Communication Networks

Prof. Laurent Vanbever

Communication Networks

Spring 2018

Tobias Bühler, TA

Slides from
Laurent Vanbever
nsg.ee.ethz.ch

ETH Zürich

February 26 2018

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?

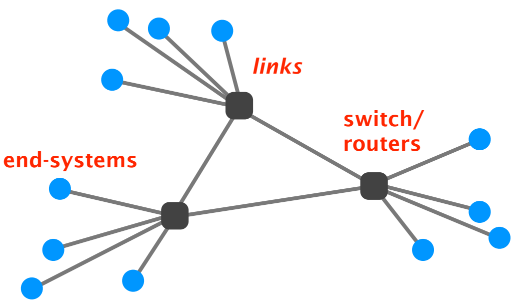
How is it shared?

How is it organized?

How does communication happen?

How do we characterize it?

Networks are composed of three basic components



Communication Networks
Part 1: General overview

What is a network made of?

#2 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	On-demand
principle	reserve the bandwidth you need in advance	send data when you need

---

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

	Reservation	On-demand
implem.	circuit-switching	packet-switching

Pros and cons of circuit switching

advantages	disadvantages
predictable performance	inefficient if traffic is bursty or short
simple & fast switching once circuit established	complex circuit setup/teardown which adds delays to transfer
	requires new circuit upon failure

Pros and cons of packet switching

advantages	disadvantages
efficient use of resources	unpredictable performance
simpler to implement than circuit switching	requires buffer management and congestion control
route around trouble	

Communication Networks
Part 1: General overview

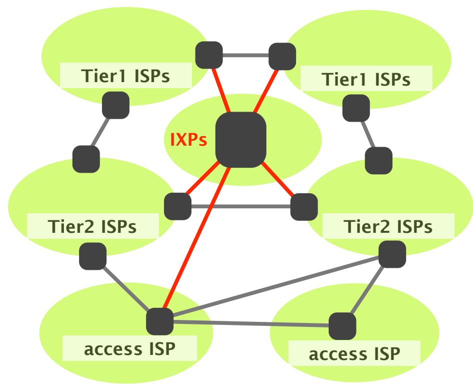
What is a network made of?

How is it shared?

#3 How is it organized?

How does communication happen?

How do we characterize it?



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?

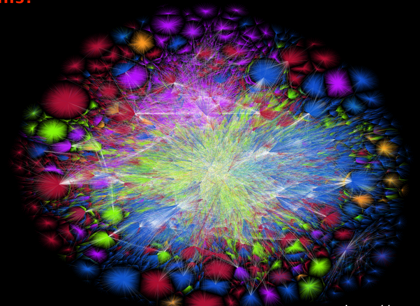
How do we characterize it?

The Internet should allow

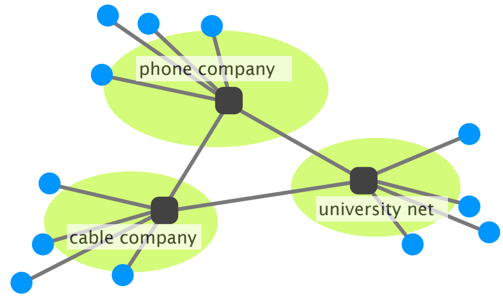
processes on different hosts
to exchange data

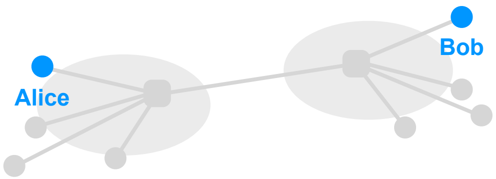
everything else is just commentary…

How do you exchange data in a network as complex as **this?**



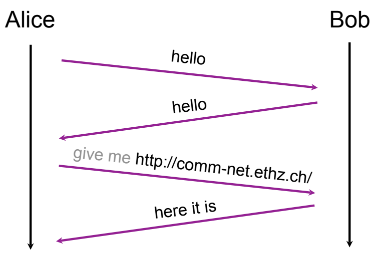
http://www.opte.org





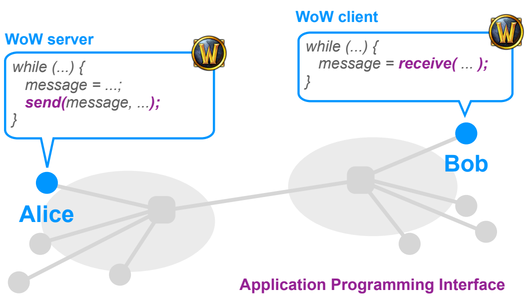
To exchange data, Alice and Bob use a set of network protocols

