Intro to Networking for HPC

A visual crash course*

inverted CSC
CERN School of Computing


North Pacific Ocean
Australia
New Zealand

I. Networking basics

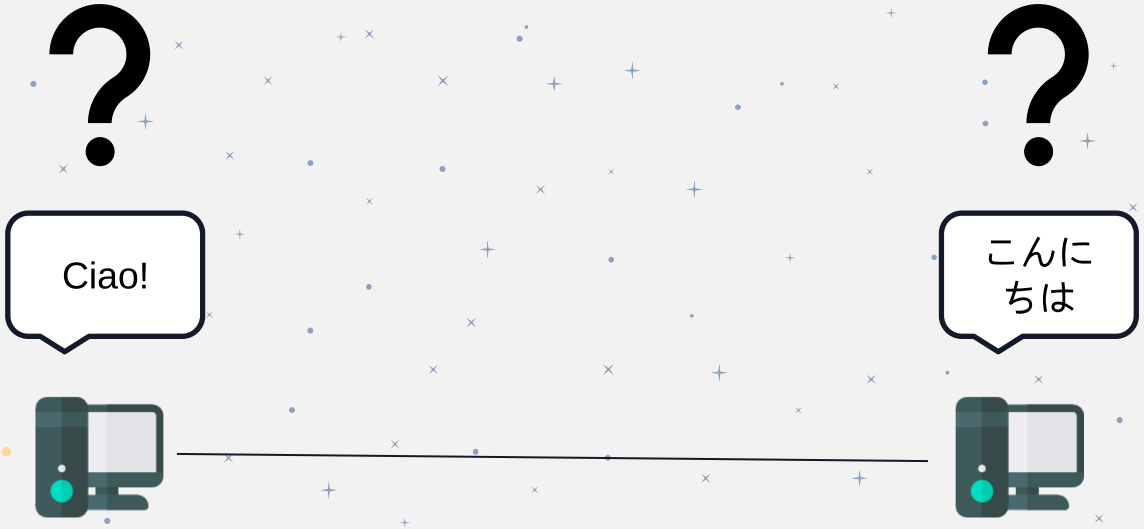
Connecting two hosts

We use a **link**

Such as copper wire

Or radio waves

with the purpose of sending bits


Ciao!
こんにちは

We need rules!

Both hosts need to use the same rules

When does a message start?
How is the data encoded?

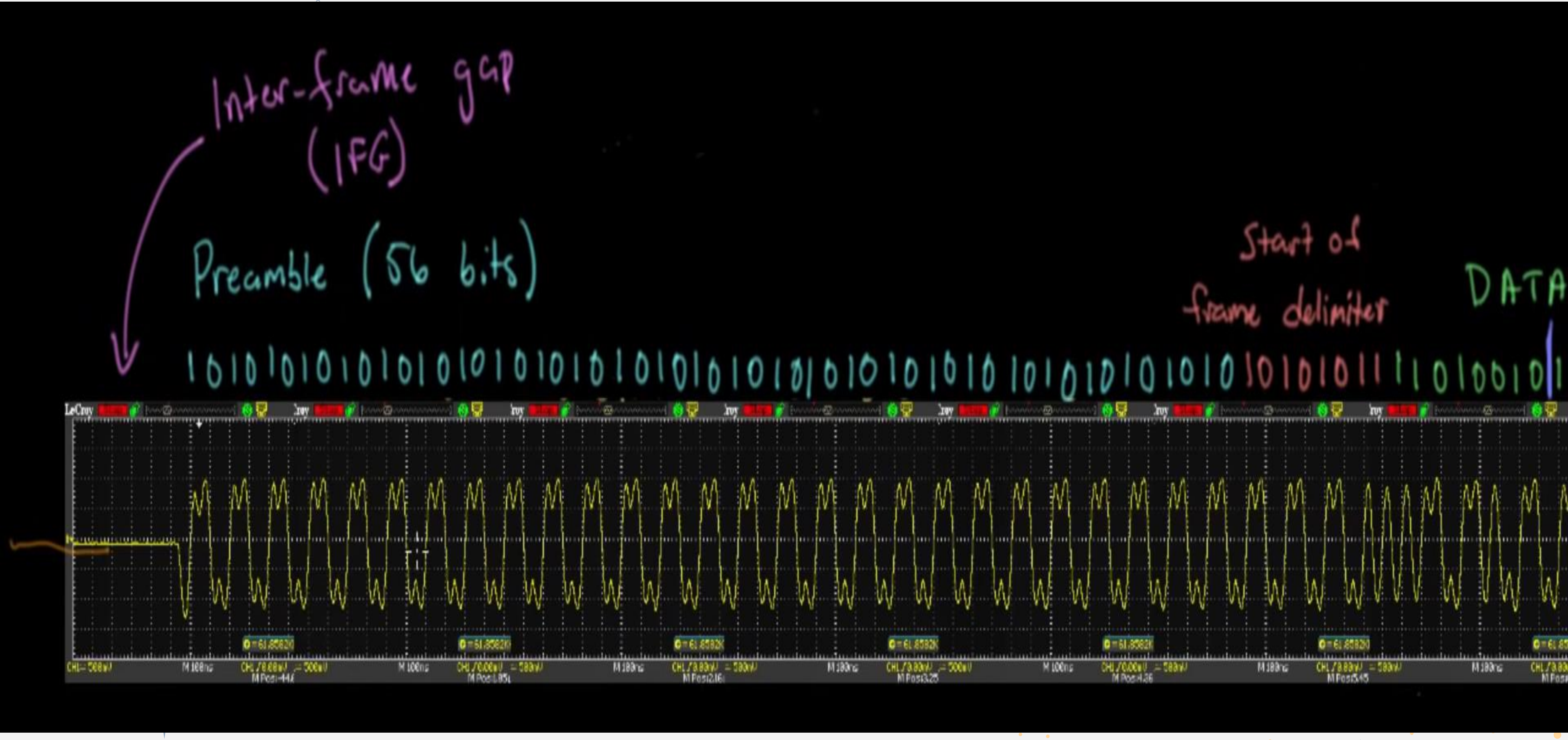
We call these rules network protocols

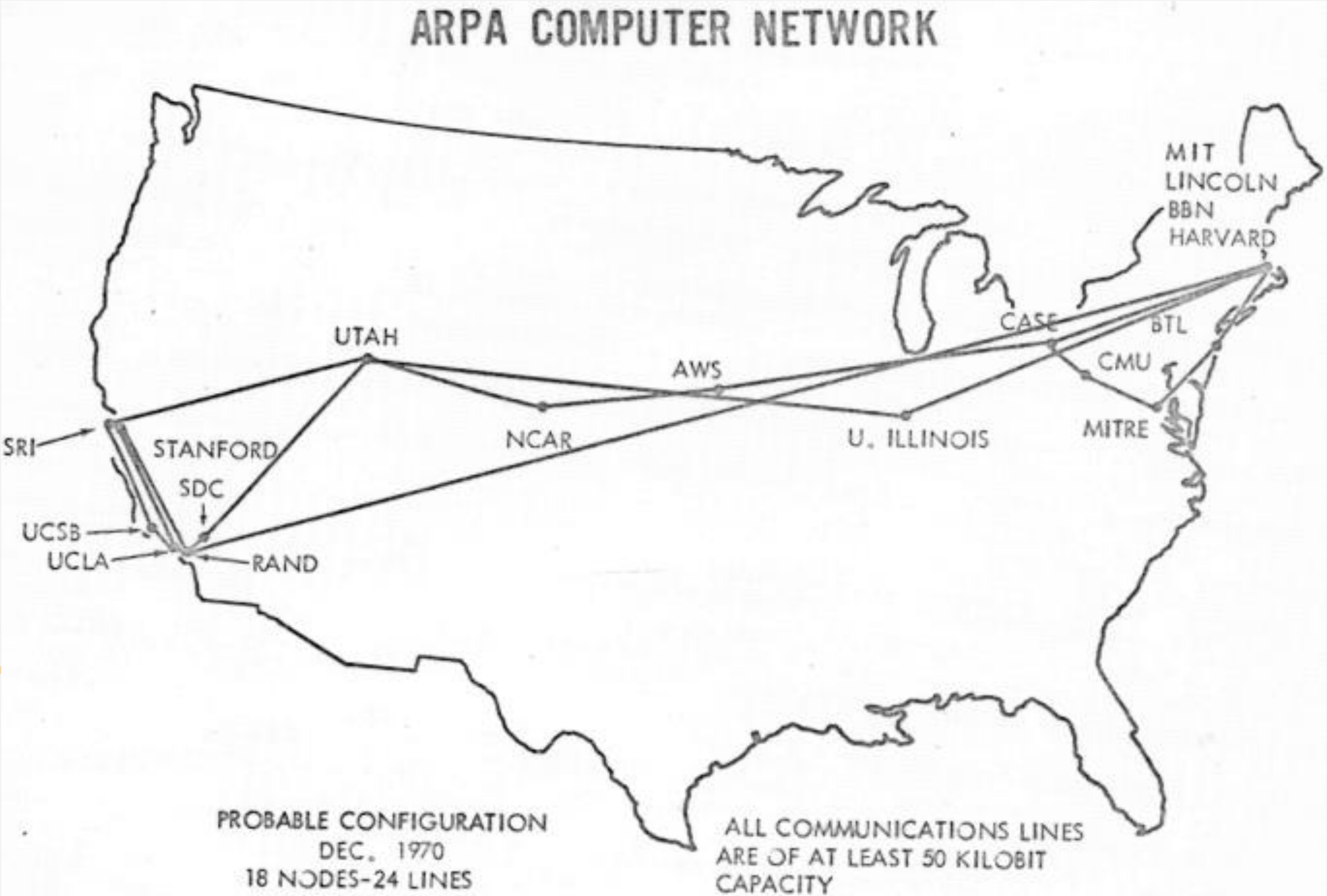
Ethernet – Language of two hosts

Alternatives:

