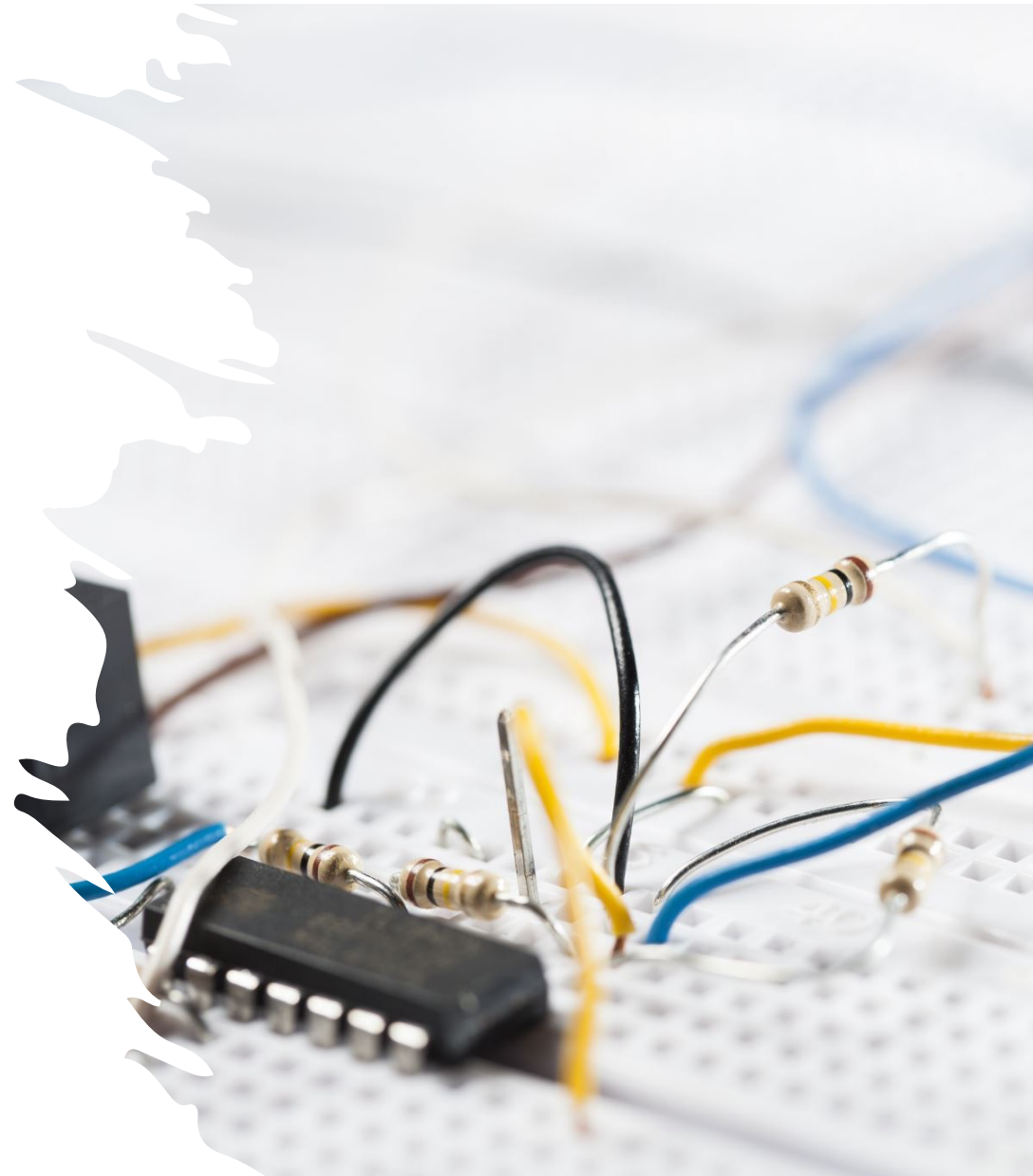


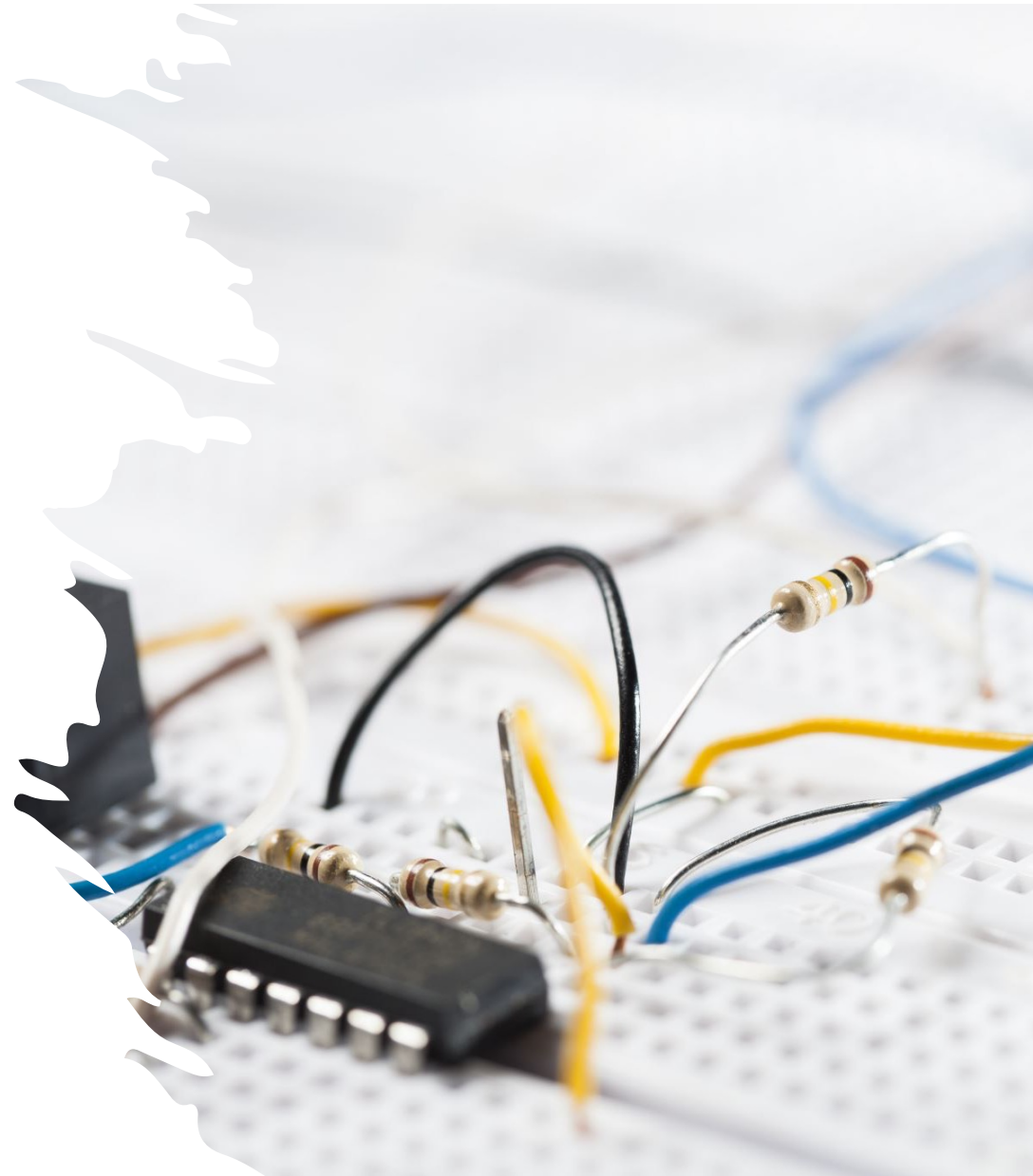
50 Years of Ethernet

Geoff Huston APNIC
June 2026



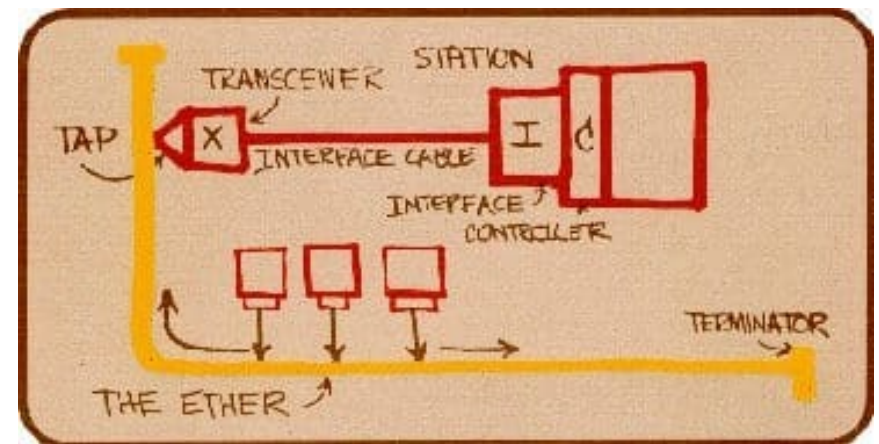
50 Years of ~~Ethernet~~ Computer Networking

Geoff Huston APNIC
June 2026



53 Years Ago

In May 1973 Bob Metcalf from the Xerox Palo Alto Research Centre published a memo that described “X-Wire”, a 3Mbps common bus local area network which became the ubiquitous “Ethernet” which has dominated the networking space for the next 53 years



Ethernet Evolution

- 1980 – 10Base5 10Mbps common bus system released using 75 ohm “thick wire” coax with vampire taps
- 1985 – 10Base2 10Mbps system using “thin wire” 50 ohm coax cable bus
- 1990 – 10BaseT 10Mbps system using twisted pair cables in a radial topology and a central hub in place of a common bus
- 1995 - 100Mbps “Fast Ethernet”
- 1998 - 1Gbit Ethernet
- 2002 - 10Gbit Ethernet
- 2015 - 40G and 100G Ethernet
- ? – Terabit Ethernet



Ethernet Evolution

- 1980 – 10Base5 10Mbps common bus system released using 75 ohm “thick wire” coaxial cables and vampire taps
- 1985 – 10Base2 10Mbps system using twisted pair cables in a radial topology and a central hub in place of a common bus
- 1995 - 100Mbps “Fast Ethernet”
- 1998 - 1Gbit Ethernet
- 2002 - 10Gbit Ethernet
- 2015 - 40G and 100G Ethernet
- ? – Terabit Ethernet

Why has Ethernet been so long-lived?



What was so special about Ethernet in 1980?

It was just a common wire!

- It was the classic minimalist network where the collective “intelligence” was placed into the devices that connected to a common wire
- It pushed functionality and cost off a common network infrastructure and out to the edge devices
- It changed traditional hub-and spoke local networks into a distributed peer topology



What was so special about Ethernet in 1980?

