CMPE 150: Introduction to Computer Networks

Katia Obraczka
Computer Engineering
UCSC Baskin Engineering

Lecture 3

CMPE 150 Winter’13
Lab Reminder

• Lab session schedule is online.
• Everyone needs to enroll in their preferred session!
• Friday’s extra lab session: 9am-1pm.
  – Also held as TA office hours.
Announcements

• Homework 1 is coming up.
  – Will be posted by the end of the week.
Last Class

• Definition of computer networks.
• Examples of networks.
• The Internet
  – Components: links, end hosts, and routers.
  – Collection of interconnected networks
• The Internet’s structure: hierarchical
  – Network edge.
  – Access networks.
    • Backbone.
• Protocols.
Today...
Introduction (Cont’d)

Fundamental concepts, terminology

Chapter 1
A Closer Look at Network Structure:

- Network edge: applications and hosts
- **Access networks**, physical media: wired, wireless communication links
- **Network core**:
  - interconnected routers
  - network of networks
The Network Edge

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

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

- **Peer-peer model:**
  - minimal (or no) use of dedicated servers
  - e.g. Skype, BitTorrent
Two Basic Problems in Communications

1. How to share a physical communication path across multiple users?
   - Key to reducing cost

2. How to communicate between users without a direct link between them?
   - Needed for constructing large networks
Early POTS Network
Early POTS Network

Circa 1903
Sharing a Communication Path

Two approaches:

1. Circuit Switching
   - Method used in Plain Old Telephone System
   - Customer buys fixed capacity for the duration of call
   - Example: 64,000 bits/second for a 5-minute phone call
   - Customer charged a per-minute fee

2. Packet Switching
   - Used in the Internet
   - Best effort model: Users send data when ready, network makes no guarantees
Circuit Switching

Each conversation takes a fixed slice of the network capacity on the phone network
Circuit Switching: FDM and TDM

Example:
4 users

FDM

TDM
Packet Switching

"I AM A DIGITAL WIRELESS INTERNET DEVICE!
IF YOU CALL ME A 'CELL PHONE' ONE MORE TIME, YOUR SERVICE WILL BE TERMINATED!"
Packet Switching

- Bit stream divided into chunks called packets
- A packet is a self-contained chunk of information transmitted on networks
  - Generated by some protocol
  - Typically contains a “header” and a “payload”
- Packet header may contain identifier of sender and receiver, sequence number to determine order, etc.
Addressing

• Sender and receiver identifier (id) are also called “addresses”.
• Hierarchical versus flat addresses.
Packet Switching: Statistical Multiplexing

Sequence of A & B packets does not have fixed pattern, bandwidth shared on demand ➔ statistical multiplexing.
Packet Switching

- Packets may be formed by encapsulating other packets
  - One protocol may take a packet generated by another protocol, add a header, and send it as a new packet
  - Occurs commonly when protocols are layered
- Also called “frames”, “datagrams”, “segments”, etc.
Packet Switching

each end-end data stream divided into packets

• user A, B packets share network resources
• each packet uses full link bandwidth
• resources used as needed

resource contention:
• aggregate resource demand can exceed amount available
• congestion: packets queue, wait for link use
• store and forward: packets move one hop at a time
  ❖ Node receives complete packet before forwarding

Bandwidth division into “pieces”

Dedicated allocation

Resource reservation
Voice on the Internet

Voice sent out as packets (network capacity used only when a packet is sent).
Does Packet Switching always win?

- “Free for all” model
  - Users get to transmit without making reservations
- Packets sent along the path go through switching points (routers)
  - Queues in routers could get long (long delay for users)
  - Routers might have to throw away packets when queues get too long (packet loss)
- Delay and loss are the two ways users pay for the cheaper service
Summary: Circuit versus Packet Switching

- 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”
Summary: Circuit Switching versus Packet Switching

- Phone networks allocate a fixed slice of their capacity to each call
  - The capacity is reserved for the duration of the call
  - Predictable quality (great for users)
  - Inefficient use of network capacity (oops, it costs more)
  - Technology called “circuit switching”

- Internet based on packet switching
  - No reservation of any resources in the network
  - “Pay as you go” model
  - Network capacity can be shared better (costs less for users)
  - Quality can be unpredictable
Protocol “Layers”

Networks are complex!
• 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)
- baggage (check)
- gates (load)
- runway takeoff
- airplane routing

- ticket (complain)
- baggage (claim)
- gates (unload)
- runway landing
- airplane routing

- a series of steps
Layering of Airline Functionality

Layers: each layer implements a service
- 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

• layering has its pros and cons
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
  - PPP, Ethernet
- **physical**: bits “on the wire”
ISO/OSI reference model

• Layering model developed by International Organization for Standardization (ISO)
  – Same folks who standardize hardware for doors and windows

• Called Open System Interconnect (OSI) model

• More comprehensive layering model than the Internet model

• Less interesting today because of the dominance of the Internet
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?
Packet Encapsulation

