Communication Networks

Prof. Laurent Vanbever

Last week on Communication Networks

Communication Networks

Part 1: General overview

#1 What is a network made of?
   How is it shared?
   How is it organized?
   How does communication happen?
   How do we characterize it?

#2 What is a network made of?
   How is it shared?
   How is it organized?
   How does communication happen?
   How do we characterize it?

Networks are composed of three basic components

- end-systems
- links
- switch/routers

There exist two approaches to sharing:
reservation and on-demand

reservation
- principle: reserve the bandwidth you need in advance

On-demand
- principle: send data when you need

In practice, the approaches are implemented using circuit-switching or packet-switching

reservation
- circuit-switching

On-demand
- packet-switching
### Pros and cons of circuit switching

<table>
<thead>
<tr>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Predictable performance</td>
<td>Inefficient if traffic is bursty or short</td>
</tr>
<tr>
<td>Simple &amp; fast switching</td>
<td>Complex circuit setup/teardown</td>
</tr>
<tr>
<td>Once circuit established</td>
<td>Adds delays to transfer</td>
</tr>
<tr>
<td></td>
<td>Requires new circuit upon failure</td>
</tr>
</tbody>
</table>

### Pros and cons of packet switching

<table>
<thead>
<tr>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Efficient use of resources</td>
<td>Unpredictable performance</td>
</tr>
<tr>
<td>Simpler to implement</td>
<td>Requires buffer management and congestion control</td>
</tr>
<tr>
<td></td>
<td>Route around trouble</td>
</tr>
</tbody>
</table>

---

Communication Networks

**Part 1: General overview**

- What is a network made of?
- How is it shared?
- How is it organized?
- How does communication happen?
- How do we characterize it?

---

The Internet should allow processes on different hosts to exchange data. Everything else is just commentary...
How do you exchange data in a network as complex as this?

To exchange data, Alice and Bob use a set of network protocols.

A protocol is like a conversational convention: who should talk next and how they should respond.

Sometimes implementations are not compliant...

Each protocol is governed by a specific interface.

In practice, there exists a lot of network protocols. How does the Internet organize this?
Modularity is a key component of any good system

Problem: can’t build large systems out of spaghetti code
hard (if not, impossible) to understand, debug, update
need to bound the scope of changes
evolve the system without rewriting it from scratch

Solution: Modularity is how we do it
…and understand the system at a higher-level

To provide structure to the design of network protocols, network designers organize protocols in layers
and the network hardware/software that implement them

Internet communication can be decomposed in 5 independent layers (or 7 layers for the OSI model)

<table>
<thead>
<tr>
<th>layer</th>
<th>service provided:</th>
</tr>
</thead>
<tbody>
<tr>
<td>L5 Application</td>
<td>network access</td>
</tr>
<tr>
<td>L4 Transport</td>
<td>end-to-end delivery (reliable or not)</td>
</tr>
<tr>
<td>L3 Network</td>
<td>global best-effort delivery</td>
</tr>
<tr>
<td>L2 Link</td>
<td>local best-effort delivery</td>
</tr>
<tr>
<td>L1 Physical</td>
<td>physical transfer of bits</td>
</tr>
</tbody>
</table>

Each layer provides a service to the layer above by using the services of the layer directly below it

Applications
... built on... |
Reliable (or unreliable) transport
... built on... |
Best-effort global packet delivery
... built on... |
Best-effort local packet delivery
... built on... |
Physical transfer of bits

Each layer has a unit of data

<table>
<thead>
<tr>
<th>layer</th>
<th>role</th>
</tr>
</thead>
<tbody>
<tr>
<td>L5 Application</td>
<td>exchanges messages between processes</td>
</tr>
<tr>
<td>L4 Transport</td>
<td>transports segments between end-systems</td>
</tr>
<tr>
<td>L3 Network</td>
<td>moves packets around the network</td>
</tr>
<tr>
<td>L2 Link</td>
<td>moves frames across a link</td>
</tr>
<tr>
<td>L1 Physical</td>
<td>moves bits across a physical medium</td>
</tr>
</tbody>
</table>
Each layer (except for L3) is implemented with different protocols

<table>
<thead>
<tr>
<th>Layer</th>
<th>Protocol</th>
</tr>
</thead>
<tbody>
<tr>
<td>L5</td>
<td>Application</td>
</tr>
<tr>
<td>L4</td>
<td>Transport</td>
</tr>
<tr>
<td>L3</td>
<td>Network</td>
</tr>
<tr>
<td>L2</td>
<td>Link</td>
</tr>
<tr>
<td>L1</td>
<td>Physical</td>
</tr>
</tbody>
</table>

- HTTP, SMTP, FTP, SIP, ...
- TCP, UDP, SCTP
- Ethernet, Wi-Fi, (A/V)DSL, WiMAX, LTE, ...
- Twisted pair, fiber, coaxial cable, ...