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

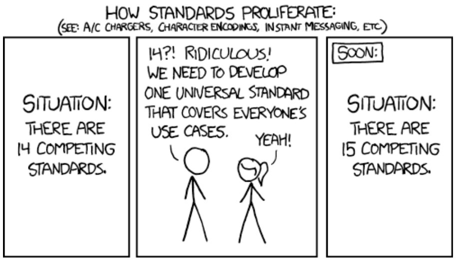


Sometimes implementations are not compliant...

Each protocol is governed by a specific interface



In practice, there exists **a lot** of network protocols.
How does the Internet organize **this?**



https://xkcd.com/927/

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

---

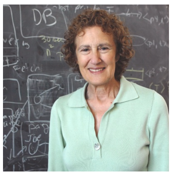


Photo: Donna Coveney

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)

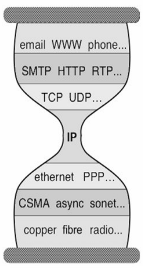
	layer
L5	Application
L4	Transport
L3	Network
L2	Link
L1	Physical

Each layer provides a service to the layer above

	layer	service provided:
L5	Application	network access
L4	Transport	end-to-end delivery (reliable or not)
L3	Network	global best-effort delivery
L2	Link	local best-effort delivery
L1	Physical	physical transfer of bits

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

	layer	role
L5	Application	exchanges messages between processes
L4	Transport	transports segments between end-systems
L3	Network	moves packets around the network
L2	Link	moves frames across a link
L1	Physical	moves bits across a physical medium

Each layer (except for L3) is implemented
with different protocols

	layer	protocol
L5	Application	HTTP, SMTP, FTP, SIP, …
L4	Transport	TCP, UDP, SCTP
L3	Network	IP
L2	Link	Ethernet, Wifi, (A/V)DSL, WiMAX, LTE, …
L1	Physical	Twisted pair, fiber, coaxial cable, …

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

	layer	protocol
L5	Application	HTTP, SMTP, FTP, SIP, …
L4	Transport	TCP, UDP, SCTP
L3	Network	IP
L2	Link	Ethernet, Wifi, (A/V)DSL, Cable, LTE, …
L1	Physical	Twisted pair, fiber, coaxial cable, …

Each layer is implemented with different protocols
and technologies

	layer	technology
L5	Application	software
L4	Transport	
L3	Network	hardware
L2	Link	
L1	Physical	

L1 L2 L3 L4 L5

hardware speed flexibility software

Software and hardware advancements

L1 L2 L3 L4 L5

hardware speed flexibility software

programmable network devices → SDN, P4

DPDK, FD.io ← highly optimized libraries, drivers

redhat.

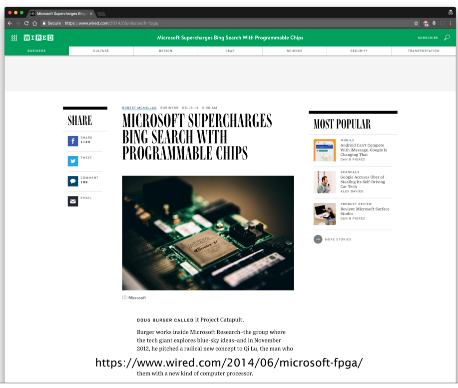
Network stack challenges at increasing speeds

The 100Gbit/s challenge

Jesper Dangaard Brouer
Red Hat inc.