Infiniband often used for HPC in the datacenters

Wi-Fi for radio waves


Inter-frame gap
(IFG)
Preamble (56 bits)
Start of
frame delimiter
DATA
LeCroy

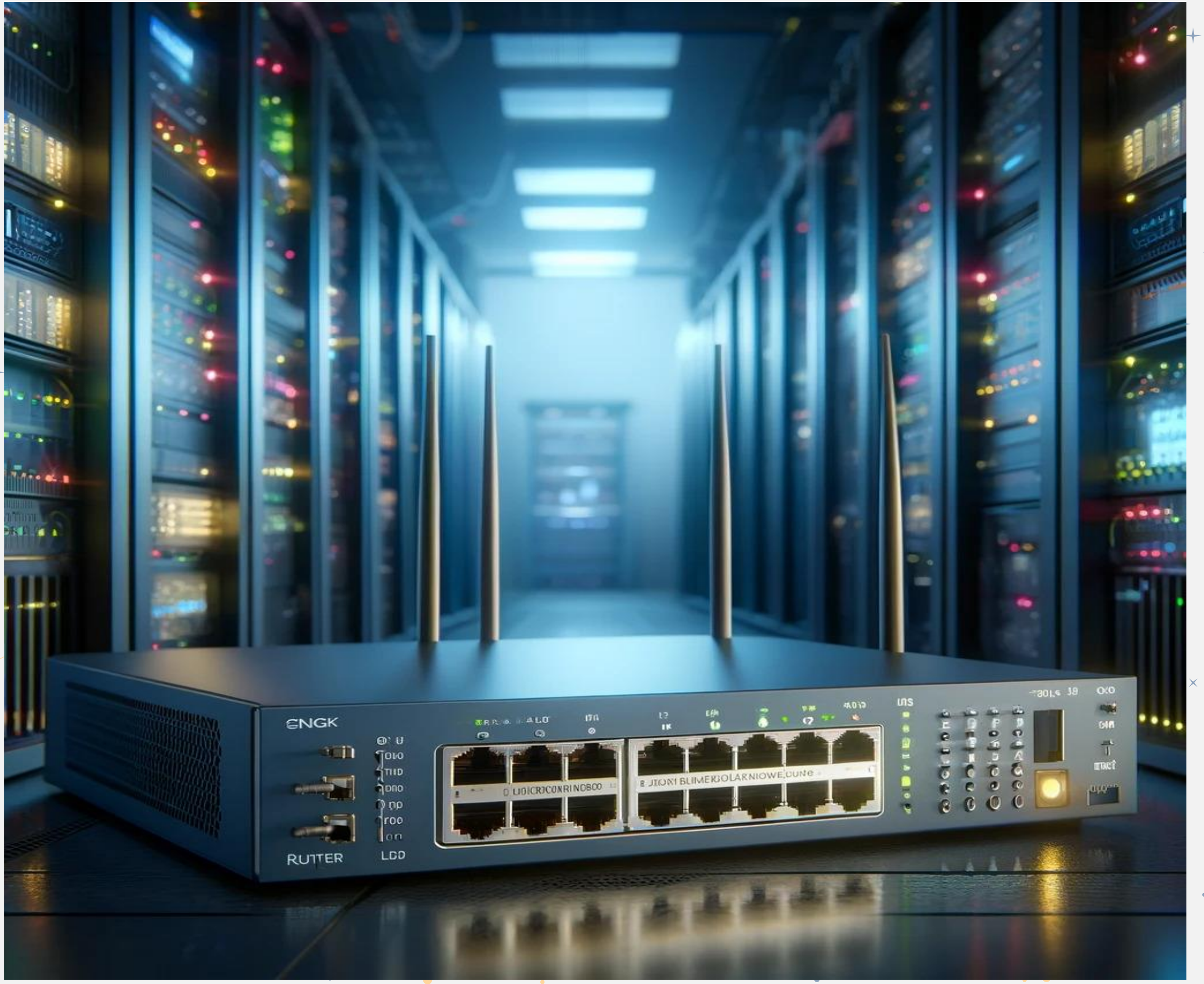

ARPA COMPUTER NETWORK
MIT
LINCOLN
BBN
HARVARD
UTAH
AWS
CASE
BTL
CMU
MITRE
U. ILLINOIS
NCAR
SRI
STANFORD
SDC
UCSB
UCLA
RAND
PROBABLE CONFIGURATION
DEC. 1970
18 NODES-24 LINES
ALL COMMUNICATIONS LINES
ARE OF AT LEAST 50 KILOBIT
CAPACITY

Internet Protocol (IP)

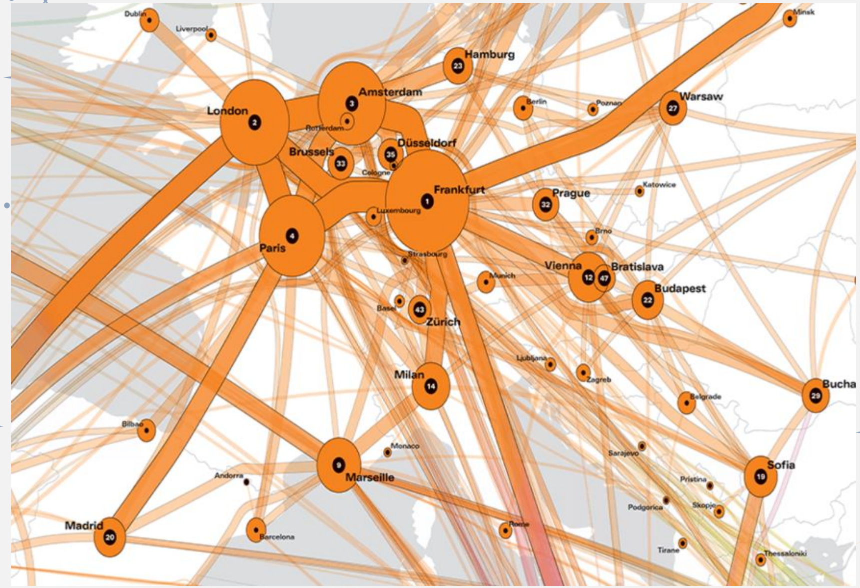
Developed to send data to hosts not directly connected

We use **routers** to forward packets

We can now have networks - little **Internets!**


ENGK
RUTER
LDD

72.32.14.16 (IP)



194.32.14.32(IP)

The internet (aka many routers) connecting networks

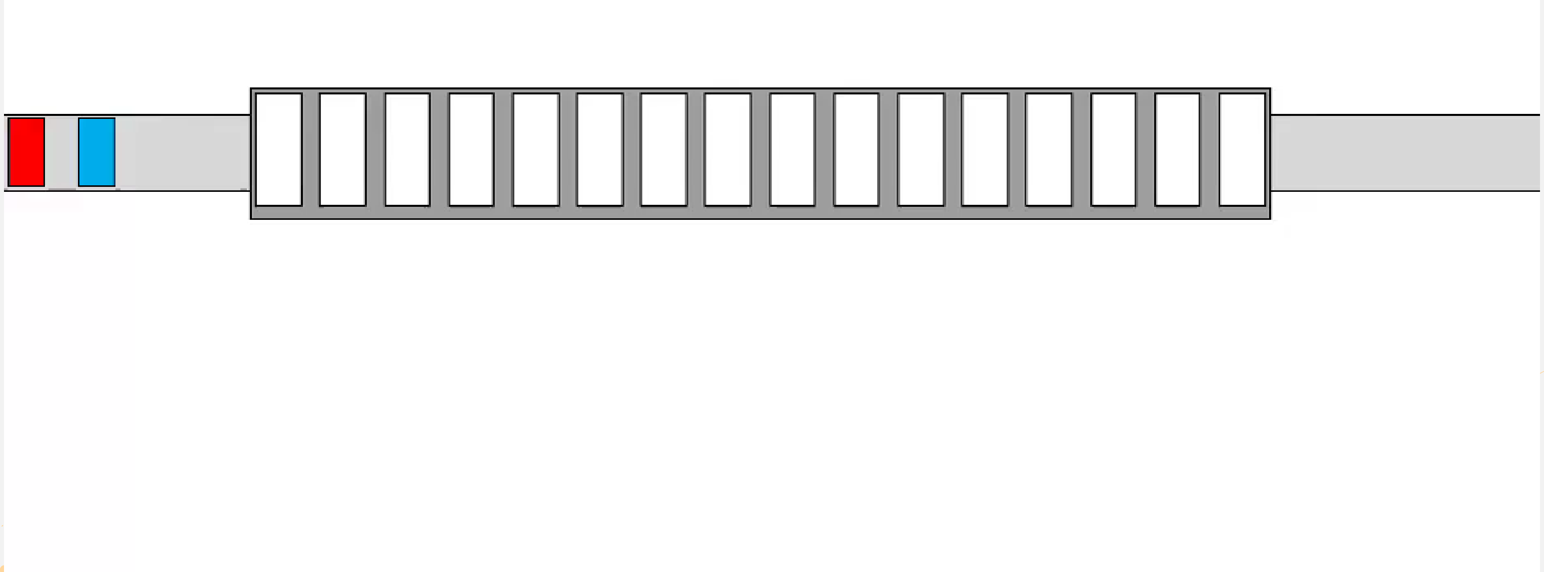
Transport Protocols

We want to run multiple network application on the same host

- We add a new protocol with a field called **PORT**

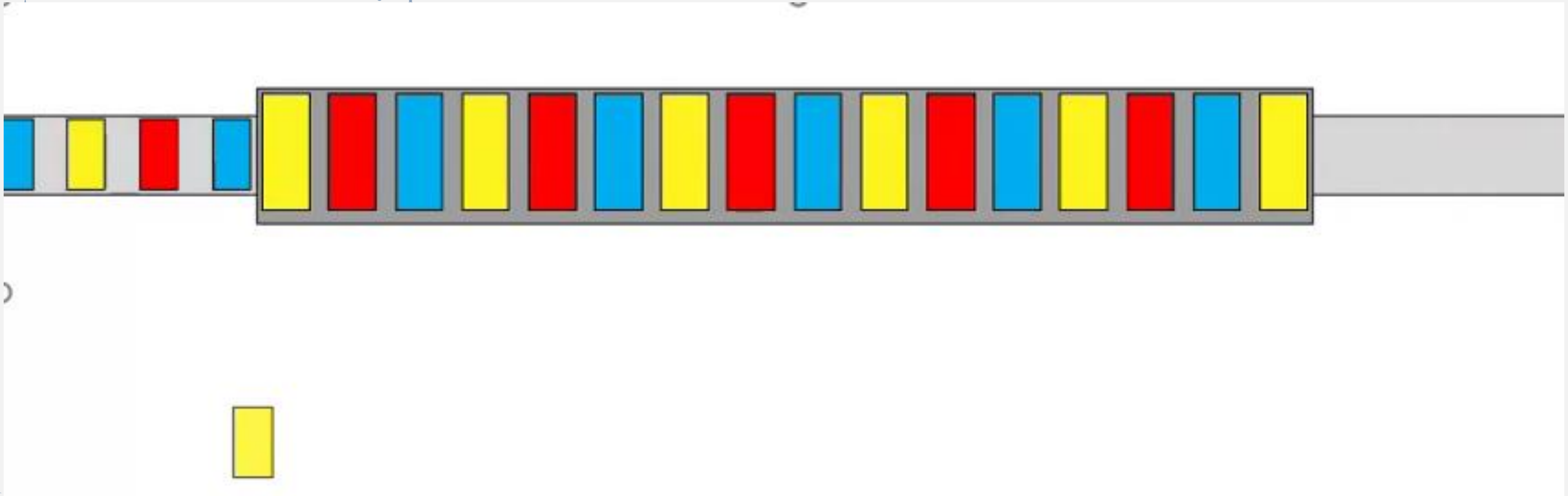
- User Datagram Protocol (UDP)
- Transmission Control Protocol (TCP)

