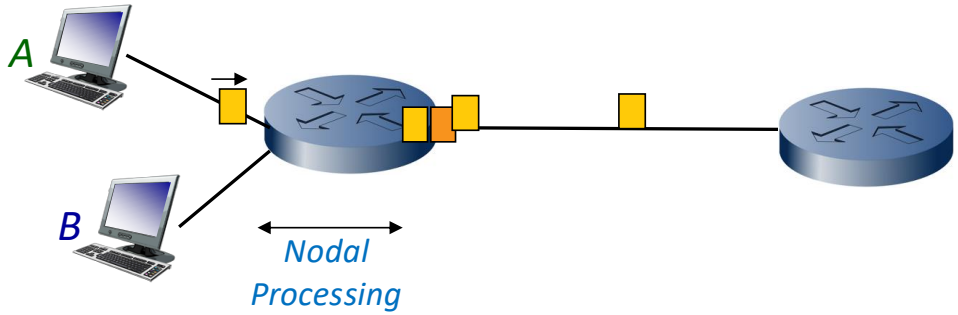
$$Delay = d_{proc} + d_{queue} + d_{trans} + d_{prop}$$

Nodal Processing Delay

Routing table process.

- Delays due to processing on the packet in a router (d_{proc})

- Examples (delay)
  - Examining the header
  - Determining the outgoing link
  - Error checking



Packets waiting to be sent.

1:10 think of car eks

Queueing Delay

happens cause link sharing.

Rates of arrival & departure vary = queue delay

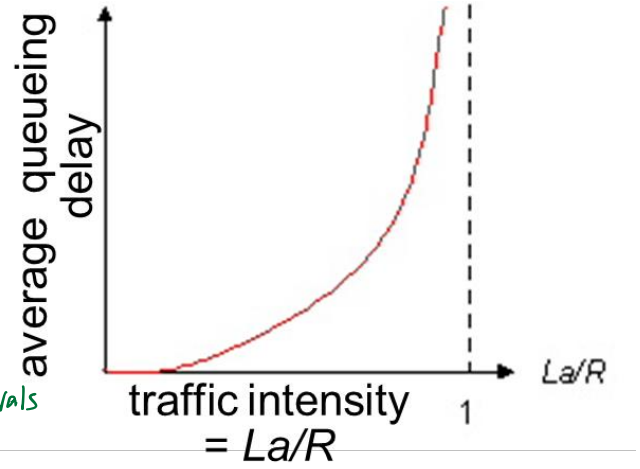
- **Traffic Intensity: La/R**
  - **R**: Link bandwidth (bps)
  - **L**: Packet length (bits)
  - **a**: Average packet arrival rate

Close to 1 - larger queue

La/R ~ 0 : Average Queuing Delay Small

La/R → 1 : Average Queuing Delay Large

La/R > 1 : More work arriving than processed



think of cashier lines → Checkout faster than arrivals = line decreases

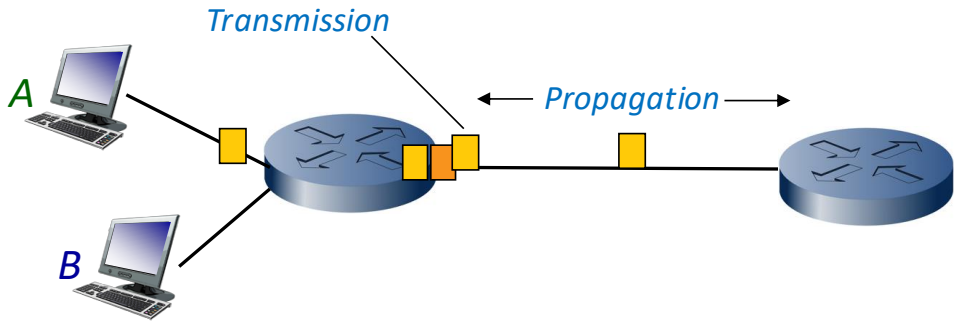
→ Arrive faster than checkout = line grows

