Communication Networks

Prof. Laurent Vanbever

Communication Networks

Spring 2022

Laurent Vanbever

nsg.ee.ethz.ch

ETH Zürich (D-ITET)

28 February 2022

Materials inspired from Scott Shenker & Jennifer Rexford

Communication Networks

Part 1: General overview

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

Communication Networks

Part 1: General overview

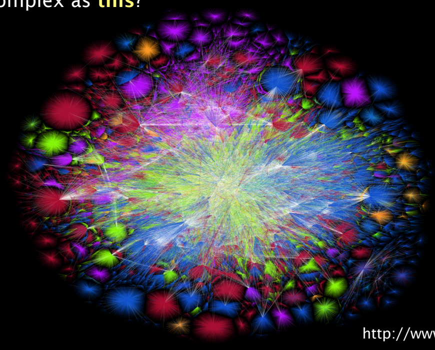
	What is a network made of?
	How is it shared?
	How is it organized?
#4	How does communication happen?
	How do we characterize it?

The Internet should allow

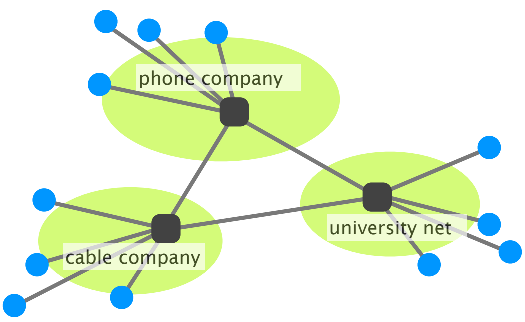
processes on different hosts
to exchange data

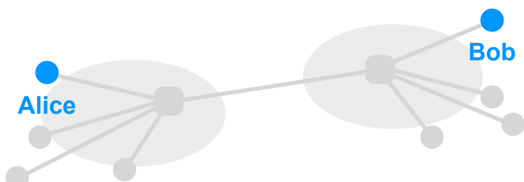
everything else is just commentary…

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



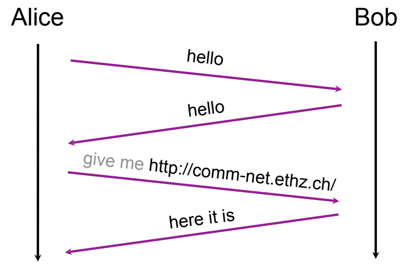
http://www.opte.org



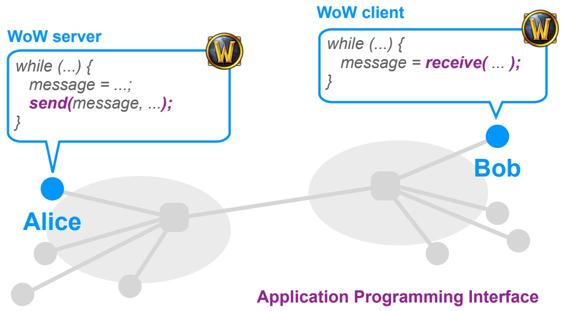


To exchange data, Alice and Bob use
a set of network protocols

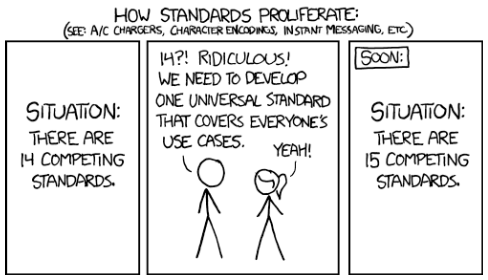
A protocol is like a conversational convention:
who should talk next and how they should respond



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

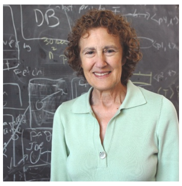


Photo: Donna Coveney

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

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)

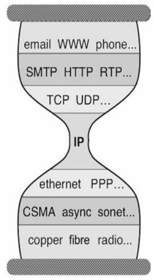
	layer
L5	Application
L4	Transport
L3	Network
L2	Link
L1	Physical

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, ...