It was self-clocked!

- Ethernet used Manchester encoding where both "1" and "0" are encoded with a voltage transition, so there was no need for a separate clock signal to time the bits on the carrier.
- There was no DC bias as there was an equal number of low/high and high/low transitions so there was no DC buildup on the wire
- It did not need to distribute a high precision stable clock signal to every attached device
- All packets used a 64-bit preamble to allow all receivers to sync their receiver clock against the packet sender's clock
- Strict clock stability needed to be sustained over one packet interval



What was so special about Ethernet in 1980?

Packets were BIG and SMALL!

- Ethernet packets were variably sized, where packets were between 64 and 1,518 bytes in total length
- This was a radical departure in building constant rate networks that supported telephony and also allowed for data, into a network designed specifically for data
- Larger packets allowed for more efficient network use for high data volume transfer, while smaller packets allowed for efficient use in small data transactions (think *telnet*!)



What was so special about Ethernet in 1980?

Distributed Collision control

- There was no single coordination control to allow a station to start transmission. The attached devices worked it out by all using the same access algorithm
- When transmitting, a sender monitored the wire to detect if there was another simultaneous sender (a “collision”). The sender aborted the collision (sending a “jam”) and reverted to waiting for a “collision avoidance period” to try again
- All packets had a minimum inter-packet gap to allow equal access!
- Performance was non-deterministic and degraded when the ethernet had too many active attached devices



What was so special about Ethernet in 1980?

