Logistics

- Teams posted
- Project description posted (w/ dates!)

- **Today by 5pm: Programming assignment 1 posted.**
  - Simple exercise with TCP and UDP programming.
  - Due October 20.

- Book assignment will be posted next week.
  - You will have two weeks from posting to turn this in.

- First reading next week too.
  - Expect summaries to be turned in within one week of posting.
Questions?

- Models? Networking so far?
Logical vs. stack view of messaging

Credit: Wikipedia
Congestion control

- Congestion control addresses dynamic loads on the network.

- If a burst of activity occurs, we can find parts of the network get bogged down and we risk things like packet loss and buffer exhaustion.

- Congestion control algorithms are built into protocols and network layers to gracefully deal with this.

- Techniques might include holding packets when a node further along the path gets saturated, or dynamically scaling message delays or timeouts.
Internetworking

- Integrated networks composed of many subnetworks connected together.
- The internet is a prime example of this.
- Internetworking requires a method to find hosts on different subnets (addressing), and a protocol for communicating between subnets (such as IP).
- These live above the actual subnet technology so that heterogeneous types of networks can all interoperate.
Protocols for the Internet

- The main internet protocols originated in the 1970s on the ARPANET project.
  - The TCP/IP protocol was an important development of that project.

- **TCP**: Transmission Control Protocol
- **IP**: Internet Protocol

- The internet protocols are an example of a successful solution to the openness issue we discussed last time. The open standards allowed for widespread adoption, and the internet we see today.
TCP/IP and the protocol stack

Layers

Application

- Messages (UDP) or Streams (TCP)

Transport

- UDP or TCP packets

Internet

- IP datagrams

Network interface

- Network-specific frames

Underlying network
TCP/IP and encapsulation
UDP Encapsulation

- Credit: Wikipedia
Successful abstraction

- The TCP/IP specification is nice because it doesn’t specify anything about what happens below the Internet datagram layer.
- IP packets can be encapsulated within packets that are transmitted over most every lower level network technology.
Transport protocols

- The dominant transport protocols are TCP and UDP.
  - TCP
    - Transport Control Protocol
    - Connection-oriented, reliable protocol
  - UDP
    - User Datagram Protocol
    - No guarantee of reliable transmission
- These are layered above IP.
IP addressing

- Network layer protocol.
- Responsible for routing and addressing.

- We are all likely familiar with IP addresses: the numeric addresses of machines on the internet, such as 128.223.32.35.

- The addressing problem was one of the early challenges to building large scale, scalable networks.
IP addressing

- IP addresses are built out of four octets, or 8-bit numbers.

**Class A:**
- 0 | Network ID | Host ID

**Class B:**
- 1 0 | Network ID | Host ID

**Class C:**
- 1 1 0 | Network ID | Host ID

**Class D (multicast):**
- 1 1 1 0 | Multicast address

**Class E (reserved):**
- 1 1 1 1 0 | unused
IP addressing

- IP assigns one address to each host in the Internet.

- The classes of addresses were intended to meet the needs of different organizations.
  - Class A: Huge organizations (e.g.: NSFNet), $2^{24}$ hosts per subnet.
  - Class B: Large organizations with more than 255 hosts.
  - Class C: Less than 255 hosts.
  - Class D: Multicast
  - Class E: Reserved for experimental use.

- In recent times, this scheme has reached the limits of scalability.
IP packet structure

- The IP protocol is complex, but the basic packet structure has the form:

```
+-----------------+-----------------+
| header          | header          |
| IP address of source | IP address of destination |
+-----------------+-----------------+
| data            |                 |
+-----------------+-----------------+ up to 64 kilobytes
```

- The IP packet has enough information for a router in the network to make decisions related to how to move the packet from the source to the destination.

- Deeper details on the packet structure in Chapter 3.4.2 and 3.4.3 for routing.
Intermediate solutions for scalability

- Network Address Translation (NAT) enabled routers (such as your wireless router at home) use reserved addresses, such as 192.168.1.x, for your devices, hiding them behind a single “real” IP address that it holds.

- This removes the need for these devices to have their own IP addresses in the global pool, allowing for single real IP addresses to be multiplexed to multiple devices.
NAT

- Example:
  - 192.168.0.10 maps to 128.223.32.44:100
  - 192.168.0.11 maps to 128.223.32.44:101

- NAT router maintains table of mappings from internal, non-routable addresses (192.168.0.x) to ports exposed to the outside world.

- NAT can also associate unique external IPs with each internal IP if a pool is available.
NAT at work

DSL or Cable connection to ISP

Modem / firewall / router (NAT)

Ethernet switch

WiFi base station access point

Laptop

Game box

Media hub

TV monitor

PC 1

PC 2

Bluetooth adapter

Bluetooth printer

Camera
IPv6

IPv4 is the current IP in use on the Internet. IPv6 was designed primarily to tackle the address exhaustion of IPv4.

- 128-bit address space.
  - Tananbaum: $7 \times 10^{23}$ IP addresses per square meter of Earth.
  - Huitema: More conservatively, 1000 IP addresses per square meter. (Including oceans)

- Either way: IPv6 has far more address space than IPv4.
What happened to IPv5?

- Why did we go from IPv4 to IPv6? No 5?
  - IP version field.

- Version number “5” was used in the late 1970s for an experimental protocol.
  - As an IP protocol, it never really went anywhere.

- IPv6 takes on the “6” simply because “5” was already taken by a different effort.
Migrating IPv4 to IPv6

- Given that IP is the fundamental protocol of the Internet, changing it is not easy.

- The approach: build “islands” of IPv6 routers between IPv4 ones, and slowly grow the IPv6 ones.

- Fortunately, the IPv4 address space is embedded in the IPv6 space, so IPv6 routers can handle IPv4 traffic.
  - First 80 bits zero, next 16 bits ones, remaining 32 bits the IPv4 address.

- There is no real technical issue here: it’s just a matter of time to get developers and device manufacturers to switch over.
Process-to-process communication

- IP is concerned with host-to-host communication.
- Distributed systems require process-to-process communication above this.
- TCP and UDP serve this purpose.
- TCP and UDP handle dispatching packets to ports on a host once IP delivers the data.
UDP

- UDP is very basic.
- UDP is very close to the IP layer, so it incurs less overhead than TCP.

- UDP is unreliable.
  - Provides checksum to deal with data integrity.
  - Does not guarantee that a packet won’t be dropped.
  - There is no acknowledgement that messages are received.

- Has a use in specific applications if the developers can benefit from the lower overhead in the presence of unreliability.
TCP

- TCP provides reliable transport.
- Stream-oriented, so arbitrarily long messages can be sent (vs. the fixed length messages of UDP).
  - Handshaking occurs to establish a bidirectional channel between processes on either end of the connection.
  - **Sequencing**: One can determine the position of packets in the larger stream.
  - **Flow control**: TCP defines a method for timing out and retrying transmission when congestion occurs.
  - Buffering is provided by TCP.
  - Checksums for data integrity checks.
Naming

- The internet also provides domain name services.
- Named entities in the network are called domains.
- The symbolic name of a domain is, obviously, a domain name.
- Host and subnet names are prefixes on these domains.
- Hierarchical naming.
- E.g.: [www.cs.uoregon.edu](http://www.cs.uoregon.edu)

- We cover details later in chapter 9.