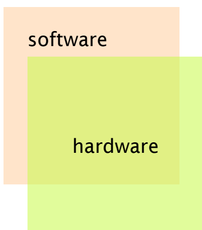
The Internet Protocol (IP) acts as an unifying, network, layer

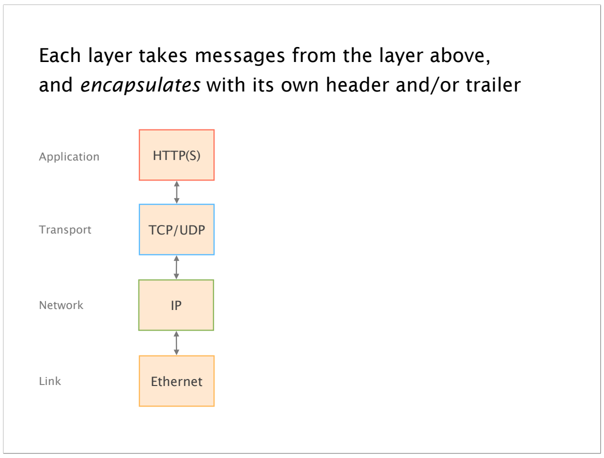
	layer	protocol
L5	Application	HTTP, SMTP, FTP, SIP, ...
L4	Transport	TCP, UDP, SCTP
L3	Network	IP
L2	Link	Ethernet, Wifi, (A/V)DSL, Cable, LTE, ...
L1	Physical	Twisted pair, fiber, coaxial cable, ...

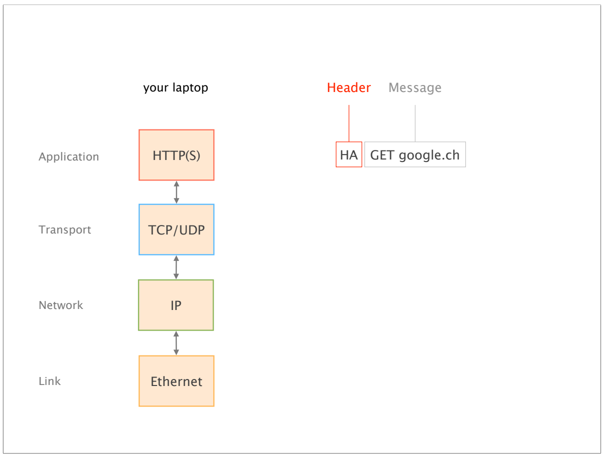
Each layer is implemented with different protocols and technologies

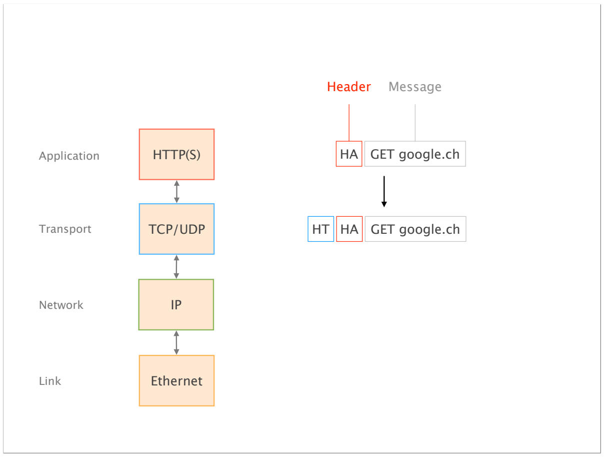
	layer
L5	Application
L4	Transport
L3	Network
L2	Link
L1	Physical