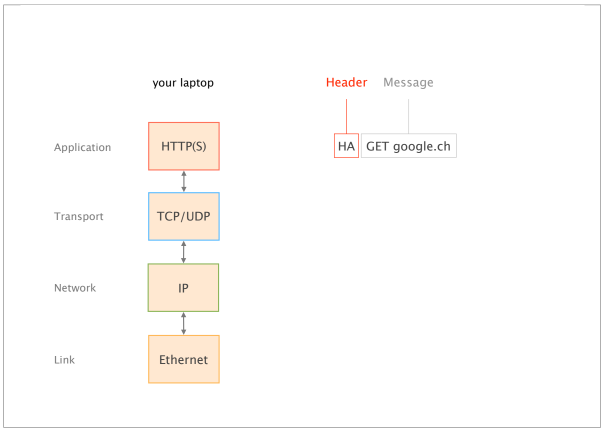
Linux Conf Au, New Zealand, January 2015

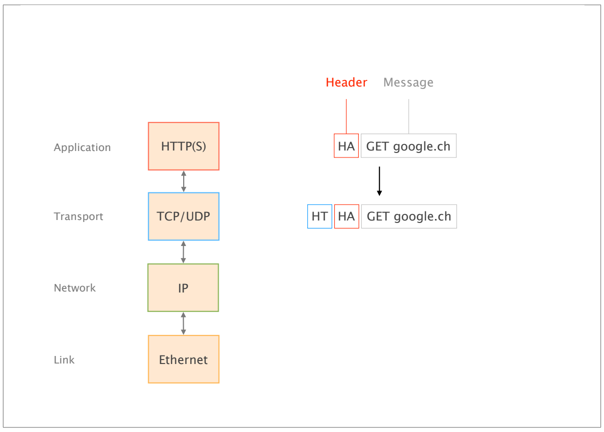
http://people.netfilter.org/hawk/presentations/LCA2015/net_stack_challenges_100G_LCA2015.pdf

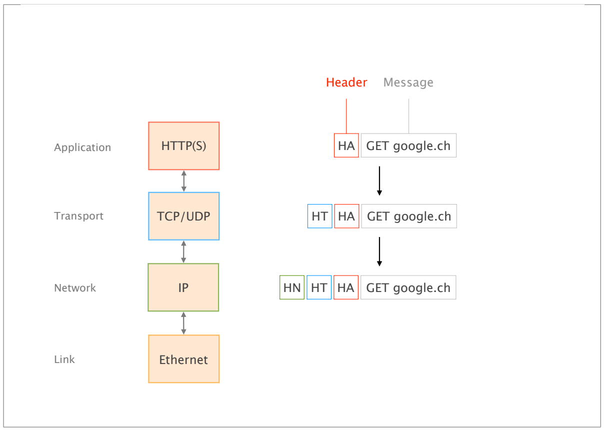


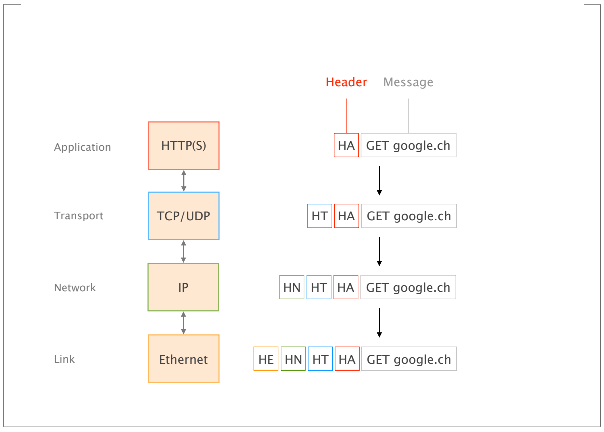
Each layer takes messages from the layer above,
and *encapsulates* with its own header and/or trailer

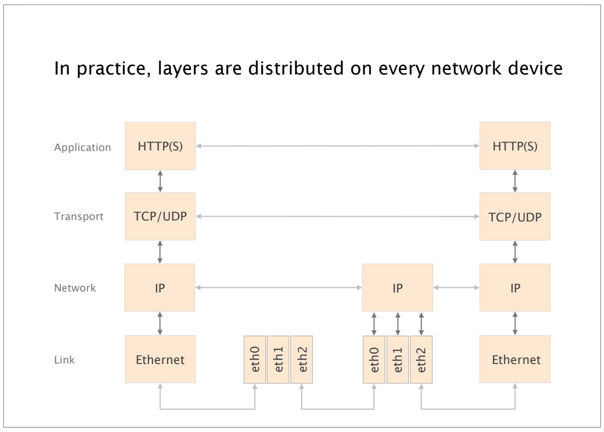
Layer	Protocol
Application	HTTP(S)
Transport	TCP/UDP
Network	IP
Link	Ethernet