The Internet Protocol (IP) acts as an unifying, network, layer

<table>
<thead>
<tr>
<th>Layer</th>
<th>Protocol</th>
</tr>
</thead>
<tbody>
<tr>
<td>L5</td>
<td>Application</td>
</tr>
<tr>
<td>L4</td>
<td>Transport</td>
</tr>
<tr>
<td>L3</td>
<td>Network</td>
</tr>
<tr>
<td>L2</td>
<td>Link</td>
</tr>
<tr>
<td>L1</td>
<td>Physical</td>
</tr>
</tbody>
</table>

- HTTP, SMTP, FTP, SIP, ...
- TCP, UDP, SCTP
- IP
- Ethernet, Wi-Fi, (A/V)DSL, Cable, LTE, ...
- Twisted pair, fiber, coaxial cable, ...

Each layer is implemented with different protocols and technologies

<table>
<thead>
<tr>
<th>Layer</th>
<th>Technology</th>
</tr>
</thead>
<tbody>
<tr>
<td>L5</td>
<td>Application</td>
</tr>
<tr>
<td>L4</td>
<td>Transport</td>
</tr>
<tr>
<td>L3</td>
<td>Network</td>
</tr>
<tr>
<td>L2</td>
<td>Link</td>
</tr>
<tr>
<td>L1</td>
<td>Physical</td>
</tr>
</tbody>
</table>

- software
- hardware

Software and hardware advancements

- Programmable network devices: SDN, P4
- DPDK, FOSS
- Highly optimized libraries, drivers

Each layer takes messages from the layer above, and encapsulates with its own header and/or trailer
In practice, layers are distributed on every network device.

Since when bits arrive they must make it to the application, all the layers exist on a host.

Routers act as L3 gateway as such they implement L2 and L3.

Switches act as L2 gateway as such they only implement L2.
Communication Networks
Part 1: General overview

What is a network made of?
How is it shared?
How is it organized?
How does communication happen?

#5 How do we characterize it?

A network connection is characterized by its delay, loss rate and throughput

Throughput
Loss
Delay

How long does it take for a packet to reach the destination
What fraction of packets sent to a destination are dropped?
At what rate is the destination receiving data from the source?

Each packet suffers from several types of delays at each node along the path

Transmission delay = packet size / link bandwidth

Example

Transmission delay = 1000 bits / 100 Mbps = 10 μsec
The propagation delay is the amount of time required for a bit to travel to the end of the link.

\[
\text{Propagation delay} = \frac{\text{link length}}{\text{propagation speed}} [\text{sec}]
\]

Example:
- Link length: 30,000 m
- Propagation speed: \(2 \times 10^8\) m/sec
- Time delay: 150 μsec

How long does it take for a packet to travel from A to B?

Time to transmit one bit = \(10^{-6}\) s

Time to transmit 800 bits = \(800 \times 10^{-6}\) s

Time when that bit reaches B:
\(10^{-6} + 10^{-3}\) s

The last bit reaches B at:
\(800 \times 10^{-6} + 10^{-3}\) s

= 1.8 ms

If we now exchange a 1 GB file split in 100B packets

\(10^7 \times 100B\) packets

The last bit reaches B at:
\((10^7 \times 800 \times 10^{-9}) + 10^{-9}\) s

= 800.1 ms

Different transmission characteristics imply different tradeoffs in terms of which delay dominates

- 10^7 x 100B pkt, 1Gbps link: transmission delay dominates
- 1 x 100B pkt, 1Gbps link: propagation delay dominates
- 1 x 100B pkt, 1Mbps link: both matter

In the Internet, we can’t know in advance which one matters!

The queuing delay is the amount of time a packet waits (in a buffer) to be transmitted on a link.

Queuing delay is the hardest to evaluate as it varies from packet to packet.

It is characterized with statistical measures, e.g., average delay & variance, probability of exceeding x.
Queues absorb transient bursts, but introduce queueing delays.

The time a packet has to sit in a buffer before being processed depends on the traffic pattern.

Queueing delay depends on:
- arrival rate at the queue
- transmission rate of the outgoing link
- traffic burstiness

Queuing delay depends on the traffic pattern.
average packet arrival rate \( a \) [packet/sec]
transmission rate of outgoing link \( R \) [bit/sec]
fixed packets length \( L \) [bit]
average bits arrival rate \( L_a \) [bit/sec]
traffic intensity \( L_a / R \) [bit/sec]

When the traffic intensity is >1, the queue will increase without bound, and so does the queuing delay

Golden rule: Design your queuing system, so that it operates far from that point

When the traffic intensity is <=1, queueing delay depends on the burst size

A network connection is characterized by its delay, loss rate and throughput

In practice, queues are not infinite. There is an upper bound on queuing delay.