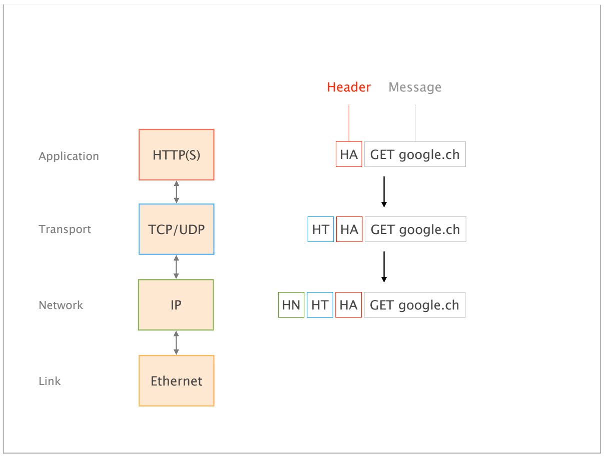
technology

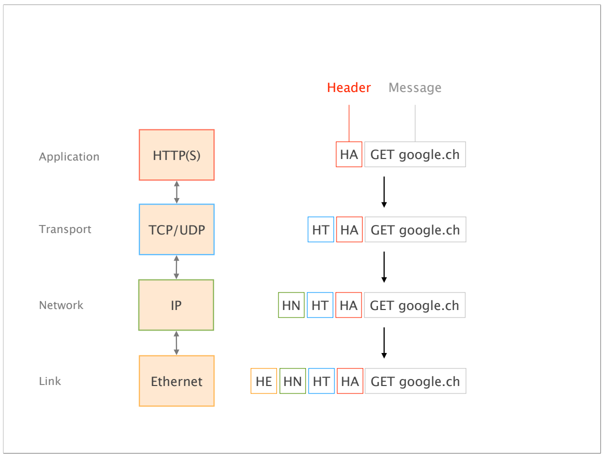


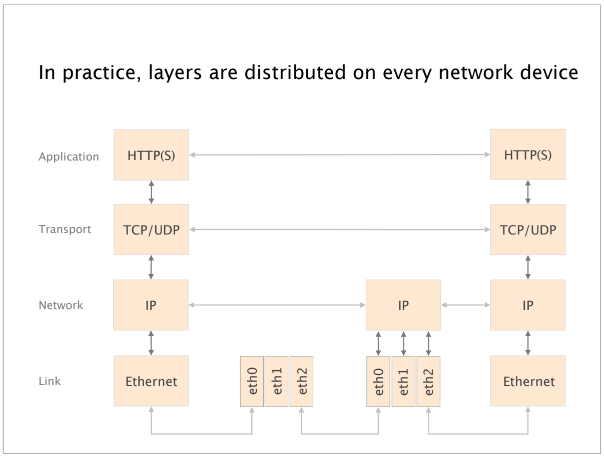

Each layer takes messages from the layer above,
and *encapsulates* with its own header and/or trailer
Application
HTTP(S)
Transport
TCP/UDP
Network
IP
Link
Ethernet

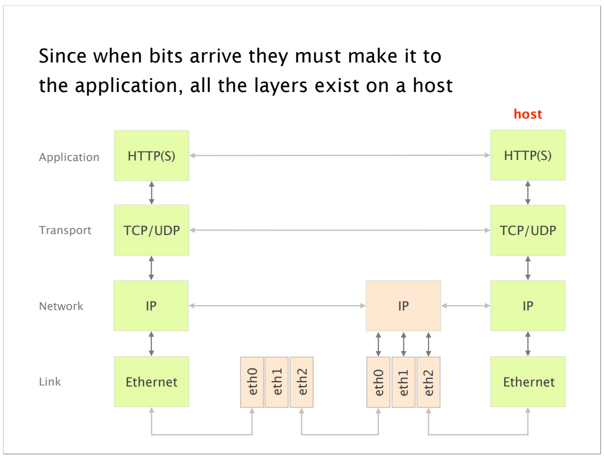

your laptop
Header
Message
Application
HTTP(S)
HA
GET google.ch
Transport
TCP/UDP
Network
IP
Link
Ethernet

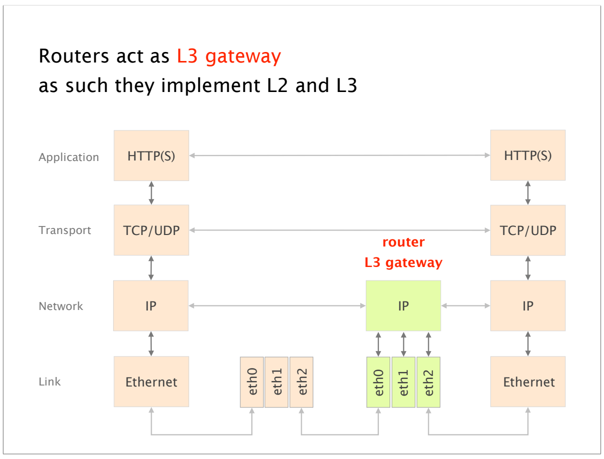

Header
Message
Application
HTTP(S)
HA
GET google.ch
Transport
TCP/UDP
HT
HA
GET google.ch
Network
IP
Link
Ethernet


