CPSC 441: Computer Communications

Instructor: Carey Williamson
Office: ICT 740
Email: carey@cpsc.ucalgary.ca

Class Location: ICT 122
Lectures: MWF 12:00 – 12:50


Slides are adapted from the book’s companion Web site, with changes by Anirban Mahanti and Carey Williamson.
Roadmap

- What is a Computer Network?
- Applications of Networking
- Classification of Networks
- Layered Architecture
- Network Core
- Delay & Loss in Packet-switched Networks
- Structure of the Internet
- Summary
Computer Network?

- “interconnected collection of autonomous computers connected by a communication technology”
- **What is the Internet?**
  - “network of networks”
  - “collection of networks interconnected by routers”
  - “a communication medium used by millions”
  - Email, chat, Web “surfing”, streaming media

- **Internet & Web**
The “nuts and bolts” view of the Internet

- millions of connected computing devices called hosts or end-systems
  - PCs, workstations, servers
  - PDAs, phones, toasters
  - running network apps
- communication links
  - fiber, copper, radio, satellite
  - links have different capacities (bandwidth)
- routers: forward packets
- packet: piece of a message (basic unit of transfer)
Roadmap

- What is a Computer Network?
- Applications of Networking
- Classification of Networks
- Layered Architecture
- Network Core
- Delay & Loss in Packet-switched Networks
- Structure of the Internet
- Summary
Applications (1)

- **end systems (hosts):**
  - run application programs
  - e.g. Web, email, ftp
  - at “edge of network”

- **client/server model**
  - client host requests, receives service from always-on server
  - e.g. Web browser/server; email client/server

- Client/server model has well-defined roles for each.
Applications (2)

- peer-to-peer model:
  No fixed clients or servers
  Each host can act as both client and server at any time

- Examples: Napster, Gnutella, KaZaA, BitTorrent
Applications (3)

- File transfer
- Remote login (telnet, rlogin, ssh)
- World Wide Web (WWW)
- Instant Messaging (Internet chat, text messaging on cellular phones)
- Peer-to-Peer file sharing
- Internet Phone (Voice-Over-IP)
- Video-on-demand
- Distributed Games
Roadmap

- What is a Computer Network?
- Applications of Networking
- Classification of Networks
- Layered Architecture
- Network Core
- Delay & Loss in Packet-switched Networks
- Structure of the Internet
- Summary
A Classification of Networks

- Local Area Network (LAN)
- Metropolitan Area Network (MAN)
- Wide Area Network (WAN)
- Wireless LAN (WLAN)
- Home Networks
- Personal Area Network (PAN)
- Body Area Network (BAN)
Local Area Network (LAN)

- company/univ local area network (LAN) connects end system to edge router
- Ethernet:
  - shared or dedicated link connects end system and router (a few km)
  - 10 Mbps, 100Mbps, Gigabit Ethernet
- widespread deployment: companies, univ, homeLANs
- LANs: chapter 5
**Metropolitan Area Network (MAN)**

“City sized”: tens of kilometers

A Cable TV Network is an example of a MAN

Typically 500 to 5,000 homes
Cable Network Architecture: Overview

cable headend

cable distribution network (simplified)

Home Environment
- Set-Top Box
- TV
- Cable Modem
- PC

Coax
Splitter
Coax
10 Mbps Ethernet

CPSC 441: Introduction 1-13
Cable Network Architecture: Overview

server(s)

cable headend

cable distribution network

home
Wide Area Network (WAN)

- Spans a large geographic area, e.g., a country or a continent
- A WAN consists of several transmission lines and routers
- Internet is an example of a WAN
Wireless Networks (WLANs)

- shared **wireless access** network connects end system to router via base station or “access point”
- **wireless LANs:**
  - 802.11b (WiFi): 11 Mbps
- **wider-area wireless access**
  - provided by telco operator
  - 3G ~ 384 kbps
    - Will it happen??
  - WAP/GPRS in Europe
  - WiMax available now

To the wired network:
- router
- base station
- mobile hosts
Home networks

Typical home network components:
- ADSL or cable modem
- router/firewall/NAT
- Ethernet
- wireless access point
“internetworking”?

- internetwork - interconnection of networks - also called an “internet”
- subnetwork - a constituent of an internet
- intermediate system - a device used to connect two networks allowing hosts of the networks to correspond with each other
  - Bridge
  - Router
- Internet is an example of an internetwork.
Roadmap

- What is a Computer Network?
- Applications of Networking
- Classification of Networks
- Layered Architecture
- Network Core
- Delay & Loss in Packet-switched Networks
- Structure of the Internet
- Summary
Layered Architecture: Why?

