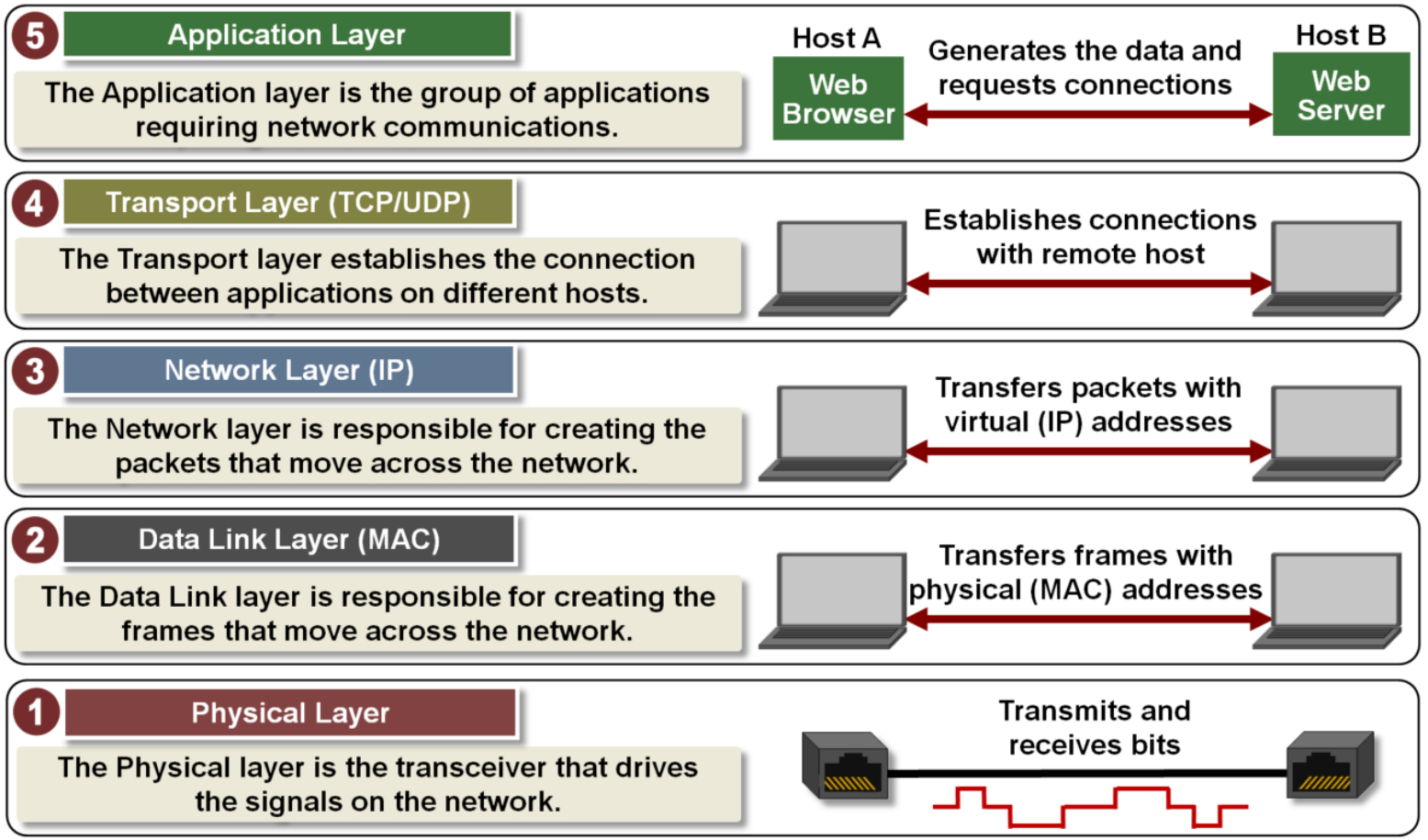
ENGSCI 233 Lecture 11.2

High-Level Networking

Today's learning objectives:

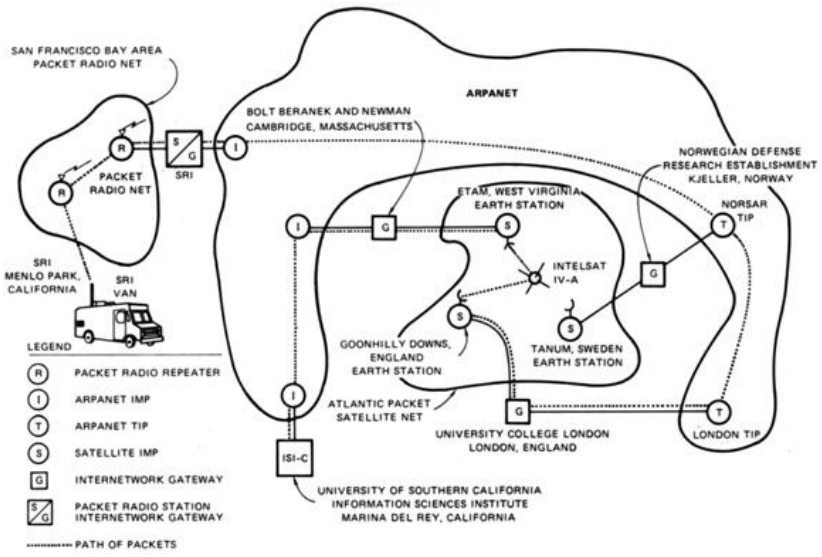
- Understand how the Internet works (at a very basic level)
- Discuss ways to manage network congestion
- Discuss ways to manage network reliability

The Internet is like an onion…



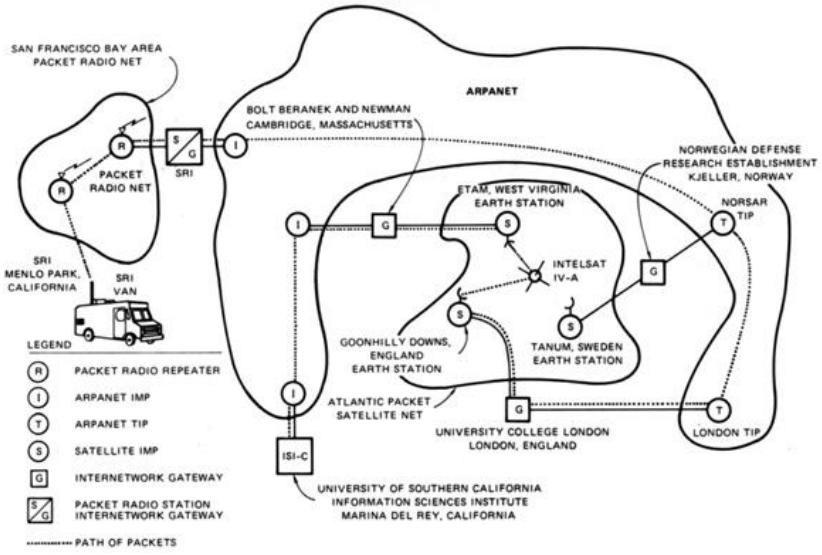
Internet is short for "internetworking."

- Originally, different networks used different protocols.
- The Internet Protocol (IP) was developed to communicate across them.



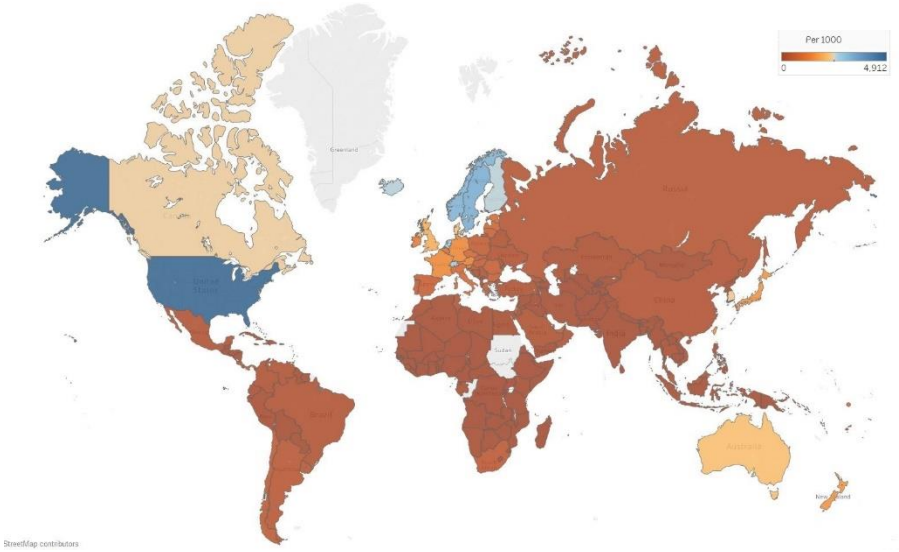
Internet Protocol is a bridge.