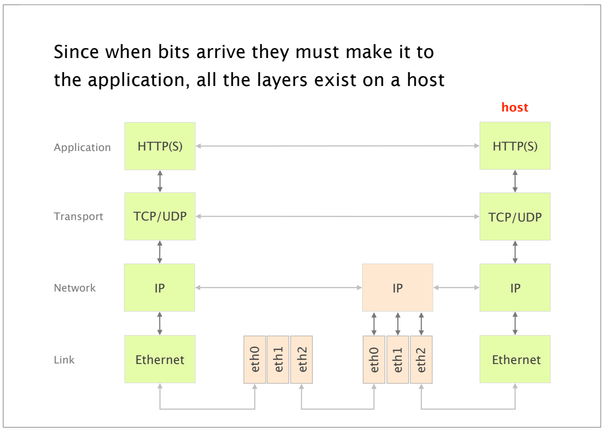

your laptop
Header
Message
Application
Transport
Network
Link
HTTP(S)
TCP/UDP
IP
Ethernet
HA
GET google.ch

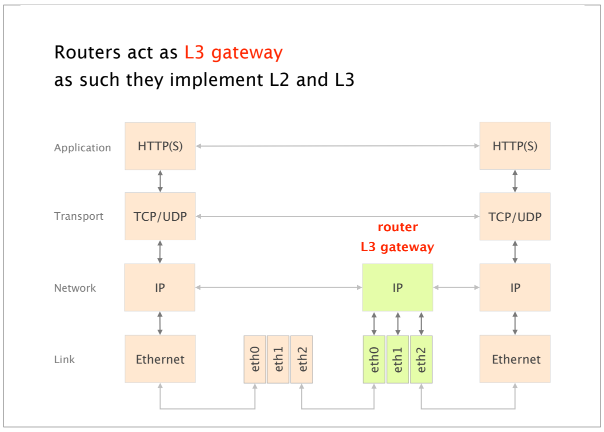

Header
Message
Application
Transport
Network
Link
HTTP(S)
TCP/UDP
IP
Ethernet
HA
GET google.ch
HT
HA
GET google.ch

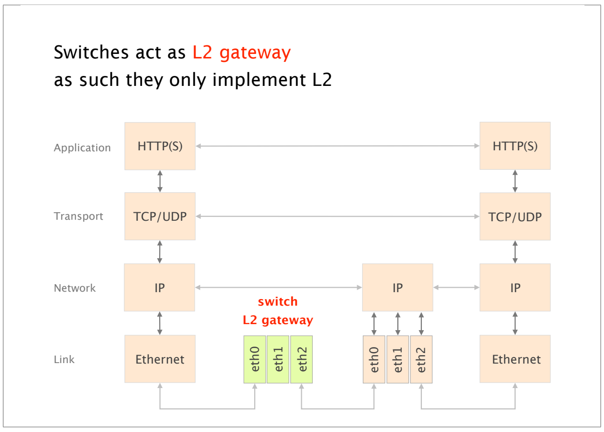

Header
Message
Application
Transport
Network
Link
HTTP(S)
TCP/UDP
IP
Ethernet
HA
GET google.ch
HT
HA
GET google.ch
HN
HT
HA
GET google.ch


Header
Message
Application
Transport
Network
Link
HTTP(S)
TCP/UDP
IP
Ethernet
HA
GET google.ch
HT
HA
GET google.ch
HN
HT
HA
GET google.ch
HE
HN
HT
HA
GET google.ch


In practice, layers are distributed on every network device
Application
Transport
Network
Link
HTTP(S)
HTTP(S)
TCP/UDP
TCP/UDP
IP
IP
IP
Ethernet
eth0
eth1
eth2
eth0
eth1
eth2
Ethernet


Since when bits arrive they must make it to
the application, all the layers exist on a host
host
Application
Transport
Network
Link
HTTP(S)
HTTP(S)
TCP/UDP
TCP/UDP
IP
IP
IP
Ethernet
eth0
eth1
eth2
eth0
eth1
eth2
Ethernet


Routers act as L3 gateway
as such they implement L2 and L3
Application
Transport
Network
Link
HTTP(S)
HTTP(S)
TCP/UDP
TCP/UDP
router
L3 gateway
IP
IP
IP
Ethernet
eth0
eth1
eth2
eth0
eth1
eth2
Ethernet


