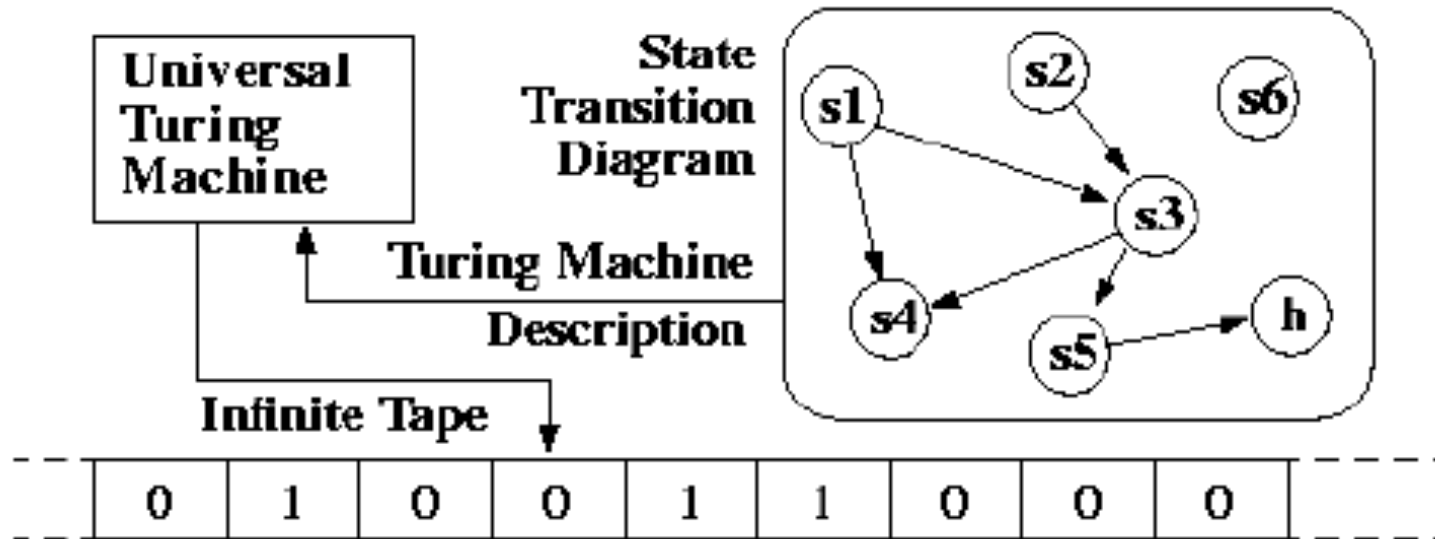




A brief history supercomputing

Lecture 1

The ideal computer

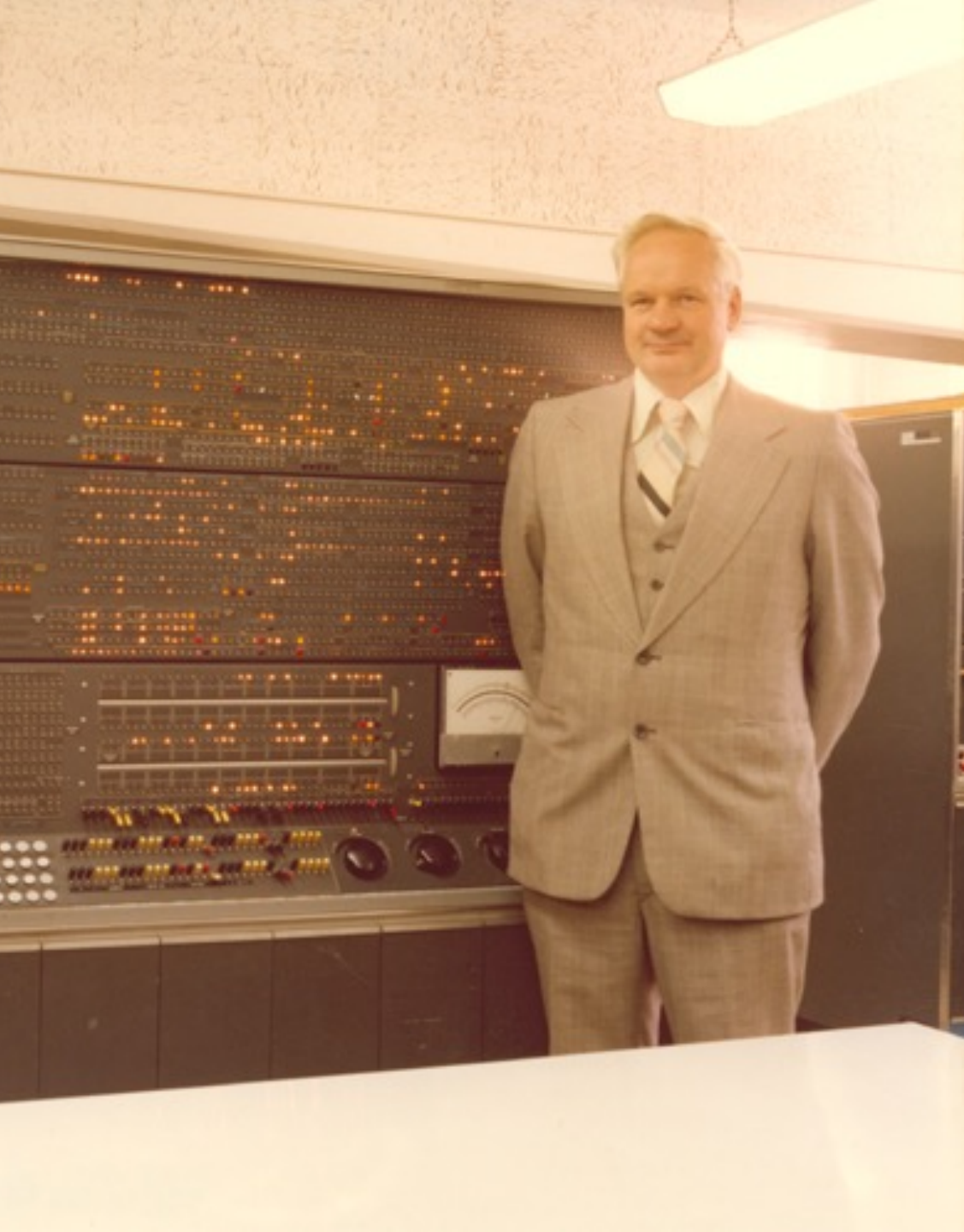


- Has an infinitely fast clock
- Executes one program instruction at a time
- Instructions and data are in an infinitely large storage device
- Access time to storage is always the same

This is the model for most runtime complexity analysis

Supercomputing vs Distributed Computing

- Supercomputers: These are large, expensive systems—usually housed in a single room—in which multiple processors are connected by a fast local network.
- Distributed Computing: These are systems in which processors are not necessarily located in close proximity to one another—and can even be housed on different continents—but which are connected via the Internet or other networks. (Think cloud computing)



Supercomputer

- Supercomputers are the highest performing computers of the day
- They are intended to solve “Grand Challenge” problems.
- In 1956 the IBM Stretch was built to be 100x faster than standard computers of the day. It cost \$76,190,000 (today’s dollars)
- Clock Speed: 1Mhz (10^6 cycles per second)
- **A clock cycle is when the CPUs transistors update. It’s a fundamental limit on computational speed.**
- The definition of a supercomputer is always relative to all the desktops and laptops of the time.
- Ran at Los Alamos Labs simulating nuclear reactions

FORTRAN

- John Backus at IBM directed the team that wrote FORTRAN (the formula translation language) in 1956.
- FORTRAN was the first widely used high-level language.
- Before FORTRAN programmers would write machine language programs directly.
- There were other high level languages at the time. But the machine language programs they produced were often 10x slower than human machine language.
- FORTRAN made sure their machine language was just as fast as machine code written by programmers.
- FORTRAN is intended for scientists (not programmers) to write fast code
- C on the other hand is intended for programmers not domain scientists

FORTRAN

- By 2010 software applications had been developed for nearly every scientific and engineering discipline.
- Some of the biggest drivers are:
 - Nuclear simulations (bombs)
 - Climate modelling (global warming)
 - Finance (automated stock trading)
 - Geology (oil)
 - Protein structure (drug discovery)
 - Molecular structure (designing new materials)
 - Artificial Intelligence (understanding you)
 - Computational Fluid Dynamics (aircraft, spacecraft, cars, blast furnaces)
 - Bioinformatics (genetics)

Government Funding



- Supercomputing has been driven by government funding through the national labs and national supercomputing centers.
- Government agencies set the grand challenges that drive innovation in supercomputing
- National labs work closely with supercomputing vendors from the design stage.
- The US Advanced Research Project Agency (ARPA) was tasked with creating revolutionary new technologies after Sputnik (to win the technology cold war)
- ARPA established: 1) supercomputing centers, 2) computer science departments in universities, and 3) developed a national computer network.

ASCI Red (1996) – Sandia National Labs

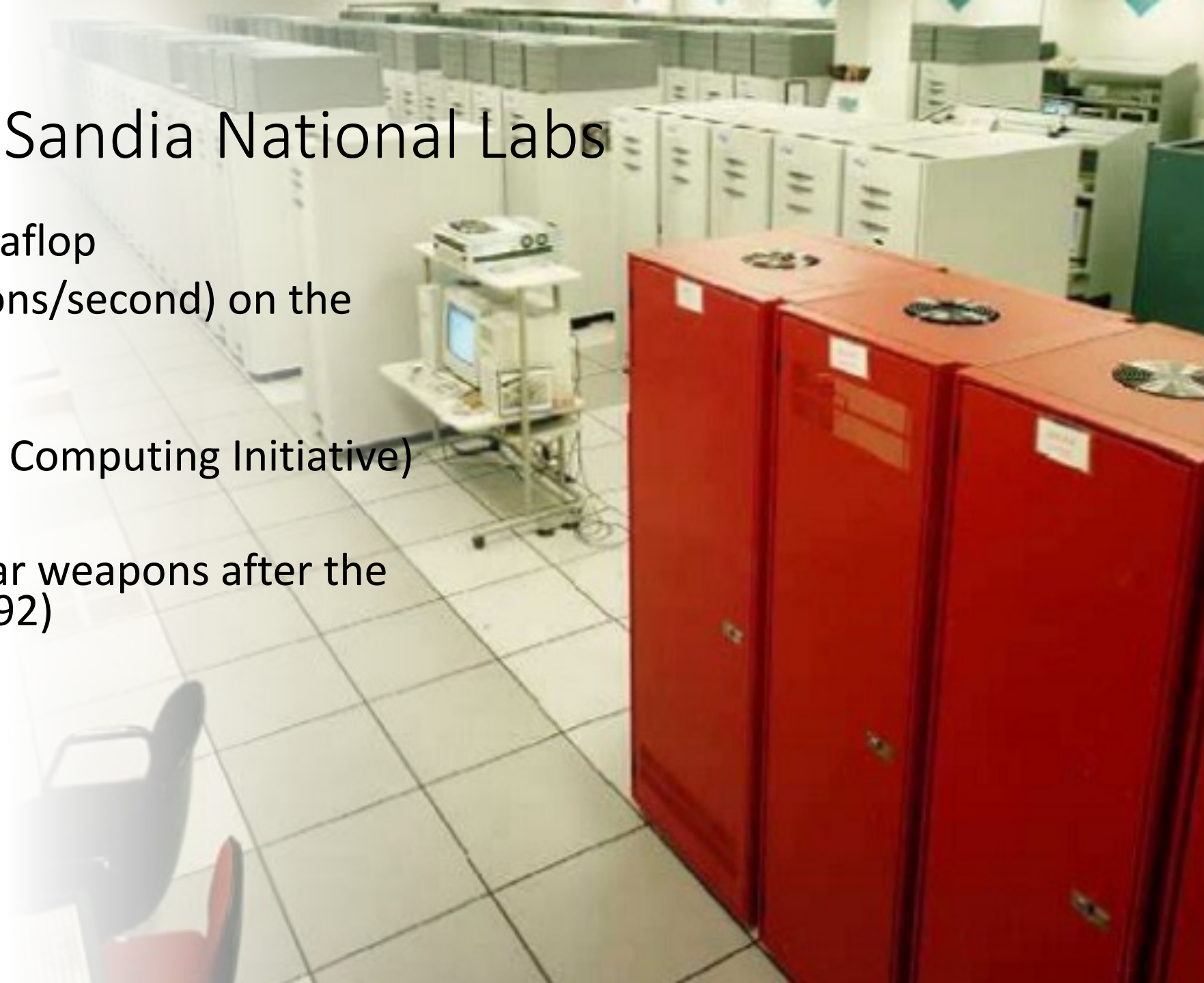
First computer to pass 1 Teraflop
(10^{12} floating point operations/second) on the
LINPACK benchmark.