It used End-to-End error handling

- The Ethernet wire did not detect or correct bit errors in packets
- If a receiver was unable to receive a packet within some application-determined time, then it had to signal that loss at the application layer.
- Think UDP, not TCP!



What was so special about Ethernet in 1980?

It used unique station addressing

- The 48-bit MAC address was coordinated across Ethernet vendors to ensure that each device had a unique MAC addresses
- No installation configuration – just plug and play



What was so special about Ethernet in 1980?

It used an open accessible standard

- The original Ethernet standard was openly accessible and standardized. It was first published in 1983 as the IEEE 802.3 standard after Xerox relinquished its patents to allow anyone to manufacture compatible equipment



What was so special about Ethernet in 1995?

It used backward-compatibility

- As Ethernet evolved, the frame format and packet size ranges in the standard spec remain invariant.
- New Ethernet networks using higher speed and different medium could be bridged with legacy ethernets, preserving historical capital investment



What was so special about Ethernet?

It was fast!

It was good!

It was open!

It was simple to build!

It operated itself!

It was really cheap!



What was so special about Ethernet?

It was fast!

It was *it was the defacto industry standard for*
It was *local networks within a few years!*

It was simple to build!

It operated itself!

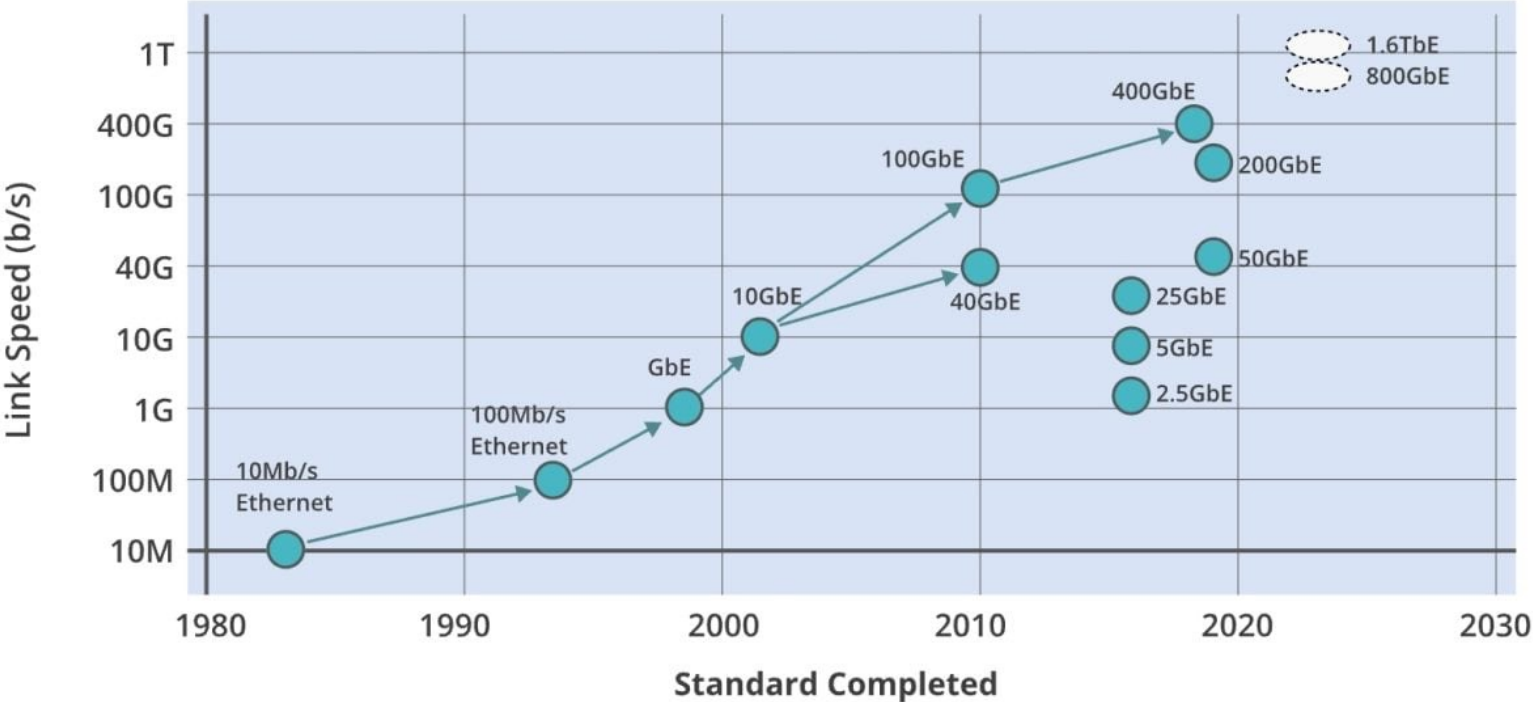
It was really cheap!