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

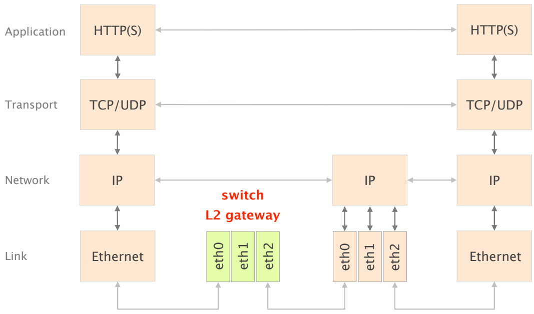

Header
Message
Application
HTTP(S)
HA
GET google.ch
Transport
TCP/UDP
HT
HA
GET google.ch
Network
IP
HN
HT
HA
GET google.ch
Link
Ethernet
HE
HN
HT
HA
GET google.ch


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


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

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

Communication Networks
Part 1: General overview

What is a network made of?

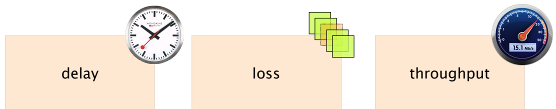
How is it shared?

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

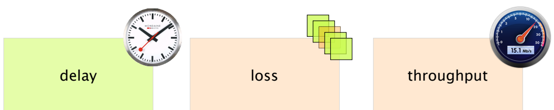


How long does it take for a packet to reach the destination

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

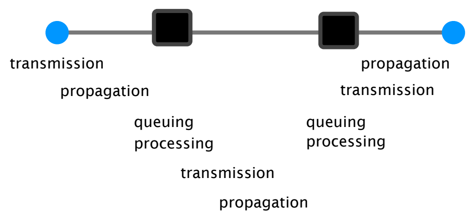
transmission delay
⊕ propagation delay — due to link properties
⊕ processing delay
⊕ queuing delay — due to traffic mix & switch internals

= 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

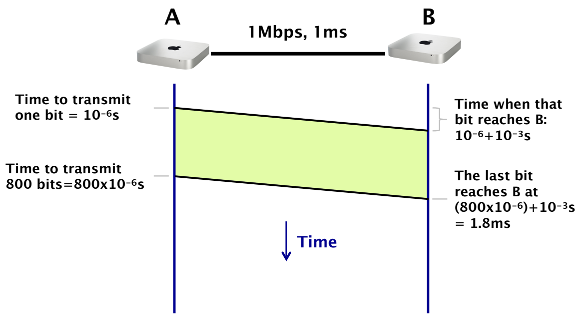
Transmission delay [sec]	=	packet size / link bandwidth	[#bits] / [#bits/sec]
Example		1000 bits / 100 Gbps	10 ns

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

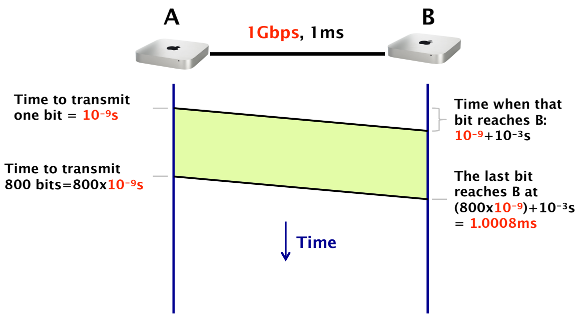
Propagation delay [sec]	=	link length / propagation speed (fraction of speed of light)	[m] / [m/sec]
Example		30 000 m / 2×10^8 m/sec (speed of light in fiber)	150 µsec

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

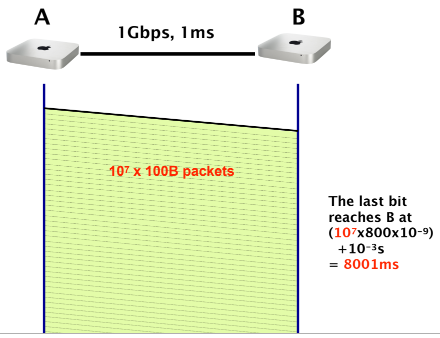
How long does it take to exchange 100 Bytes packet?