A network connection is characterized by its delay, loss rate and throughput

If the queue is persistently overloaded, it will eventually drop packets (loss)

The throughput is the instantaneous rate at which a host receives data

\[
\text{Average throughput} = \frac{\text{data size}}{\text{transfer time}} \quad \text{[bits/sec]}
\]
To compute throughput, one has to consider the bottleneck link...

Average throughput = transmission rate of the bottleneck link

min(R_S, R_L)

To compute throughput, one has to consider the bottleneck link and the intervening traffic.

If 4*min(R_S, R_L) > R, the bottleneck is now in the core, providing each download R/4 of throughput.

A network connection is characterized by its delay, loss rate and throughput.

As technology improves, throughput increase & delays are getting lower except for propagation (speed of light).

Because of propagation delays, Content Delivery Networks move content closer to you.

A brief overview of Internet history:

The Internet history starts in the late 50’s, with people willing to communicate differently:

- Telephone network is the communication system entirely based on circuit switching
- People start to want to use networks for other things, (not personal) computers, ... but knew that circuit-switching will not make it too inefficient for bursty loads and not resilient

From this wish arose three crucial questions:

- How can we design a more efficient network?
- How can we design a more resilient network?
- How can we connect all these networks together?

Paul Baran
Len Kleinrock
Bob Kahn
RAND
UCLA
DARPA

The 60s saw the creation of packet switching and the Advanced Research Projects Agency Network.

The 70s saw the creation of Ethernet, TCP/IP and the e-mail:
- 1971: Network Control Program (predecessor of TCP/IP)
- 1972: Email & Telnet
- 1973: Ethernet
- 1974: TCP/IP (paper by Vint Cerf & Bob Kahn)

The 80s saw the creation of the Web and the Internet going commercial:
- 1989: Arpanet is decommissioned
- 1990: Birth of the Web (Tim Berners Lee at CERN)
- 1993: Search engines invented (Excite)
- 1994: NSFNet decommissioned
- 1998: Google re-invents search

In the 90s, TCP/IP went mainstream:
- 1983: NCP to TCP/IP Flag day
- 1983: Domain Name Service (DNS)
- 1985: NSFNet (TCP/IP) succeeds to ARPANET
- 1986: Van Jacobson saves the Internet (with congestion control)

The 90s saw the creation of the Web as well as the Internet going commercial:
- 1989: Arpanet is decommissioned
- 1990: Birth of the Web (Tim Berners Lee at CERN)
- 1993: Search engines invented (Excite)
- 1994: NSFNet decommissioned
- 1998: Google re-invents search

The new millennium brings the Web 2.0, focus on user-generated content:
- 1998: IPv6 standardization
- 2004: Facebook goes online
- 2006: Google buys YouTube
- 2007: Netflix starts to stream videos
- 2007: First iPhone

Fast Internet access everywhere, every device needs an Internet connection:
- 2009: Mining of the Bitcoin genesis block
- 2018: Only 26% of the Alexa Top 1000 websites reachable over IPv6
- Soon?: Encrypted transport protocols (for example QUIC)

The first message ever exchanged on the Internet was “lo”
Leonard Kleinrock @UCLA tries to log in a Stanford computer on Oct. 29, 1969.

In the 80s, TCP/IP went mainstream:
- 1983: NCP to TCP/IP Flag day
- 1985: NSFNet (TCP/IP) succeeds to ARPANET
- 1986: Van Jacobson saves the Internet (with congestion control)

The new millennium brings the Web 2.0, focus on user-generated content:
- 1998: IPv6 standardization
- 2004: Facebook goes online
- 2006: Google buys YouTube
- 2007: Netflix starts to stream videos
- 2007: First iPhone

Fast Internet access everywhere, every device needs an Internet connection:
- 2009: Mining of the Bitcoin genesis block
- 2018: Only 26% of the Alexa Top 1000 websites reachable over IPv6
- Soon?: Encrypted transport protocols (for example QUIC)
Please join the *Slack* workspace

Web, smartphone and desktop clients available

Ask questions and receive important notifications
E.g. related to the theoretical exercises

Create a (private) channel for your group
During the practical assignments

Contribute to public channels in English
You can contact me in German (@buehlert)

Two practical assignments
in the second half of the semester

Group of maximum three students
Registration will open soon

Internet Hackathon in week 9 (~ 6-10pm)
More information follow shortly

**This Thursday**
First Exercise Session (IFW A 36)

Next Monday on
Communication Networks

**Routing!**