Life is hard, packets get lost sometimes



We need **reliable** transfer of data – retransmission

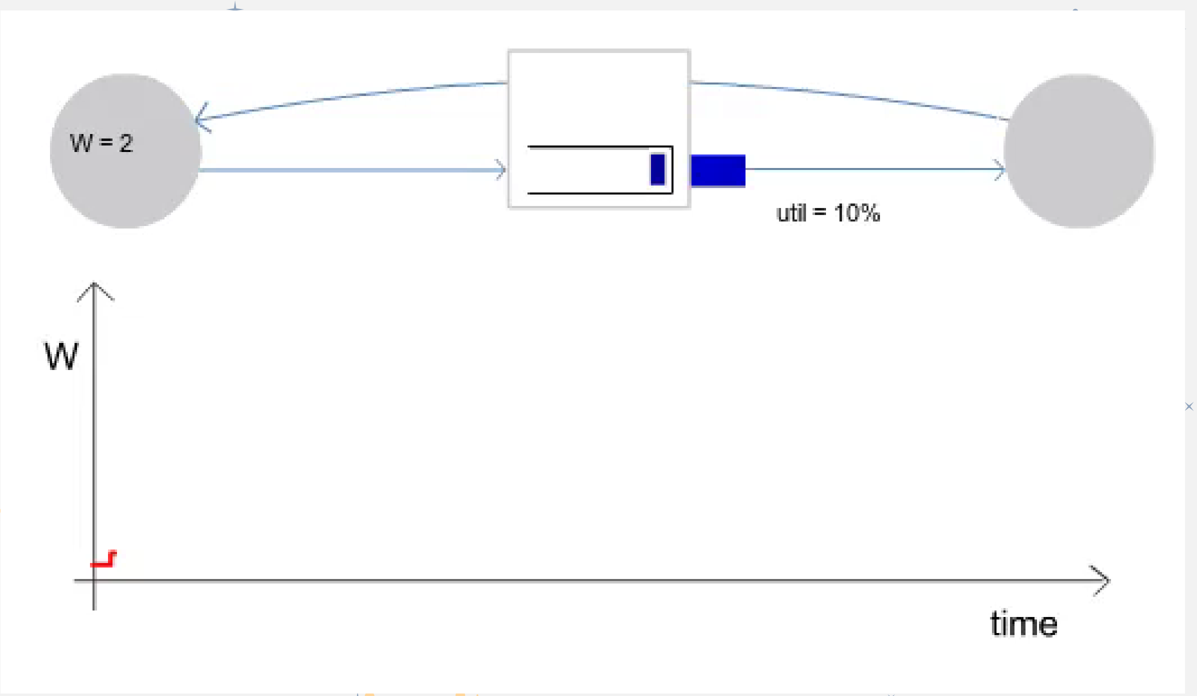
Life is hard, packets get lost sometimes



We need **reliable** transfer of data - retransmission

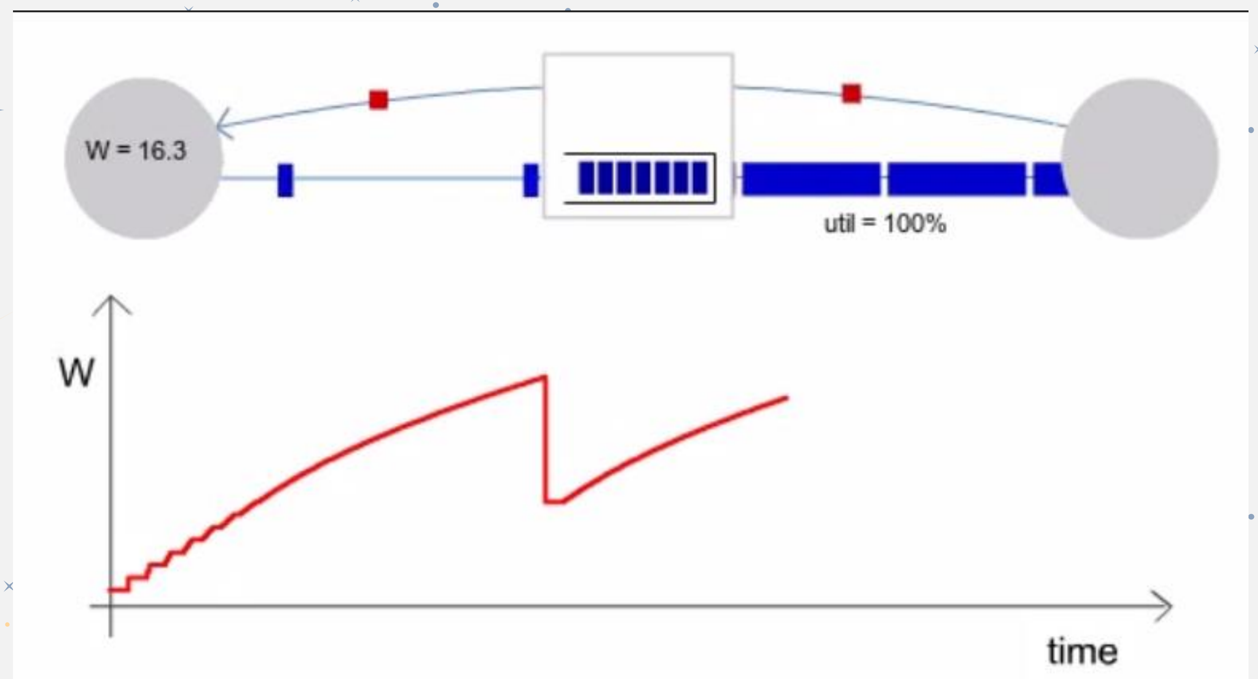
Congestion Control

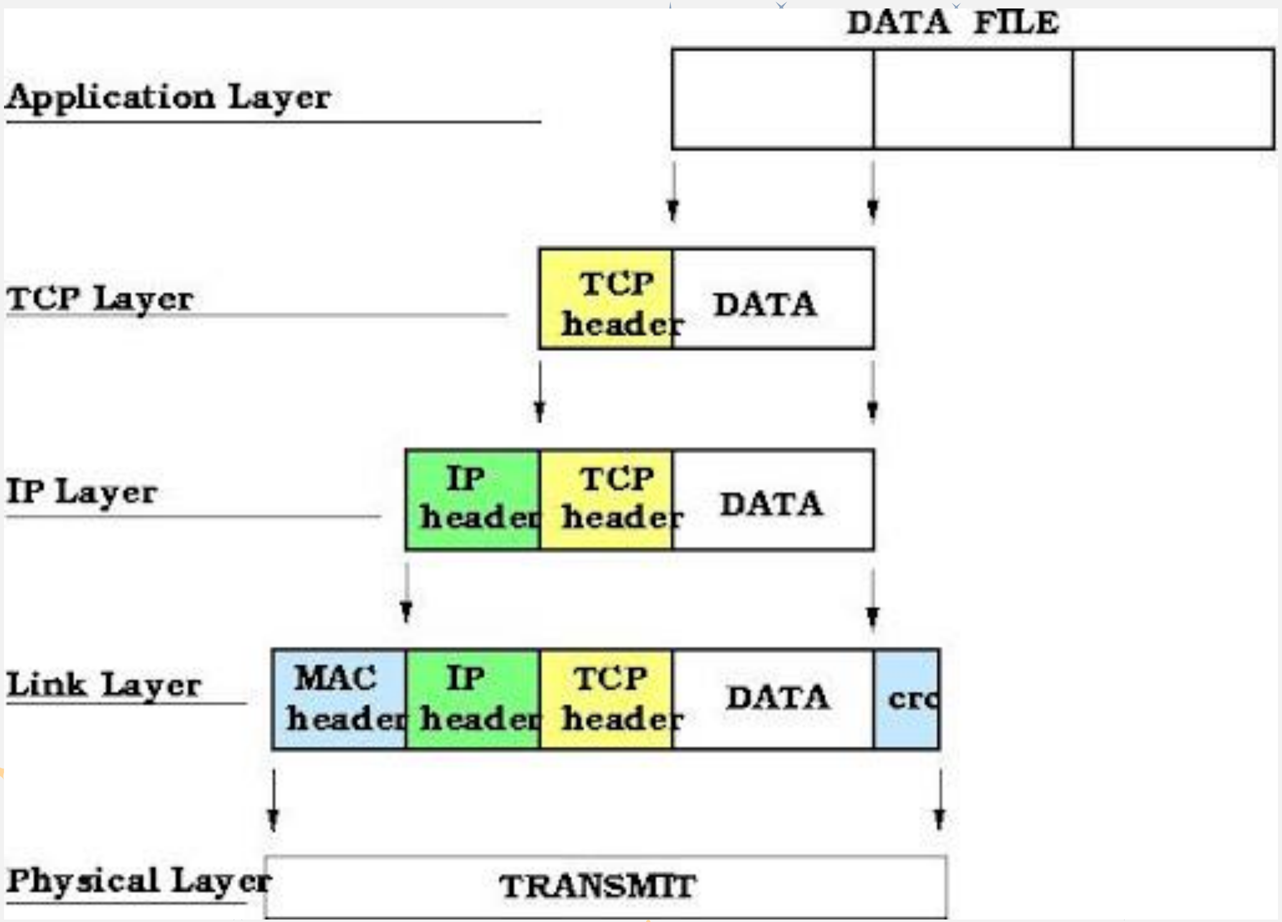
Reliability => Retransmission => Congestion



