Principles of Internet Design

CS204: Advanced Computer Networks
Oct 4, 2023
Adapted from Jiasi’s CS 204 slides for Spring 23
Agenda

• Internet History
• More Overview
  • Modularity through layering
  • Data, control, and management plane
  • Best-effort packet delivery
• Design philosophy of the Internet
  • What we have now
  • How it came about

Q: What are the general principles of Internet design?
Internet History
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 demo
  - NCP (Network Control Protocol) first host-host protocol
  - first e-mail program
  - ARPAnet has 15 nodes
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 (ATM precursor)
- **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
- **1985**: 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

2005-present
- ~1 billion hosts
  - Smartphones and tablets
- Aggressive deployment of broadband access
- Increasing ubiquity of high-speed wireless access
- Emergence of online social networks:
  - Facebook: nearly 2 billion active users
  - Tiktok: 1 billion monthly active users
- Service providers (Google, Microsoft) create their own networks
  - Bypass Internet, providing “instantaneous” access to search, email, etc.
- E-commerce, universities, enterprises running their services in "cloud" (e.g., Amazon EC2)
- Live video conference (e.g., Zoom)
More review on Internet
Modularity through layering
Protocol Stack

- Application
  - Applications
  - Reliable streams
  - Messages
- Transport
  - Best-effort *global* packet delivery
- Network
- Link
  - Best-effort *local* packet delivery
IP Suite: End Hosts vs. Routers

HTTP message
TCP segment
IP packet
Ethernet interface
SONET interface
Ethernet interface
SONET interface
Ethernet interface
SONET interface
Ethernet interface
SONET interface
Ethernet interface
Layer Encapsulation

User A

Get index.html

Connection ID

Source/Destination

Link Address

User B
The “Narrow Waist” of IP

The waist facilitates interoperability

The Hourglass Model

The waist facilitates interoperability
The narrow waist of IP

Q: Why does the Internet protocol stack resemble an hourglass?

Q: Why does the Internet protocol stack resemble an hourglass?

• Theory 1
  • IP is a global address, so no need for two naming systems?

• Theory 2
  • Lower layers are diverse (e.g. wireless, optical, cable)
  • Higher layers are also diverse (e.g. voice, video, file transfer)
    → IP layer in the middle must be more general (and hence unique)?

• Theory 3
  • Analytic birth/death model?
Any Alternatives to TCP/IP?

- Named data networking (NDN)
  - Universal names for content, instead of IP addresses
    - e.g., weather/riverside/yesterday, video/horror/freddy/chunk1
  - Forwarding and routing based on name prefixes
Data, Control, and Management Planes
Inside the Network

Forward packets from the sender to the receiver
Split into Data vs. Control Plane

- **Data plane**: packets
  - Handle individual packets as they arrive
  - Forward, drop, or buffer
  - Mark, shape, schedule, ...

- **Control plane**: events
  - Track changes in network topology
  - Compute paths through the network
  - Reserve resources along a path

Motivated by need for high-speed packet forwarding
Adding the Management Plane

• Making the network run well
  • Traffic reaches the right destination
  • Traffic flows over short, uncongested paths
  • Unwanted traffic is discarded
  • Failure recovery happens quickly
  • Routers don’t run out of resources

• A control loop with the network
  • Measure (sense): topology, traffic, performance, …
  • Control (actuate): configure control and data planes

• Time scales?
Best-Effort Packet-Delivery Service
Host-Network Division of Labor

- **Packet switching**
  - Divide messages into a sequence of packets
  - Headers with source and destination address
- **Best-effort delivery**
  - Packets may be lost
  - Packets may be corrupted
  - Packets may be delivered out of order
Host-Network Interface: Why Packets?

• Data traffic is bursty
  • Logging in to remote machines
  • Exchanging e-mail messages

• Don’t want to waste bandwidth
  • No traffic exchanged during idle periods

• Better to allow multiplexing
  • Different transfers share access to same links

• Packets can be delivered by most anything
  • RFC 1149: IP Datagrams over Avian Carriers
Host-Network Interface: Why Best-Effort?