Switches act as L2 gateway
as such they only implement L2
Application
Transport
Network
Link
HTTP(S)
HTTP(S)
TCP/UDP
TCP/UDP
IP
IP
IP
switch
L2 gateway
Ethernet
eth0
eth1
eth2
eth0
eth1
eth2
Ethernet

Let's see how it looks like in practice
on a host, using Wireshark https://www.wireshark.org

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

delay | loss | 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

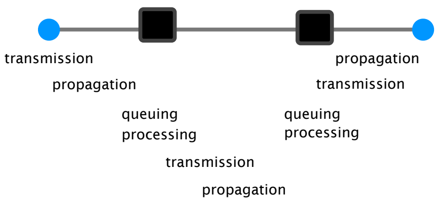
delay | loss | throughput

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

	transmission	delay	due to link properties
⊕	propagation	delay	
⊕	processing	delay	due to traffic mix & switch internals
⊕	queuing	delay	
=	total	delay	

Overall, the main culprits for the overall delay are
the transmission, propagation and queuing delays

	transmission	delay	
⊕	propagation	delay	
⊕	processing	delay	*tend to be tiny*
⊕	queuing	delay	
=	total	delay	



The transmission delay is the amount of time
required to push all of the bits onto the link

Transmission delay [sec] = packet size [#bits] / link bandwidth [#bits/sec]

Example: 1000 bits / 100 Mbps = 10 µsec

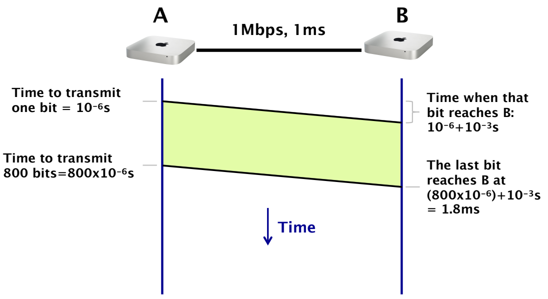
The propagation delay is the amount of time required for a bit to travel to the end of the link

Propagation delay [sec] = $\frac{\text{link length [m]}}{\text{propagation speed [m/sec]}}$ (fraction of speed of light)

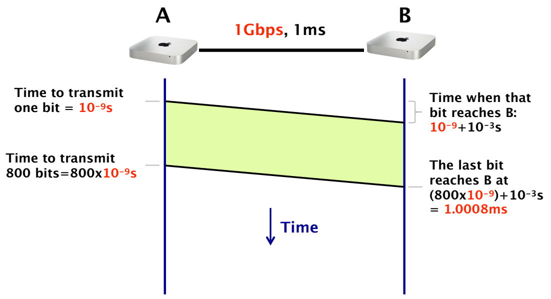
Example $\frac{30\ 000\ \text{m}}{2\text{x}10^{8}\ \text{m/sec}}$ (speed of light in fiber) 150 μsec

How long does it take for a packet to travel from A to B?
(not considering queuing for now)

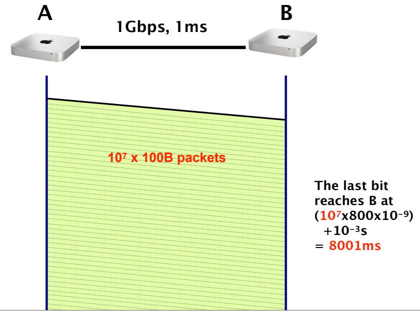
How long does it take to exchange 100 Bytes packet?



If we have a 1 Gbps link,
the total time decreases to 1.0008ms



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



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

10^{7}x100B pkt	1Gbps link	transmission delay dominates
1x100B pkt	1Gbps link	propagation delay dominates
1x100B pkt	1Mbps link	both matter

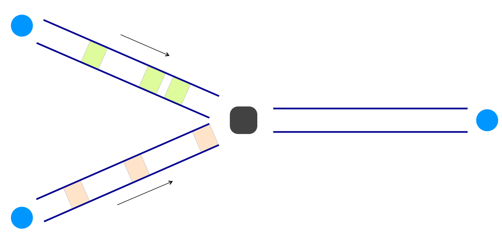
In the Internet, we **can't know** in advance which one matters!

