Communication Networks
Spring 2017

Laurent Vanbever
www.vanbever.eu

ETH Zürich (D-ITET)
February, 27 2017

Materials inspired from Scott Shenker & Jennifer Rexford
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?
Networks are composed of three basic components:

- **end-systems**
- **links**
- **switch/routers**
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?
There exist two approaches to sharing: reservation and on-demand.

- **Reservation**: reserve the bandwidth you need in advance.
- **On-demand**: send data when you need it.
In practice, the approaches are implemented using 

circuit-switching or 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>Which 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 than circuit switching</td>
<td>Requires buffer management and congestion control</td>
</tr>
<tr>
<td>Route around trouble</td>
<td></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?
List any technologies, principles, applications...
used after typing in:

> www.google.ch

and pressing enter in your browser
You have a lot of networking knowledge already!

... and this, across all the layers

<table>
<thead>
<tr>
<th>Layer</th>
<th>Protocol</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transport</td>
<td>TCP</td>
<td>72%</td>
</tr>
<tr>
<td>Application</td>
<td>DNS</td>
<td>65%</td>
</tr>
<tr>
<td>Network</td>
<td>IP</td>
<td>56%</td>
</tr>
<tr>
<td>Network</td>
<td>Router</td>
<td>44%</td>
</tr>
<tr>
<td>Application</td>
<td>HTTP</td>
<td>41%</td>
</tr>
<tr>
<td>Application</td>
<td>HTML</td>
<td>30%</td>
</tr>
<tr>
<td>Server</td>
<td></td>
<td>26%</td>
</tr>
<tr>
<td>Link</td>
<td>Ethernet</td>
<td>19%</td>
</tr>
<tr>
<td>Transport</td>
<td>UDP</td>
<td>17%</td>
</tr>
<tr>
<td></td>
<td>Encryption</td>
<td>15%</td>
</tr>
</tbody>
</table>

+72 other!
This week on

Communication Networks
Communication Networks

Part 1: General overview

What is a network made of?

How is it shared?

How is it organized?

#4  How does communication happen?

#5  How do we characterize it?
Communication Networks
Part 1: General overview

What is a network made of?

How is it shared?

How is it organized?

#4 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?
phone company

cable company

university net
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.

Hello

Give me http://comm-net.ethz.ch/

Here it is
Sometimes implementations are not compliant...

Alice

Bob

hello

give me http://...
give me http://...
give me http://...
give me http://...
give me http://...
give me http://...
Each protocol is governed by a specific interface

Alice

WoW server

while (...) {
    message = ...;
    send(message, ...);
}

Bob

WoW client

while (...) {
    message = receive(...);
}

Application Programming Interface
In practice, there exists a lot of network protocols. How does the Internet organize this?
HOW STANDARDS PROLIFERATE:
(SEE: A/C CHARGERS, CHARACTER ENCODINGS, INSTANT MESSAGING, ETC)

SITUATION:
THERE ARE 14 COMPETING STANDARDS.

14?! RIDICULOUS!
WE NEED TO DEVELOP ONE UNIVERSAL STANDARD THAT COVERS EVERYONE'S USE CASES.  YEAH!

SOON:

SITUATION:
THERE ARE 15 COMPETING STANDARDS.

https://xkcd.com/927/
Modularity is a key component of any good system

<table>
<thead>
<tr>
<th>Problem</th>
<th>Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>can’t build large systems out of spaghetti code</td>
<td>Modularity is how we do it</td>
</tr>
<tr>
<td>hard (if not, impossible) to understand, debug, update</td>
<td>…and understand the system at a higher-level</td>
</tr>
</tbody>
</table>
Modularity, based on abstraction, is the way things get done.

— Barbara Liskov, MIT
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)

L5  Application
L4  Transport
L3  Network
L2  Link
L1  Physical
Each layer provides a service to the layer above

<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</td>
<td>Application exchanges <strong>messages</strong> between processes</td>
</tr>
<tr>
<td>L4</td>
<td>Transport transports <strong>segments</strong> between end-systems</td>
</tr>
<tr>
<td>L3</td>
<td>Network moves <strong>packets</strong> around the network</td>
</tr>
<tr>
<td>L2</td>
<td>Link moves <strong>frames</strong> across a link</td>
</tr>
<tr>
<td>L1</td>
<td>Physical moves <strong>bits</strong> 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 HTTP, SMTP, FTP, SIP, …</td>
</tr>
<tr>
<td>L4</td>
<td>Transport TCP, UDP, SCTP</td>
</tr>
<tr>
<td>L3</td>
<td>Network IP</td>
</tr>
<tr>
<td>L2</td>
<td>Link Ethernet, Wifi, (A/V)DSL, WiMAX, LTE, …</td>
</tr>
<tr>
<td>L1</td>
<td>Physical Twisted pair, fiber, coaxial cable, …</td>
</tr>
</tbody>
</table>
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>

- L5: Application protocols (HTTP, SMTP, FTP, SIP, ...)
- L4: Transport protocols (TCP, UDP, SCTP)
- L3: IP
- L2: Ethernet, Wifi, (A/V)DSL, Cable, LTE, ...
- L1: Twisted pair, fiber, coaxial cable, ...
Each layer (except for L3) is implemented with different protocols and technologies.
Network stack challenges at increasing speeds
The 100Gbit/s challenge

Jesper Dangaard Brouer
Red Hat inc.

Linux Conf Au, New Zealand, January 2015

Microsoft Supercharges Bing Search With Programmable Chips

Robert McMillan  

Microsoft Supercharges Bing Search With Programmable Chips

Microsoft

Doug Burger called it Project Catapult.