- Networks are complex with many pieces
  - Hosts, routers, links, applications, protocols, hardware, software
- Can we organize it, somehow?
- Let’s consider a Web page request:
  - Browser requests Web page from server
  - Server should determine if access is privileged
  - Reliable transfer page from server to client
  - Physical transfer of “bits” from server to client
Motivation Continued ...

Application Services

Communication Service

Network Services

Application logic

Reliable delivery

Transfer “bits”

Web Server

Web Client
Motivation Continued ...

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 network technology doesn’t affect rest of system
- layering considered harmful? (design vs implementation)
Layers, Protocols, Interfaces

Application Services

Communication Service

Network Services

Application logic protocol

Reliable delivery protocol

Transfer “bits” protocol

Web Server

Web Client
Layered Architecture

- Networks organized as a stack of layers?
  The purpose of a layer is to offer services to
  the layer above it using a well-defined
  interface (programming language analogy:
  libraries hide details while providing a service)
  Reduces design complexity

- Protocols: “horizontal” conversations at any
  layer $n$ (i.e., between peer layers)

- Data Transfer: each layer passes data &
  control information to the layer below;
  eventually physical medium is reached.
Layered Architecture (cont’d)

- A set of layers & protocols is called a Network Architecture. These specifications enable hardware/software developers to build systems compliant with a particular architecture.
  
  E.g., TCP/IP, OSI
Layering: Design Issues

- How many layers? What do they each do?
- How to identify senders/receivers?
  - Addressing
- Unreliable physical communication medium?
  - Error detection
  - Error control
  - Message reordering
- Sender can swamp the receiver?
  - Flow control
- Multiplexing/Demultiplexing
Network Reference Models

- **Open Systems Interconnection (OSI) Model**
  
  Classic 7-layer model (covered in Wed tutorial)

- **TCP/IP Model**
  
  Streamlined practical 4-layer protocol stack
# Reference Models (2)

<table>
<thead>
<tr>
<th>OSI</th>
<th>TCP/IP</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>Application</td>
</tr>
<tr>
<td>6</td>
<td>Presentation</td>
</tr>
<tr>
<td>5</td>
<td>Session</td>
</tr>
<tr>
<td>4</td>
<td>Transport</td>
</tr>
<tr>
<td>3</td>
<td>Network</td>
</tr>
<tr>
<td>2</td>
<td>Data link</td>
</tr>
<tr>
<td>1</td>
<td>Physical</td>
</tr>
</tbody>
</table>

TCP/IP does not have a session layer, as indicated by the shaded area.
TCP/IP Model: History

- Originally used in the ARPANET
- ARPANET required networks using leased telephone lines & radio/satellite networks to interoperate
- Goals of the model are:
  - Seamless interoperability
  - Wide-ranging applications
  - Fault-tolerant to some extent
The Application Layer

- Residence of network applications and their application control logic
- Examples include:
  - HTTP (Hyper-Text Transfer Protocol)
  - FTP (File Transfer Protocol)
  - Telnet
  - SMTP (Simple Mail Transfer Protocol)
  - DNS (Domain Name Service)
The Transport Layer

- Concerned with end-to-end data transfer between end systems (hosts)
- Transmission unit is called segment
- TCP/IP networks such as the Internet provides two types of services to applications
  - “connection-oriented” service - Transmission Control Protocol (TCP)
  - “connectionless” service - User Datagram Protocol (UDP)
TCP: Connection-oriented Service

- Handshaking between client & server programs
  Parameters for ensuing exchange
  Maintain connection-state
- Packet switches do not maintain any connection-state; state is at end systems hence “connection-oriented”
- Similar to a phone conversation
- TCP is bundled with reliability, congestion control, and flow control.
UDP: Connectionless Service

- No handshaking
- Send whenever and however you want
- A “best effort” service
  - No reliability
  - No congestion & flow control services
- Useful for network applications that prefer quick delivery of most packets rather than guaranteed (slow) delivery of all packets (e.g., VOIP, video streaming)
The Internet Layer

- End systems inject **datagrams** in the networks
- A transmission path is determined for each packet (routing)
- A “best effort” service
  - Datagrams might be lost
  - Datagrams might be arrive out of order
- Analogy: Postal system
The Host-to-Network Layer