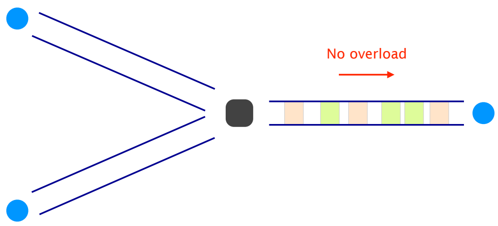
The queuing delay is the amount of time a packet waits (in a buffer) to be transmitted on a link

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

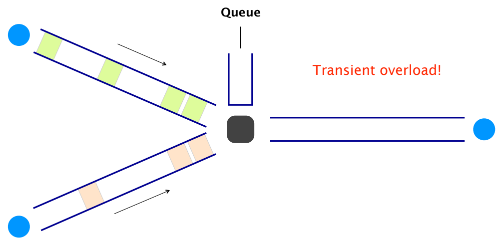
It is characterized with statistical measures
e.g., average delay & variance, probability of exceeding *x*

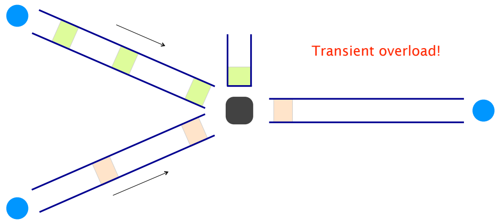
Queuing delay depends on the traffic pattern

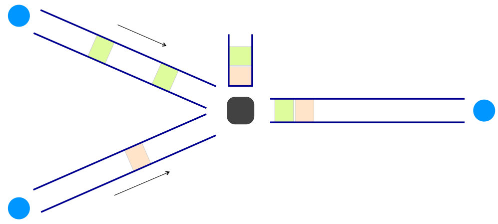


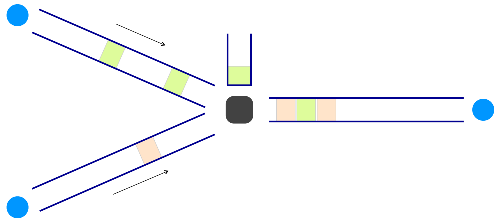


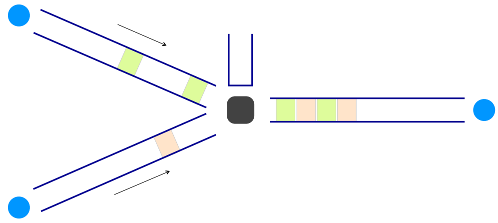
Queuing delay depends on the traffic pattern



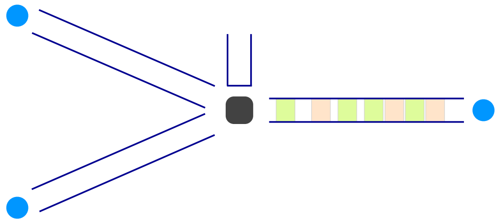








Queues absorb transient bursts,
but introduce queueing delays



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

Queueing delay depends on:

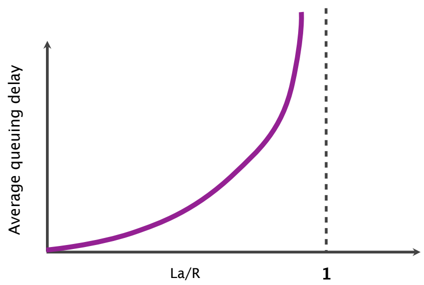
- arrival rate at the queue
- transmission rate of the outgoing link
- traffic burstiness

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	*La/R*	

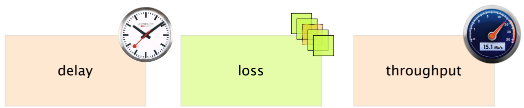
When the traffic intensity is >1, the queue will increase without bound, and so does the queuing delay

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

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

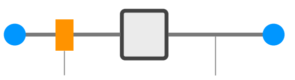


A network *connection* is characterized by its delay, loss rate and throughput



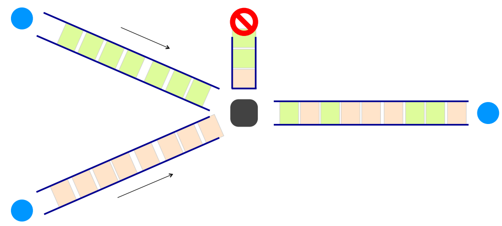
In practice, queues are not infinite.
There is an upper bound on queuing delay.