In

depends length of medium & speed of medium

Transmission & Propagation Delays

- Transmission Delay
  - Time required to transmit all packet bits on the link
  - L/R
    - L: Packet Length
    - R: Link Bandwidth



- Propagation Delay
  - Time required to propagate from the beginning to end of the link
  - d/s
    - d: Length of the link
    - s: Speed of propagation in the medium

Delay

- Ideal to include every source but not always possible (lack information)

- Packet Switching: Four sources of delay
  - **Transmission Delay**

    d_{trans}: transmission delay:
    - *L: Packet length (bits)*
    - *R: Link bandwidth (bps)*
    - $d_{trans} = L/R$

  - **Propagation Delay**

    d_{prop}: propagation delay:
    - *d: Length of physical link*
    - *s: Propagation speed in medium (~$2x10^8$ m/sec)*
    - $d_{prop} = d/s$

  - **Nodal Procession Delay**

    d_{proc}: nodal processing
    - *Check bit errors*
    - *Determine output link*
    - *Typically < msec*

  - **Queueing Delay**

    d_{queue}: queuing delay
    - *Time waiting at output link for transmission*
    - *Depends on congestion level of router*

(usually never enough info)
↳ grocery store ex - only knowing # people doesn't help determine wait time.

Jitter

- variation of delay between packets.

P1 - 1 ms
P2 - 2 ms
1 ms jitter

- Packet delay variations
  - Packet delay difference between for different packets
- Important in real-time applications
- Mitigation by buffers at the receiver side

Important because - affects delay calculations
- affects time sensitive content.
↳ realtime ex streams
↳ only certain pixel loads

- Solution

Calculate delay in load & z buffer

solves ex netflix / youtube ⤶

not always possible for realtime (stream / games)
↳ more complex solution required

Loss (Dropped Packets)

- When outta queue space

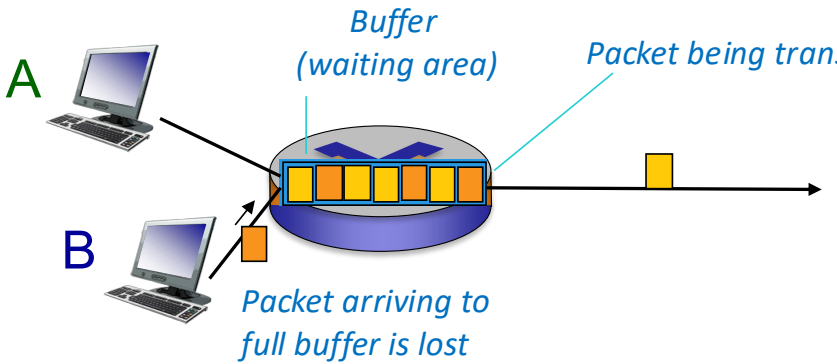
queue overflow

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

Detecting loss -

cause incomplete data

- Need to detect loss quick & early to react



Solution - numbering packets

↳ what if still incoming

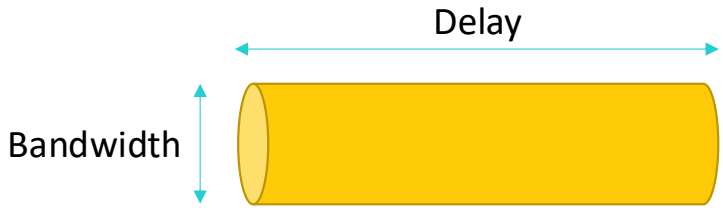
↳ maybe different paths

Solutions require knowing which packet lost.

