CSE 222
Graduate Networking

Winter 2001

Lecture 2: Internet Design

Geoffrey M. Voelker

Overview

- Current Internet architecture at a high level
  - Should seem like a review
- Clark Internet design philosophy paper

Network Components

- Hosts: Computer, PDA, light switch…
- Link: Transmission media
  - Wired or wireless
  - Broadcast or switched
- Switch: Crossroads, move bits between links
  - Packet switching: stateless store and forward (Internet)
  - Circuit switching: stateful, cut through (POTS)
- Protocol: Agreement on how information is to be formatted and transmitted

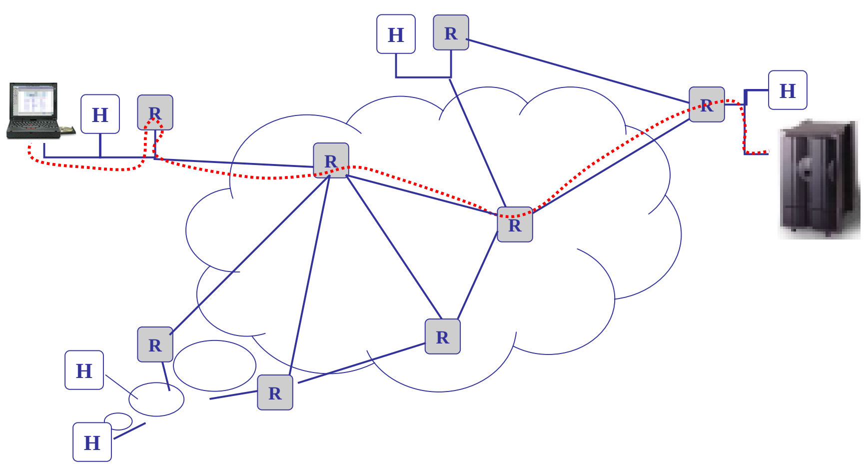
Internetwork

- An internetwork is a network of networks
  - Could do this in more than one way
  - IP is just one, POTS is another
- Routers (gateways) move packets between networks
  - Packet switched or circuit switched
  - Could also translate
- The Internet
  - Routers move IP packets from one network to another
    - » IP "dial tone"
  - Minimal requirements on underlying network
  - Can use almost any potential network or link layer
    - » Modem, Ethernet, token ring, ATM, ADSL, cell phone, cable…

How do we communicate in the Internet?

- Need to specify how to handle
  - Addressing
  - Routing
  - Service models
  - Failures
  - Management
  - Accounting
  - Etc.

Using the Web



Addressing

- Domain name: www.ucsd.edu
  - Global (across networks), human readable
- IP: 132.239.50.184
  - Global, what is actually used in routing
- Ethernet: 00-90-27-BD-BC-F7
  - Local, used within a particular network

- When making a connection…
  - Domain name is converted to an IP address
  - IP address used in packets
  - Packets sent on Ethernet to Ether address of the gateway

Routing

- When a gateway receives a packet, it needs to decide what to do with it
  - If it is to itself, deliver it
  - Otherwise, figure out which link to send it to (packet switching)
- Routing tables
  - Map IP prefixes to output links using forwarding tables
  - Per-hop, router does not need to know final destination
  - Automatically updated in response to failures, changes
- Internet routing done at two levels
  - Within a domain, across domains
- Much more later…

Service Models

- What can you expect from a particular protocol?
  - Unreliable (UDP)
  - Reliable (TCP)
  - Timely (RTP)
    - » No guarantees on Internet, though!

Failures

- Internet delivery is best effort – no guarantees
- Routers go down
  - Use a different route (how do routers figure out alternates?)
- Data corruption
  - Can happen in many places – any piece of hardware a packet travels over (link, router I/O bus, router memory, etc.)
  - Hardware CRCs, software checksums to detect
- Reordering
  - Sequence numbers (how large?)
  - Buffer packets at end points (how much?)
- Drops
  - Router link buffers can fill up (congestion)
  - Need to drop (which ones?)

Clark: Design Philosophy of the DARPA Internet Protocols

- Great paper
  - Not many papers explaining the motivation and reasoning that went into the design of systems that we take for granted
- Note that this was written 15 years after the project began
  - And the paper itself is already 13 years old!
- The setting
  - Multiple research and military networks
  - How do we connect them so that they can talk to each other?
  - Hard to imagine, but this was *before* LANs

Before going into the text…

- Meta-points
  - The Internet, TCP/IP, etc., were designed and engineered – there is no natural law that says the Internet had to look the way it does now
    - » It could well have been done differently
  - "The Internet", "TCP/IP", etc., continually evolve
    - » The Internet today is not the same Internet as 1988, 1973
    - » TCP/IP have changed considerably over the years
    - » We're using IPv4, with IPv6 in the works
  - Seemingly straightforward decisions can have very subtle correctness and performance implications
    - » EOL and PSH
    - » Acknowledging bytes vs. packets (debates still rage…)

Primary Goal

- "Effective technique for multiplexed utilization of existing interconnected networks"
  - Minimal assumptions about underlying networks
    - » No support for broadcast, multicast, real-time, reliability, etc.
    - » Nets could support it, but not necessary
    - » Extra support could actually get in the way (X.25 example)
  - Packet switched, store and forward
    - » Matched application needs, nets already packet switched
    - » Enables find-grained resource sharing
  - "Gateways" interconnect networks
    - » Routers today

Why is this hard?