(ASCI: Accelerated Strategic Computing Initiative)

Designed to simulate nuclear weapons after the
Nuclear Test Ban Treaty (1992)

RAM: 1212 gigabytes

850kw



Industry

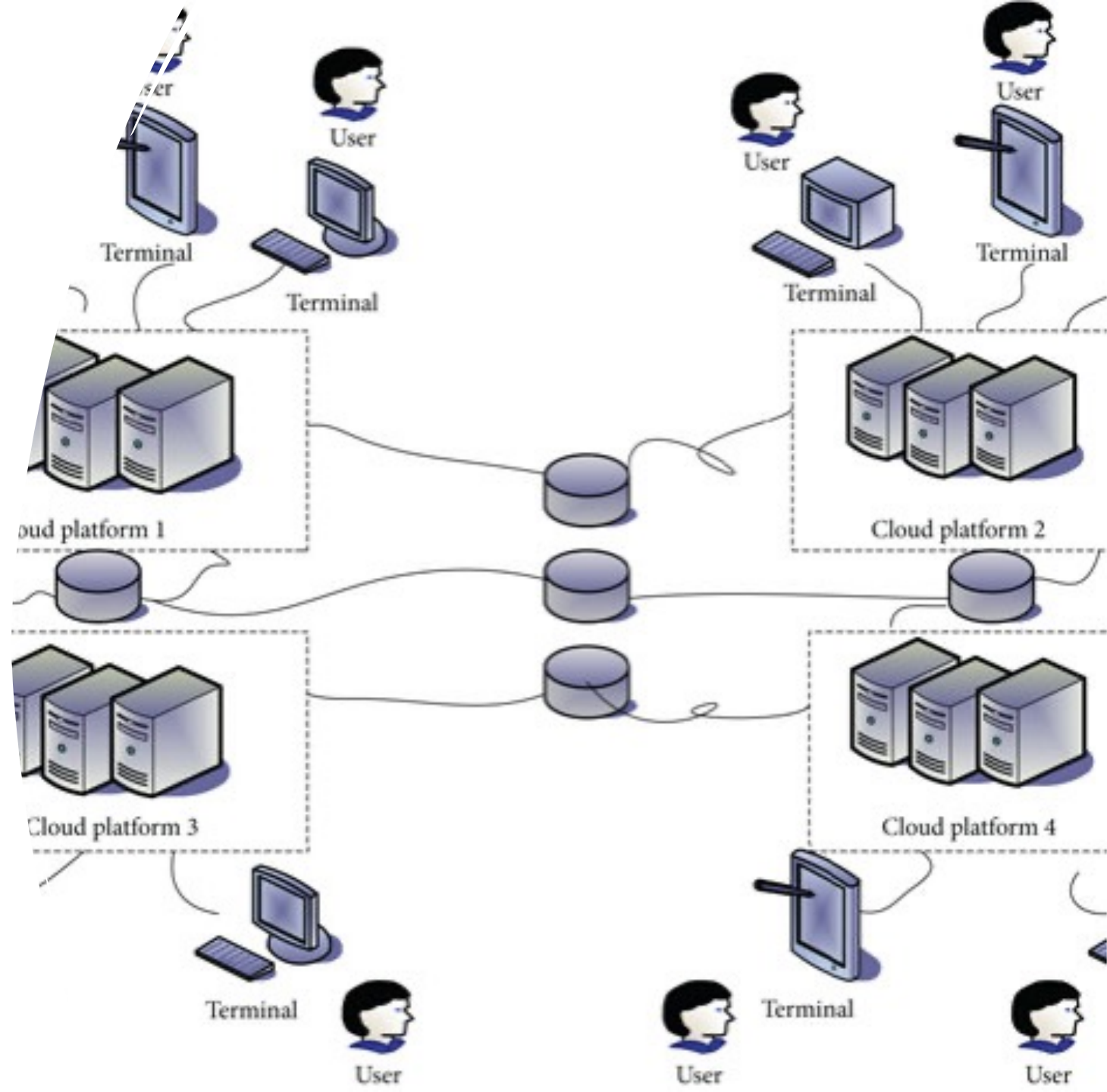
- The non-government sources of supercomputing demand are

Examples

- 1) Pharmaceuticals ()
- 2) Oil and Gas (computational fluid dynamics of oil given huge amounts of seismic data)
- 3) Computational Finance (high frequency trading) – 60-75% of trading

Distributed Computing Example

- When NASA's Jet Propulsion Laboratory needed to analyse images taken by the Mars Rovers, then the Amazon cloud was a good fit. Slow connections between monocomputers don't matter to much because the image processing is largely independent.
- Each computer processes one image or set of images.



Supercomputers

- Weather prediction is a classic example.
- To predict the weather even a few hours in the future requires processing enormous amounts of data gathered from millions of locations each recording temperature, wind, humidity, solar radiation, etc.
- The computation has to be spread over many computers but very fast communications are needed so they can access the data they need.
- Weather models run continually so they need new data all the time.



Evolution of Supercomputers

- Increasing processor speed (1960 - now)
 - Processing vectors (1970s - now)
 - Additional processors for single memory systems (monocomputer)
 - Fast networks for distributed computation over many monocomputers (compute clusters or multicomputers) (1990s - now)
 - GPUs (2000s - now)
-
- Supercomputing has evolved from a primarily hardware challenge (1960s-1990s – Cray Era) to creating software that runs on compute clusters (1990s-today – Multicomputer Era)
 - (GPU accelerators recapitulate this pattern but over the past 10 years)

Discussion (5 mins)

Nuclear simulations involve modeling particles in regions of space and how they interact with their neighbours.

Are nuclear simulations better run on a distributed computing system or a supercomputer and why?



Four Eras. - Vector Operations

- The first thirty years of supercomputing was dominated by companies like Cray Research (Seymore Cray), Control Data Corp (CDC) and International Business Machines (IBM) (American companies).
- This included the beginning of the Vector operation “era”
- Vector operations increase CPU speed by applying the same instruction to more than one variable during the clock cycle.

CPU Vector Operations (Advanced Vector Extensions [AVX] and Single Instruction Multiple Data Streaming Extensions [SSE])

```
ssh vanilla@hopper.alliance.unm.edu
```

```
lscpu
```

```
lscpu | grep avx
```

```
ssh xena
```

```
lscpu | grep avx
```

```
ssh wheeler
```

```
lscpu | grep avx
```

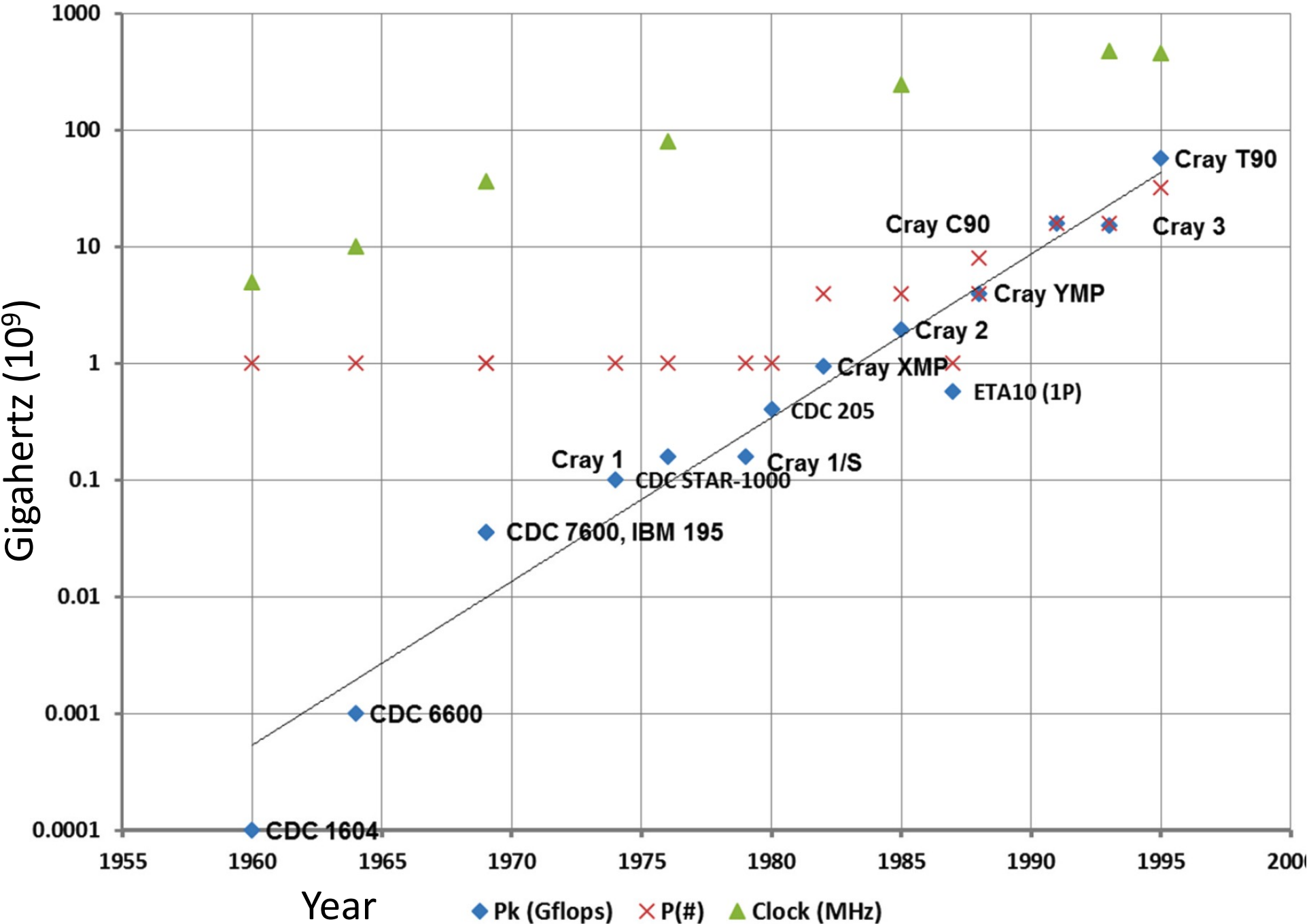
```
lscpu | grep sse
```

Cray, CDC, and IBM
Supercomputer clock
Speeds

1995 was the last
serious Cray
supercomputer.