Burger works inside Microsoft Research—the group where the tech giant explores blue-sky ideas—and in November 2012, he pitched a radical new concept to Qi Lu, the man who

https://www.wired.com/2014/06/microsoft-fpga/
Each layer takes messages from the layer above, and *encapsulates* with its own header and/or trailer.
GET google.ch

HTTP(S)

TCP/UDP

IP

Ethernet

Header

Message

HA

GET google.ch

HT

HA

GET google.ch
GET google.ch

HTTP(S)

TCP/UDP

IP

Ethernet

Application

Transport

Network

Link

Header

Message
HTTP(S) → TCP/UDP → IP → Ethernet

Header: GET google.ch
Message:

- HE
- HN
- HT
- HA

GET google.ch

- HE
- HN
- HT
- HA

GET google.ch

- HE
- HN
- HT
- HA

GET google.ch
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
Let’s see how it looks like in practice on a host, using Wireshark

https://www.wireshark.org
Now, it’s your turn

...to design a Internet protocol

instructions given in class
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.

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?
A network *connection* is characterized by its delay, loss rate and throughput
Each packet suffers from several types of delays at each node along the path

\[
\begin{align*}
\text{transmission delay} & \quad \text{due to link properties} \\
\oplus \text{ propagation delay} & \quad \text{due to traffic mix & switch internals} \\
\oplus \text{ processing delay} & \quad \text{due to traffic mix & switch internals} \\
\oplus \text{ queuing delay} & \quad \text{due to traffic mix & switch internals} \\
\hline
\text{total delay} & \quad \text{due to link properties}
\end{align*}
\]
Overall, the main culprits for the overall delay are the transmission, propagation and queuing delays tend to be tiny.
The transmission delay is the amount of time required to push all of the bits onto the link.

Transmission delay = \frac{\text{packet size}}{\text{link bandwidth}}

Example:

\frac{1000 \text{ bits}}{100 \text{ Mbps}} = 10 \mu\text{sec}
The propagation delay is the amount of time required for a bit to travel to the end of the link.

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

\text{propagation speed} \quad [\text{m/sec}]

(fraction of speed of light)

Example:

\[
\frac{30\,000\text{ m}}{2.1 \times 10^8\text{ m/sec}} = 150 \mu\text{sec}
\]

(speed of light in fiber)
How long does it take for a packet to travel from A to B? (not considering queuing for now)
How long does it take to exchange 100 Bytes packet?

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 = 1.8ms
If we have a 1 Gbps link, the total time decreases to 1.08ms.
If we now exchange a 1GB file split in 100B packets.

The last bit reaches B at $(10^7 \times 800 \times 10^{-9}) + 10^{-3}s = 8001ms$. 
Different transmission characteristics imply different tradeoffs in terms of which delay dominates.

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

\[ 10^7 \times 100B \text{ pkt} \quad 1\text{Gbps link} \quad \text{transmission delay dominates} \]

\[ 1 \times 100B \text{ pkt} \quad 1\text{Gbps link} \quad \text{propagation delay dominates} \]

\[ 1 \times 100B \text{ pkt} \quad 1\text{Mbps link} \quad \text{both matter} \]
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 \( x \).
Queuing delay depends on the traffic pattern
Queuing delay depends on the traffic pattern

Queue

Transient overload!
Transient overload!
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
average packet arrival rate \( a \) [packet/sec]

transmission rate of outgoing link \( R \) [bit/sec]

fixed packets length \( L \) [bit]

average bits arrival rate \( La \) [bit/sec]

traffic intensity \( \frac{La}{R} \)
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 $\leq 1$, 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.

$$\text{queuing delay upper bound: } N \times \frac{L}{R}$$
If the queue is persistently overloaded, it will eventually drop packets (loss)
A network *connection* is characterized by its delay, loss rate and throughput.
The throughput is the instantaneous rate at which a host receives data

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

\[
\text{Average throughput} = \min(R_S, R_L)
\]

= transmission rate of the bottleneck link
To compute throughput, one has to consider the bottleneck link... and the intervening traffic

if $4 \times \min(R_S, R_L) > R$ 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.

http://wwwnui.akamai.com/gnet/globe/index.html
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, defense, (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 resilient network? (also) lead to the invention of packet switching
- How can we design a more efficient network? (also) lead to the invention of packet switching
- How can we connect all these networks together? lead to the invention of the Internet as we know it

Paul Baran
RAND

Len Kleinrock
UCLA

Bob Kahn
DARPA
The 60s saw the creation of packet switching and the Advanced Research Projects Agency Network.
The first message ever exchanged on the Internet was “lo”

Oct. 29 1969

Leonard Kleinrock @UCLA tries to log in a Stanford computer

UCLA

We typed the L… Do you see it?

Yes! We see the L

We typed the O… Do you see it?

Yes! We see the O

We typed the G. system crashes

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
In the 80s, TCP/IP went global

<table>
<thead>
<tr>
<th>Year</th>
<th>Event Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1983</td>
<td>NCP to TCP/IP Flag day</td>
</tr>
<tr>
<td>1983</td>
<td>Domain Name Service (DNS)</td>
</tr>
<tr>
<td>1985</td>
<td>NSFNet (TCP/IP) succeeds to ARPANET</td>
</tr>
<tr>
<td>198x</td>
<td>Internet meltdowns due to congestion</td>
</tr>
<tr>
<td>1986</td>
<td>Van Jabobson saves the Internet (with congestion control)</td>
</tr>
</tbody>
</table>
The 90s saw the creation of the Web as well as the Internet going commercial.

- **1989** Arpanet is decommissioned
- **1989** Birth of the Web
  - Tim Berners Lee (CERN)
- **1993** Search engines invented (Excite)
- **1995** NSFNet is decommissioned
- **1998** Google reinvents search
Next Monday on Communication Networks

Routing!