Congestion Control

Reliability => Retransmission => Congestion





DATA FILE
Application Layer
TCP Layer
TCP header
DATA
IP Layer
IP header
TCP header
DATA
Link Layer
MAC header
IP header
TCP header
DATA
crc
Physical Layer
TRANSMIT

Application protocols

Upon TCP/IP you can build your own

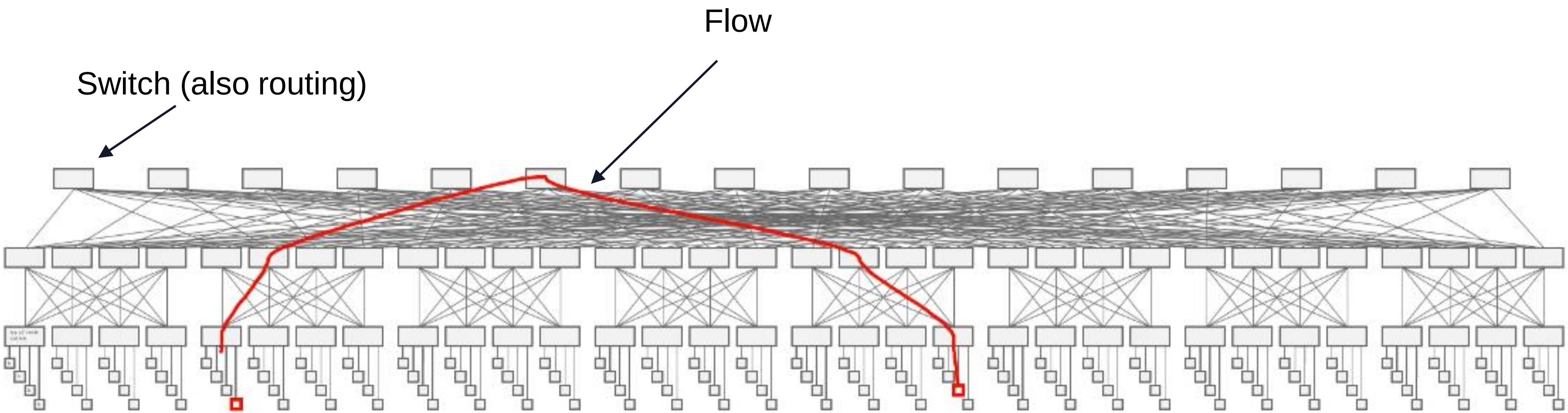
HTTP invented by **Tim Berns** at CERN
the basis for the World Wide Web

Domain Name Protocol (DNS)
(142.250.186.142 <-> google.com)

Message Passing Interface(MPI) for
parallel computing

II. The Datacenter

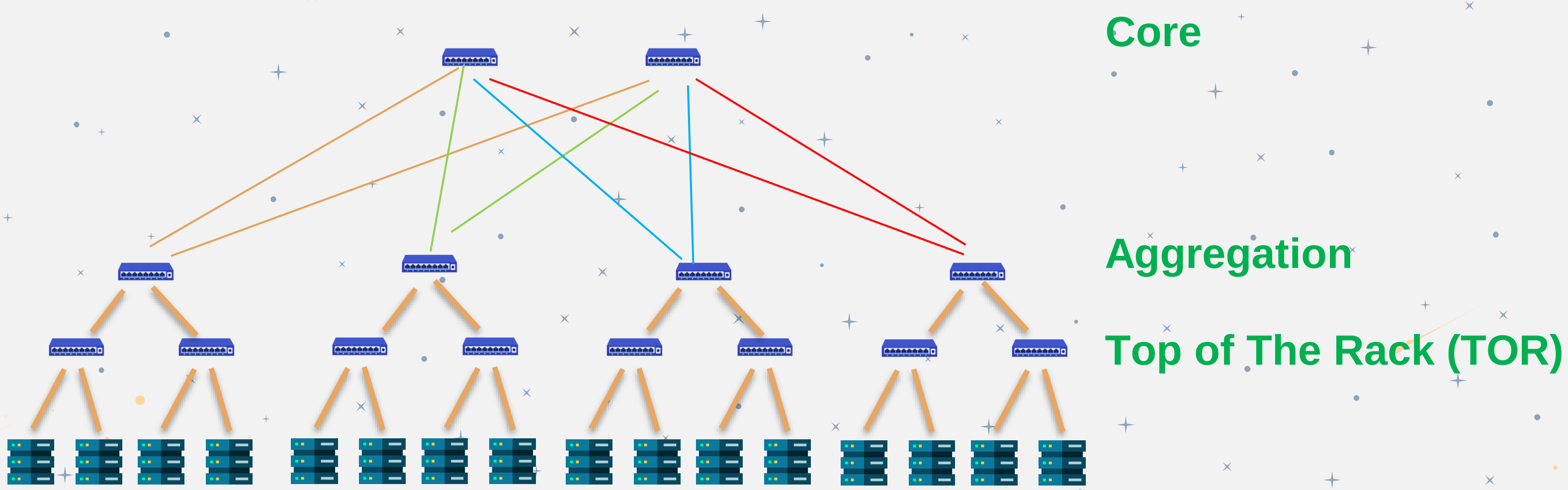

Switch


Flow
Switch (also routing)

Each server should be able to speak with any other server

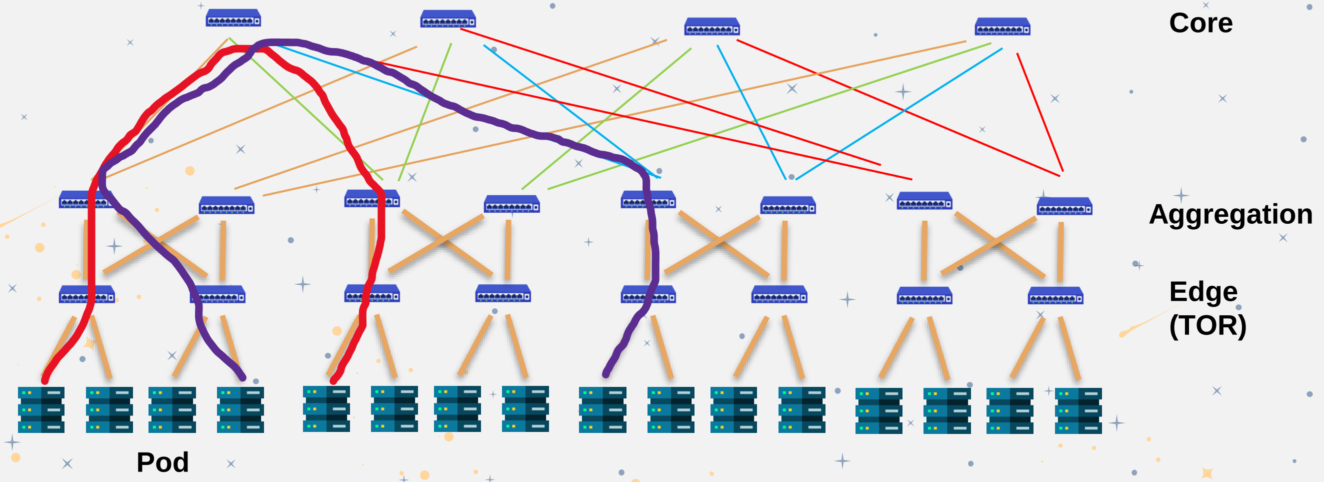
If most of the traffic is intra- data center then there is a very uneven distribution of bandwidth

Hierarchical



Fat Tree

Special case of Clos networks (1952 telecommunications)



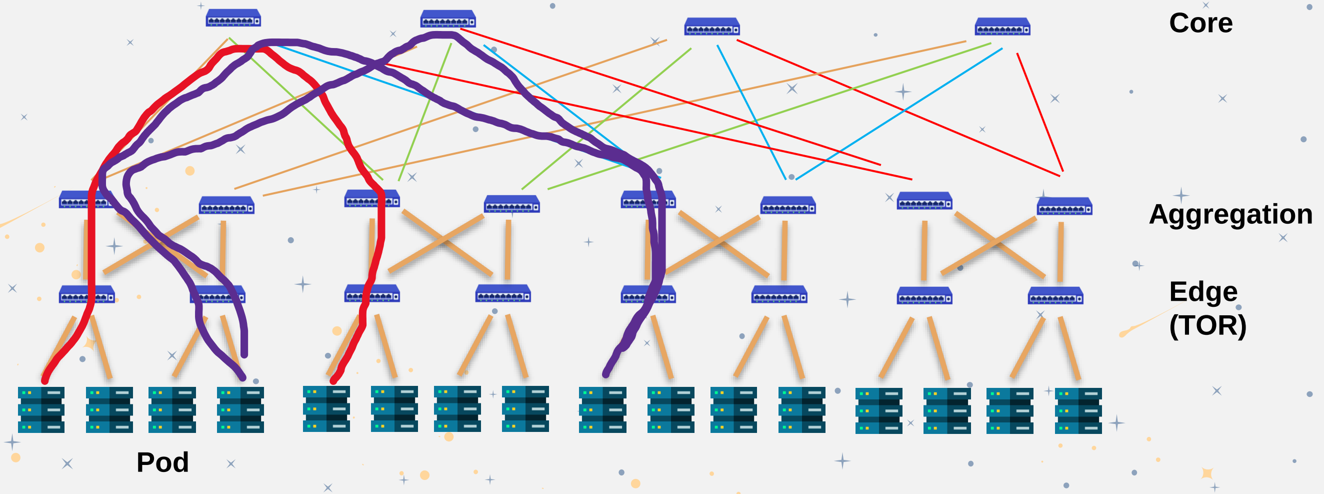
Al-Fares, M., Loukissas, A., & Vahdat, A. (2008). A scalable, commodity data center network architecture. *ACM SIGCOMM computer communication review*, *38*(4), 63-74.