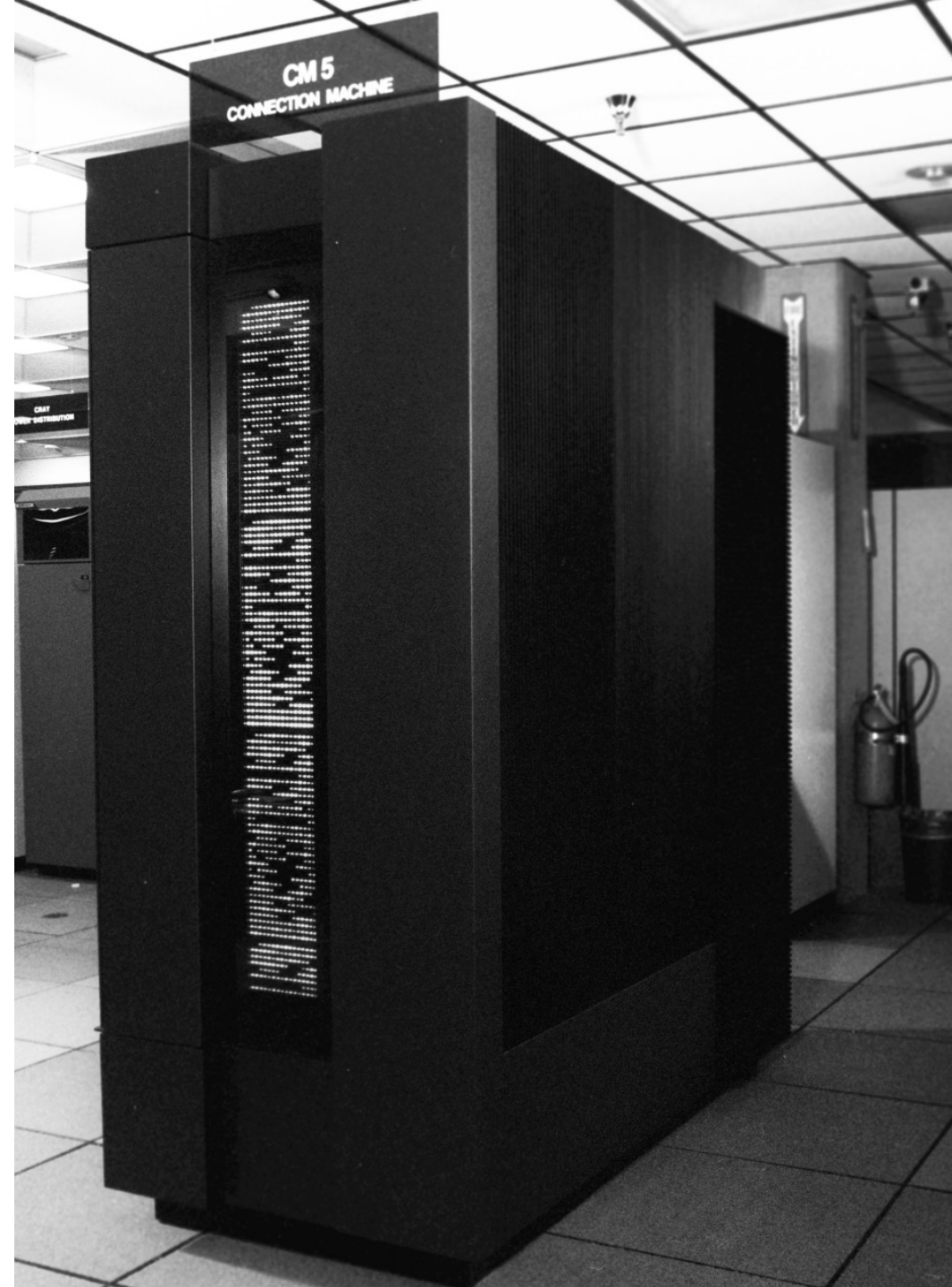
Era of increasing clock
speeds.

The clock speed race
stopped dominating
speed



End of the Monocomputer Era

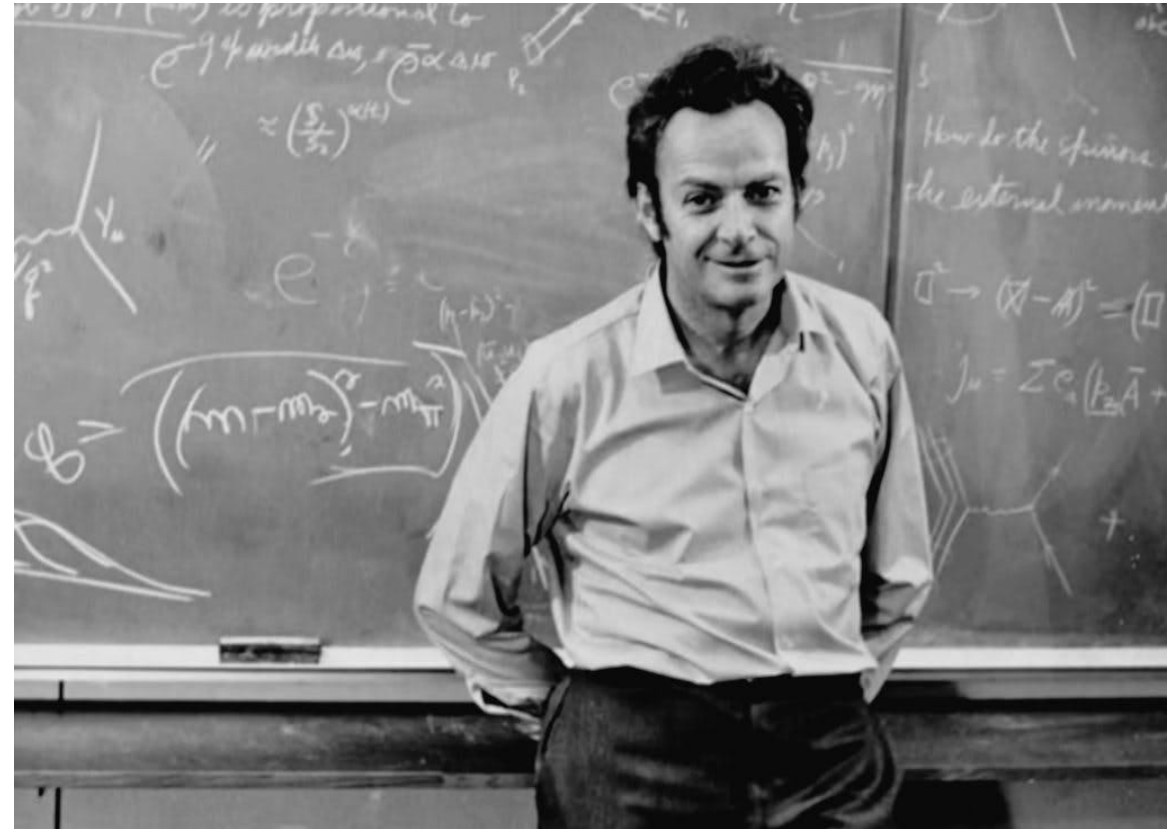
- In 1993 Thinking Machines built their 32 node multicomputer (called connection machines at the time)
- Ran on UNIX
- Achieved a LINPACK benchmark of 60 gigaflops.
- The best Cray monocomputer that year had 16 CPUs and a LINPACK score of 16 gigaflops.
- Seymore Cray: “If you were plowing a field, which would you rather use? Two strong oxen or 1024 chickens?”



Thinking Machines Corp

- Founded by MIT student Danny Hillis
- That's good. Now I don't have to explain to people that I work with a bunch of loonies. I can just tell them the name of the company."
- Feynman was a quantum physicist interested in computation.
- One of the first examples of parallel computations was Feynman's reorganization of the IBM computing units during the Manhattan project so they could process calculations in parallel.