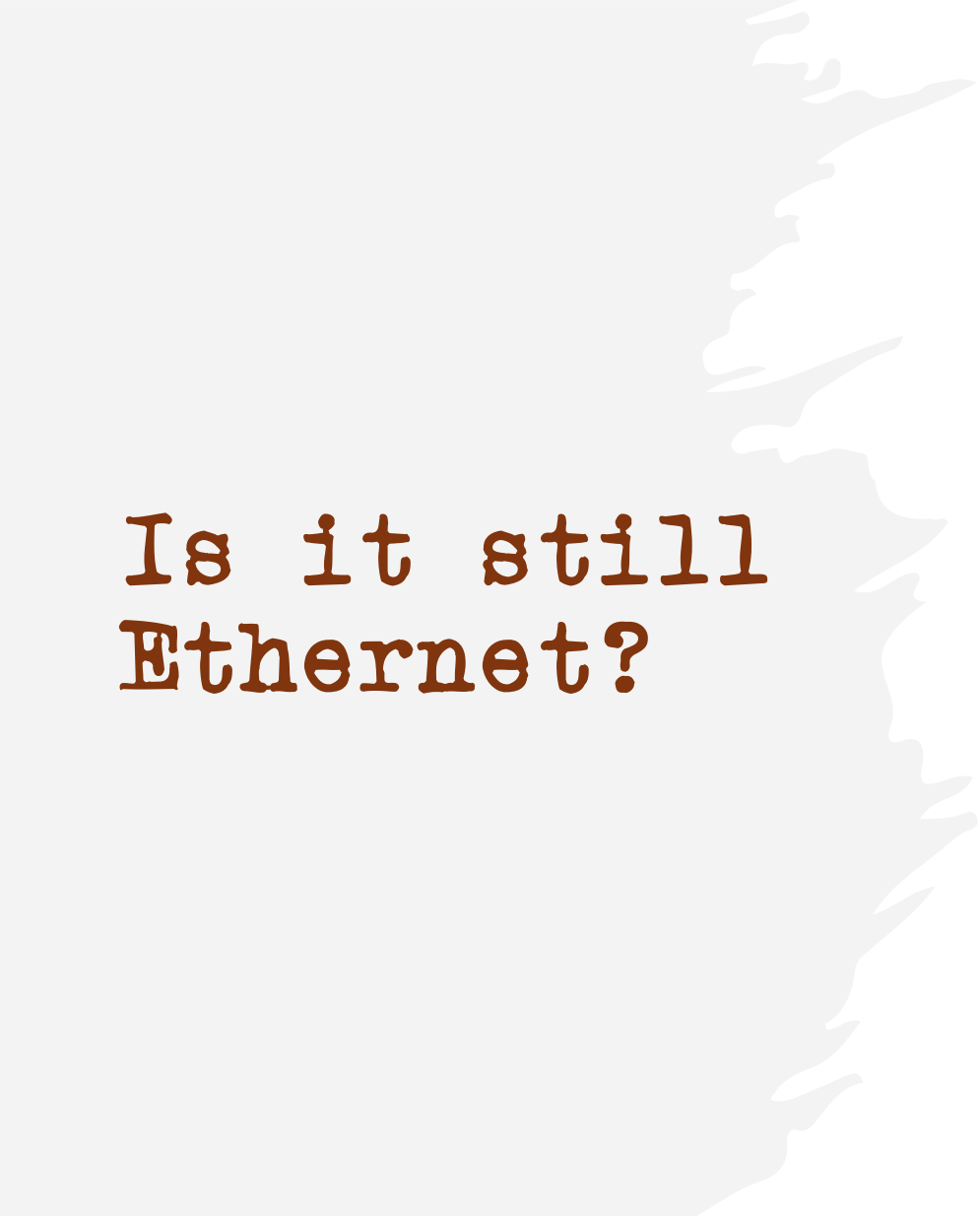
It was really VERY cheap!



Ethernet Evolution

ETHERNET SPEEDS

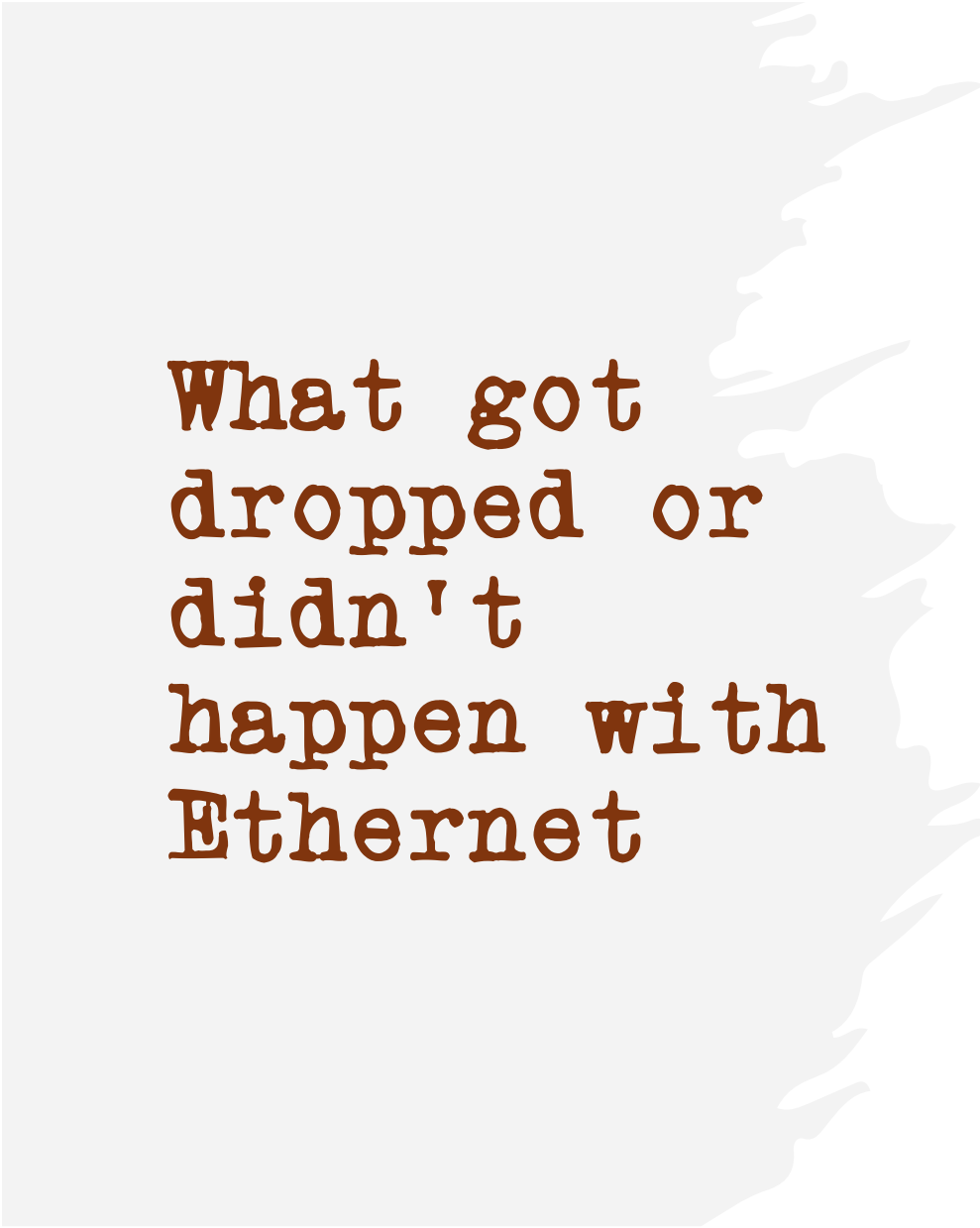




Is it still Ethernet?