Fat Tree

Special case of Clos networks (1952 telecommunications)

Theorem: there exists an **optimal** arrangement of flows

Most of the datacenters use it

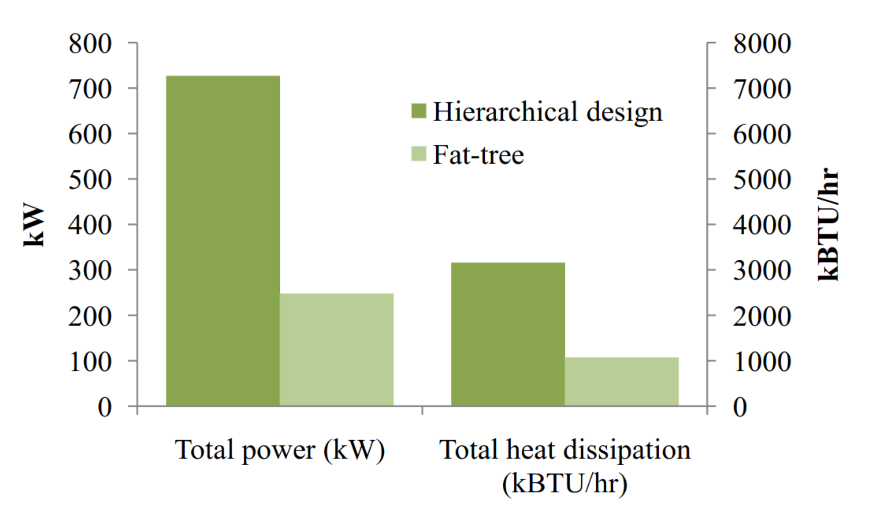


Al-Fares, M., Loukissas, A., & Vahdat, A. (2008). A scalable, commodity data center network architecture. *ACM SIGCOMM computer communication review*, *38*(4), 63-74.


NORTH AND SOUTH AMERICA
LONDON
PARIS

Why Fat Tree?

- **Scales better**
- **We can use consumer grade Ethernet fabric**



	Hierarchical design			Fat-tree		
Year	10 GigE	Hosts	Cost/ GigE	GigE	Hosts	Cost/ GigE
2002	28-port	4,480	$25.3K	28-port	5,488	$4.5K
2004	32-port	7,680	$4.4K	48-port	27,648	$1.6K
2006	64-port	10,240	$2.1K	48-port	27,648	$1.2K
2008	128-port	20,480	$1.8K	48-port	27,648	$0.3K

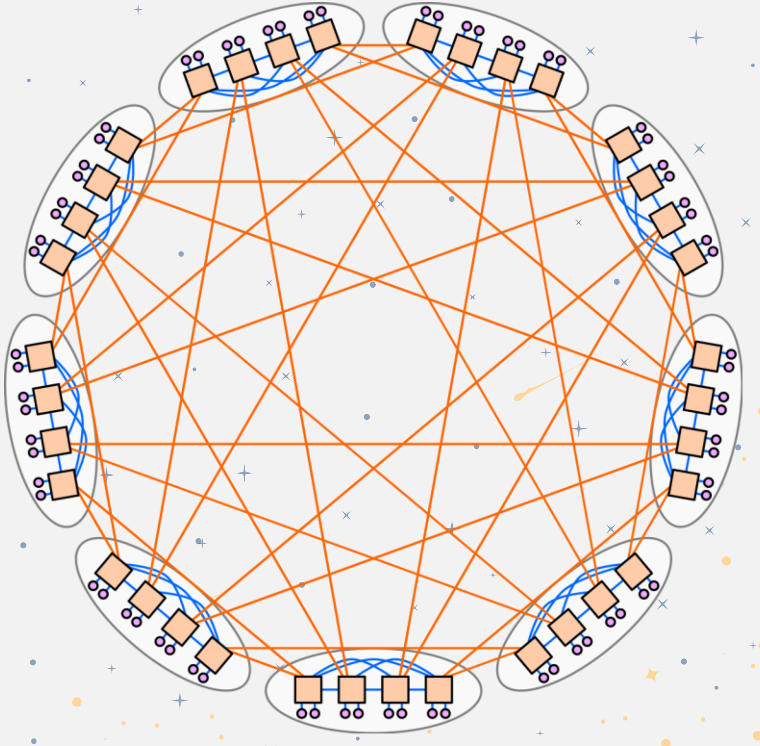
Al-Fares, M., Loukissas, A., & Vahdat, A. (2008). A scalable, commodity data center network architecture. *ACM SIGCOMM computer communication review*, *38*(4), 63-74.

DragonFly

"cables dominate network cost"[1]

Dragonfly was designed to reduce this cost

Used only in some HPC datacenters



[1]Kim, John, et al. "Technology-driven, highly-scalable dragonfly topology." *ACM SIGARCH Computer Architecture News* 36.3 (2008): 77-88.

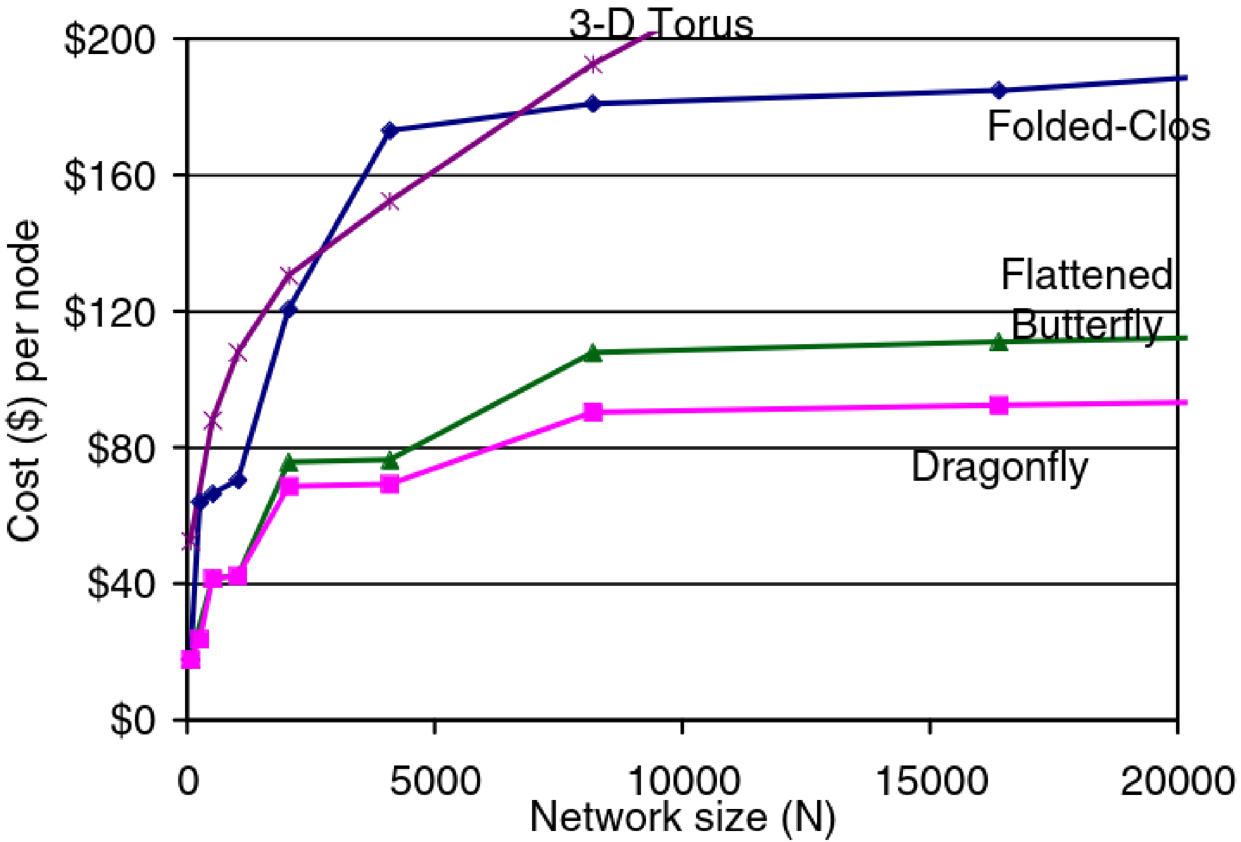
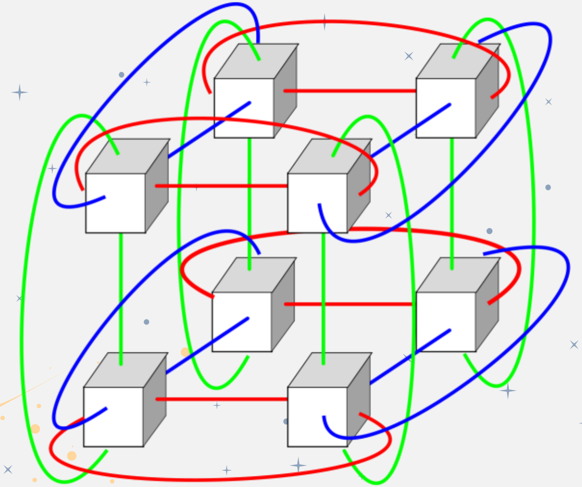
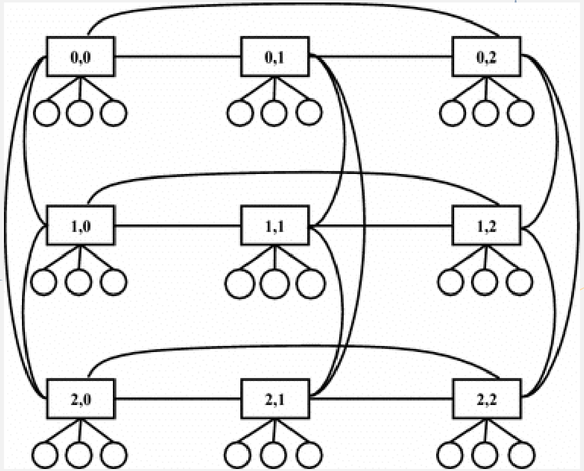


Figure 19. Cost comparison of the dragonfly topology to alternative topologies.