- Allows packets to be split up and reassembled
- Throws out old/stuck packets
- Assigns globally unique virtual addresses



There are two IP versions.

- IPv4 was the original practical version.
  - 32-bit addresses
    - Written like 192.168.0.1
  - There are more than 4 billion computers now!
- IPv6 deals with address exhaustion.
  - 128-bit addresses
    - 2001:0db8:0000:0000:0000:ff00:0042:8329
  - First 64 bits give network, second 64 give device



Some IP addresses are special.

- Private addresses only work in the local network
  - 10.x.x.x, 172.16.x.x, 192.168.x.x
- Multicast addresses are received by many
  - 224.0.0.0 – 239.255.255.255
- Loopback addresses refer to the local computer
  - 127.x.x.x, 0:0:0:0:0:0:0:1

How does the data get there?

- Networks are connected by devices called *routers*
- You probably have a home router

- Network backbones are made of routers.

```
 1     2 ms     2 ms     2 ms  net-208-gw-new.net.auckland.ac.nz [130.216.209.254]
 2     2 ms     3 ms     1 ms  cx-beta-sxj-400.net.auckland.ac.nz [172.18.0.66]
 3     2 ms     2 ms     2 ms  cxj-chi-to-cxj-beta.net.auckland.ac.nz [172.18.0.177]
 4     1 ms     1 ms     1 ms  br-cpf4-south.net.auckland.ac.nz [130.216.252.173]
 5     2 ms     2 ms     2 ms  br-asr2-br-cpf4-bond1.net.auckland.ac.nz [130.216.252.169]
 6     1 ms     1 ms     1 ms  210.7.37.37
 7   132 ms   132 ms   133 ms  reannz-1-lo-jmb-706.sttlwa.pacificwave.net [207.231.240.33]
 8   133 ms   132 ms   145 ms  abilene-1-lo-jmb-706.sttlwa.pacificwave.net [207.231.240.8]
 9   166 ms   165 ms   165 ms  ae-1.4079.rtsw.minn.net.internet2.edu [162.252.70.173]
10   221 ms   174 ms   174 ms  ae-1.4079.rtsw.eqch.net.internet2.edu [162.252.70.106]
11   175 ms   174 ms   211 ms  ae-0.4079.rtsw3.eqch.net.internet2.edu [162.252.70.163]
12   180 ms   181 ms   180 ms  ae-1.4079.rtsw.clev.net.internet2.edu [162.252.70.130]
13   187 ms   189 ms   188 ms  et-4-0-0.4079.rtsw.alba.net.internet2.edu [162.252.70.92]
14   190 ms   189 ms   190 ms  i2-re-chic-nox-mghpcc-gw1.nox.org [192.5.89.253]
15   199 ms   198 ms   198 ms  192.5.89.53
16   199 ms   198 ms   199 ms  nox1sumgw1-mit-re.nox.org [18.2.4.110]
17   199 ms   199 ms   199 ms  dmz-rtr-1-external-rtr-3.mit.edu [18.0.161.13]
18   199 ms   199 ms   199 ms  dmz-rtr-2-dmz-rtr-1-2.mit.edu [18.0.162.6]
19   206 ms   218 ms   200 ms  18.4.213.65
20   200 ms   199 ms   199 ms  bucket.mit.edu [18.25.0.50]
```

How do we know the route?