- Somehow, host has to connect to the network and be able to send IP Datagrams
- How?
Internet protocol stack

- **application**: supporting network applications
  - FTP, SMTP, STTP
- **transport**: host-host data transfer
  - TCP, UDP
- **network**: routing of datagrams from source to destination
  - IP, routing protocols
- **link**: data transfer between neighboring network elements
  - PPP, Ethernet
- **physical**: bits “on the wire”
Layering: logical communication

Each layer:
- distributed
- "entities" implement layer functions at each node
- entities perform actions, exchange messages with peers
Layering: *logical* communication

- take data from app
- generate “segment” according to transport protocol
- add addressing, reliability check info to form “datagram”
- send datagram to peer
- wait for peer to ack receipt
Layering: physical communication
## Protocol layering and data

Each layer takes data from above
- adds header information to create new data unit
- passes new data unit to layer below

<table>
<thead>
<tr>
<th>source</th>
<th></th>
<th>destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>M</td>
<td></td>
<td>M</td>
</tr>
<tr>
<td>(\text{M} \rightarrow \text{T} \rightarrow \text{M})</td>
<td>application</td>
<td>application</td>
</tr>
<tr>
<td>(\text{M} \rightarrow \text{N} \rightarrow \text{M})</td>
<td>transport</td>
<td>transport</td>
</tr>
<tr>
<td>(\text{M} \rightarrow \text{L} \rightarrow \text{M})</td>
<td>network</td>
<td>network</td>
</tr>
<tr>
<td>(\text{M} \rightarrow \text{P} \rightarrow \text{M})</td>
<td>link</td>
<td>link</td>
</tr>
<tr>
<td></td>
<td>physical</td>
<td>physical</td>
</tr>
</tbody>
</table>

message  
segment  
datagram  
frame
Roadmap

- What is a Computer Network?
- Applications of Networking
- Classification of Networks
- Layered Architecture
- Network Core
- Delay & Loss in Packet-switched Networks
- Structure of the Internet
- Summary
The Network Core

- mesh of interconnected routers
- the fundamental question: how is data transferred through net?
  - circuit-switching: dedicated circuit per call: telephone net
  - packet-switching: data sent thru net in discrete “chunks”
Network Core: Circuit-Switching

End-to-end resources reserved for “call”

- Link bandwidth, switch capacity
- Dedicated resources with no sharing
- Guaranteed transmission capacity
- Call setup required
- “Blocking” may occur
Network Core: Circuit-Switching

- Capacity of medium exceeds the capacity required for transmission of a single signal
  How can we improve “efficiency”? Let’s multiplex.

- Divide link bandwidth into “pieces”: frequency division - FDMA
  time division - TDMA
  code division - CDMA (cellular networks)
  wavelength division - WDM (optical)
Circuit-Switching: FDMA and TDMA

### FDMA

- Frequency
- Time

### TDMA

- Frequency
- Time

Example:
- 4 users

Colors represent different users or channels.
Network Core: Packet-Switching

- “store-and-forward” transmission
- source breaks long messages into smaller “packets”
- packets share network resources
- each packet briefly uses full link bandwidth
- resource contention
  
  aggregate resource demand can exceed amount available
  
  congestion: packets queue, wait for link use
  
  analogy: Calgary commute at rush hour
Sequence of A & B packets does not have fixed pattern \(\Rightarrow\) statistical multiplexing.

In TDM each host gets same slot in revolving TDM frame.
Packet-switching versus circuit-switching

Is packet switching a “slam dunk” winner?

- Great for bursty data resource sharing
- Excessive congestion: 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 ongoing research problem
Packet-switching: store-and-forward

- Takes $L/R$ seconds to transmit (push out) packet of $L$ bits on to link or $R$ bps
- Entire packet must arrive at router before it can be transmitted on next link: *store and forward*
- delay = $3L/R$

**Example:**
- $L = 7.5$ Mbits
- $R = 1.5$ Mbps
- delay = 15 sec
Packet-Switching: Message Segmenting

Now break up the message into 5000 packets

- Each packet 1,500 bits
- 1 msec to transmit packet on one link
- *pipelining*: each link works in parallel
- Delay reduced from 15 sec to 5.002 sec
Packet-switched networks: forwarding

- **datagram network:**
  
  *destination address* in packet determines next hop
  
  routes may change during session (flexible?)
  