Delay × Bandwidth Product

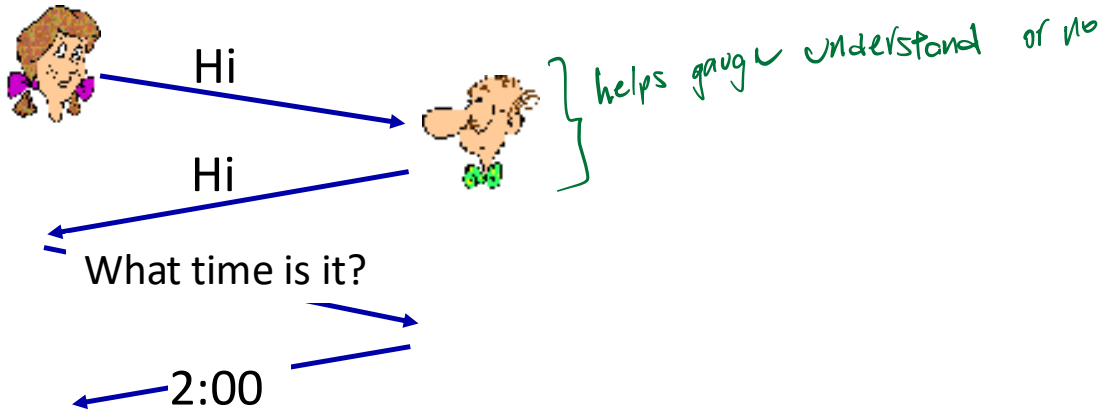
Helps control how much (the rate) of sends

- Delay × Throughput
  - If you consider network as a pipe, the delay bandwidth product provides the volume of the pipe



Introduction:
Protocol Layers & Reference Models

Example: A Human Protocol



Protocols – standards/agreements

- Protocols – which side saying what
  - **Protocols** define **format**, **order** of **messages sent and received** among network entities, and **actions taken** on message transmission, receipt
  - Examples: TCP, IP, HTTP, Skype, 802.11

↳ Application layer protocol

- Internet Standards
  - RFCs: Request For Comments
  - IETF: Internet Engineering Task Force

Example RFC

https://tools.ietf.org/html/rfc7235

Internet Engineering Task Force (IETF)
Request for Comments: 7235
Obsoletes: 2616
Updates: 2617
Category: Standards Track
ISSN: 2070-1721

R. Fielding, Ed.
Adobe
J. Reschke, Ed.
greenbytes
June 2014

Hypertext Transfer Protocol (HTTP/1.1): Authentication

Abstract

The Hypertext Transfer Protocol (HTTP) is a stateless application-level protocol for distributed, collaborative, hypermedia information systems. This document defines the HTTP Authentication framework.

Status of This Memo

APPROVED

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 5741.

Reference Model

- Define layers of abstraction
  ↳ separate potential issue

Dealing with complex systems

- Explicit structure allows identification and relationship of pieces
  - Layered **reference model** for discussion

- Modularization eases maintenance, updating of system
  - Change of implementation of layer's service transparent to rest of system

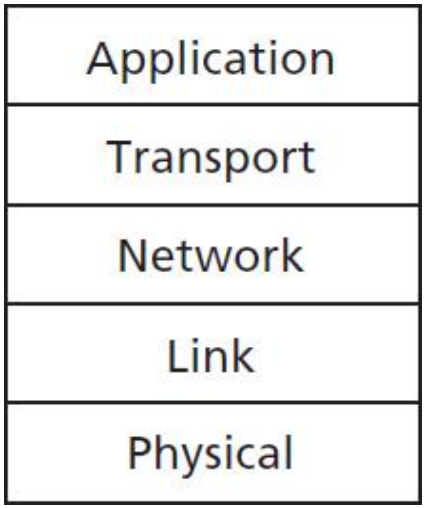
Layers: Each layer implements a service

- Via its own internal-layer actions
- Relying on services provided by layer below

- **Could layering be harmful?** – if layer dependent on previous

Internet Protocol Stack

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



OSI/ISO Reference Model

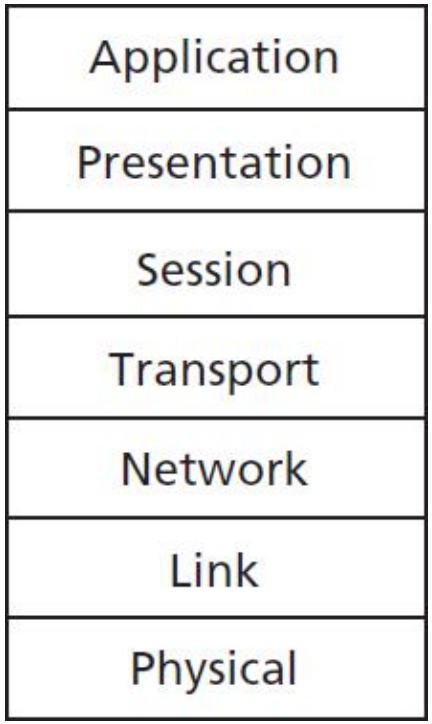
Additional layers to Internet Protocol Stack