- Routers use a *routing table* to know where to send traffic
- Different groups of addresses can be routed different ways
- High-level protocols discover the route
- Choosing the best route uses algorithms like Dijkstra's

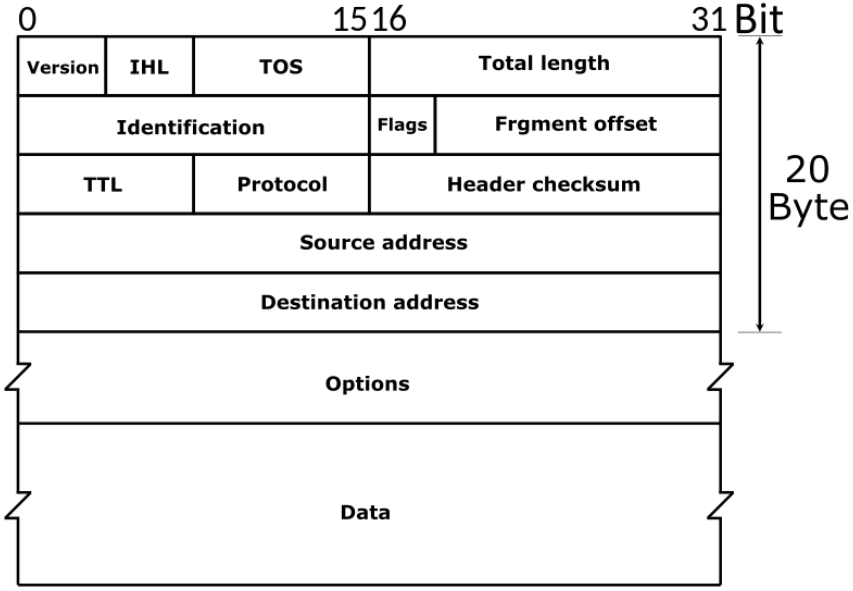
Example routing table contents

Network destination	Netmask	Gateway	Interface	Metric
0.0.0.0	0.0.0.0	192.168.0.1	192.168.0.100	10
127.0.0.0	255.0.0.0	127.0.0.1	127.0.0.1	1
192.168.0.0	255.255.255.0	192.168.0.100	192.168.0.100	10
192.168.0.100	255.255.255.255	127.0.0.1	127.0.0.1	10
192.168.0.1	255.255.255.255	192.168.0.100	192.168.0.100	10

How do you find the MAC address?

- The address resolution protocol (ARP) operates at the same layer as IP
  - Neighbor discovery protocol for IPv6
- Send a message to all MAC addresses, looking for the owner of an IP address

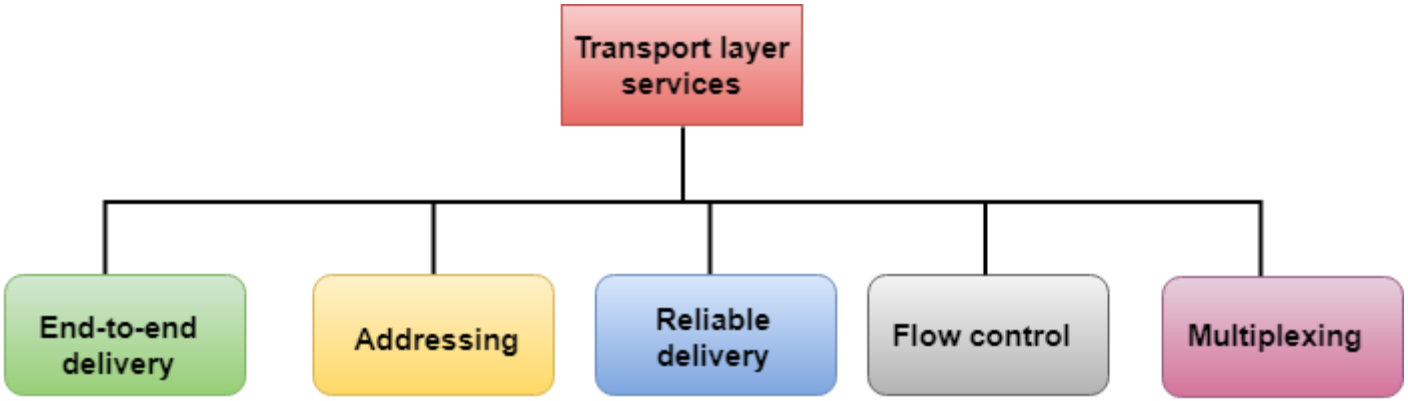
What data do we actually send?