  no “per flow” state, hence more scalable

- **virtual circuit network:**
  
  each packet carries tag (virtual circuit ID), tag determines next hop
  
  fixed path determined at *call setup time*
  
  path is not a dedicated path as in circuit switched (i.e., store & forward of packets)
  
  *routers maintain per-call state*

- **datagram networks need per packet routing.**
Network Taxonomy

Telecommunication networks

Circuit-switched networks
- FDM
- TDM

Packet-switched networks
- Networks with VCs
- Datagram Networks
Roadmap

- What is a Computer Network?
- Applications of Networking
- Classification of Networks
- Layered Architecture
- Network Core
- Delay & Loss in Packet-switched Networks
- Structure of the Internet
- Summary
How do loss and delay occur?

packets queue in router buffers

- packet arrival rate to link exceeds output link capacity
- packets queue, wait for turn
- if queue is full, arriving packets dropped (Drop-Tail)

free (available) buffers: arriving packets dropped (loss) if no free buffers

packet being transmitted (delay)

packets queueing (delay)
Four sources of packet delay

1. Processing delay:
   check bit errors
   determine output link

2. Queueing delay:
   time waiting at output link for transmission
   depends on congestion level of router
Delay in packet-switched networks

3. Transmission delay:
- \( R = \text{link bandwidth (bps)} \)
- \( L = \text{packet length (bits)} \)
- time to send bits into link = \( L/R \)

4. Propagation delay:
- \( d = \text{length of physical link} \)
- \( s = \text{propagation speed in medium} (\sim 2 \times 10^8 \text{ m/sec}) \)
- propagation delay = \( d/s \)

Note: \( s \) and \( R \) are very different quantities!
Nodal processing delay

\[ d_{\text{nodal}} = d_{\text{proc}} + d_{\text{queue}} + d_{\text{trans}} + d_{\text{prop}} \]

- \( d_{\text{proc}} \) = processing delay
  - typically a few microsecs or less
- \( d_{\text{queue}} \) = queuing delay
  - depends on congestion
- \( d_{\text{trans}} \) = transmission delay
  - \( = \frac{L}{R} \), significant for low-speed links
- \( d_{\text{prop}} \) = propagation delay
  - a few microsecs to hundreds of msecs
Queueing delay (revisited)

- $R =$ link bandwidth (bps)
- $L =$ packet length (bits)
- $a =$ average packet arrival rate

Traffic intensity = $\frac{La}{R}$

- $\frac{La}{R} \sim 0$: average queueing delay small
- $\frac{La}{R} \to 1$: delays become large
- $\frac{La}{R} > 1$: more “work” arriving than can be serviced, average delay infinite!
"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 that will reach router $i$ on path towards destination
  - router $i$ will return packets to sender
  - sender times interval between transmission and reply.

![Diagram showing traceroute probes]

CPSC 441: Introduction
Roadmap

- What is a Computer Network?
- Applications of Networking
- Classification of Networks
- Layered Architecture
- Network Core
- Delay & Loss in Packet-switched Networks
- Structure of the Internet
- Summary
Internet structure: network of networks

- roughly hierarchical
- at center: “tier-1” ISPs (e.g., UUNet, BBN/Genuity, Sprint, AT&T), national/international coverage treat each other as equals

![Diagram](image)

- Tier-1 providers interconnect (peer) privately
- Tier-1 providers also interconnect at public network access points (NAPs)
Tier-1 ISP: e.g., Sprint

Sprint US backbone network
Internet structure: network of networks

- “Tier-2” ISPs: smaller (often regional) ISPs
  Connect to one or more tier-1 ISPs, possibly other tier-2 ISPs

Tier-2 ISP pays tier-1 ISP for connectivity to rest of Internet
- tier-2 ISP is customer of tier-1 provider

Tier-2 ISPs also peer privately with each other, interconnect at NAP
Internet structure: network of networks

“Tier-3” ISPs and local ISPs
last hop (“access”) network (closest to end systems)

Local and tier-3 ISPs are customers of higher tier ISPs connecting them to rest of Internet
Internet structure: network of networks

- a packet passes through many networks!
Introduction: Summary

Covered a “ton” of material!
- Internet overview
- what’s a protocol?
- network edge, core, access network
  - packet-switching versus circuit-switching
- Internet/ISP structure
- performance: loss, delay
- layering and service models
- Internet history (tutorial)

You now have:
- context, overview, “feel” of networking
- more depth, detail to follow!