INTRODUCTION

What is a computer network?

Components of a computer network:

- hosts (PCs, laptops, handhelds)
- routers & switches (IP router, Ethernet switch)
- links (wired, wireless)
- protocols (IP, TCP, CSMA/CD, CSMA/CA)
- applications (network services)
- humans and service agents

Hosts, routers & links form the *hardware* side.

Protocols & applications form the *software* side.

Protocols can be viewed as the “glue” that binds everything else together.
A physical network:
Protocol example: low to high

- NIC (network interface card): hardware
  → e.g., Ethernet card, WLAN card
- device driver: part of OS
- ARP, RARP: OS
- IP: OS
- TCP, UDP: OS
- OSPF, BGP, HTTP: application
- web browser, ssh: application

→ multi-layered glue

What is the role of protocols?

→ facilitate communication or networking
Simplest instance of networking problem:

Given two hosts $A$, $B$ interconnected by some network $N$, facilitate communication of information between $A$ & $B$.

Information abstraction

- representation as objects (e.g., files)
- bytes & bits
  $\rightarrow$ digital form
- signals over physical media (e.g., electromagnetic waves)
  $\rightarrow$ analog form
Minimal functionality required of $A$, $B$

- encoding of information
- decoding of information

$\rightarrow$ data representation & a form of translation

Additional functionalities may be required depending on properties of network $N$

- information corruption
  $\rightarrow 10^{-9}$ for fiber optic cable
  $\rightarrow 10^{-3}$ or higher for wireless
- information loss: packet drop
- information delay: like toll booth, airport
- information security
Network $N$ connecting two or more nodes can be

- point-to-point links
- multi-access links
- internetworks

→ physical vs. logical topology
→ e.g., peer-to-peer, VPN

Network medium may be

- wired
- wireless

Node (e.g., hosts, routers) may be

- stationary
- mobile
**Point-to-point links**

\[ A \quad \quad \quad \quad \quad \quad \quad B \]

- various “cables”
- line of sight wireless communication
  \[ \quad \rightarrow \quad \text{directional antennas} \]
- no addressing necessary
  \[ \quad \rightarrow \quad \text{special case} \]
Multi-access links

- bus (e.g., old Ethernet)
- wireless media
  → omni-directional antennas
- broadcast mode (physical; not logical)
  → listen to everything: promiscuous mode
- access control: i.e., bus arbitration
  → resolve contention and recover from interference
- addressing necessary
Internetwork

- recursive definition
  - point-to-point and multi-access: internetwork
  - composition of one or more internetworks
- addressing necessary
- path selection between sender/receiver: routing
- how much to send: congestion control
- protocol translation: internetworking
- location management: e.g., Mobile IP
LAN (local area network) vs. WAN (wide area network) distinction:

- **LAN**: point-to-point, multi-access
- **WAN**: internetwork

→ geographical distinction is secondary
→ often go hand-in-hand
→ counter example?
Myriad of different LAN technologies co-existing in a WAN. For example:

- Fast Ethernet (100 Mbps)
- Gigabit Ethernet (1000 Mbps)
- WLAN (54 or 11 Mbps)
- FDDI (Fiber Distributed Data Interface)
- wireless Ethernet (11 Mbps, 54 Mbps)
- SONET
- ATM
- modem/DSL

→ WAN is a collection of LANs
Each LAN, in general, speaks a different language.

→ message format

→ procedural differences

Internetworking solves this problem by translating everything to IP . . .

→ technical definition of Internet

But:

→ IP is not necessary

→ e.g., large systems of layer 2 switches

→ trend: L2 (70s & 80s) → IP (90s) → L2 (Y2K+)

→ IP remains central glue
Remark on addresses (aka names):

Communicating entities are *processes* residing on nodes $A$ and $B$ running some operating system.

Thus an *address* must also identify which process a message is destined for on a host.

$\rightarrow$ e.g., port number abstraction
Key Issues

Fault-tolerance

The larger the network, the more things can go wrong.

E.g.: link/node failures, message corruption, software bugs

→ managing downtime: tier-1 providers

→ 99.999%

Two types of failures:

• independent

• correlated
In a network system with \( n \) components, assume a component fails with independent probability \( p \)

\[ \text{expected number of failures: } n \cdot p \]

\[ \text{probability of no failures: } (1 - p)^n \]

\[ \text{probability of } k \text{ simultaneous failures: } p^k \]

Thus correlated failures have miniscule probability.

\[ \text{exponentially small in } k \]
In reality, failures are not independent.

→ e.g., power outage, natural disasters

We have:

→ Murphy’s Law

• issue of reliable communication

• reliable network services

→ main principle: redundancy

• Examples:

  – routing of messages: alternate/back-up routes

  – domain name servers: duplication

  – transmission by space probes: forward error correction (FEC)

  → also used for multimedia traffic
Network security

Features:

• confidentiality: encryption
• integrity: message has not been tampered
• authentication: sender really is who she claims to be

→ “CIA”

→ foundation: cryptography

→ end-to-end

→ networking problem?
Modern security vulnerabilities:

- denial of service (DoS) attack
  - e.g., SYN flooding
- distributed DoS (DDoS) attack
  - e.g., commercial, personal, infrastructure
- worm attacks: e.g., CodeRed, Blaster, ...
  - buffer overflow: mainly bugs in MS DLLs
- spam mail (security issue?)
• with fault-tolerance impacts QoS (quality of service)
  → Aug. 04: US broadband deployment exceeds dial-up
• security: trade-off with overhead
  → what is the desired operating point?
  → too much $\Rightarrow$ too slow
  → too little $\Rightarrow$ too vulnerable

For example: secure routing (S-BGP)
  $\rightarrow$ “BBN vs. Cisco”
Big picture:

-→ points in the same spectrum
-→ malicious (Byzantine) vs. non-malicious
-→ availability
-→ service assurances
Performance

Issues:

• excessive traffic can cause congestion (analogous to highways)

• traffic volume exhibits large fluctuations
  → burstiness

• multimedia traffic is voluminous (even for single user)

• ubiquitous access
  → wired/wireless Internet

Potential for bottleneck development

→ spontaneous or persistent

→ similar consequences as failures
Different applications require different levels of service quality.

Challenges:

- how to provide customized QoS
- many users and applications: scalability
- must interoperate with legacy Internet

Current state:

- overprovisioning
  - “throw bandwidth at the problem”
  - tier-1 ISPs use sophisticated traffic engineering
- still no Internet QoS
  - changing with VoIP and content deployment
- not economic
  - few tier-1 providers make money
Data networking, telephony, and content convergence

→ Y2K+ trend

• VoIP (Voice-over-IP): wired world
  → traditional TDM-based telephony system is entirely separate network
  → economic factors are dictating convergence
  → from KaZaA to Skype

• cellular voice networks: 2G, 2.5G, 3G
  → what is 4G?
  → telcos/cellular providers are concerned
  → take-over by WLAN + IP?
  → strategy: active participation
• peer-to-peer: rampant content dissemination
  → from audio to movies
  → content providers need to get into the action
  → do not want to get into the action

$6$ question:
  → what will the wireless/wireline future hold?

Mixture of high bandwidth/low bandwidth networks, wireline/wireless, . . .