Application Layer

TCP Layer

IP Layer

Link Layer

Physical Layer

DATA FILE

TCP layer

IP layer

MAC header

Physical Layer

TRANSMIT
Encapsulation

source

destination

message $M$
segment $H_t M$
datagram $H_n H_t M$
frame $H_l H_n H_t M$

application
transport
network
link
physical

link
physical

network
link
physical

switch

router

CMPE 150 Winter’13
Network Performance
Performance Metrics

- Delay
- Loss
- Throughput
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

- Packets queueing (delay)
- Packet being transmitted (delay)

free (available) buffers: arriving packets dropped (loss) if no free buffers
Four sources of packet delay

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

\(d_{\text{proc}}\): nodal processing
- check bit errors
- determine output link
- typically < msec

\(d_{\text{queue}}\): queueing delay
- time waiting at output link for transmission
- depends on congestion level of router
Four sources of packet delay

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

\( d_{\text{trans}} \): transmission delay:
- \( L \): packet length (bits)
- \( R \): link bandwidth (bps)
- \( d_{\text{trans}} = \frac{L}{R} \)

\( d_{\text{prop}} \): propagation delay:
- \( d \): length of physical link
- \( s \): propagation speed in medium (~2x10^8 m/sec)
- \( d_{\text{prop}} = \frac{d}{s} \)

\( d_{\text{trans}} \) and \( d_{\text{prop}} \) very different
“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 process]
“Real” Internet delays and routes

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

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

trans-oceanic link

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

- 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
**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

![Diagram of server sending bits into a pipe that can carry fluid at rates Rs and Rc bits/sec]
Throughput (more)

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

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

bottleneck link  
link on end-end path that constrains end-end throughput
Throughput: Internet scenario

- per-connection end-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
Internet History

1961-1972: Early packet-switching principles

- **1961**: Kleinrock - queueing theory shows effectiveness of packet-switching
- **1964**: Baran - packet-switching in military nets
- **1967**: ARPAnet conceived by Advanced Research Projects Agency
- **1969**: first ARPAnet node operational
- **1972**:
  - ARPAnet public demonstration
  - NCP (Network Control Protocol) first host-host protocol
  - first e-mail program
  - ARPAnet has 15 nodes
Internet History

1972-1980: Internetworking, new and proprietary nets

- **1970**: ALOHAnet satellite network in Hawaii
- **1974**: Cerf and Kahn - architecture for interconnecting networks
- **1976**: Ethernet at Xerox PARC
- **late 70’s**: proprietary architectures: DECnet, SNA, XNA
- **late 70’s**: switching fixed length packets
- **1979**: ARPAnet has 200 nodes

Cerf and Kahn’s internetworking principles:
- minimalism, autonomy - no internal changes required to interconnect networks
- best effort service model
- stateless routers
- decentralized control

define today’s Internet architecture
Internet History

1980-1990: new protocols, a proliferation of networks

- **1983**: deployment of TCP/IP
- **1982**: SMTP e-mail protocol defined
- **1983**: DNS defined for name-to-IP-address translation
- **1985**: FTP protocol defined
- **1988**: TCP congestion control
- New national networks: Csnet, BITnet, NSFnet, Minitel
- 100,000 hosts connected to confederation of networks
Internet History

1990, 2000’s: commercialization, the Web, new apps

• Early 1990’s: ARPAnet decommissioned
• 1991: NSF lifts restrictions on commercial use of NSFnet (decommissioned, 1995)
• early 1990s: Web
  – hypertext [Bush 1945, Nelson 1960’s]
  – HTML, HTTP: Berners-Lee
  – 1994: Mosaic, later Netscape
• late 1990’s: commercialization of the Web

Late 1990’s – 2000’s:
• more killer apps: instant messaging, P2P file sharing
• network security to forefront
• est. 50 million host, 100 million+ users
• backbone links running at Gbps
Internet History

2007:
• ~500 million hosts
• Voice, Video over IP
• Peer-to-peer applications: BitTorrent (file sharing) Skype (VoIP), PPLive (video)
• more applications: YouTube, gaming, social networks
• wireless, mobility
How to find Internet protocol specifications?

- Called RFCs (Request for Comments)
- The first version of IP (Internet Protocol) is RFC 791 (from 1981)
- First version of HTTP protocol: RFC 1945 (May 1996)
Internet structure: network of networks

- roughly hierarchical
- at center: small # of well-connected large networks
  - “tier-1” commercial ISPs (e.g., Verizon, Sprint, AT&T, Qwest, Level3), national & international coverage
  - large content distributors (Google, Akamai, Microsoft)
  - treat each other as equals (no charges)

Tier-1 ISPs & Content Distributors, interconnect (peer) privately
... or at Internet Exchange Points (IXPs)
Internet structure: network of networks

“tier-2” ISPs: smaller (often regional) ISPs

- connect to one or more tier-1 (provider) ISPs
  - each tier-1 has many tier-2 customer nets
  - tier 2 pays tier 1 provider

- tier-2 nets sometimes peer directly with each other (bypassing tier 1), or at IXP
Internet structure: network of networks

- “Tier-3” ISPs, local ISPs
- customer of tier 1 or tier 2 network
  - last hop (“access”) network (closest to end systems)
Internet structure: network of networks

- a packet passes through *many* networks from source host to destination host