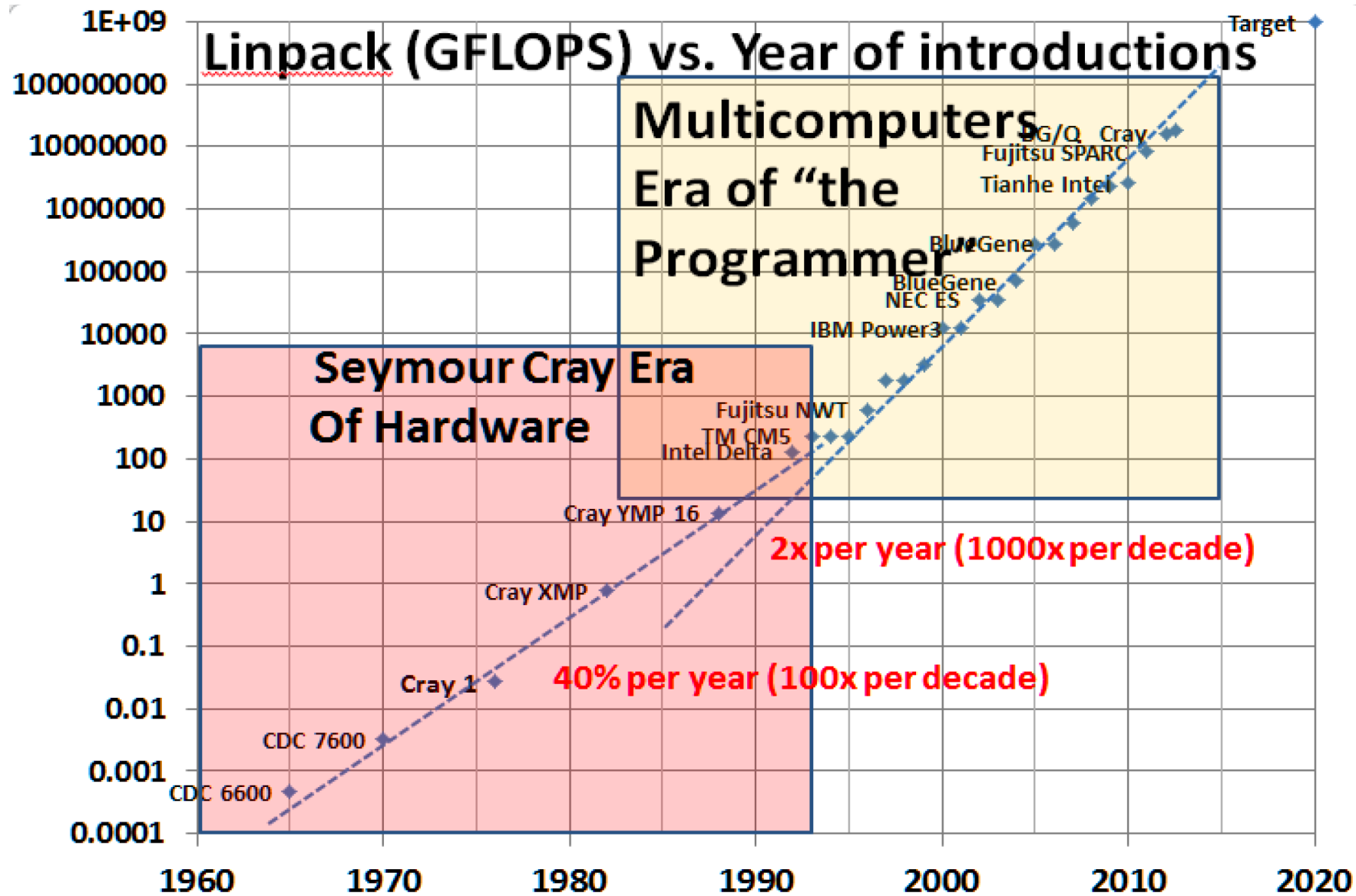
Connectionist Machines



Richard Feynman, Caltech

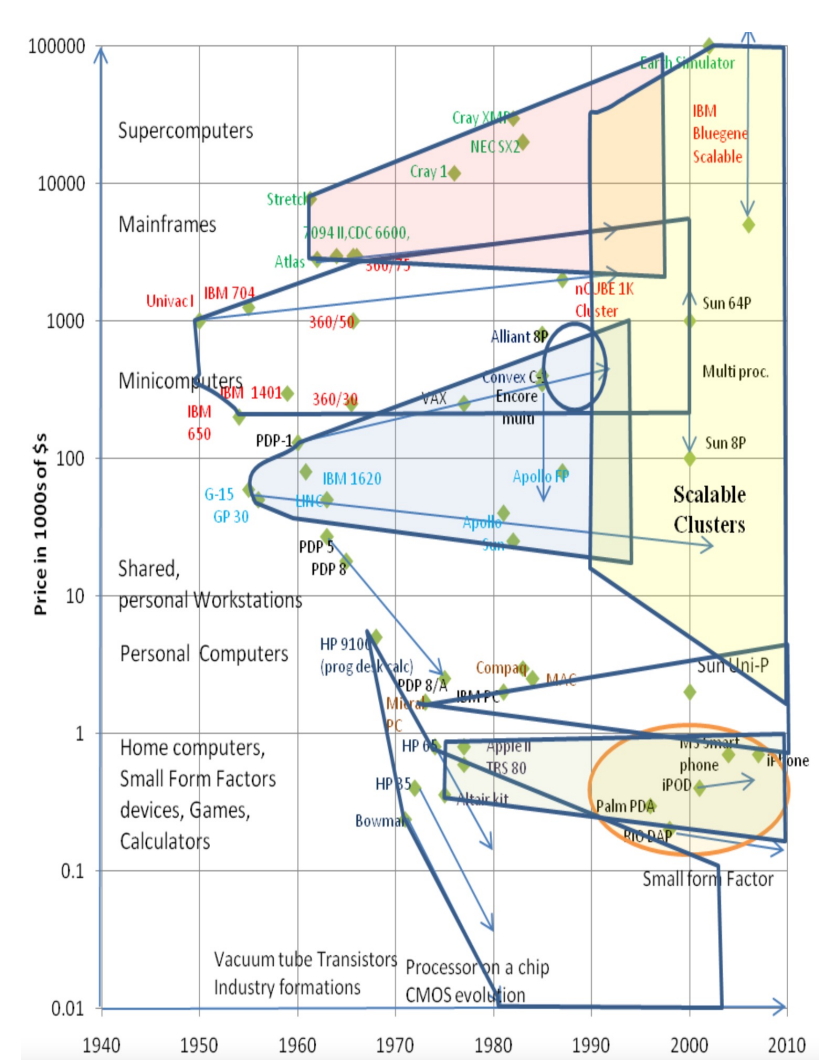
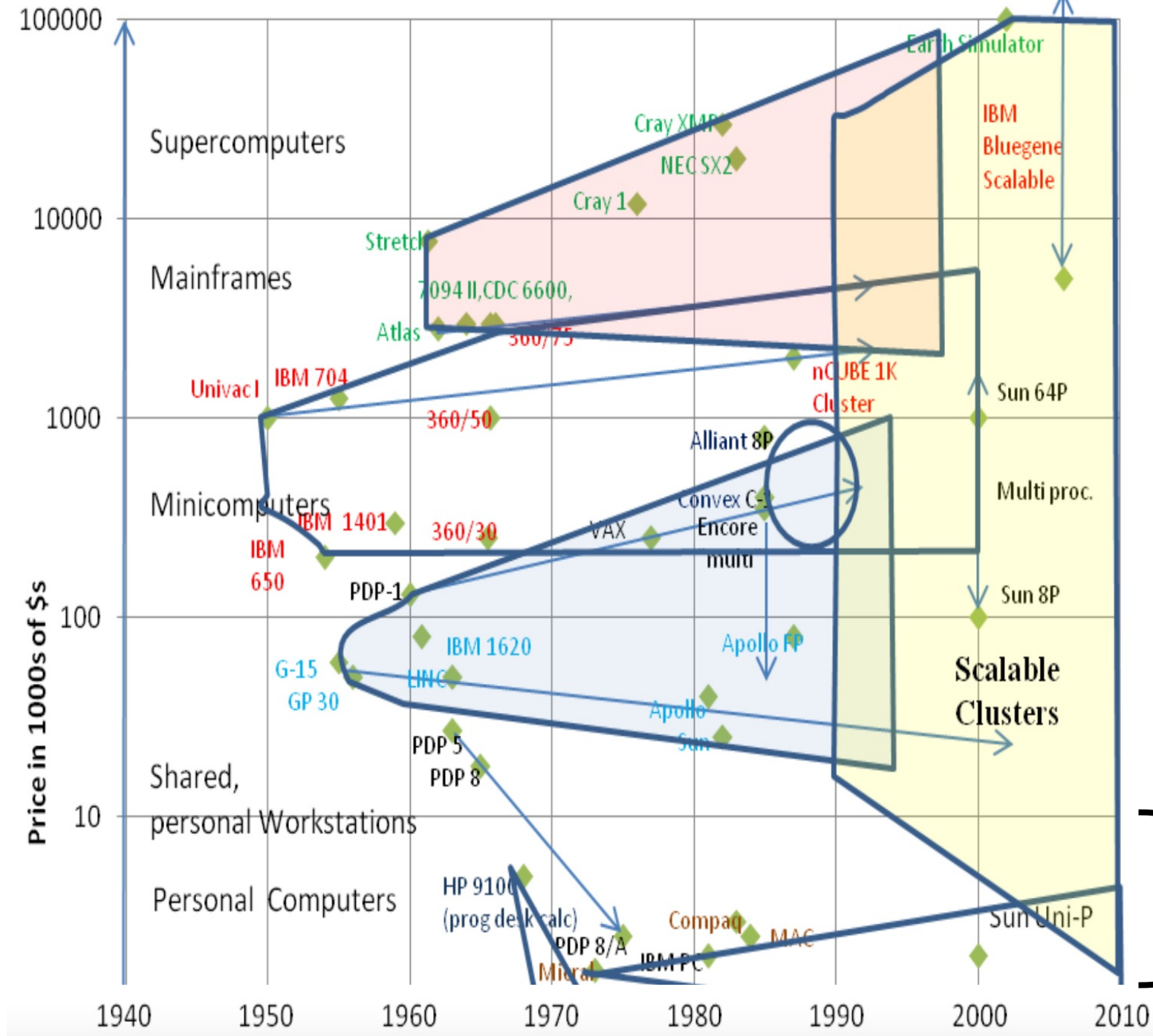
Clock Speed vs Cluster Size

- Between 1983 (5Mhz, 64 node, Caltech Cosmic Cube) and 2013 the number of nodes in a single cluster reach 3 million.
- That's a 50,000 fold increase over 30 years.
- Clock speed increased from from 5 Mhz to 5 Ghz over the same period. 1000x increase.



Four Eras

- The first thirty years of supercomputing was dominated by companies like Cray Research (Seymore Cray), Control Data Corp (CDC) and International Business Machines (IBM) (American companies).
- During this time Japan entered the supercomputing business with Hitachi, Fujitsu, and NEC.
- From 1995 until the present scaling up (monocomputers) became less important than scaling out (multicomputers).
- This change was partly enabled by CMOS and BIOS standards that allowed many manufacturers to create compatible computers (the PC revolution).

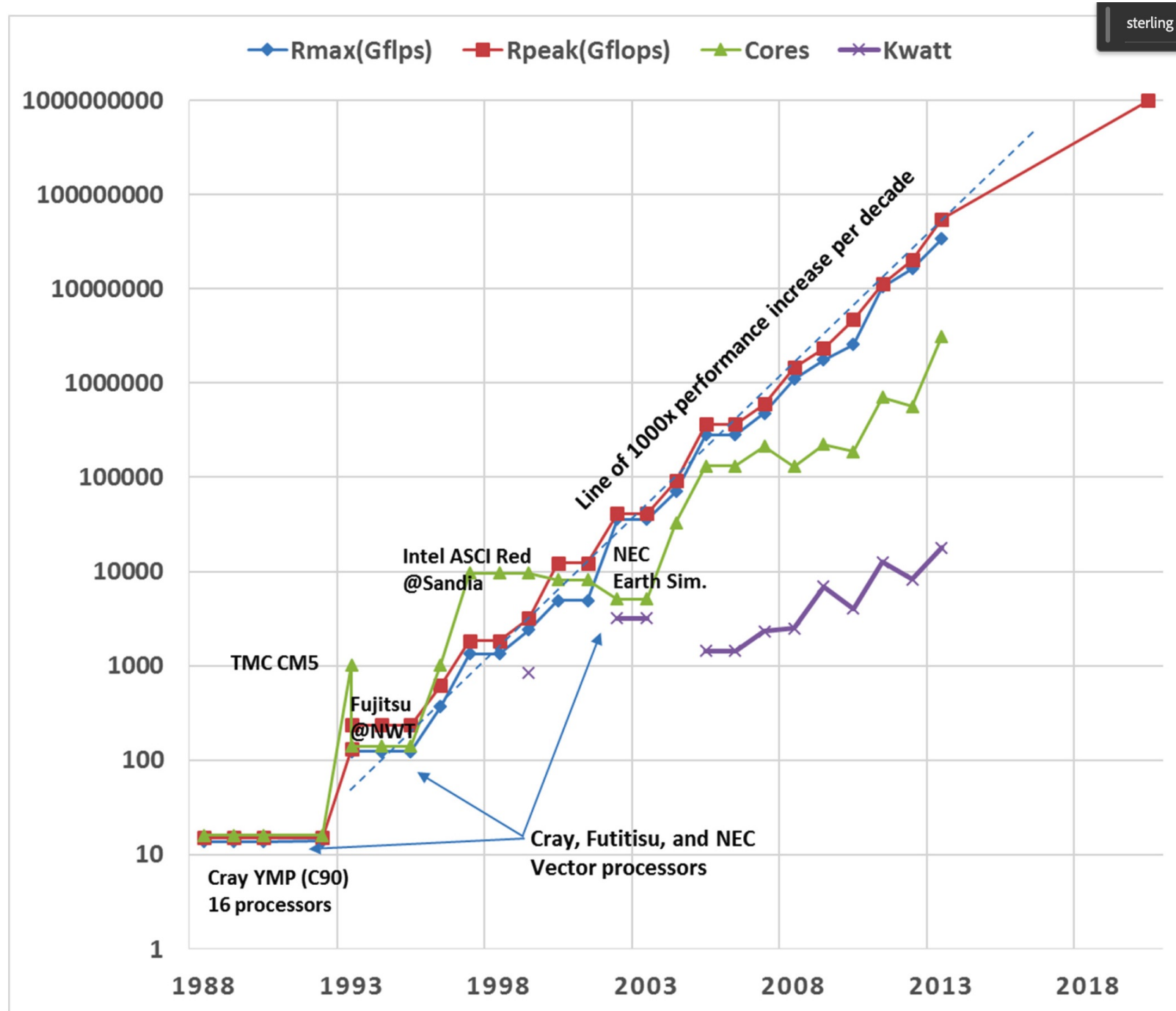


cost/performance
driven down

Gordon Bell, Bell's Law for the birth and death of computer classes: a theory of the computer's evolution

Increasing Cores drives super computing performance

High Performance LINPACK FLOPS



Beowulf Clusters

- **Beowulf clusters are a collection of PCs running Linux networked together.**
- Version 1.0 of the Message Passing Interface (MPI) was developed in 1993.
- It set a standard for FORTRAN and C programs to communicate with each other over a regular ethernet network.
- In particular, it defined how many computers to talk to many other computers and have all the messages synchronized.
- Stirling and Beckers Beowulf system packaged MPI, ran on commodity CMOS machines (PCs), and democratised supercomputing.

Beowulf Clusters

- In 1994 Donald Becker and Tom Stirling, both at NASA, built a cluster using available PCs and networking hardware.
 - 16 Intel 486DX PCs connected with 10 Mb/s Ethernet
 - Achieved 1 GFLOP/s on \$50,000 system
 - Linux had been released in 1991.
-
- Beowulf: 'Because my heart is pure, I have the strength of a thousand men.'

In 1993 NASA developed the first Beowulf cluster.

At the time Cray Computers dominated supercomputers with Monolithic machines.

Thomas Sterling and Donald Becker created the first cluster supercomputer.

Adopted by NASA Goddard Space Center.

Used Linux and PVM (Parallel Virtual Machine)

Later MPI supplanted PVM.



Goddard Space Center, 'Wiglaf'
Beowulf Cluster, 1994

In 1993 NASA
cluster.

At the time Cray
supercomputer

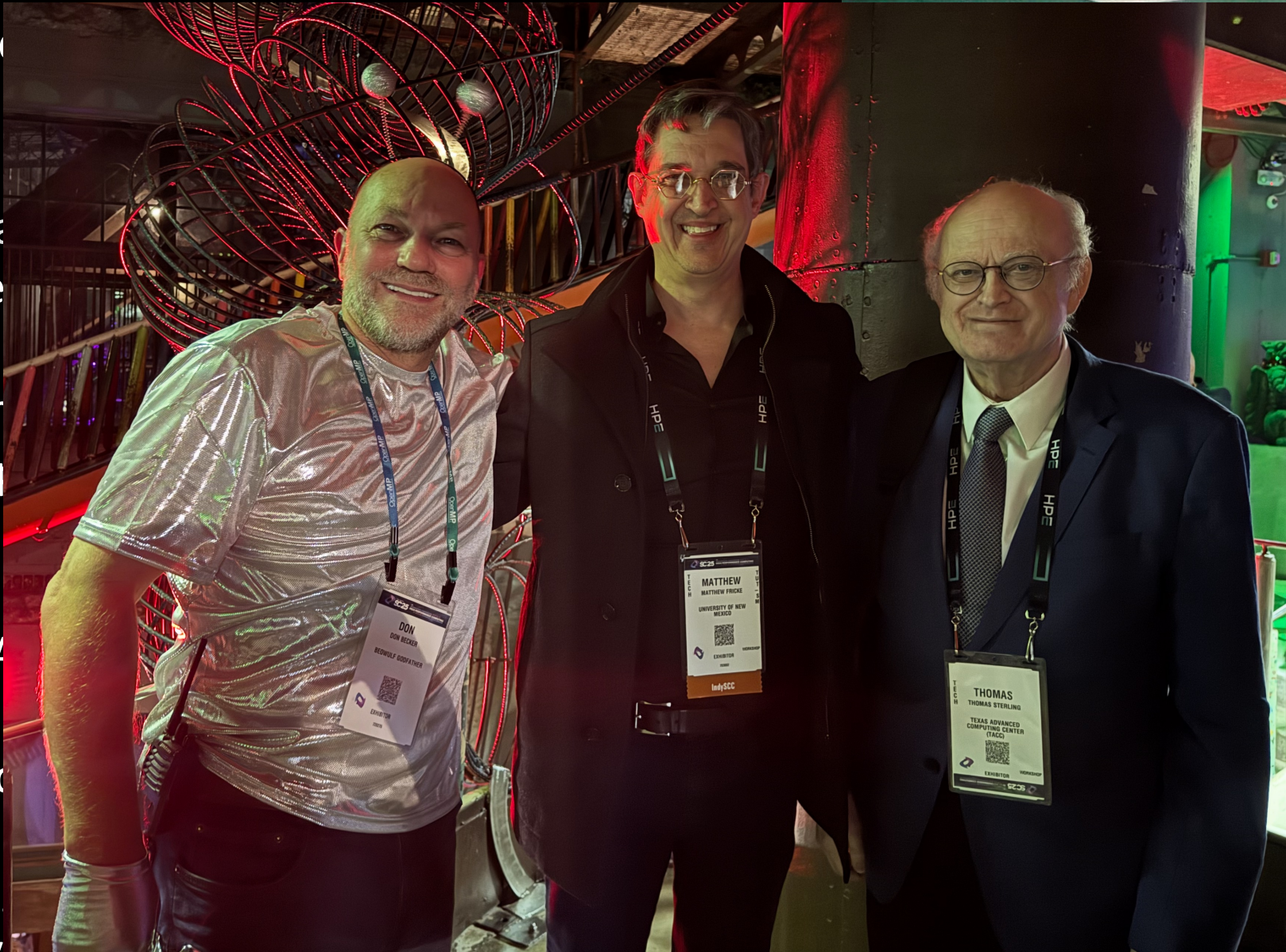
Thomas Sterling
first cluster sup

Adopted by NA

Used Linux and

Later MPI (Me
supplanted PViv.