- Every network characteristic is different between two arbitrary networks
  - Addressing
    - » Each network media has a different addressing scheme
  - Bandwidth
    - » Modems to terabits
  - Latency
    - » Seconds to nanoseconds
  - Packet size
    - » Dozens to thousands of bytes
  - Loss rates
    - » Differ by many orders of magnitude

Possibilities

- All nets communicate using a common format
  - Internet: IP over everything
  - To talk across networks, you send IP packets
  - Internal to a network, can use whatever you want
    - » Raw Ethernet, ATM, etc.
- Translate packets from one network to another
  - Convert Ethernet to ATM
  - Convert IP to OSI
  - X.25

Secondary Goals

- Survivability
- Multiple communication services
- Accommodate a variety of networks
- Distributed management of resources
- Cost-effective
- Minimal effort to attach hosts to net
- Resource accounting

- These are in order of priority -- a different priority ordering would likely result in a different design

Survivability

- Internet
  - Assume anything can fail between two end points
    - » Fate-sharing (state is lost only when end point is lost)
  - Designed to tolerate 1% drop rate
    - » High or low? Depends on the app, assumptions, goals, etc.
    - » 10% things break down
  - Routing is simpler, does not have to adapt to failure
    - » For a given datagram
- POTS (the *other* global network)
  - Ultra reliable switches
  - Self-healing
    - » Hardware switch over in the middle of a phone call

Survivability Implications

- End points maintain all essential state
  - Routers are stateless ("soft state")
  - End points responsible for recovering from failures
- Host machines are trusted
  - Have to rely upon hosts to implement the protocols correctly
    - » For performance as well as correctness
  - Easy to be malicious
    - » Ex: source addresses (everything in an IP packet) are trusted (IP spoofing)
- Can be difficult to determine source of failures
  - Not much feedback from network back to end point
  - Makes performance optimizations more difficult

Types of Service

- Bottom line: Best effort datagram service
- Building block
  - Unreliable (UDP)
  - Reliable service (TCP)
    - » Even two types of this: what are they?
  - Real-time
    - » Can you do real-time without help from network?
  - Multicast
- Reason why TCP and IP became separate protocols
  - IP basis for all other protocols
  - Originally were combined

Network Variety

- Fundamental goal was to interconnect networks…
- Internet successful in part because its design met this goal extremely well
  - Topology: point-to-point, bus, ring, radio, satellite, etc.
  - Characteristics: modem to Tbit speeds, us to sec delays
- Does not mean that IP on a given network is efficient
  - X.25 supports reliable delivery
  - ATM uses 53 byte cells, poor fragmentation for IP packets
    - » IP gets layered on top of cells, more effort at end points

Other Goals

- Distributed management
  - Different parts of network owned, controlled, managed by separate, Difficult to support, though
    - » Hard to do things across entities (e.g., optimize routes)
    - » Not sure what optimize means (to an ISP, customer?)
  - Problem in 88, more of a problem today
- Cost-effective (compared to?)
  - IP routers cheap compared to POTS switches
  - But higher drop rates, inefficient routing, end-to-end reliability all impose an overhead/inefficiency on the network
  - Still a source of religious debate

Other Goals (2)

- Attachment cost
  - $100M/year on protocol stacks by major OS vendors
    - » Is that cheap or expensive given 100s of millions of users?
  - Bugs still result in misbehaving hosts!
    - » Both correctness and performance
    - » There are tools that can tell you what OS, TCP/IP stack you are using based upon the bug signatures in implementations
- Accountability
  - Who pays for all of this?
  - What is the economic model of the future?
  - No one knows (plenty of ideas, though)
  - Another RD (religious debate)

Discussion

- What surprised you most about the Internet design and architecture when you read the paper?
- What is different about the Internet today than in '88 when Clark wrote this paper?
- What are limitations to the Internet design?

- When thinking about the Internet, imagine it as the culmination of many iterations of design and trial and error – not as an architecture that was preordained

For Next Time...

- Send me email if you aren't on the list yet
- Read Saltzer84 and Clark90 papers
- Browse Chapter 3, read Chapter 4
- Optional
  - Zimmerman80 – OSI reference model
  - Shenker95 – Should there be a new service model for the Internet to support multimedia applications?