Removes potential redundancies

- **Presentation:** Allow applications to interpret meaning of data
  - Encryption
  - Compression
  - Machine-specific conventions

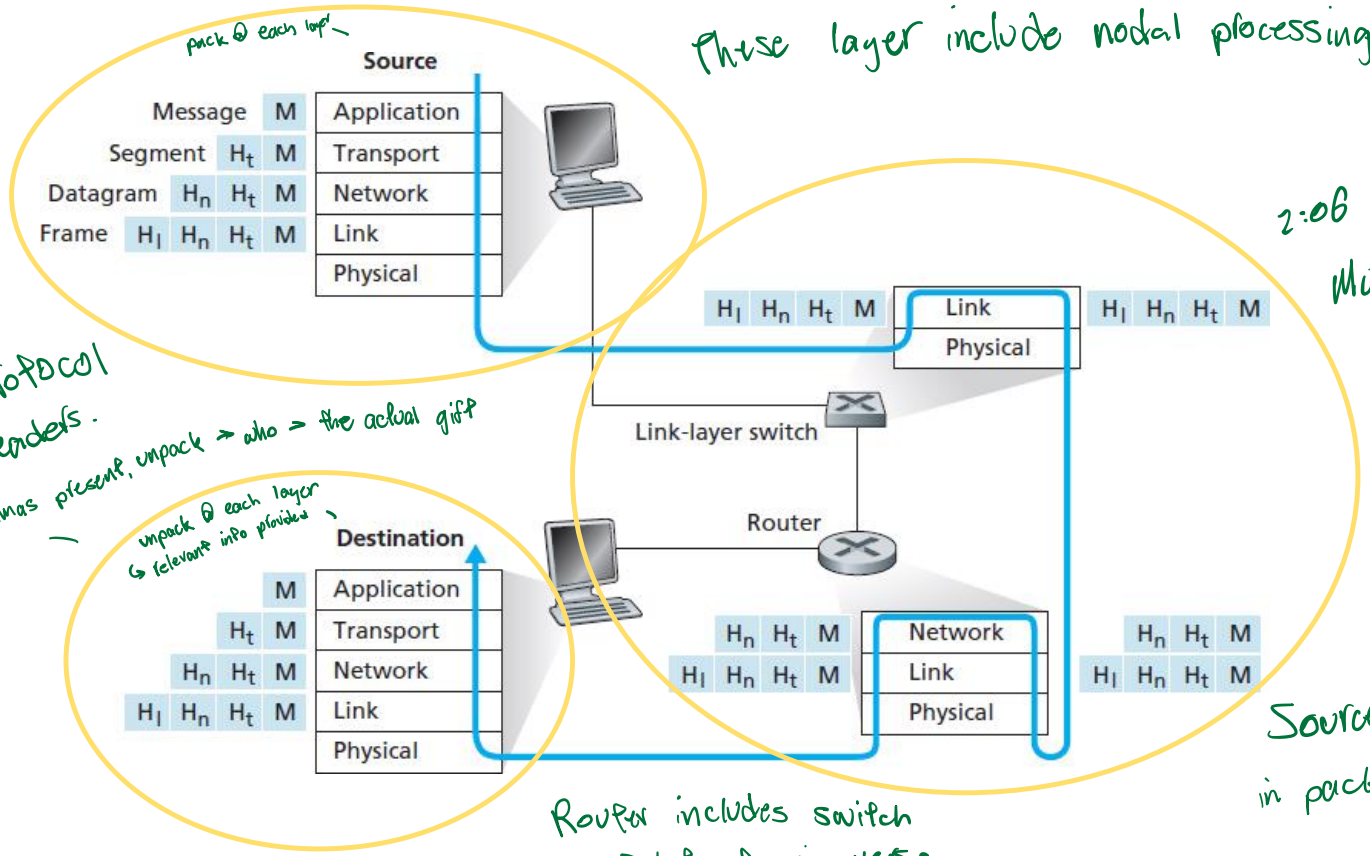
- **Session:** Synchronization, checkpointing, recovery of data exchange

Internet stack missing these layers. These services, **if needed**, must be implemented in application.