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

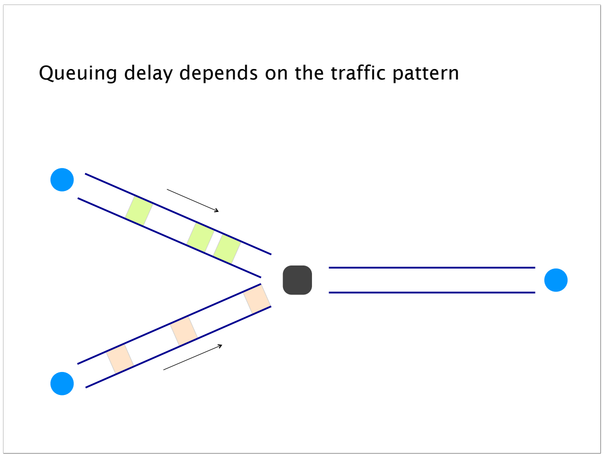
10^7x100B pkt	1Gbps link	transmission delay dominates
1x100B pkt	1Gbps link	propagation delay dominates
1x100B pkt	1Mbps link	both matter

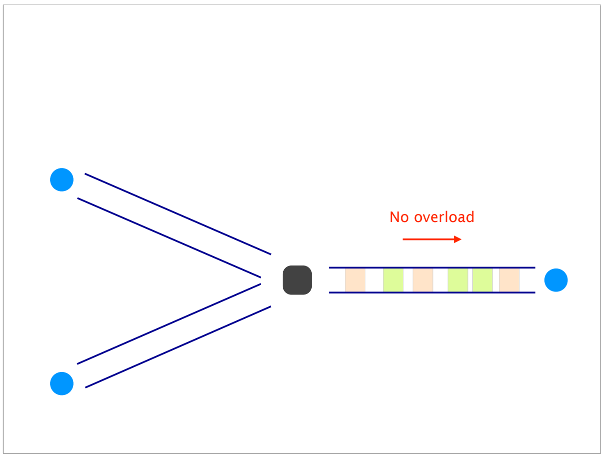
In the Internet, we **can't know** in advance which one matters!

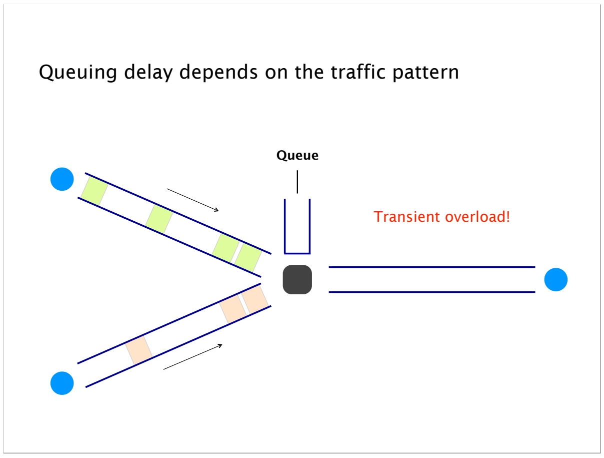
The queuing delay is the amount of time a packet waits (in a buffer) to be transmitted on a link

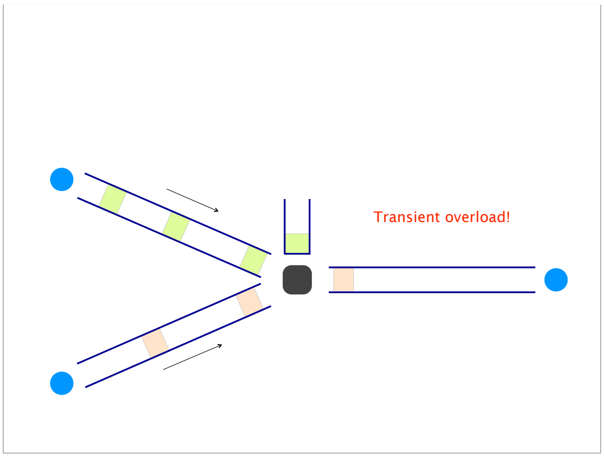
Queuing delay is the hardest to evaluate
as it varies from packet to packet