- The frame format for packets has remained constant
- The MAC address registry is still used
- The packet size range has remained constant

But...



What got dropped or didn't happen with Ethernet

- No more common bus wire
- No more passive cable system
- No Self-clocking preamble for every packet
- No more CSMA/CD common bus contention system
- We just can't agree on Jumbograms
- Never really figured out robust real time and related QoS prioritization
- We needed a network protocol to help us scale – massive bridged networks were operationally unstable

Local vs Wide Area networking

- Back then we used X.25 for wide area networking and Ethernet for local networking. What a clumsy distinction!
- What we wanted was a universal adaptation layer that would allow an application to operate over any context, local or wide area
- We wanted flexibility:
 - to allow new media systems to be integrated into the network model
 - To allow new end-to-end transport systems to be integrated into the same networking model

The Internet Protocol met those needs



The emergence of IP

- The Internet Protocol rose to prominence at much the same time as Ethernet local networks

What was so special about IP?

It was fast!

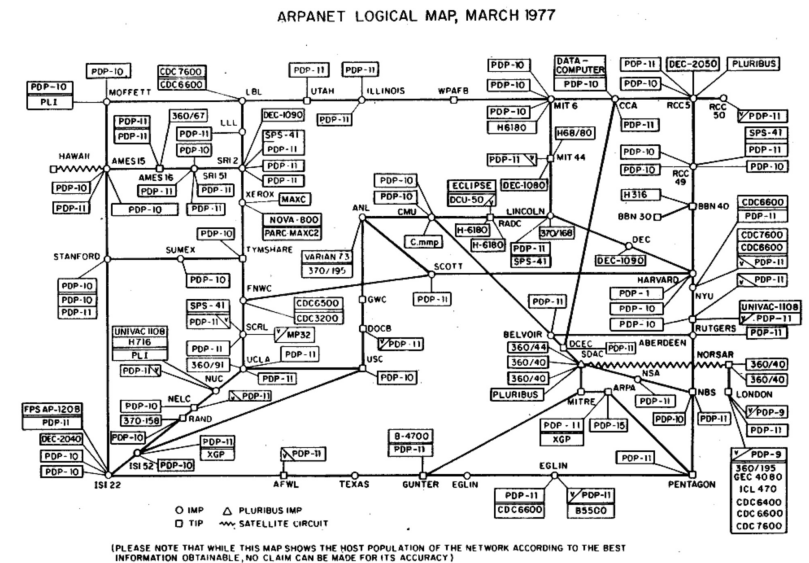
It was good!

It was open!

It was simple to build!

It (sort of) operated itself!

It was really cheap!



What was so special about IP?

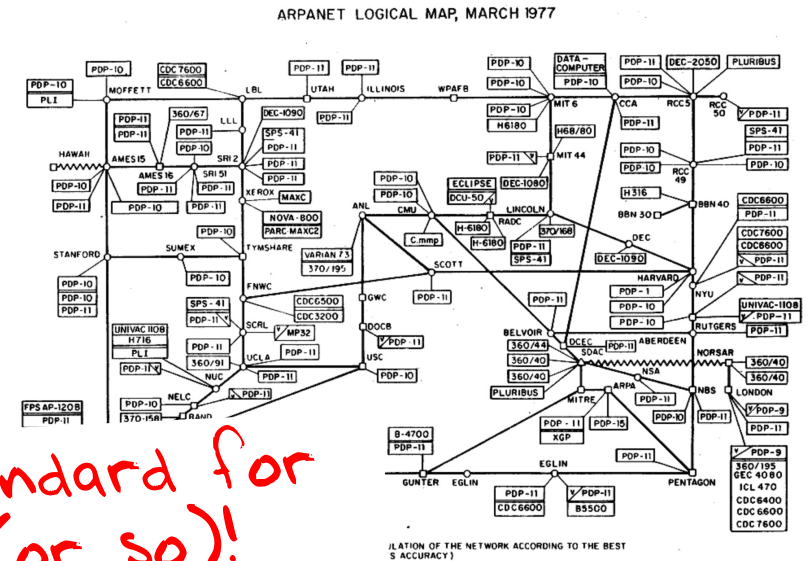
It was fast!

It was the defacto industry standard for data networks within a decade (or so)!

It was simple to build!

It (sort of) operated itself!

It was really cheap!



Edge vs Middle

- The IP networking architecture could be seen as “large scale Ethernet”
 - Strip functionality out of the network’s control plane and reduce the network to its basic function of datagram forwarding
 - Push all the control functions over to the connected devices
- Virtual circuits as a network state vs IP datagram core and end-to-end edge functionality
- “stateful” vs “stateless” networks