Recieved data - Application layer messages
↳ OS - Transport (+ headers) segments
2:02 ↳ Network
↳ link layer header

Encapsulation



These layer include nodal processing delay

pack @ each layer

Send ∨ protocol

Stack → headers.

kinda like Christmas present, unpack → who → the actual gift

unpack @ each layer
↳ relevant info provided

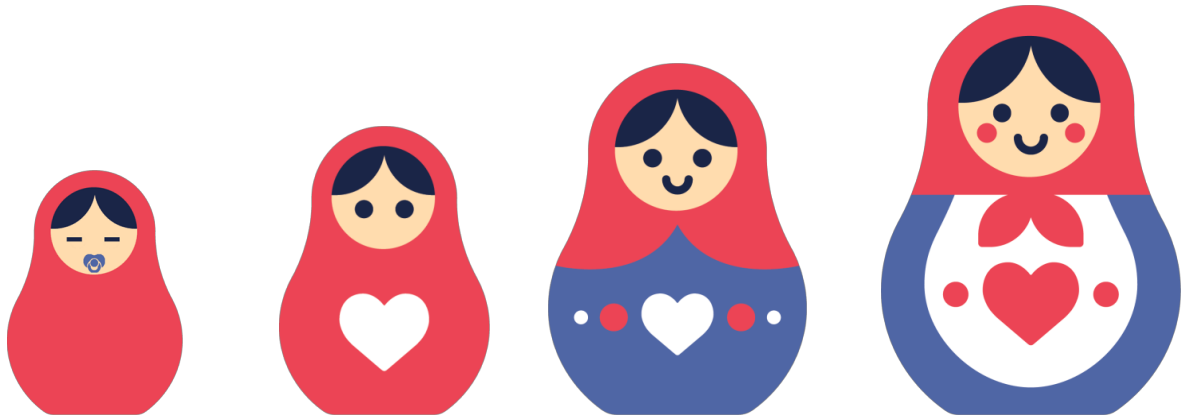
2:06

Multiplexing
↳ on switch remove previous frame header

Source & Destination in packet

Router includes switch
→ but not vice versa

Encapsulation



Introduction:
Network Security

Network Security

- Internet not originally designed with (much) security in mind
  - Original vision: A group of mutually trusting users attached to a transparent network

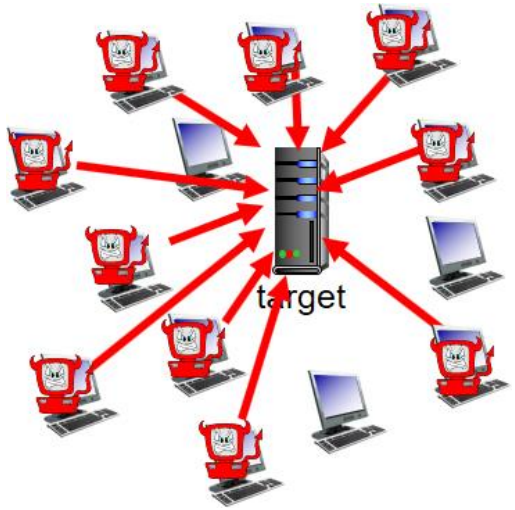
- Important
  - How networks could be attacked
  - How we can defend networks against attacks
  - How to design architectures that are immune to attacks

Networks Under Attack

- Malware for hosts spread through the network
  - Virus: Self-replicating infection by receiving/executing object (e.g. e-mail attachment) runs cause users
  - Worm: Self-replicating infection by passively receiving object that gets itself executed runs itself
- Spyware: Can record keystrokes, web sites visited, upload info to collection site
- Spoofing (IP Spoofing)