• Never having to say you’re sorry...
  • Don’t reserve bandwidth and memory
  • Don’t do error detection & correction
  • Don’t remember from one packet to next
• Easier to survive failures
  • Transient disruptions are okay during failover
• Can run on nearly any link technology
  • Greater interoperability and evolution
Intermediate Transport Layer

• But, *applications want efficient, accurate transfer of data in order, in a timely fashion*
  • Let the end hosts handle all of that
  • (An example of the “end-to-end argument”)
• Transport layer can optionally...
  • Detect and retransmit lost packets
  • Put out-of-order packets back in order
  • Detect and handle corrupted packets
  • Avoid overloading the receiver
  • <insert your requirement here>
Design Philosophy of the Internet

What do we have now, and why?
Inter-networking

• **Goal**: scalable network infrastructure that connects different smaller networks together, to enable hosts on different networks to talk to each other.

• LAN approach: connect everyone!

• Key challenges with the LAN approach:
  1. Scaling up
  2. Heterogeneity
Why scaling up doesn’t work
What is LAN heterogeneity?

• **Sources of heterogeneity**
  • Addressing
  • Bandwidth and latency
  • Packet size
  • Loss rates
  • Packet routing

• **Gateways provide translation between LANs**
Options for gateway functionality

1. Translation: translate between different LAN “languages”
   • Updates: translation may fail if LANs get updated or new features are added
   • Scalability: have to translate between many LANs

2. Unified network layer: define some common “words” that everyone has to understand
   • This is the current design
Key Principles of the original Internet design

1. Internet communication must continue despite loss of networks or gateways.
2. The Internet must support multiple types of communications service.
3. The Internet architecture must accommodate a variety of networks.
4. The Internet architecture must permit distributed management of its resources.
5. The Internet architecture must be cost effective.
6. The Internet architecture must permit host attachment with a low level of effort.
7. The resources used in the internet architecture must be accountable.
Consequences of the robustness design goals

1. Internet communication must continue despite loss of networks or gateways.
   • State contained in the end host, only soft state in the network
     • Soft state = information that times out (goes away) unless refreshed
     • Easily recover from errors
       • E.g., routing protocols automatically update themselves periodically
     • Complicated functionality (e.g. reliability in the transport layer) implemented in the end host
     • Network gateways kept simple

   • Fate sharing of end hosts
     • If end hosts go down, state is lost
     • If gateway fails, network can recover (soft state)

   • Conservative transmission / liberal reception
     • “Be conservative in what you send; be liberal in what you accept”
     • E.g. sender receives ACK for unknown packets; silently drops
Consequences of the universality design goals

2. The Internet must support multiple types of communications service.
   • Different transport-layer protocols (e.g. TCP, UDP)
   • Datagram as fundamental unit supporting the transport protocols

3. The Internet architecture must accommodate a variety of networks.
   • Best-effort service of datagrams
     • No special treatment of different packets (ignoring QoS)
     • No loss recovery (at the network layer)
     • Makes it easier to add new networks
   • IP-over-everything
     • Common set of names (IP addresses) and routing protocols so that gateways know how to behave
Consequences of the mgmt. design goals

4. The Internet architecture must permit distributed management of its resources.
   - Multiple tier-1 ISPs
   - Different intra-domain and inter-domain routing mechanisms
Consequences of the cost design goals

5. The Internet architecture must be cost effective.
   • What is considered cost in this context? Extra header, retransmission, ...
   • None of these seems to be problematic now

6. The Internet architecture must permit host attachment with a low level of effort.
   • The cost for attachment was once considered high: Each node has to implement all the protocols that are desired
   • This turns out to be the right choice! Again, “smart edge, dumb core”
Consequences of the design goals

7. The resources used in the internet architecture must be accountable.
   - Understanding and monitoring the usage of the resources

These goals, are listed by the author in the order of importance
   - Given the possibility of a hostile environment, survivability was put as a first goal, and accountability as a last goal
   - Would you re-order the goals in the modern era?
Discussions: Any other goals moving forward?

1. Internet communication must continue despite loss of networks or gateways.
2. The Internet must support multiple types of communications service.
3. The Internet architecture must accommodate a variety of networks.
4. The Internet architecture must permit distributed management of its resources.
5. The Internet architecture must be cost effective.
6. The Internet architecture must permit host attachment with a low level of effort.
7. The resources used in the internet architecture must be accountable.