Lots of other topologies for specialized workloads



3d Torus

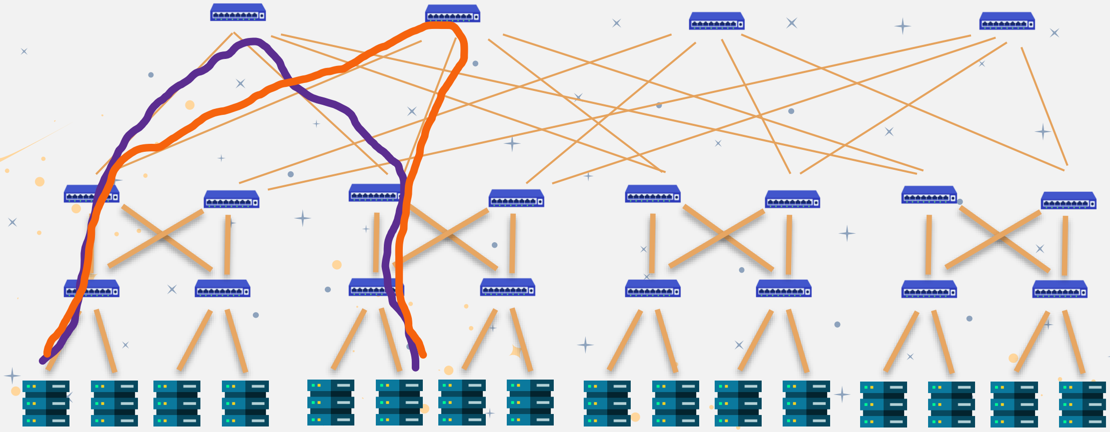


Butterfly

How are paths chosen?

Each flow is placed on one randomly chosen path between the endpoints by using the Equal-cost multipath (ECMP)

A form of load balancing the links



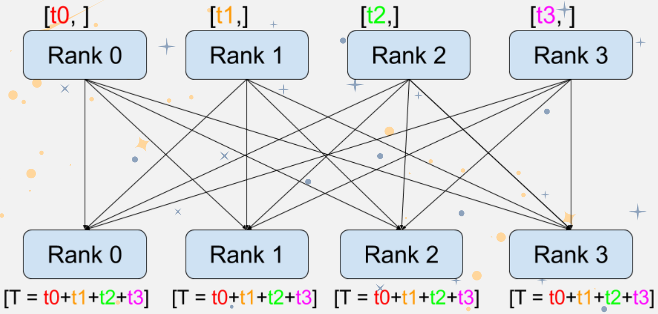
III. Achieving High Throughput and Low latency

GPUs create more traffic than ever

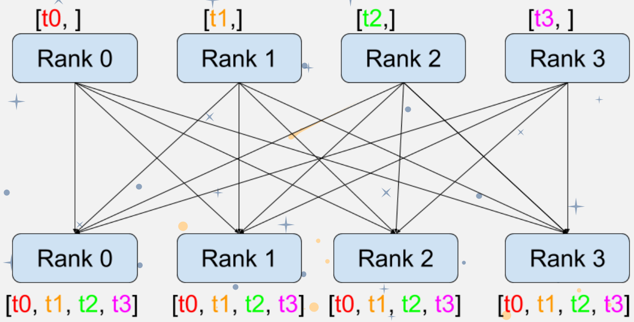
Traffic Patterns between GPUS cause overloading of the network

Usually implemented in communication libraries (e.g. NCCL, RCCL, MPI)

All-reduce



All-gather



For distributed computation – OpenMPI, PyTorch

When you have 1000Os of GPUS –

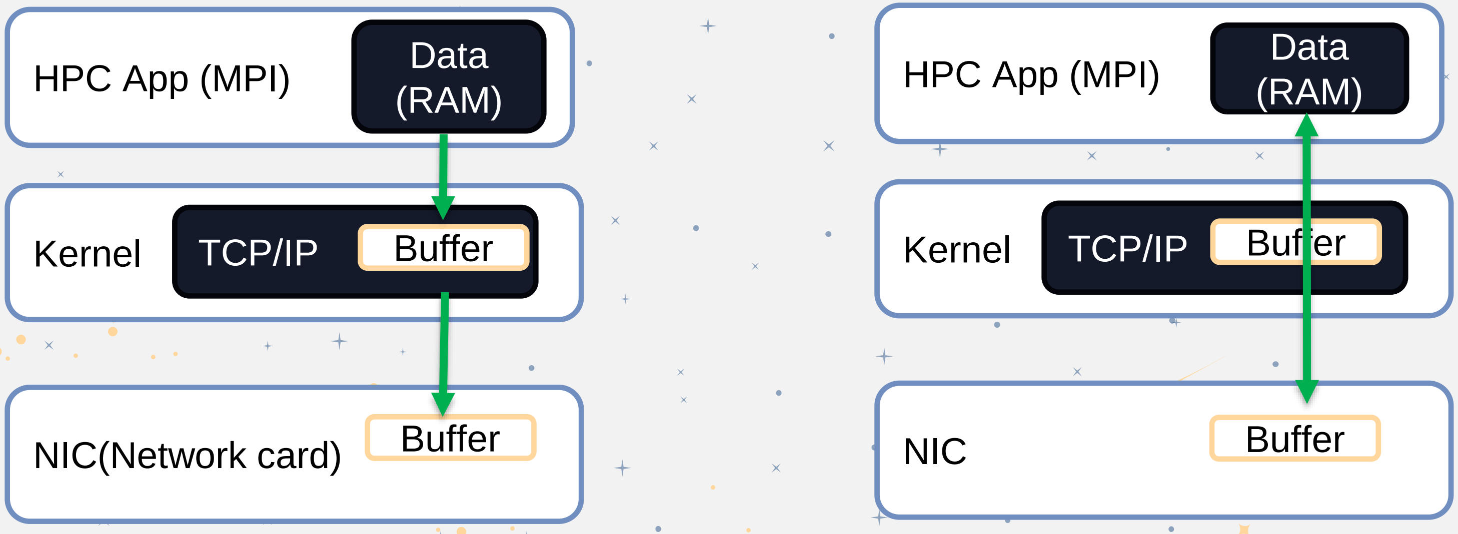
Communication becomes a bottleneck

	kernel_type	sum	percentage
0	COMPUTATION	3430656	61.3
1	COMMUNICATION	2167936	38.7
2	MEMORY	408	0.0
3	COMPUTATION overlapping COMMUNICATION	0	0.0
4	COMPUTATION overlapping MEMORY	0	0.0

Remote Direct Memory Access (RDMA)

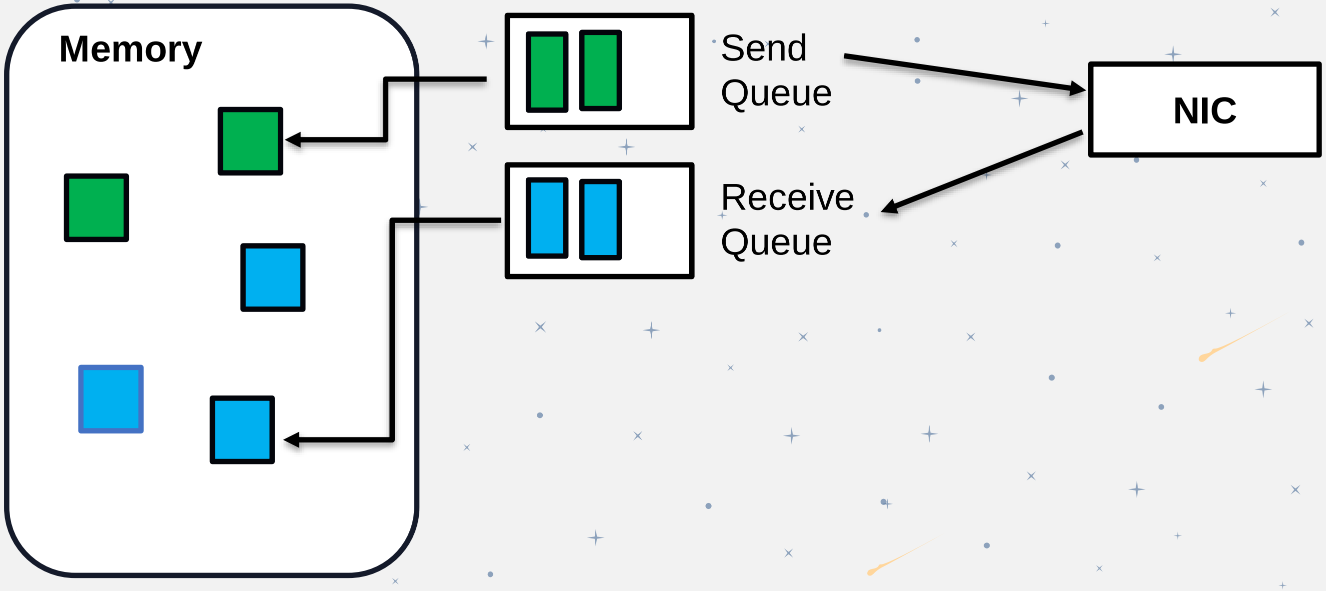
Packets bypass the kernel and CPU (vs through the kernel in TCP/IP)