Denial of Service

- **Denial of Service (DoS):** attackers make resources (server, bandwidth) unavailable to legitimate traffic by overwhelming resource with bogus traffic
  - Select target
  - Break into hosts around the network (see botnet)
  - Send packets to target from compromised hosts



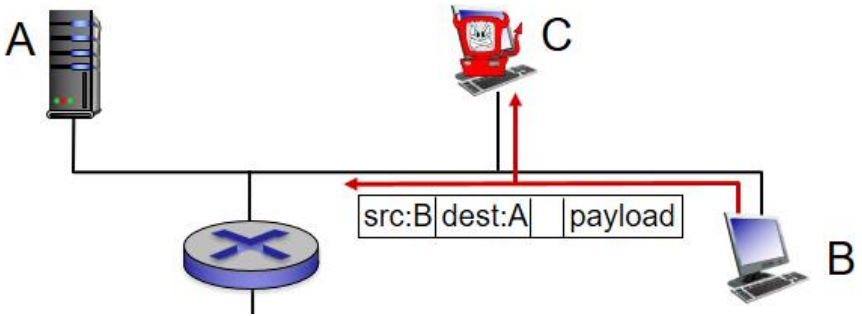
Packet Sniffing

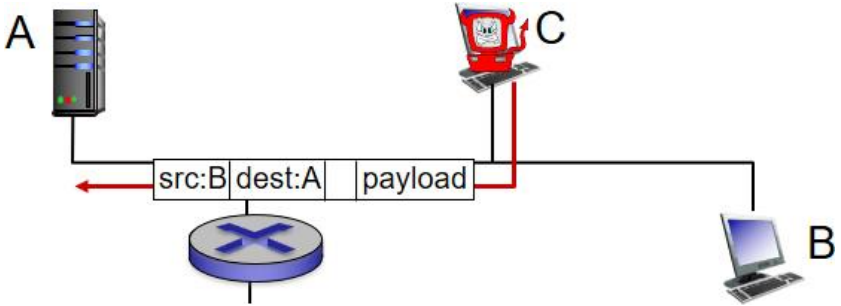
- slower
https ensures encryption
↳ makes sure talking right users

- Because open protocols
↳ no encryption means easy

Packet Sniffing:

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





Network Security

- Internet not originally designed with (much) security in mind
  - Original vision: A group of mutually trusting users attached to a transparent network

- Important
  - **How networks could be attacked**

Introduction:
Brief History of Networking

History of Computer Networks

Kleinrock, Queuing Theory
(Effectiveness of Packet Switching)

Baran (Packet Switching for Secure Voice in Military Nets)
Davies & **Scantlebury**

ARPANet Conceived

First **ARPANet** node

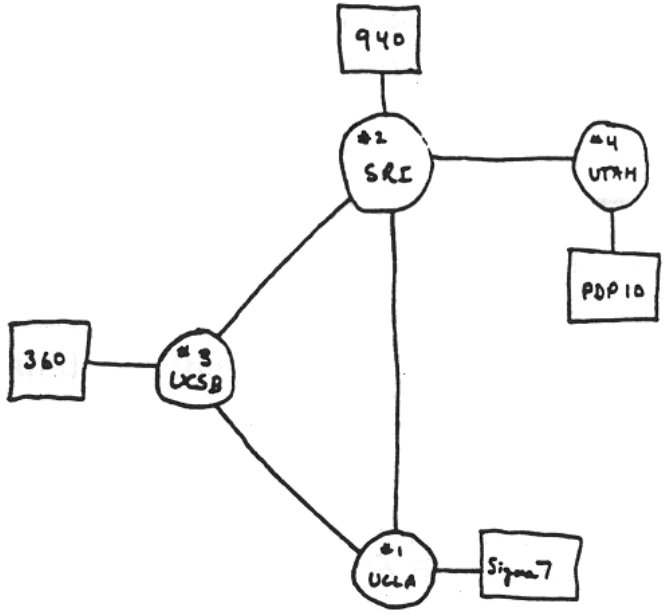
ARPANet public demo

NCP

Email

1961 — 1970

Packet Switching Principles



THE ARPA NETWORK

Reference: Computer Networking : A Top-Down Approach. James F. Kurose, Keith W. Ross

