CS193i, Stanford | Handout #3
Spring, 2000-01 | Nick Parlante

Networking

What is the Internet?

What does it accomplish?

How does it look to its users?

How does it work?

How is it built? How do you write code to use it?

Bits -- K, MB

Units

"1k" = ~1000 bytes = ~1000 characters

1 page text = 2k

small image = 10k (compressed format, such as GIF or JPEG)

screen size image = 100k

"1 Meg" = ~1000k (aka "1 MB")

Typical computer program 1-3 Megs

"1 Gig" = 1000 Meg

Hard disk: $200 currently buys a 40 GB drive --> $0.005 / MB (was 6 GB in 2000, 4 GB in 1999 $0.05 / MB)

$0.42.MB of RAM memory (was $1.5 / MB in 2000, $3 / MB in 1999)

Price Drops

It's the exponential price decrease of "bit handling" hardware is pretty impressive.

They're just bits

This is why, for example, a free email service is a possible business --> it's just bits, and they're cheap and getting cheaper all the time.

Network Basics

Call

Depends on some addressing scheme -- "namespace" for the computers

Send / Receive

Copy bytes from the sender to the receiver

Technologies

Switched circuit

Packets

Measures

Bandwidth

"Bandwidth" -- bits per second (Kbps, Mbps, Gbps) used by EE types. It's easy to confuse millions-of-bits-per-second Mbps with meagabytes-per-second MBps. It's unfortunate that "bit" and "byte" begin with the same letter.

Bytes per second (K per second, MB per second) used by software types. Figure roughly 10 bits per byte when converting to allow for overhead and so the math is easy.

Latency

"Latency" -- the initial delay for the first usable bit to go from the source to the destination. The time required for a "dialog" type protocol between two computers can be largely bound by latency rather than bandwidth, even though bandwidth is the more frequently quoted figure.

Ping-pong "roundtrip" -- e.g. a simple ping-pong type dialog between two computers (send:"hi I'm Alice"... respond:"Hi, I'm Bob") will take at least 2x the latency, no matter how fast the bandwidth is.

Road-trip example

Load up your car with 1000 DVD-RAM disks of data (1000 x 5GB = 5000 GB). Drive it from SF to NYC. in 3 days = 86400 x 3 = 259200 seconds.

Bandwidth = 19MB/second = not bad. Latency = 3 days = terrible.

Could use 80 GB tapes to increase bandwidth, but won't help with latency.

"It's the latency, stupid"

http://rescomp.stanford.edu/~cheshire/rants/Latency.html

Article by Stuart Cheshire: latency is at times more important and is generally harder to improve than bandwidth

Current Technologies

Typical phone line / modem type application = 53 Kbps =~5K / sec

Cable Modem -- 500kbps -- 50k / sec

Digital Subscriber Loop (DSL) 500kbps -- 50k / sec (The local phone company monopolies have not been a good thing for development in this area -- fortunately the Cable companies scared them.)

10T Ethernet connection. Uses a 10Mbps signaling. Ideal case 2 computers in a room. May approach 900k/sec. Around 500k/sec more likely, with reductions the more traffic (from other computers) the ethernet must carry at the same time. Latency can approach 1 thousandth of second -- 1 ms.

USB 12Mbps -- 1.2MB/sec (10x faster USB 2.0 is in the works)

Standard SCSI -- 5MB/ sec

Fast SCSI -- 10MB/sec (a current 9 GB hard drive can just about saturate this on sustained reads. Hard drives have latencies on the order of 10ms.)

Ultra Wide SCSI -- 40MB/sec

Ultra2 Wide SCSI -- 80MB/sec

Firewire aka 1304 400Mbps -- 40MB/sec (800 Mbps and 1.6Gbps are rumored)

Fast Ethernet 100T (5-10MB/sec) and 1000T (50-100MB/sec) are increasingly in use. Capacity depends very much on how much "sharing" is going on on the network link and the ability of the computers involved to keep the pipe stuffed with data.

Serial vs. Parallel hardware. Paradoxically, the fastest future directions appear to be serial based. (Cheaper cables, remove the difficulty of keeping the parallel signals in synch.)

Copper can go up to around 1 Gbps. Wire quality, shielding, and length all matter. (There are 1 Gbps ethernet standards for both copper and fiber, but copper versions can't span very far -- say 100 Meters)

Fiber optic can go to around (in the lab) 1000 Gbps. Or 10^12 bps -- on the order of a million simultaneous channels of video. More expensive than copper, but getting cheaper. A little harder to install, has fancier connectors, can't go around sharp bends. Current fiber standards run around 5 Gbps.

Data Rates

1 hour stereo CD = 600 MB (not compressed)

Uncompressed CD sound bandwidth = 160k/sec (16 bit samples, 44kHz, stereo)

1 hour of MP3 sound = 60 MB (compressed, sounds almost as good as CD)

MP3 sound bandwidth = 15k / sec (depends very much on compression settings)

Uncompressed video is on the order of 20MB/sec -- video compresses very, very well

Compressed VHS video = ~1.5M bps = 150k bytes / second. (this is how those little satellite TV dishes work). Also, how Tivo works.

Old Style: Central Switched Circuit

"Connection Oriented"

Old

Old Phone Company Way

Phone calls are not necessarily handled this way any more, but the switched circuit concept will come up later.