Scaling

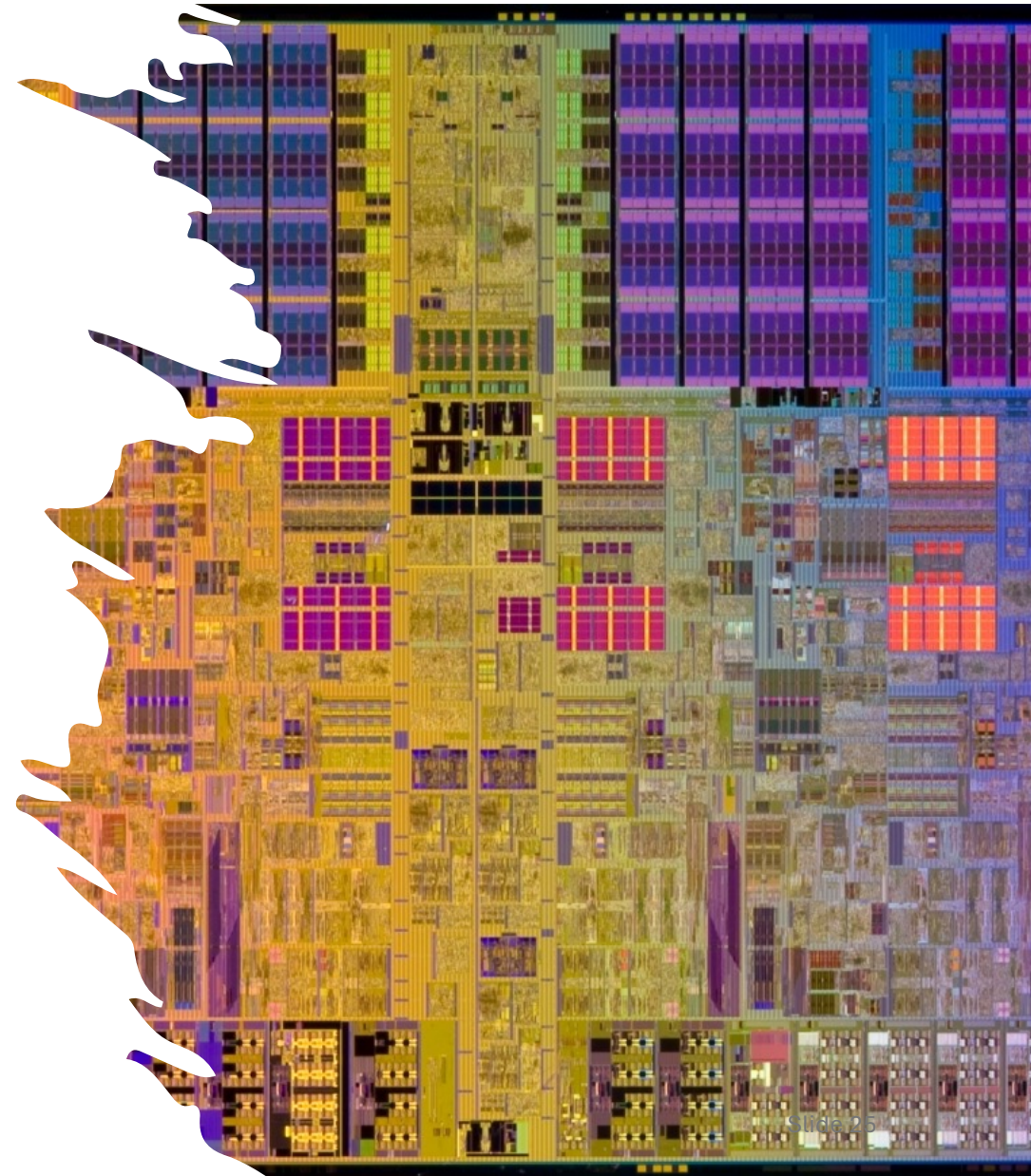
- All this is about “how to grow”
- How can you construct a single network architecture that can scale a billion-fold or more?
 - Push complexity to the edge as much as possible
 - Keep the core as simple as you can



Behind scaling pressure is...

Moore's Law

- The edge-based model of networking has benefitted greatly from the march of progress of Moore's Law:
 - The unit cost of processing and storage plummets under the pressure of Moore's Law in chip fabrication capabilities
 - The unit cost of communications as similarly plummeted because of massive improvements in DSP capabilities



The Result

- Shift in the mode of operation of communications systems from **scarcity rationing** to **overwhelming abundance**
- A shift in the mode of service delivery from “just in time” retrieval to “just in case” pre-provisioning through the rise in Content Data Networks
- Which has caused shared IP service delivery networks to shrink to **last mile switched access networks**



Today's Picture

- Hop-by-hop forwarding address-based networking is being replaced by name-based service delivery systems
 - We are living the reality of name-based networking
- The Internet has shrunk into a collection of last mile feeder systems that connect edge devices to services that are preloaded into local data centres
- There is no “core” of today’s Internet – just a collection of private feeder systems that feed collections of CDN at the edge
- Is there even a “network” left in today’s Internet?
- Does routing even matter any more in what’s left of the public Internet?



Exactly where are we?

- We started this journey building a telephone network for computers to communicate between each other
- We leveraged Moore's Law to push the technology burden to the edge and reduce the dependence on shared common public networks
- Now one-way content distribution lies at the core of today's Internet
- This content distribution role is an enterprise service framework rather than a public carriage service
- The internal parts of the carriage network are now being privatized and removed from most forms of public regulatory oversight



Tomorrow

- What if we can continue to improve the cost performance of silicon systems for the next 50 years?
- What new models are enabled in the nature of digital services by such a set of changes?



What about AI and AI Data Centres?

Specialized data centers for AI token processing don't make a huge amount of sense if the edge just keeps on getting faster and cheaper

Even specialized storage centres don't make sense if storage keeps on getting smaller and cheaper

If Moore's Law can be sustained over the coming decades, then 2076 will bear absolutely no resemblance today -- in exactly the same way as today's digital technology landscape has almost nothing in common with 1976!