Initially added to **InfiniBand fabric**

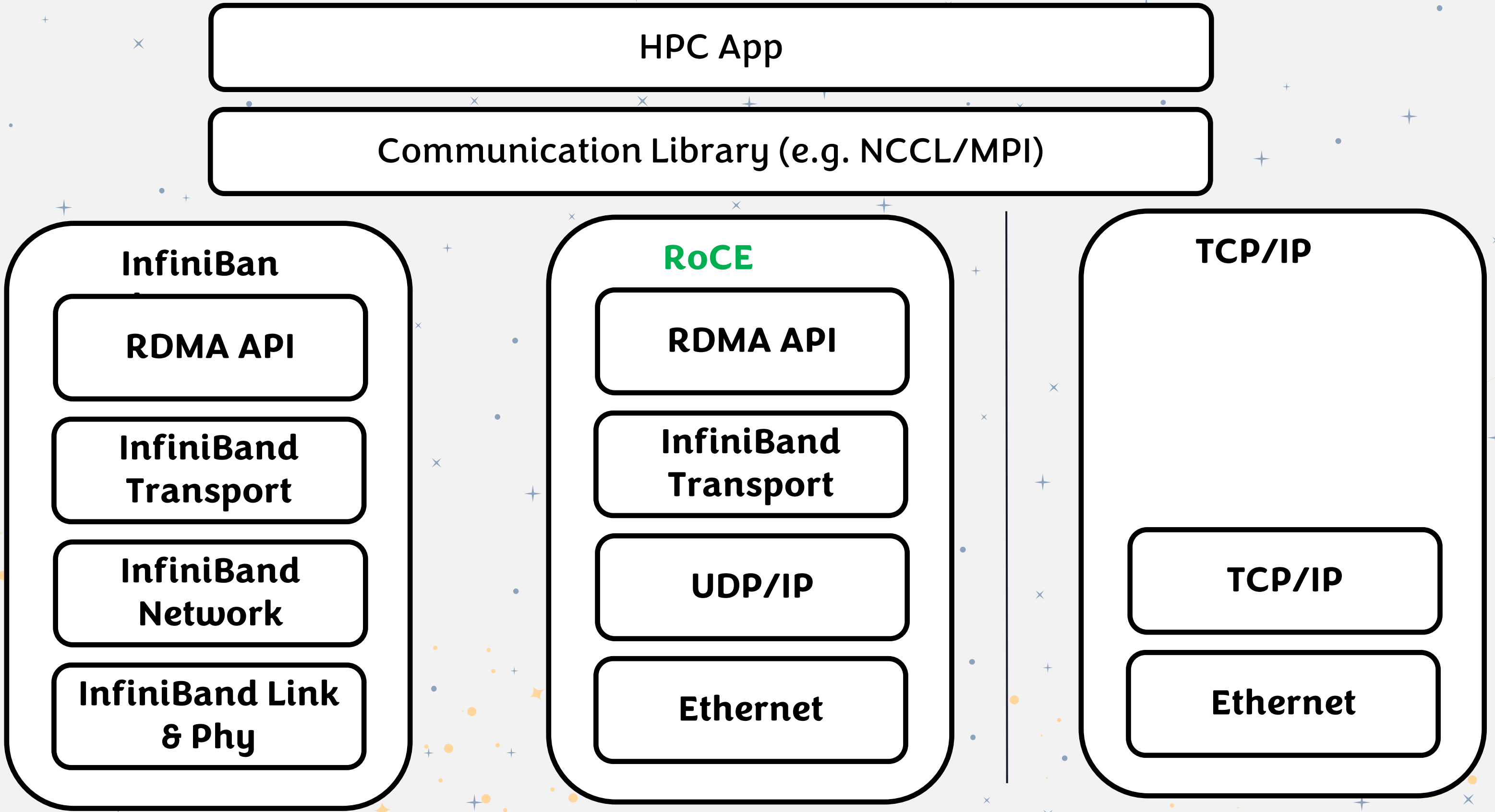


Recio, Renato, et al. A remote direct memory access protocol specification. No. rfc5040. 2007.

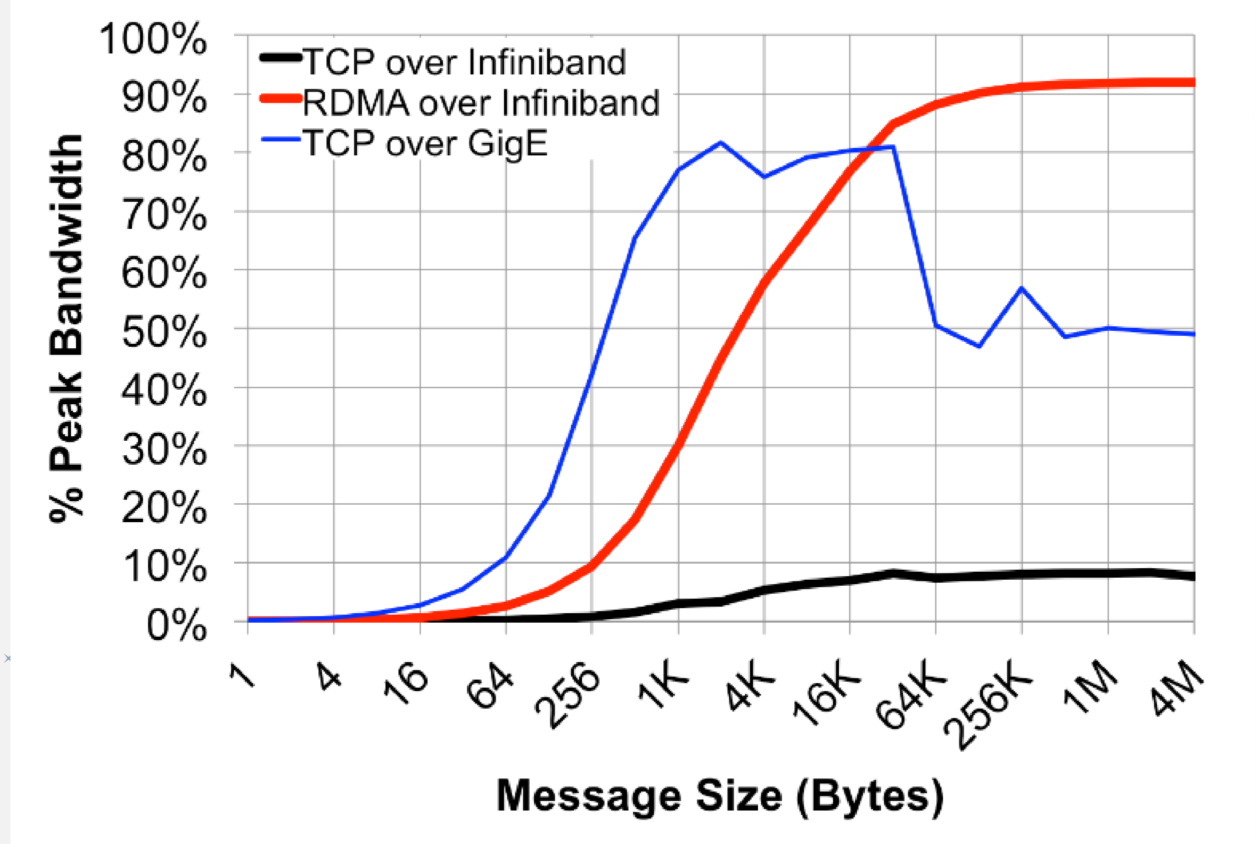
Looking closer



How does the network stack look like?



Why RDMA?



TCP over Infiniband
RDMA over Infiniband
TCP over GigE
% Peak Bandwidth
100%
90%
80%
70%
60%
50%
40%
30%
20%
10%
0%
1
4
16
64
256
1K
4K
16K
64K
256K
1M
4M
Message Size (Bytes)

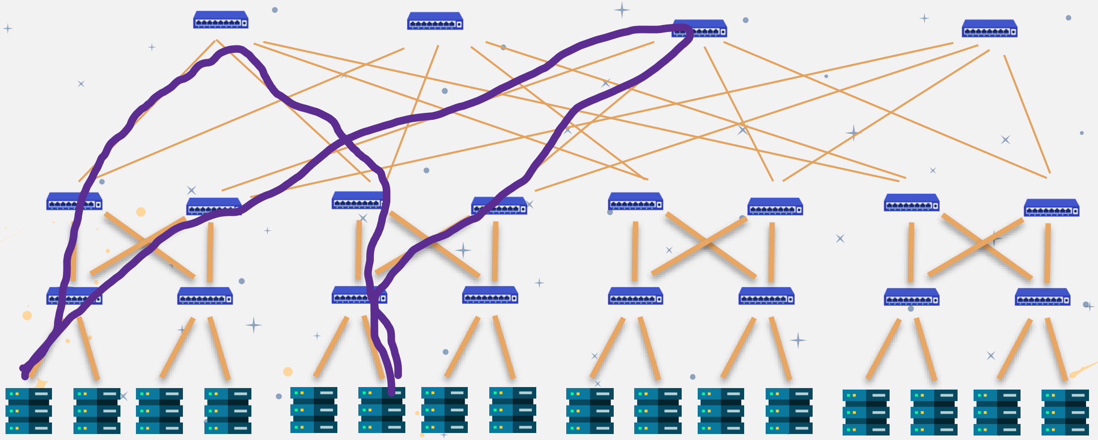
Rise of multipath protocols

The protocols so far are single-path transports (e.g. RoCE or TCP)

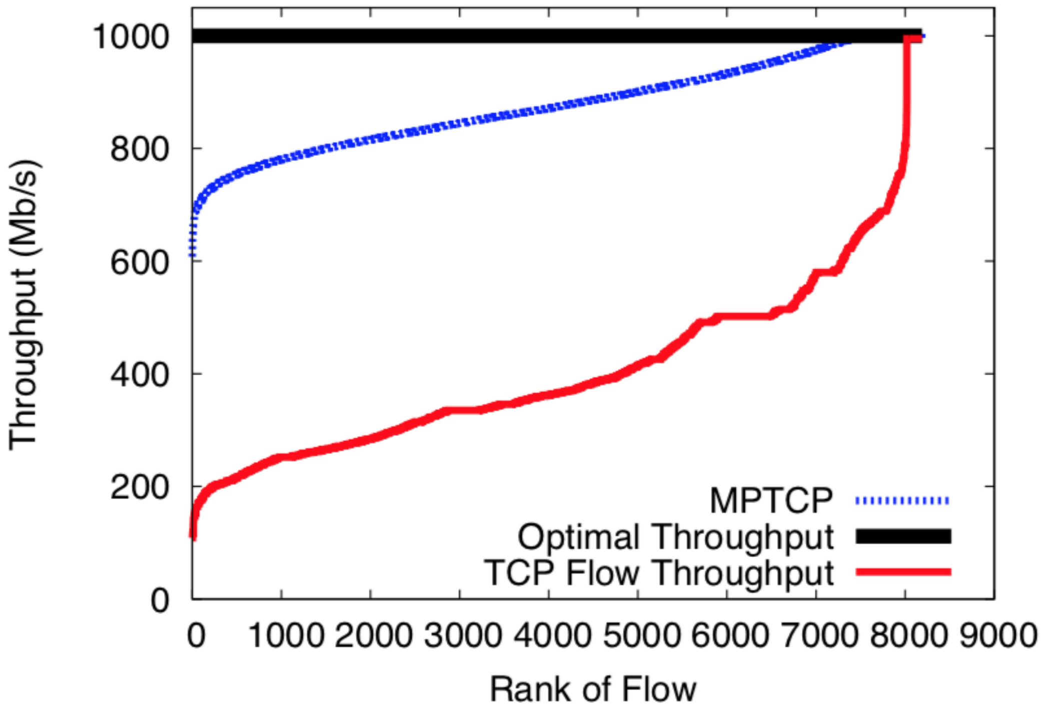
each flow is placed on one randomly chosen path by ECMP

What if we break a flow into multiple smaller flows?