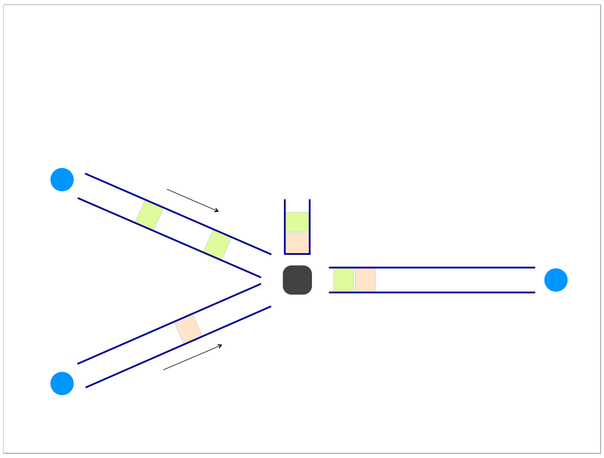
It is characterized with statistical measures
e.g., average delay & variance, probability of exceeding *x*

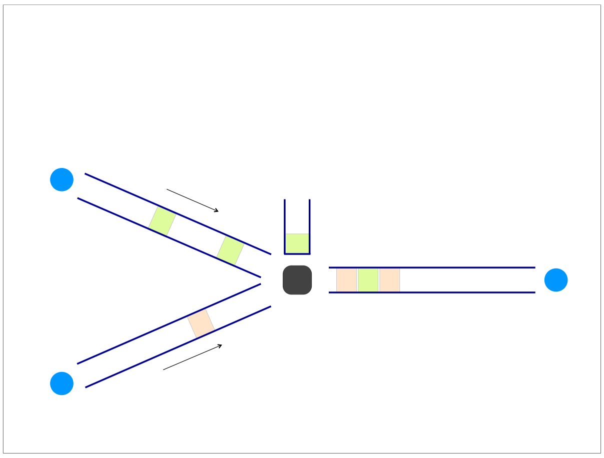

Queuing delay depends on the traffic pattern

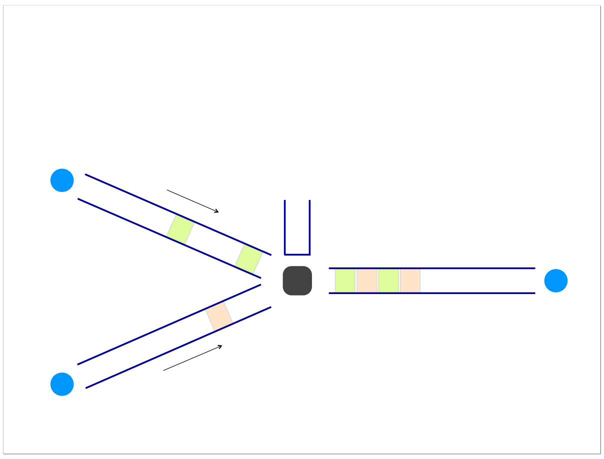

No overload

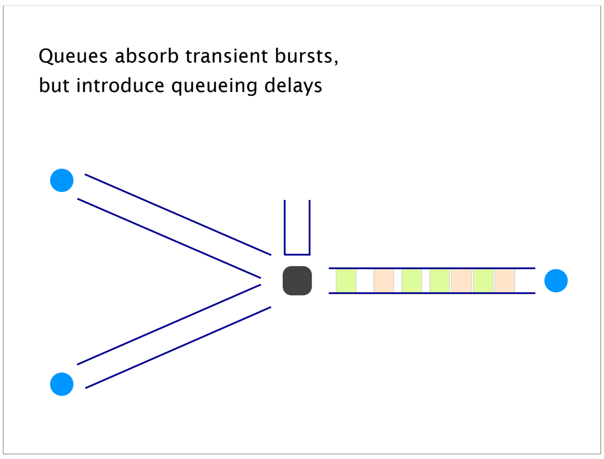

Queuing delay depends on the traffic pattern
Queue
Transient overload!