queue
N+1 packets

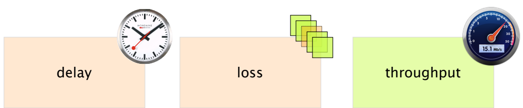


queuing delay upper bound: N*L/R

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



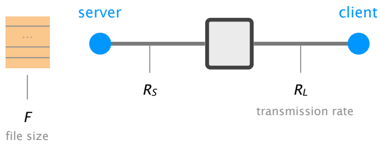
A network *connection* is characterized by its delay, loss rate and throughput



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

$$\text{Average throughput [\#bits/sec]} = \frac{\text{data size [\#bits]}}{\text{transfer time [sec]}}$$

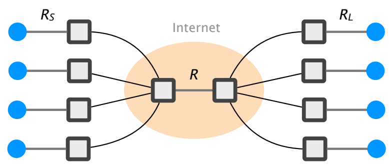
To compute throughput, one has to consider the bottleneck link



Average throughput $\min(R_S, R_L)$

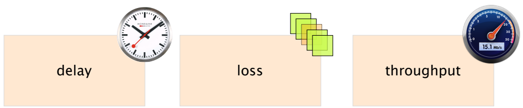
= transmission rate of the bottleneck link

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



if $4 \cdot \min(R_S, R_L) > R$ the bottleneck is now in the core, providing each download R/4 of throughput

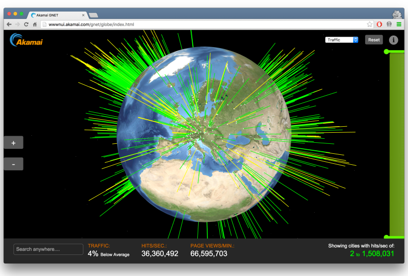
A network *connection* is characterized by its delay, loss rate and throughput



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

Paul Baran
RAND

How can we design a more resilient network?
lead to the invention of packet switching

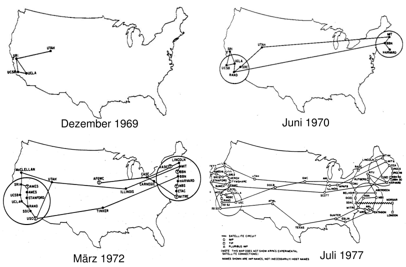
Len Kleinrock
UCLA

How can we design a more efficient network?
(also) lead to the invention of packet switching

Bob Kahn
DARPA

How can we connect all these networks together?
lead to the invention of the Internet as we know it

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?
Stanford: *Yes! We see the L*

UCLA: We typed the O… Do you see it?
Stanford: *Yes! We see the O*

UCLA: We typed the G. system crashes

http://ftp.cs.ucla.edu/csd/first_words.html

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 mainstream

1983	NCP to TCP/IP Flag day Domain Name Service (DNS)
1985	NSFNet (TCP/IP) succeeds to ARPANET
198x	Internet meltdowns due to congestion
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
	Birth of the Web Tim Berners Lee (CERN) — Swiss made
1993	Search engines invented (Excite)
1995	NSFNet is decommissioned
1998	Google reinvents 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 Mobile Internet access

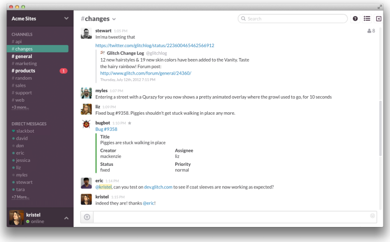
Fast Internet access everywhere, every device needs an Internet connection

2009 ↓	Mining of the Bitcoin genesis block
	Fast mobile Internet access: 4G/LTE
	Internet of Things (IoT) boom Cars & refrigerators in the Internet
2018	Only 26% of the Alexa Top 1000 websites reachable over IPv6 http://www.worldipv6launch.org/measurements/
Soon?	Encrypted transport protocols For example QUIC

Communication Networks

Course organization

Please join the Slack workspace



Web, smartphone and desktop clients available

Please join the Slack workspace

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 Englisch

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!