Call

The "call" operation sets up an end-to-end connection.

In the old days, this was a wire connection all the way between the phones.

Good

Latency and bandwidth available don't vary during the call -- important for some applications --e.g. live 2-way video stream.

If the call is long, the set up cost is relatively cheap since it's a 1-time cost amortized over all the traffic for the call

"Routing" during the call is not hard -- all done up front

Bad

If the call is short, the setup is relatively costly

- The call is using those resources, even during periods of silence -- potentially overall inefficient.
- The switching logic is centralized into the "center"

LANs

- Local Area Network -- computers in physical proximity. 200 computers within 1/4 mile radius -- one floor of a building scale.
- Problem: want to get bytes from one computer to another
- Could have centralized phone system type setup - switching logic gets messy

Packets

- A relatively small bundle of bytes (like a small file) — perhaps 1k in size.
- The source divides what is to be sent into a series of relatively small packets.
- A typical text email message takes up a few packets. Audio takes a stream of thousands of packets.
- Packet = "header" information for routing + "body" of bytes to be transmitted. The body can contain, text, voice, video....the transport system is independent of the purpose.
- Each packet starts at the source and is communicated hop by hop across the routers to the destination. The packets are re-assembled at the destination.

Packets Analysis

"bursty"

- The rate at which the packets arrive at the destination will tend to be "bursty" and a little unreliable: silence, then 5 packets show up in one group, more silence, 1 packet, silence, 8 packets, ...
- Fine for text, web pages
- Bad for real time telephone, video
- Q: how do you do audio or video? Related Q: why is there a 1 second delay when you start up your CD player? One solution: buffer up enough data to smooth out the bursts. This is how RealAudio and Quicktime work.

Good

- Potentially efficient: the packets from many concurrent connections can share resources along the way. Over time, the silence in one connection is utilized by traffic from another.

Ship Container analogy

- Suppose you are sending stuff from SF to Hawaii
- Circuit switched: we reserve you an entire container. You put your stuff in it. There tends to be unused space in the container.
- Packets: you give your stuff to the shipper. The shipper packs your stuff along with the stuff of others into the containers. All the containers on the ship are full.

vs. Connection

Compare to "connection oriented":. Large startup cost to build a dedicated "connection".. Inefficient use of the silent periods in the conversation. Efficient if we are talking 100% of the time. Latency and bandwidth are predictable.

Routing

Route all packets independently (TCP/IP) vs. route all packets along per-connection assigned path (ATM). Disadvantage that each packet has its own routing computation. However, tends to use the overall resources efficiently since bursty traffic can interleave.

Point: Different from and typically more efficient than a telephone connection

Point: Efficient = **cheap**

QOS

Quality of Service (QOS) extensions to packet communication -- try to guarantee latency/bandwidth for some of the traffic. Complicates routing. TCP/IP does not do a good job currently with QOS.

Broadband Ethernet LAN technology -- One of the Cleverest things ever

(editorial) An admirably elegant way to cheaply connect a group of computers -- original design credited to Bob Metcalfe at Xerox PARC.

Original design was for a radio based network -- read the story in Accidental Empires, by Robert X Cringely.

All computers on the LAN have an address

There is one wire, and all the computers are connected to it -- they share the wire -- only one computer is supposed to transmit at once.

Much simpler than the "central hub" shape of a telephone type network. Simpler = cheaper.

All computers listen to the wire at all times

Ethernet uses a single cable as the shared medium

Could use the same sort of strategy with any shared medium -- e.g. a radio frequency

Sender divides their message into small "packets" of, say, around 1000 bytes. Every packet begins with the address of the recipient.

Ethernet Protocol

The sender listens, waiting for a period of silence.

When there’s a period of silence, sender broadcasts the packet

Everybody listens all the time -- ignoring packets not for them

Sometimes two transmission overlap, and so the packets "collide" and get garbled. This can happen because (speed of light and all that) both senders can start sending before each other's signals have propagated down the wire to each other. The network card can usually detect this collisions and so know to re-transmit.

Have a “wait/re-transmit” protocol to re-send packets -- wait a random amount of time -- one of Metcalfe's breakthrough ideas. Also part of what makes ethernet a little unpredictable in terms of performance, especially latency.

Ethernet Conclusions

Shared

there's just the one wire and everybody uses it

Distributed and cooperative

no central control -- surprisingly decentralized

Insecure

not to hard to listen and pick up packets not intended for you

Random

Performance degrades as more computers use the shared medium

A little unpredictable how long it will take to get a message through

Compare to the telephone system "connection" oriented service -- connection services is more predictable but typically less efficient overall.

Works really well overall

The uncertainty of ethernet has proved to not be a big practical problem. Ethernet works so well that it's irresistible. IBM's competing "Token Ring" technology provides less uncertain traffic -- the rumor goes that that's why "uncertainty" was getting so much press as an important problem. IBM did not like using Xerox technology, so had to think of a reason to promote their technology.

WAN

Wide Area Network

Connect separate LANs

Routers

To understand the Internet, want to define the idea of a router -- a device that connects two LANs

The router must be able to communicate on both the LANs -- it will translate between the two

Listen to packets on both sides

Retransmit packets on one LAN from the other when necessary -- eg. when the sender is on one LAN, and the recipient is on the other