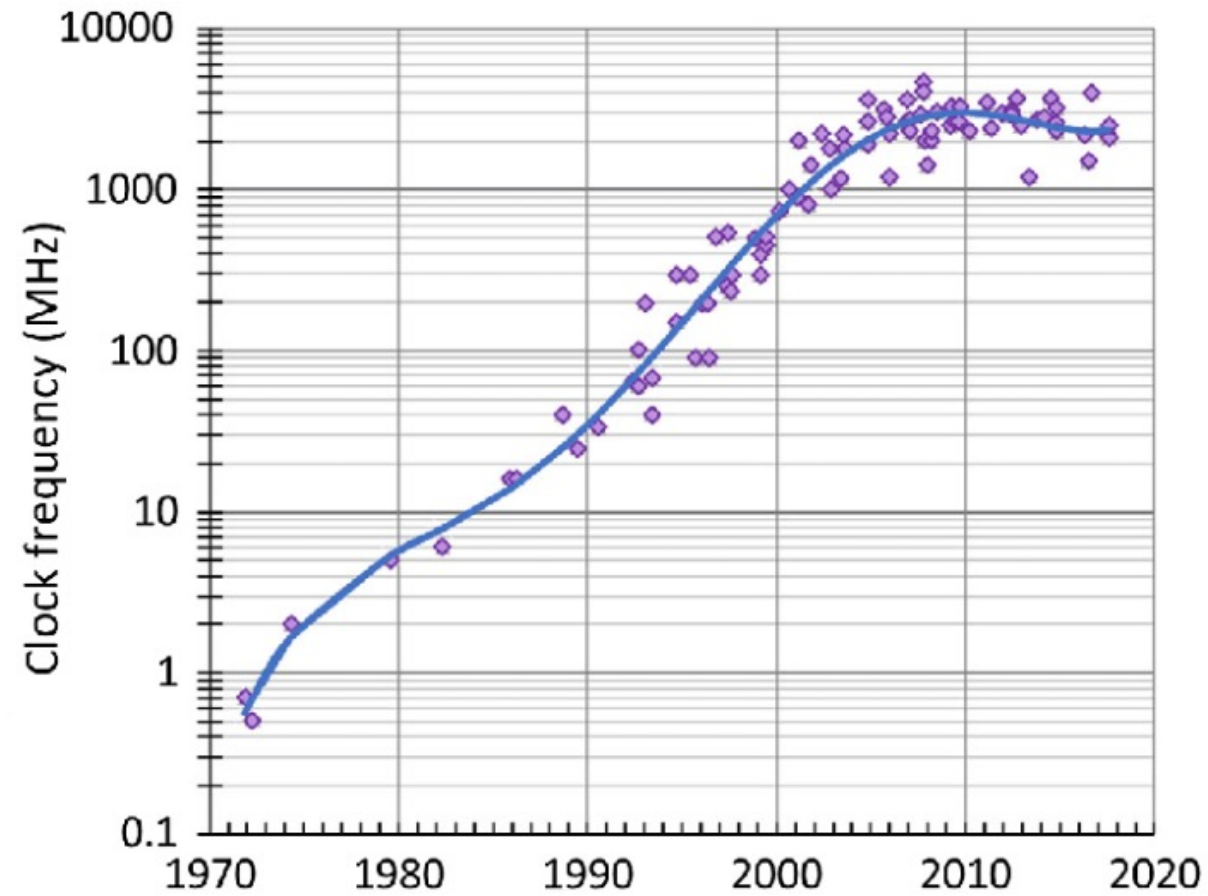
The Alternate Tomorrow ...

- What if we can't continue to improve the cost performance of silicon systems for the next 50 years?



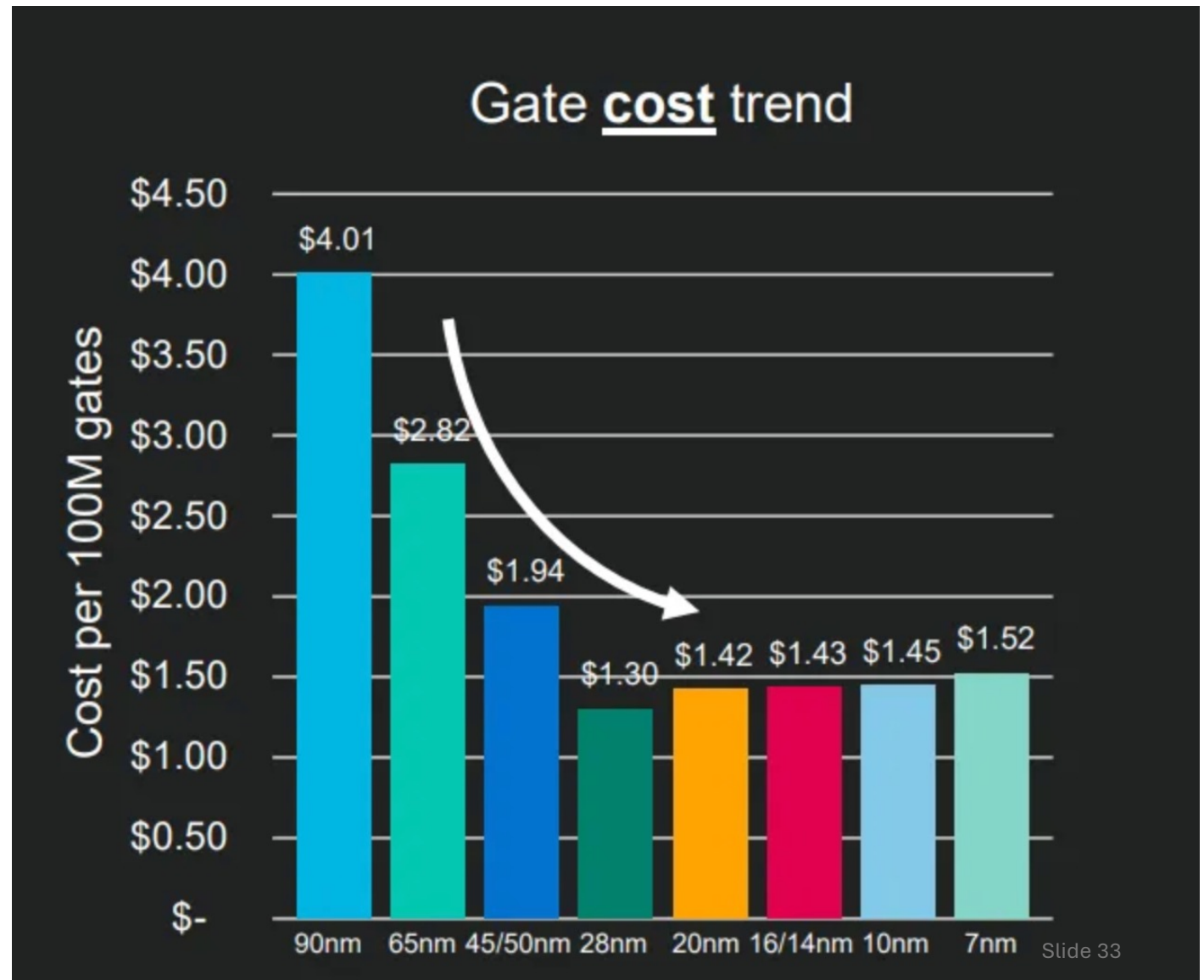
Faster?

- Nope!



Cheaper?

- Nope!