11/18/25



2 21
Goddard Space Center, 'Wiglaf'
Beowulf Cluster, 1994

In 1993 NASA developed the first **Beowulf** cluster.

These are Linux based clusters of many computers.

At the time Cray Computers dominated supercomputers with Monolithic machines.

Seymore Cray was not impressed with Linux Clusters:

“If you were plowing a field, which would you rather use? Two strong oxen or 1024 chickens?”



Seymore Cray with Cray-1 (1976)

In 1993 NASA develop

These are Linux based

At the time Cray Comp
with Monolithic mach

Seymore Cray was not

“If you were p
would you rat
oxen or 1024



Roadrunner 1999

Result of a “bake-off” between CARC and Linux and NCSA’s Windows NT Wolfpack cluster.

Roadrunner was made the first NSF nationally available Linux cluster.



Design and Analysis of the Alliance/University of New Mexico Roadrunner Linux SMP SuperCluster

Authors: [David A. Bader](#), [Arthur B. MacCabe](#), [Jason R. Mastaler](#), [John K. McIver, III](#), [Patricia A. Kovatch](#) | [Authors Info & Claims](#)

[IWCC '99: Proceedings of the 1st IEEE Computer Society International Workshop on Cluster Computing](#) • Page 9

Published: 02 December 1999 [Publication History](#)

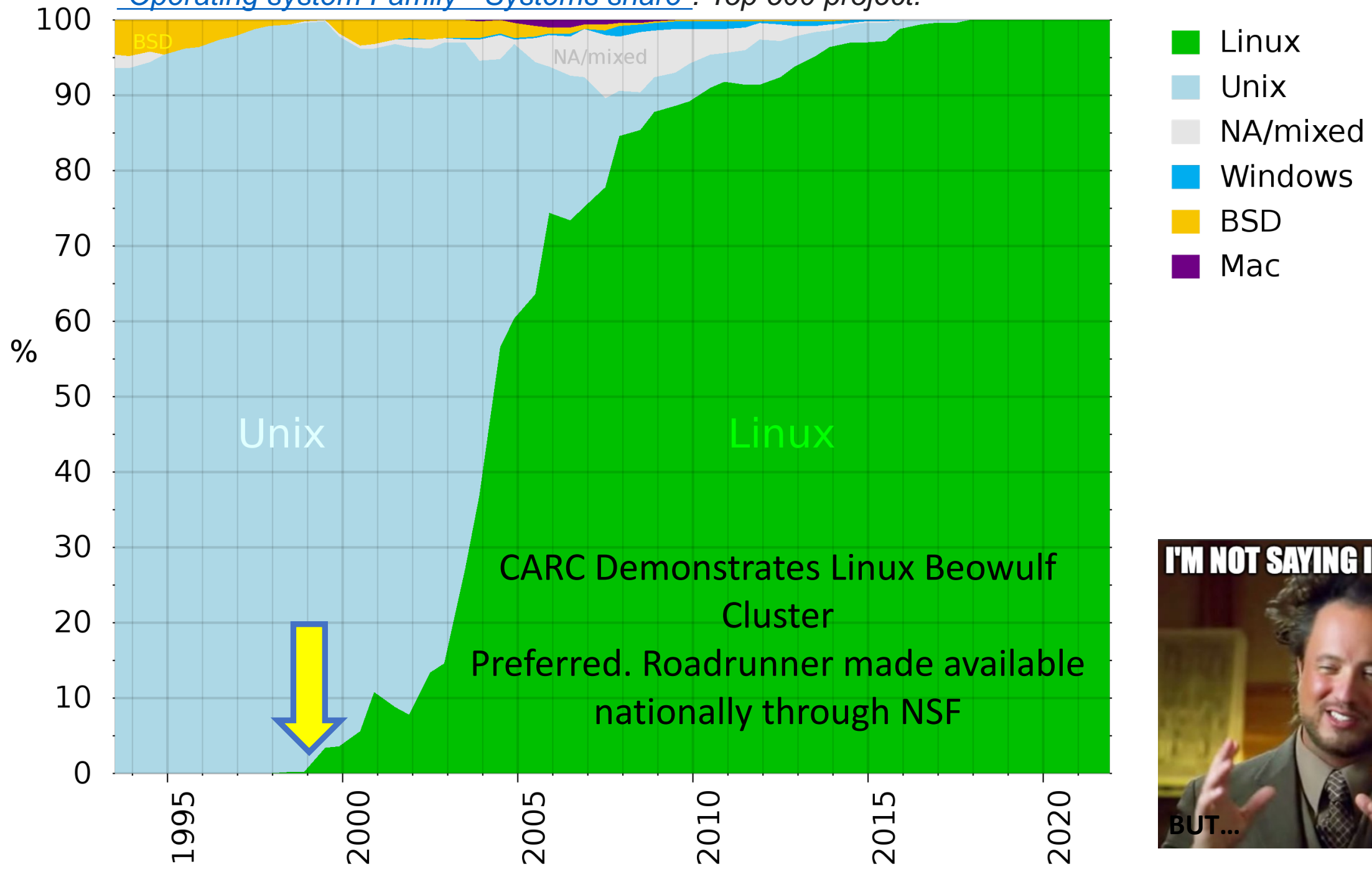
1 0

[Publisher Site](#)

Abstract

This paper will discuss high performance clustering from a series of critical topics: architectural design, system software infrastructure, and programming environment. This will be accomplished through an overview of a large scale, high performance SuperCluster (named Roadrunner) in production at The University of New Mexico (UNM) Albuquerque High Performance Computing Center (AHPCC). This SuperCluster, sponsored by the U.S. National

"Operating system Family - Systems share". Top 500 project.





ScaLAPACK: A Portable Linear Algebra Library for Distributed Memory Computers - Design Issues and Performance * (Technical Paper)

J. Choi[‡], J. Demmel[‡], I. Dhillon[‡], J. Dongarra[§],
S. Ostrouchov[†], A. Petit[‡], K. Stanley[‡], D. Walker[¶], and R. C. Whaley[†]

Abstract

This paper outlines the content and performance of ScaLAPACK, a collection of mathemat-



New Mexico

Jack Dongarra
(Far left)