History of Computer Networks

Kleinrock, Queuing Theory
(Effectiveness of Packet Switching)

Baran (Packet Switching for Secure Voice in Military Nets)
Davies & **Scantlebury**

ARPANet Conceived

First **ARPANet** node

ARPANet public demo

NCP

Email

TCP
UDP
IP

SMTP

FTP

TCP Congestion Control

DNS

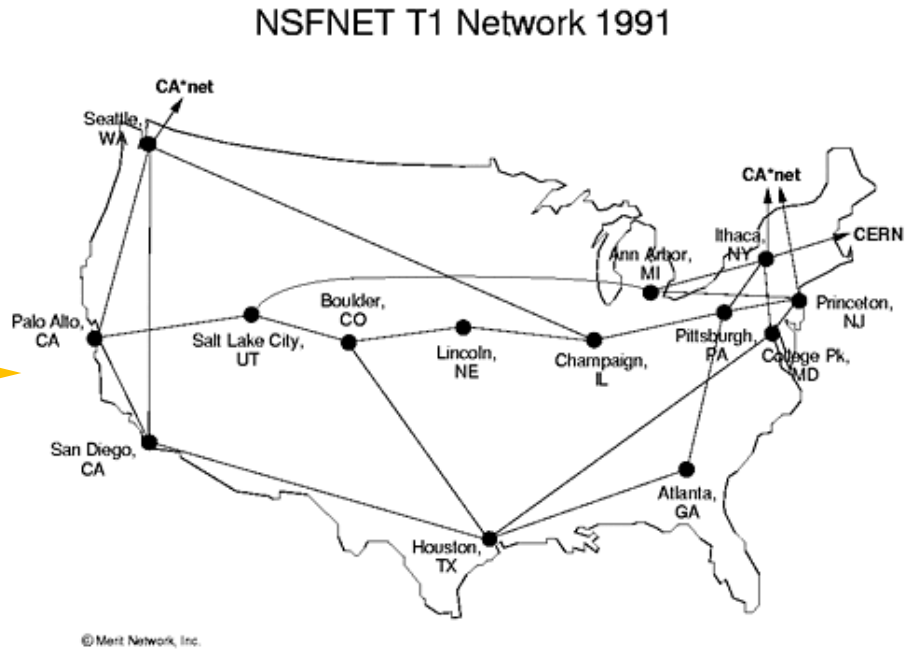
1961 — 1970 — DARPA, Telenet, Cyclades, Tymnet, etc. — 1980 — TCP/IP Deployment — 1990