The Alternate Tomorrow

- What if we can't continue to improve the cost performance of silicon systems for the next 50 years?
- What if faster and more capable compute engines requires specialized data centers with extreme demands of concentrated power and performance ?
- What if advances in computing head to the world of quantum computing and are not amenable to miniaturization and cost reduction?
- What if Moore's Law runs out?



The Heart of AI

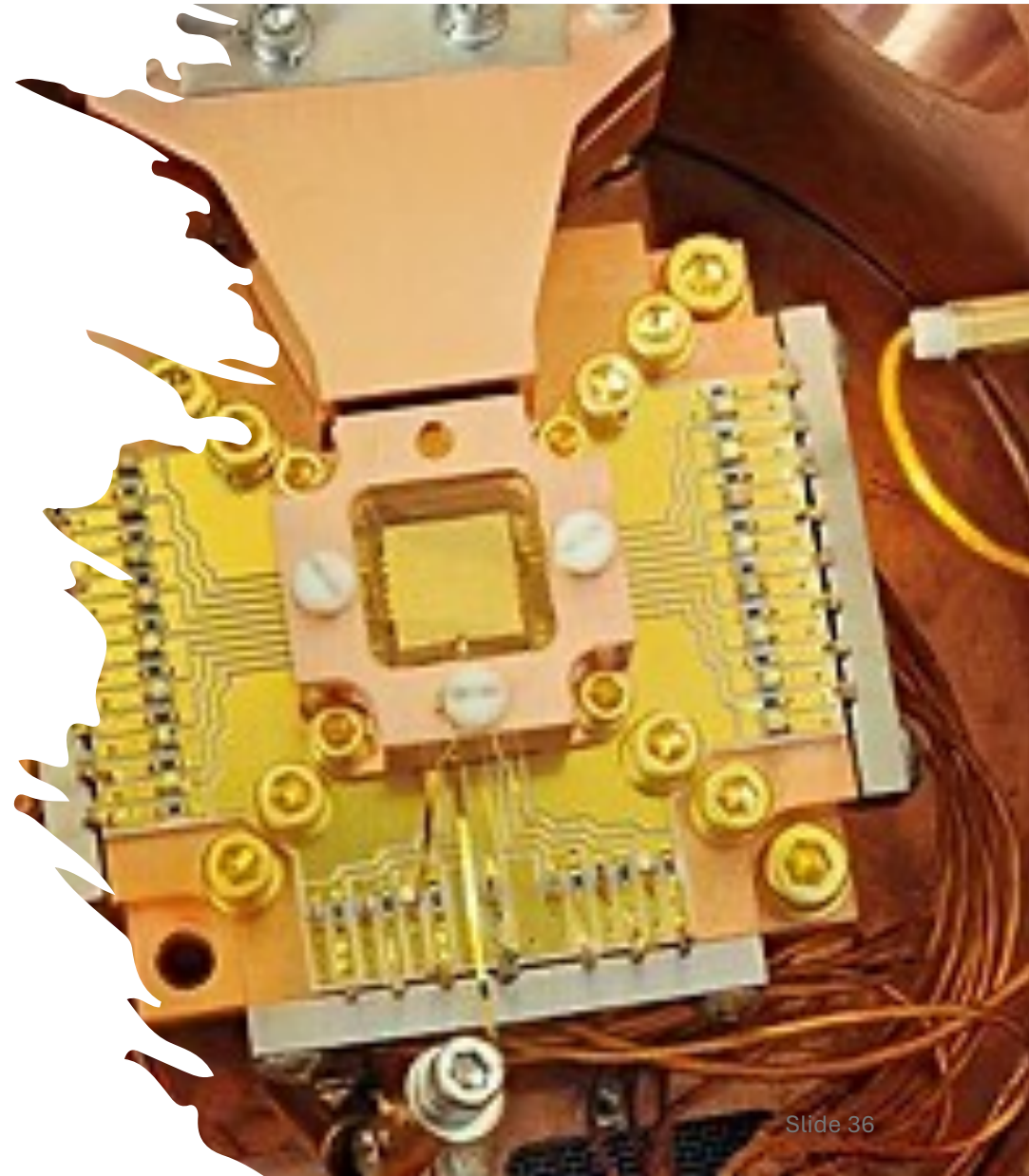
- We are in the middle of a massive trillion-dollar build of a very small set of highly specialized large-scale environments that support large scale neural network computing platforms
- This is the core of new form of a **centralized** compute and network service model
- The investors funding these highly specialized (and expensive) technology temples are actually betting that Moore's law is finished and future needs cannot be met by more of smaller, faster, cheaper devices



Quantum Computing

- So far, we've seen quantum computers being constructed as high cost highly specialized devices operating in extreme environments in order to preserve quantum properties
- It's an entirely open question as to how this will scale?
- And at what cost?

Hint: So far scaling of quantum is not looking good!



The pendulum of compute

- Expensive technology tends to use centralized shared access models to defray these costs against a large pool of users
 - Telephony
 - Mainframes
- Abundant inexpensive technology tends to use highly duplicated and dispersed deployment models that rely on networks to support service coherence
 - Mobiles
 - IoT





What's the problem?

- In the quest for more complex compute power at scale we are seeing the return of proprietary solutions in applications and service delivery platforms that expose as little as possible to the underlying network platform
- Its all “Black Box” engineering in white wrappers!

What's the problem?

Perhaps the problem is a looming demise of open technologies and open technology standards

- All content distributions systems now use strong encryption to hide the application from the network and the platform
- Applications are no longer constructed on a platform of common libraries provided by the platform
- Applications are now paranoid and avoid exposing their behaviour wherever and whenever possible
- Applications are increasingly reluctant to use standard open technologies in standard and open ways



What's the problem?

- In its place we are seeing a resurgence of various closed technologies that create a set of datacentre-to-application bindings that are impervious to all third parties
- These closed architectures make minimal assumptions about a common network protocol, a common name space or even a common name space
- What happens to the efforts that support open technologies, open standards and open networking in such a world?



Where does all this head?

Where does all this head?

i just don't know!

Where does all this head?

i just don't know!

But i'm not convinced that it's going to be good for us - as a society or as humans!

Thanks!