Clev Moler's PhD
Student

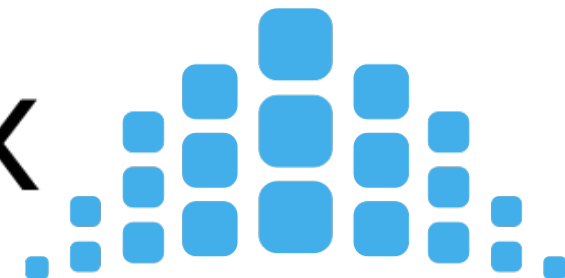
11/18/25



WAREWULF



Rocky Linux



slurm
workload manager



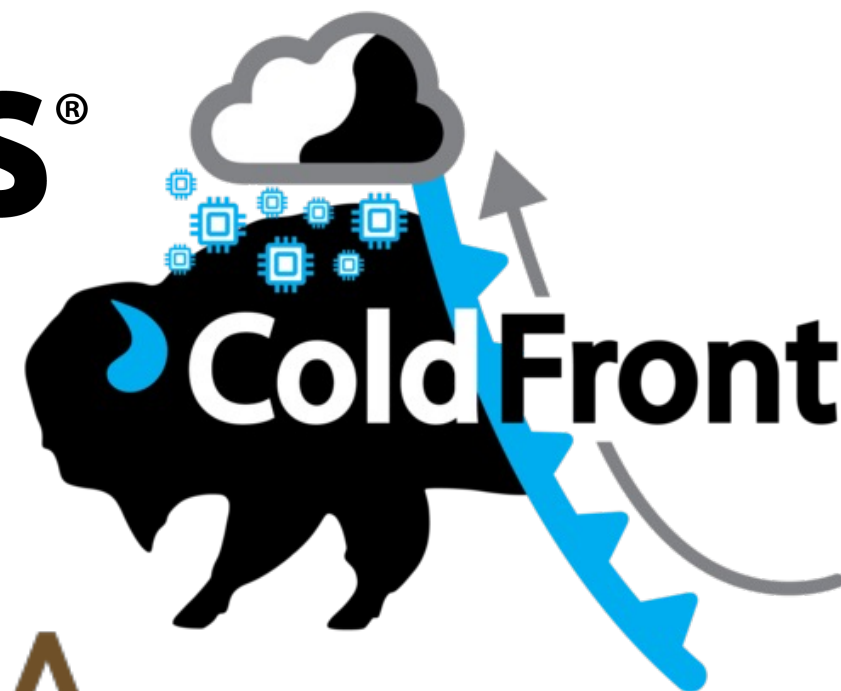
APPTAINER

ubccr/mokey

FreeIPA self-service account management portal



BeeGFS®



13

Contributors

19

Issues

199

Stars

47

Forks



Spack



FreeIPA

Open Source Identity Management Solution

Rocky 9 – Why & How



bug-for-bug
compatible



Richard Stallman
Free Software Foundation, GNU



Linus Torvalds
Linux, Git



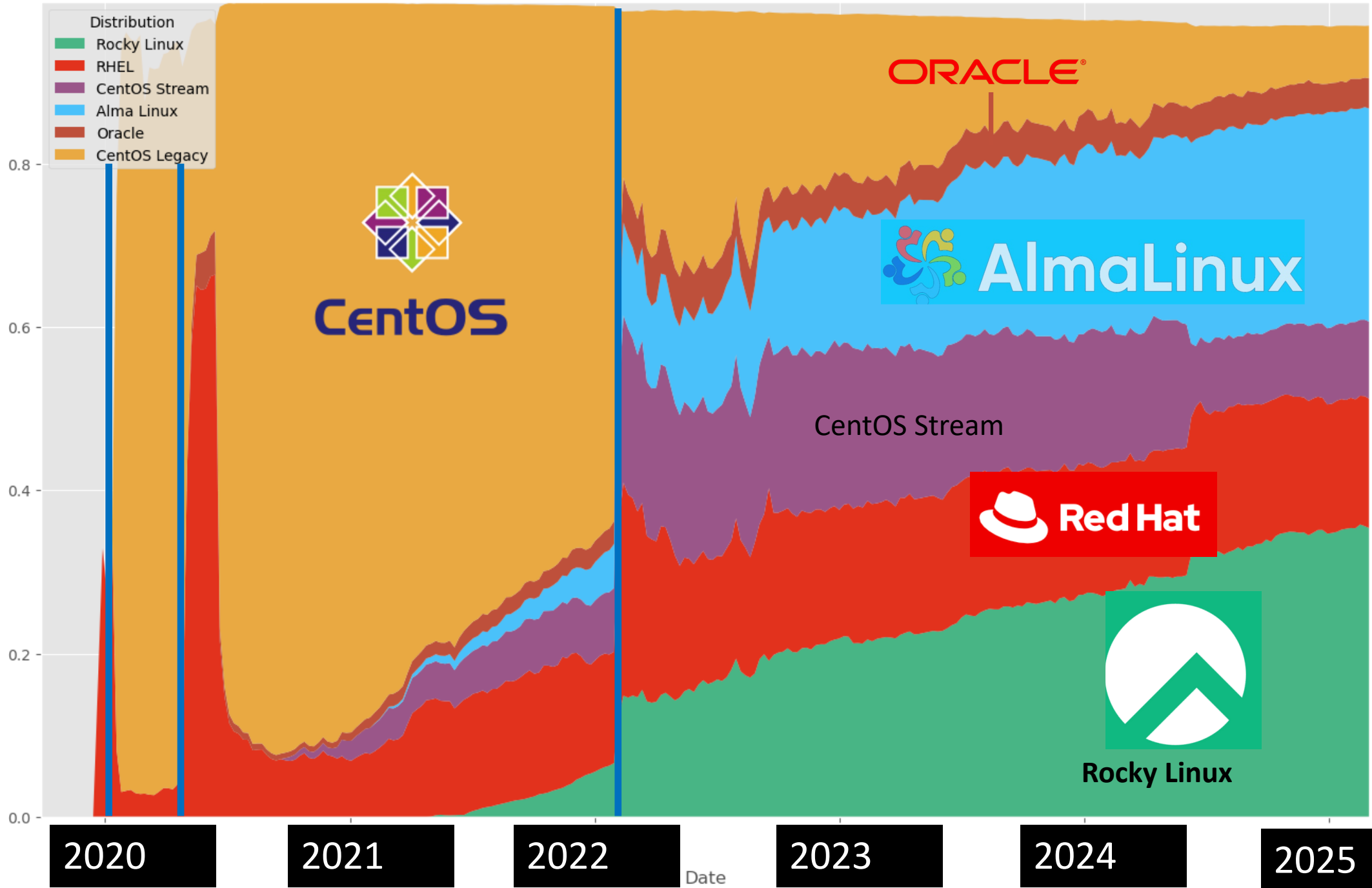
Lennart Poettering
Red Hat
Systemd



Gregory Kurtzer (CIQ)
Singularity, Rocky, Warewulf

<https://thenewstack.io/centos-creator-gregory-kurtzer-discusses-his-new-distro-rocky-linux/>

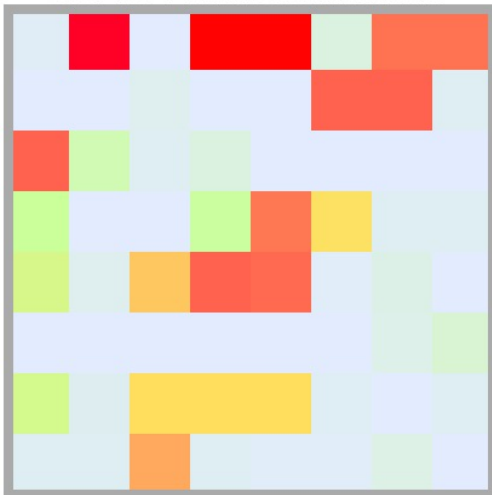
Share of Instances



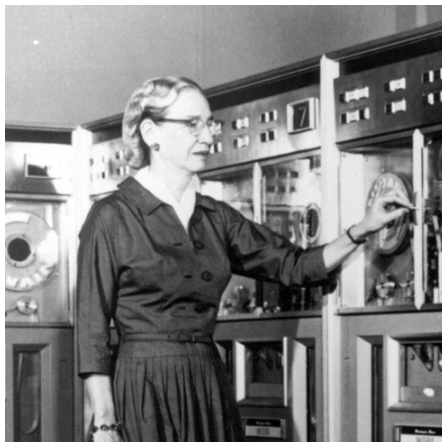
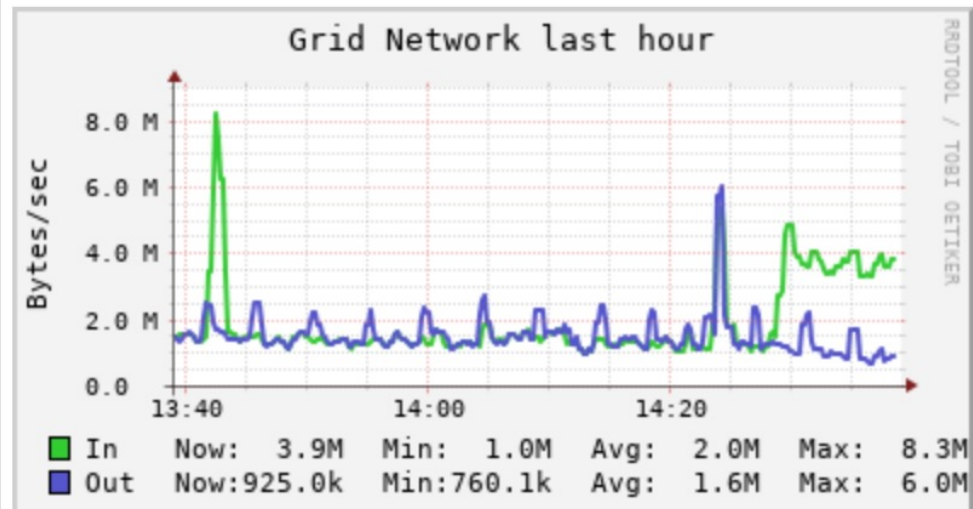
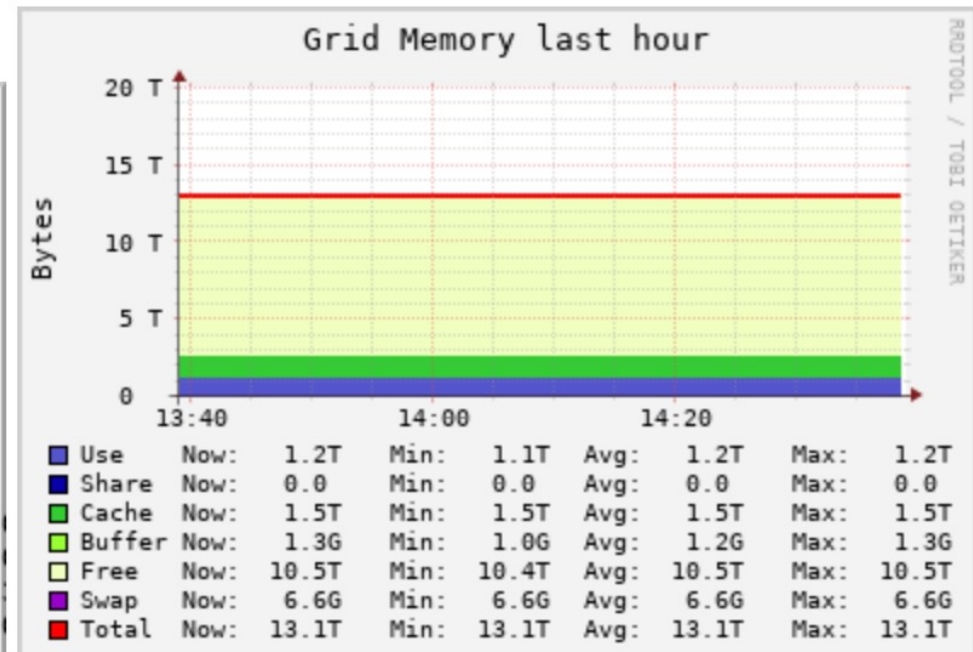
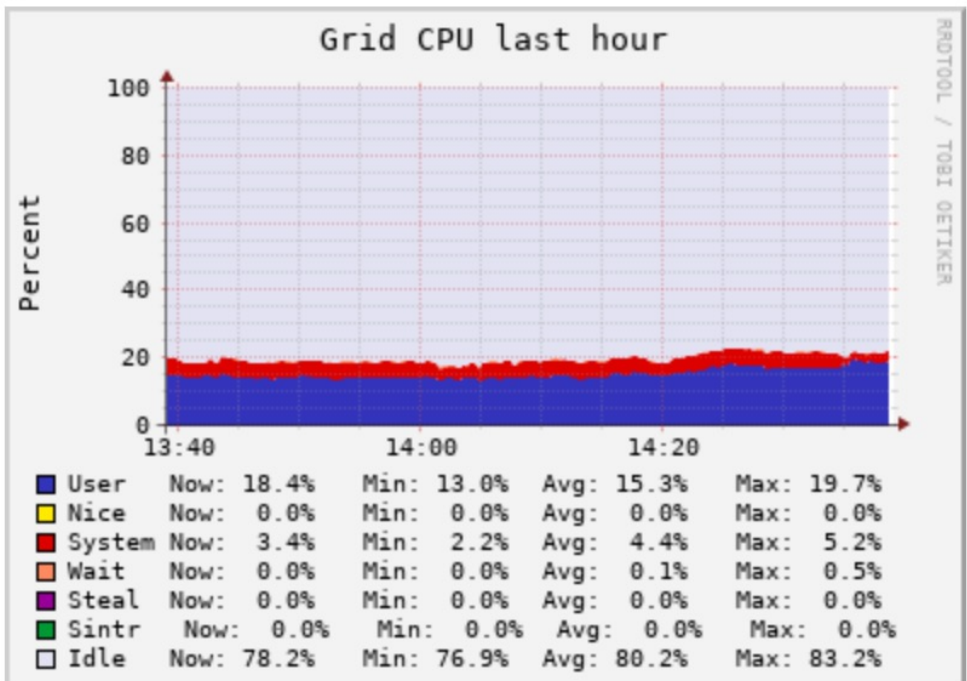
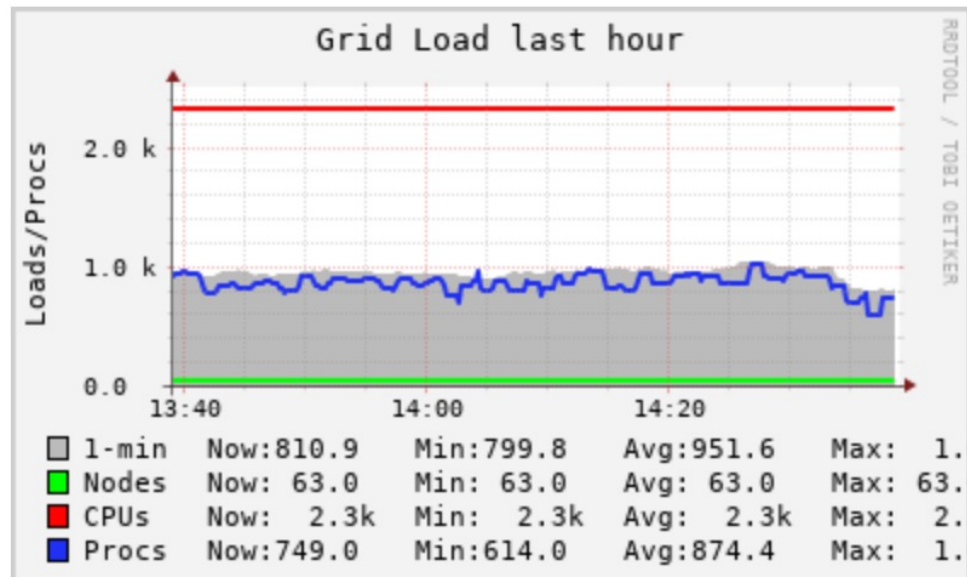
CPUs Total: **2336**
 Hosts up: **63**
 Hosts down: **0**

Current Load Avg (15, 5, 1m):
40%, 37%, 35%
 Avg Utilization (last hour):
41%

Server Load Distribution



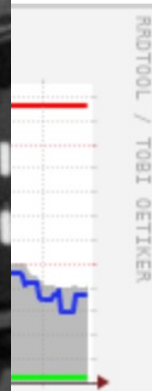
Hopper



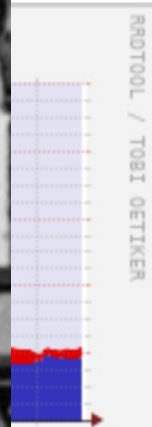
CPU Total:

2226

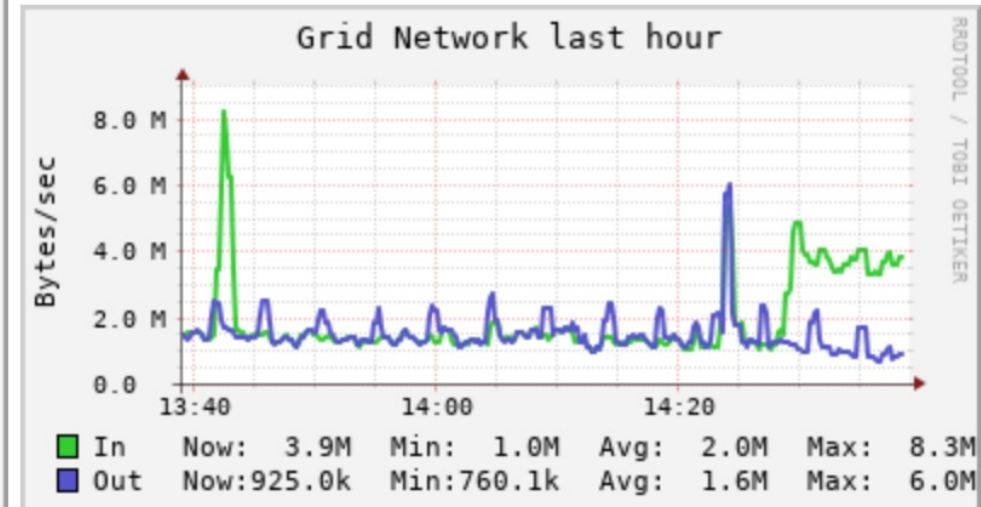
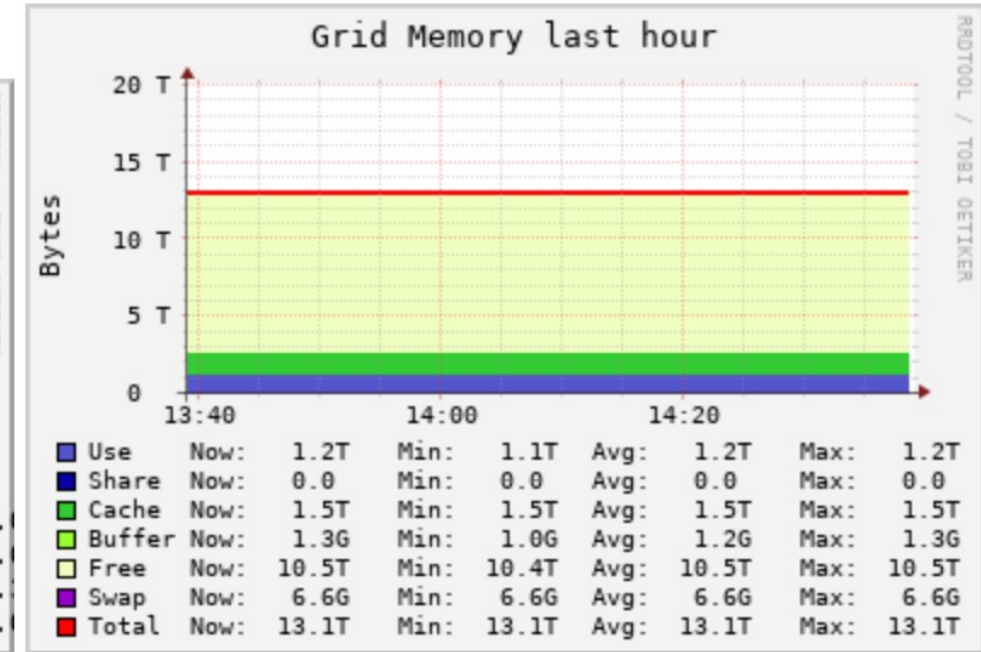
Hopper



Max: 1.
Max: 63.
Max: 2.
Max: 1.

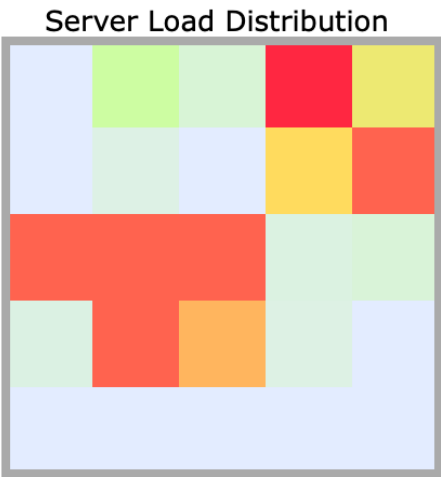


9.7%
0.0%
5.2%
0.5%
0.0%
0.0%
3.2%

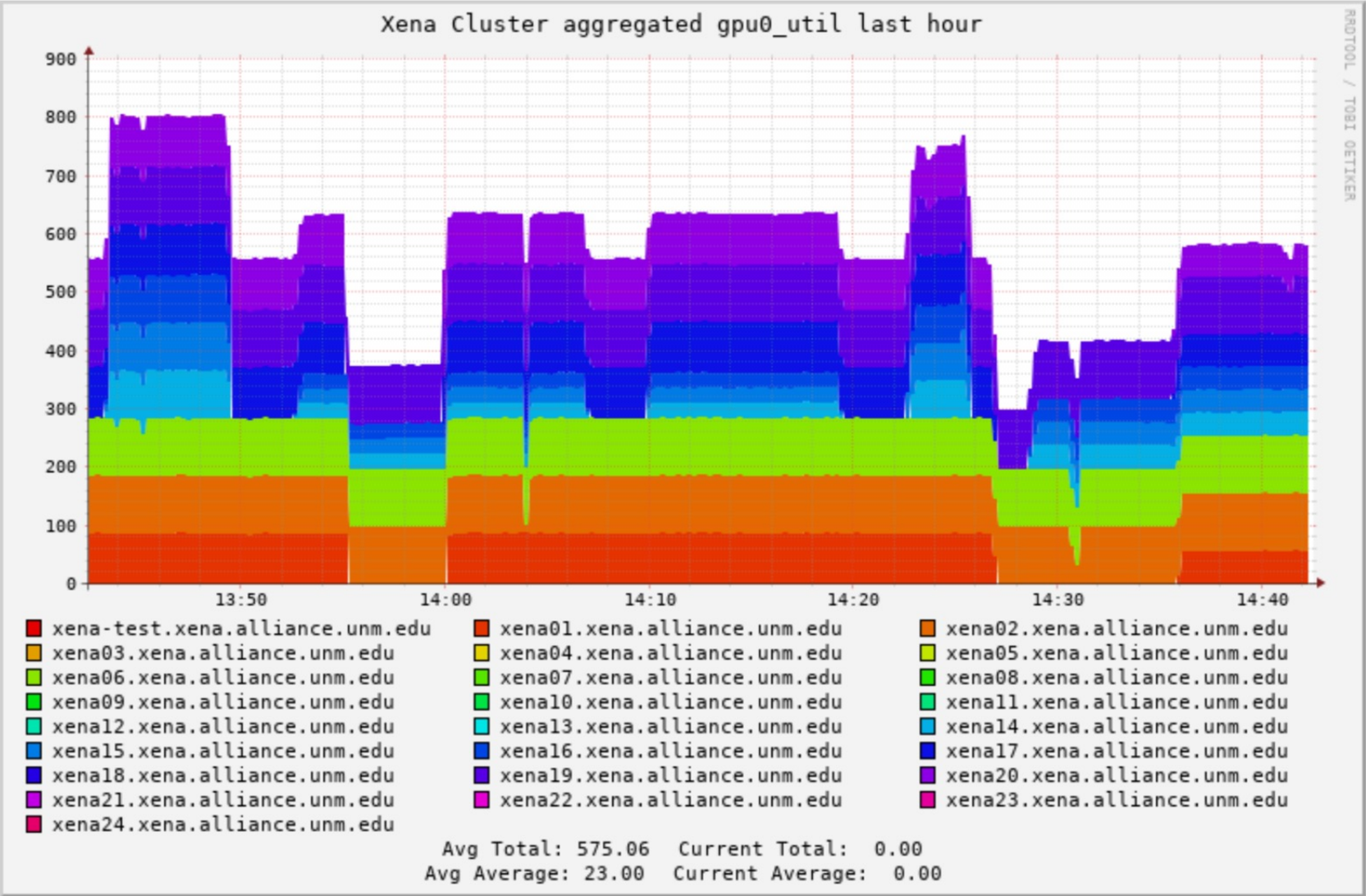


CPU's Total:416
Hosts up:26
Hosts down:0

Current Load Avg (15, 5, 1m):
39%, 39%, 39%
Avg Utilization (last hour):
38%



Stacked Graph - gpu0_util (%)



CPU's Total:

416

Stacked Graph - gpu0_util (%)

Hosts up:

26

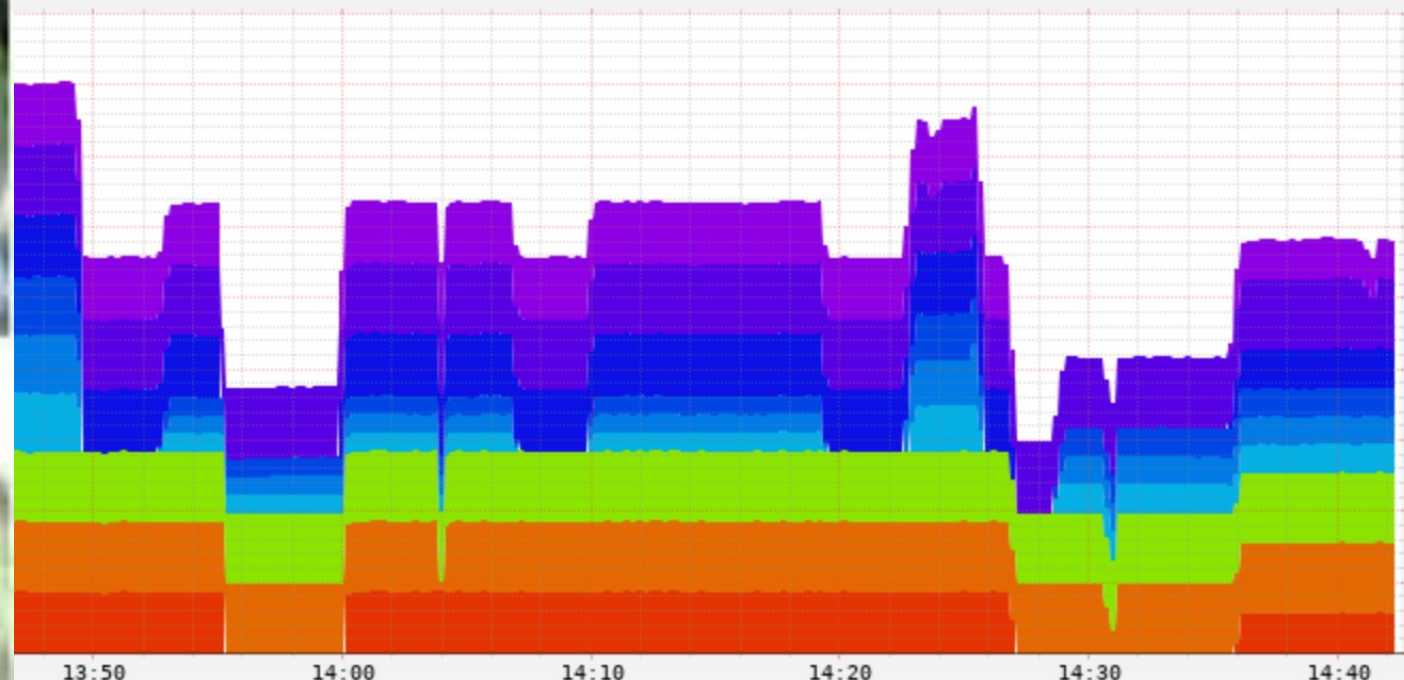
H

C

A



Xena Cluster aggregated gpu0_util last hour



xena.alliance.unm.edu
a.alliance.unm.edu
a.alliance.unm.edu
a.alliance.unm.edu
a.alliance.unm.edu
a.alliance.unm.edu
a.alliance.unm.edu
a.alliance.unm.edu
a.alliance.unm.edu
a.alliance.unm.edu

xena01.xena.alliance.unm.edu
xena04.xena.alliance.unm.edu
xena07.xena.alliance.unm.edu
xena10.xena.alliance.unm.edu
xena13.xena.alliance.unm.edu
xena16.xena.alliance.unm.edu
xena19.xena.alliance.unm.edu
xena22.xena.alliance.unm.edu

xena02.xena.alliance.unm.edu
xena05.xena.alliance.unm.edu
xena08.xena.alliance.unm.edu
xena11.xena.alliance.unm.edu
xena14.xena.alliance.unm.edu
xena17.xena.alliance.unm.edu
xena20.xena.alliance.unm.edu
xena23.xena.alliance.unm.edu