ALOHANet

Packet Switching Principles

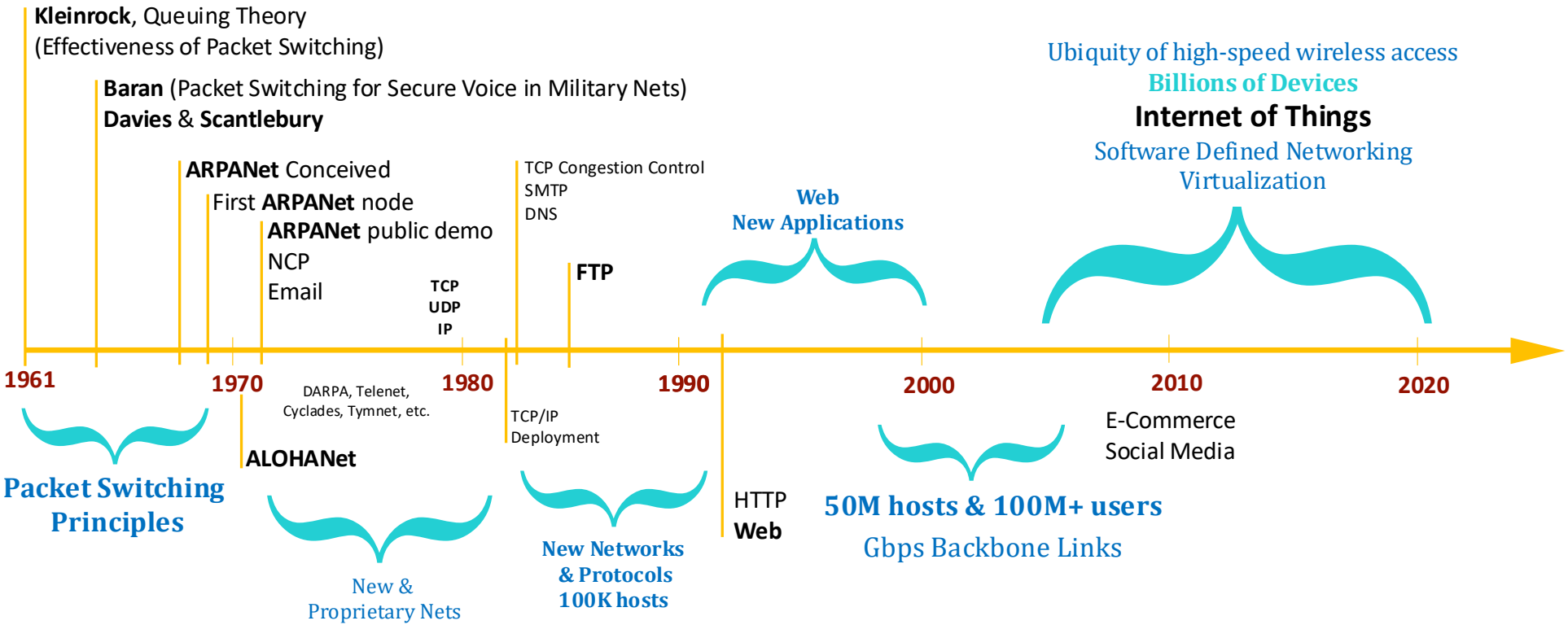
New & Proprietary Nets

New Networks & Protocols 100K hosts



Reference: Computer Networking : A Top-Down Approach. James F. Kurose, Keith W. Ross

History of Computer Networks

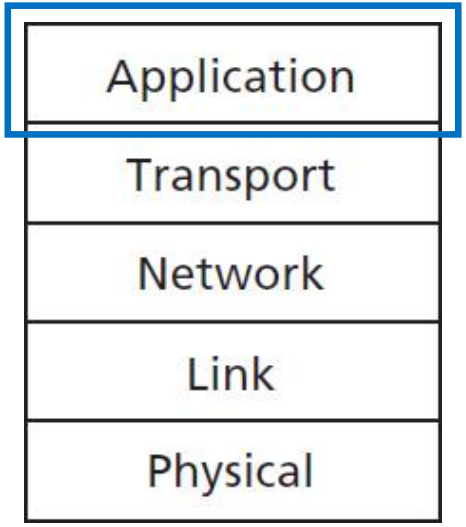


https://www.internetsociety.org/internet/history-internet/brief-history-internet/

Reference: Computer Networking : A Top-Down Approach. James F. Kurose, Keith W. Ross

Summary

- The Internet
- Basics
  - Network Edge
  - Network Core
  - Network Performance
- Protocol Layers and Reference Models
- Network Security: Networks Under Attack
- Brief History of Networking



Acknowledgements

- The following materials have been used in preparation of this presentation:

  [1] Textbook and (edited) Slides: Computer Networking: A Top-Down Approach
  James Kurose, Keith Ross
  7th and 8th Edition, Pearson
  http://gaia.cs.umass.edu/kurose_ross/

  [2] Reference: Computer Networks: A Systems Approach
  https://www.systemsapproach.org/book.html

- Recommended Additional Resources:

  [1] Interactive Exercises (Chapter One)
  http://gaia.cs.umass.edu/kurose_ross/interactive/

  [2] History of Networking (Podcast)
  https://rule11.tech/history-of-networking/