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

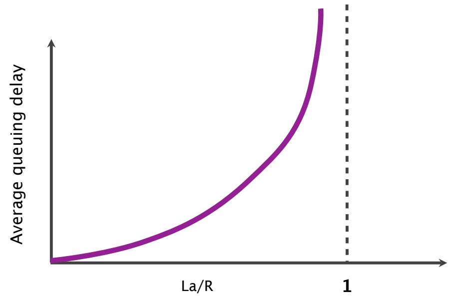
average packet arrival rate	*a*	[packet/sec]
transmission rate of outgoing link	*R*	[bit/sec]
fixed packets length	*L*	[bit]
average bits arrival rate	*La*	[bit/sec]
traffic intensity	*La/R*	

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

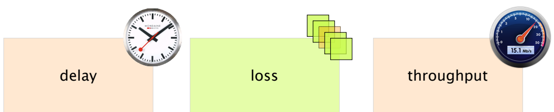
Golden rule

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

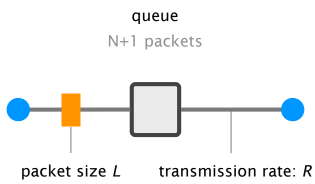
When the traffic intensity is <=1,
queueing delay depends on the burst size



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

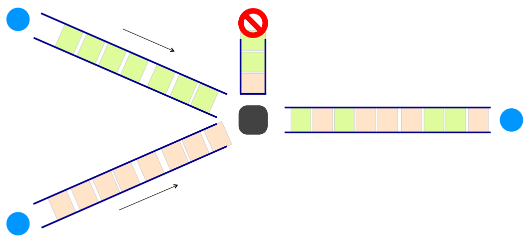


In practice, queues are not infinite.
There is an upper bound on queuing delay.

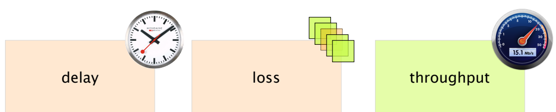


queuing delay upper bound: N*L/R

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



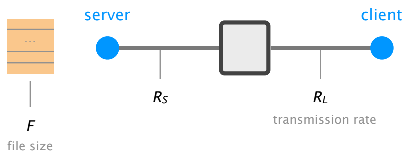
A network *connection* is characterized by its delay, loss rate and throughput



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

Average throughput [#bits/sec]	=	data size / transfer time	[#bits] / [sec]

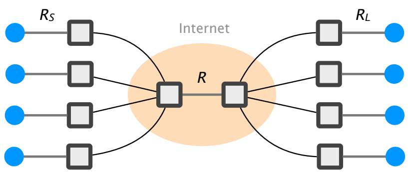
To compute throughput, one has to consider the bottleneck link



Average throughput $\min(R_S, R_L)$

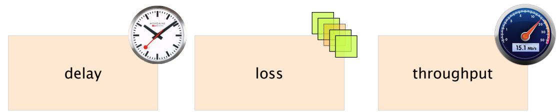
= transmission rate of the bottleneck link

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



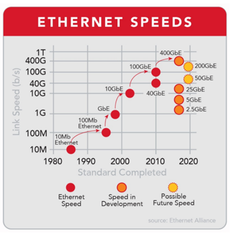
if $4 \cdot \min(R_S, R_L) > R$ the bottleneck is now in the core, providing each download R/4 of throughput

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



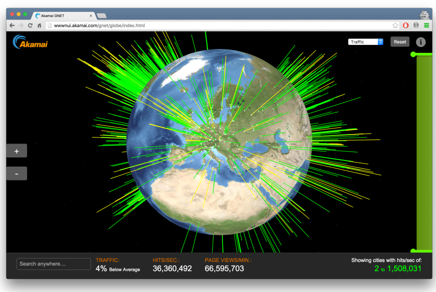
As technology improves, throughput increase & delays are getting lower except for propagation

(speed of light)



source: ciena.com

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



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?

Communication Networks

Spring 2022

Laurent Vanbever
nsg.ee.ethz.ch

ETH Zürich (D-ITET)
28 February 2022

Materials inspired from Scott Shenker & Jennifer Rexford