This is called **Multipath**



[1] Raiciu, Costin, et al. "Improving datacenter performance and robustness with multipath TCP." *ACM SIGCOMM Computer Communication Review* 41.4 (2011): 266-277.

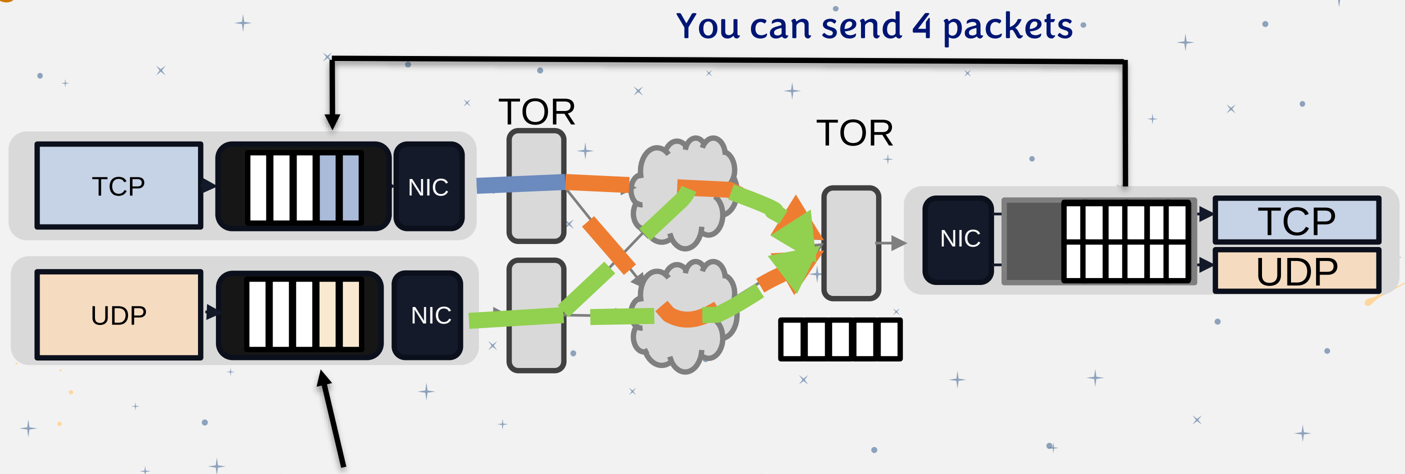


1000
800
600
400
200
0
Throughput (Mb/s)
0 1000 2000 3000 4000 5000 6000 7000 8000 9000
Rank of Flow
MPTCP
Optimal Throughput
TCP Flow Throughput

IV. Recent developments

Receiver driven control loop

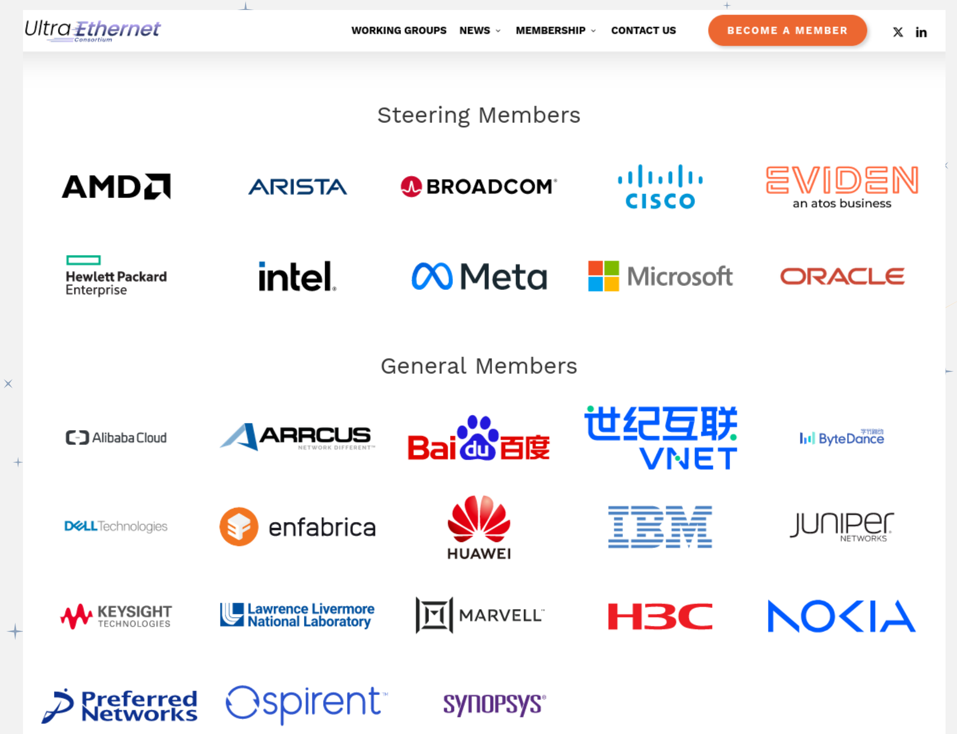
The receiver host controls the "speed" at which the packets are sent by the sender host



The sending host will send packets only when asked by the receiver

Ultra Ethernet Transport

Plans to bypass NVIDIA's monopoly with InfiniBand on datacenter networking



Most big companies are working
On an open network stack over Ethernet for HPC

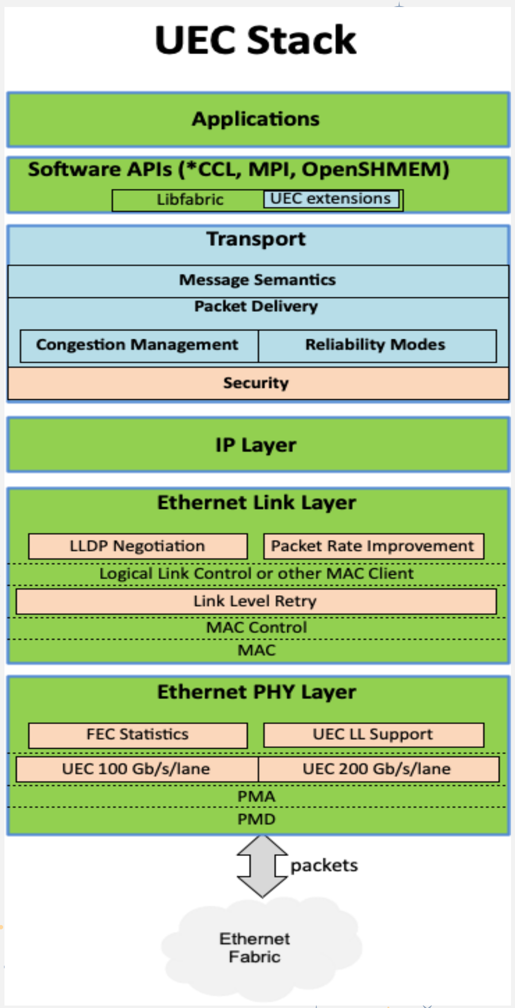
Ultra Ethernet Transport

Multi-path packet spraying

Transport optimized for RDMA

Multiple transport delivery services

and so on...



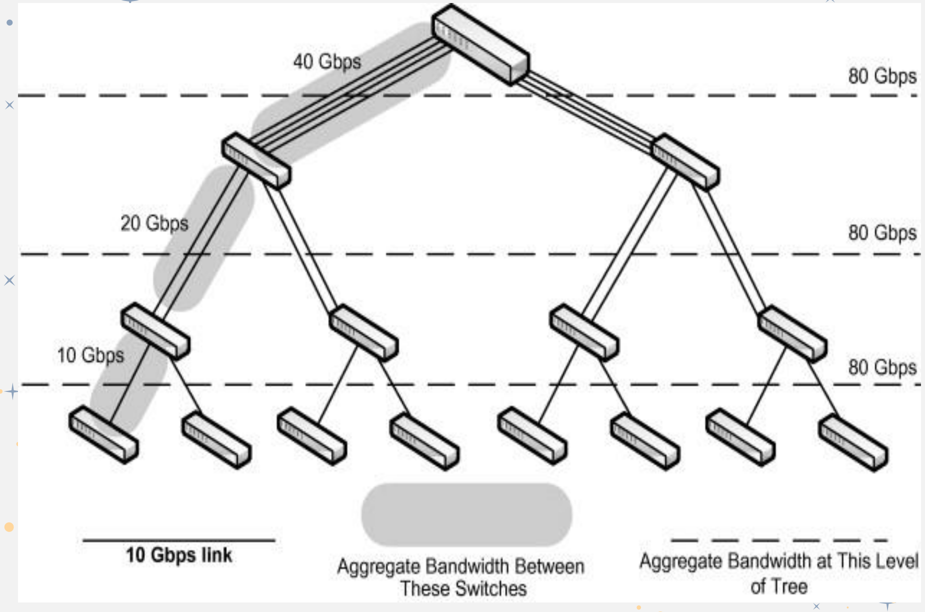
Questions?

Oversubscription

worst-case achievable bandwidth among the end hosts

---

total bisection bandwidth of a particular communication topology



1:1 indicates that the aggregate bandwidth not decrease from one tier to the next as we approach the core.

5:1 means that only 20% of available host bandwidth is available for some communication patterns.