- Packet ID and fragment offset to deal with fragmentation
- TTL = time to live
  - Number of router hops before abandoning transmission
- Protocol helps deal with next layer up
- Checksum for header, but nothing for data

How do we maintain a connection?

- IP can get data from one computer to another, but what does it mean?
- The transport layer tells the destination computer how to combine packets into useful data.



UDP is as reliable as the network.

- UDP = User Datagram Protocol
- *Ports* define different purposes for communication (application layer)
  - Port 80 is used for web browsing via HTTP
- We do get a data checksum this time
- No effort to deal with missing or out-of-order packets
- Good for streaming audio/video, or simple requests

UDP datagram header

Offsets	Octet	0								1								2								3							
Octet	Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	0	Source port																Destination port															
4	32	Length																Checksum															

TCP adds reliability, but at a cost.

- Transmission Control Protocol
- Have to exchange connection requests and acknowledgements to start a connection
- Reassembles packets in order
- Resends packets if not acknowledged
  - Vulnerable to denial of service
- Controls congestion

TCP segment header

Offsets	Octet	0								1								2								3							
Octet	Bit	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
0	0	Source port																Destination port															
4	32	Sequence number																															
8	64	Acknowledgment number (if ACK set)																															
12	96	Data offset				Reserved 0 0 0				N S	C W R	E C E	U R G	A C K	P S H	R S T	S Y N	F I N	Window Size														
16	128	Checksum																Urgent pointer (if URG set)															
20	160	Options (if *data offset* > 5. Padded at the end with "0" bytes if necessary.)																															
⋮	⋮																																
60	480																																

The application layer has cats.

- There are a wide range of different protocols
- All relate to exchanging useful information
  - Some help run the internet
  - Some make web browsing, email, and other common apps work
  - Some have special purposes

Some protocols form the plumbing.

- BGP (border gateway protocol)
  - How routers exchange routing information
- DHCP (dynamic host configuration protocol)
  - How new devices on a network get IP addresses
- DNS (domain name system)
  - How domain names (google.com) are translated to IP addresses (172.217.25.174)
- NTP (network time protocol)
  - Synchronizes time across a network

Web browsers use HTTP and HTTPS.

- Hyper text transport protocol
- Implements commands to get information from a server, or change data
  - GET – retrieves data from a specified location
  - POST – sends data for server use/interpretation
  - PUT – places data at a specified location
- Secure version adds encryption to prevent eavesdropping and spoofing

There are many application protocols.

- Email
  - POP, SMTP, IMAP
- Instant messaging
  - XMPP, IRC
- Telephone
  - SIP
- File transfer
  - FTP, BitTorrent
- Stock market data
  - FIX
- Remote desktop
  - RDP

Next week: Computer Software Architecture

Thank you!

Image References

Slide 1: *Map of the Internet*, by Randall Munroe, from https://xkcd.com/195/ (CC BY-NC 2.5)
Slide 3: by Microchip Technology, from https://microchipdeveloper.com/tcpip:tcp-ip-five-layer-model
Slides 4-5: by SRI International, from https://commons.wikimedia.org/wiki/File:SRI_First_Internetworked_Connection_diagram.jpg (CC BY-SA 3.0)
Slide 6: by Jroehl, from https://commons.wikimedia.org/wiki/File:List_of_countries_by_IPv4_address_allocation_per_1000_population.jpg (CC BY-SA 4.0)
Slide 7: from https://19216811.vn/en
Slide 9: from https://en.wikipedia.org/wiki/Routing_table (CC BY-SA 3.0)
Slide 10: by Raimond Spekking, from https://commons.wikimedia.org/wiki/File:Intel_Centrino_Wireless-N_1000-0659.jpg (CC BY-SA 4.0)
Slide 11: by Michel Bakni, from https://commons.wikimedia.org/wiki/File:IPv4_Packet-en.svg (CC BY-SA 4.0)
Slide 12: from https://www.minitool.com/lib/transport-layer.html
Slide 13: from https://en.wikipedia.org/wiki/User_Datagram_Protocol (CC BY-SA 3.0)
Slide 14: from https://en.wikipedia.org/wiki/Transmission_Control_Protocol (CC BY-SA 3.0)
Slide 15: by Neil Drumm, from https://www.flickr.com/photos/drumm/2989736147 (CC BY-SA 2.0)