Avg Total: 575.06 Current Total: 0.00
Avg Average: 23.00 Current Average: 0.00

	Easley (2025)	Hopper (2023)	Wheeler (2016)	Xena (2016)	Taos (2020)
OS	Rocky 9.5	Rocky 8	CentOS 7	CentOS 7	CentOS 7
Interconnect	Infiniband 400G	InfiniBand 200G/100G	InfiniBand 40G	InfiniBand 56G	InfiniBand 56G
# of Nodes	65	61	304	32	9
GPUs	8 H100*s, 24 L40*s	37 A100s	0	33 K40s	0
CPU Cores	4,160	2,176	2,432	720	180
RAM/node	256 GB – 2 TB	96 GB – 1.5 TB	48 GB	64 GB – 3 TB	64 GB – 1 TB
Local disk/node	1 TB	448 GB	Diskless	1 TB	1 TB
CPU Architecture	Sapphire Rapids*	Intel Cascade Lake	Intel Nehalem EP	Intel Haswell	Intel Broadwell

*AMX – Native 2D registers and matrix operations, H100s for comp chem, L40s for ML

	Easley (2025)	Hopper (2023)	Wheeler (2016)	Xena (2016)	Taos (2020)
OS	Rocky 9.5	Rocky 8	CentOS 7	CentOS 7	CentOS 7
Interconnect	Infiniband 400G	InfiniBand 200G/100G	InfiniBand	InfiniBand	InfiniBand
# of Nodes	65	61			
GPUs	8 H100*s, 24 L40*s	37 A100s			
CPU Cores	4,160	2,176			
RAM/node	256 GB – 2 TB	96 GB – 1.5 TB			
Local disk/node	1 TB	448 GB			
CPU Architecture	Sapphire Rapids*	Intel Cascade Lake	Intel Nehalem EP	Intel Haswell	Intel Broadwell



Wheeler 2400 cores in 306 nodes, 12 racks
 Estimated 25 Terra Floating Point Operations/second (Tflops)

*AMX – Native 2D registers and matrix operations, H100s for comp chem, L40s for ML

Easley (2025)

Hopper (2023)

Wheeler (2016)

Xena (2016)

Taos (2020)

OS

Rocky 9.5

Rocky 8

CentOS 7

CentOS 7

CentOS 7



4000 Cores in 1 Rack

133.5 Tflops

4 compute nodes per 2U slot

Wheeler 2400 cores in 306 nodes, 12 racks

Diskless

1 TB

1 TB

Intel Nehalem

Intel H

Intel Broadwell

5.37x

*AMX — Native 2D registers and matrix operations, H100s for comp chem, L40s for ML

The GPU Era

- In 2001 Nvidia introduced the GeForce 3 video card.
- This card allowed the pixel renderers to be programmed instead of being hardcoded.
- Over time Nvidia made their graphic processing units (GPUs) more programmable with additional programmable engines for vertex and geometry shading.
- These different engines for programming different parts of the graphics pipeline became too complex.
- Nvidia came up with a general model for programming pixel math on the GPU (2009 GeForce 8800)
- The language Nvidia invented to program these pixel operations was the Compute Unified Compute Architecture (CUDA)
- GPUs are part of the reason clusters are moving towards having fewer but more powerful nodes.



GPUs for Computation

- GPUs proved to be ideal for matrix math.
- The GPU architecture applies the same operation to thousands of elements at the same time.
- This is a bit like the earlier CPU vectorization operations on steroids (512 vector operations in the latest Intel CPUs, The Nvidia H100 GPU has 14,592 CUDA cores)
- The H100 can execute almost 15,000 instructions per clock cycle.
- This doesn't work for all types of computations, if you can't apply the same operation to a vector of values the GPU will be slower than a CPU.

Scaling up

- NVIDIA DGX-2H (\$400,000 each, 81k CUDA cores, 10240 tensor cores)
https://www.nvidia.com/content/dam/en-zz/es_em/Solutions/Data-Center/dgx-2/nvidia-dgx-2h-datasheet.pdf



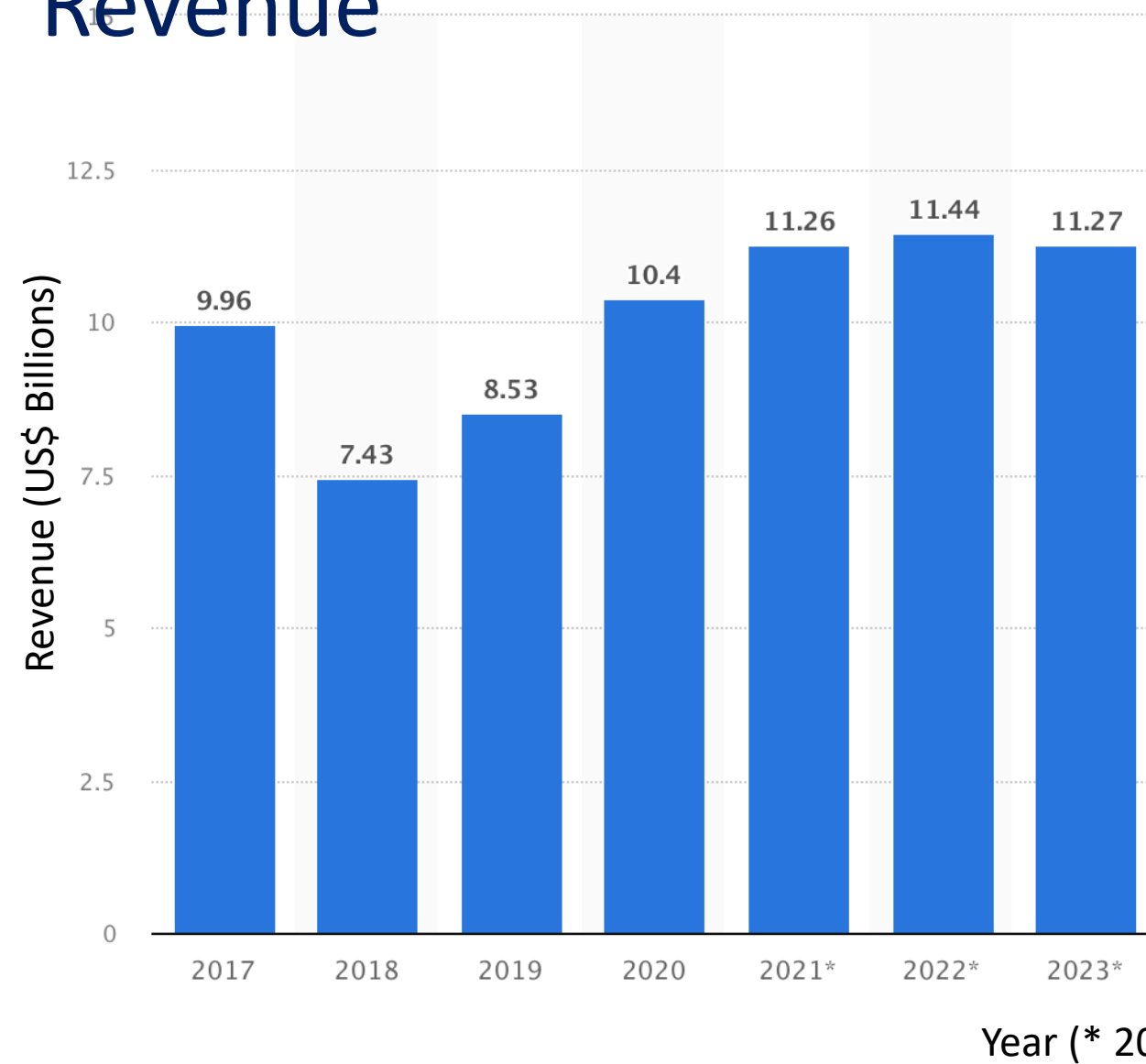
Scaling out

- Stampede 2
- <https://www.tacc.utexas.edu/systems/stampede2>
- \$30,000,000, 285,000 CPUs

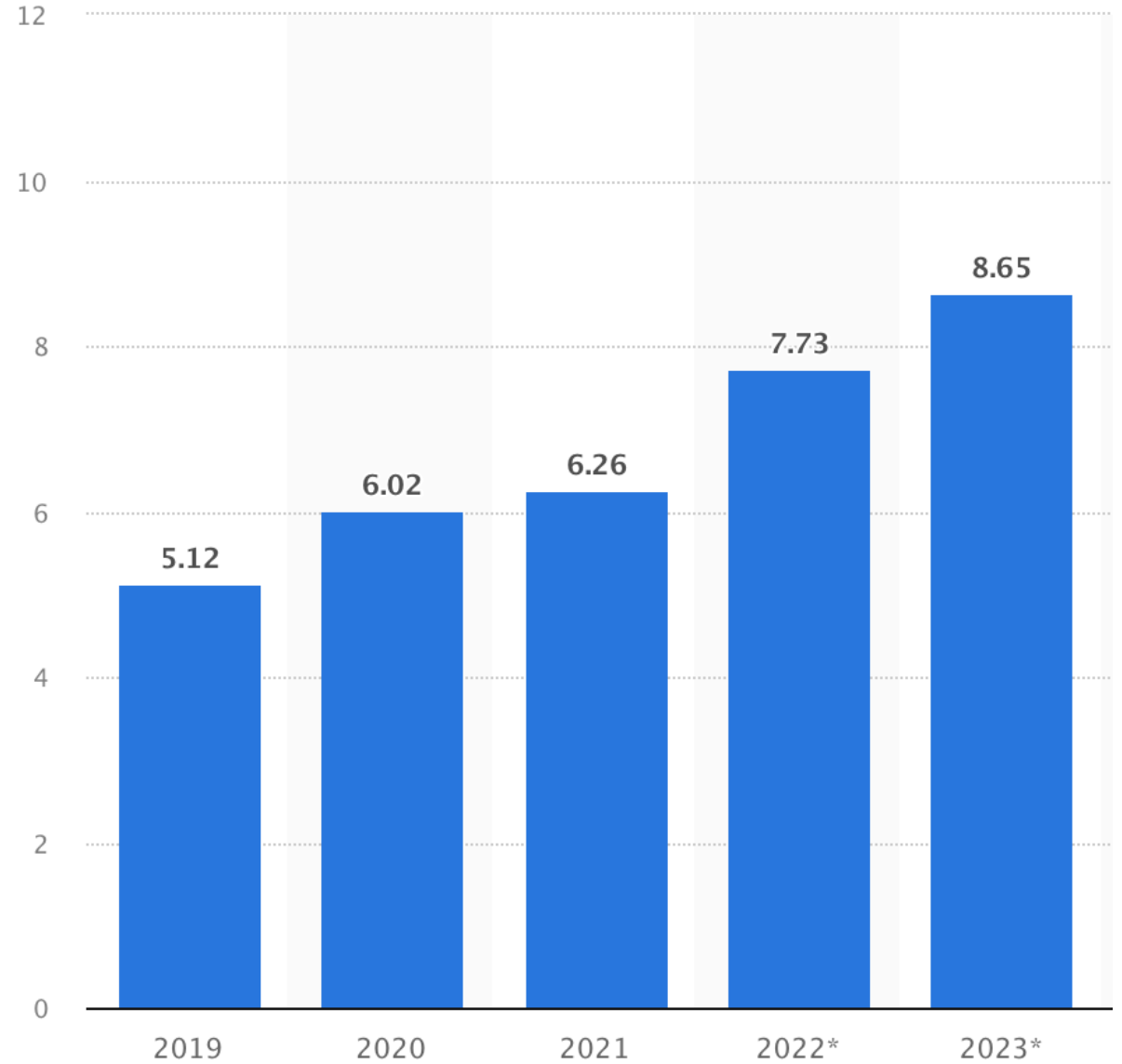


US Revenue

Desktop Computers



Supercomputers



Frontier and Aurora

- Frontier and Aurora are the two fastest supercomputers in the world.
- Frontier is at Oak Ridge National Labs
 - 606,208 AMD Epyc Cores, 37,888 AMD Radeon MI250X GPUs (8,335,360 CUDA cores).
 - 22 Mwatt system
 - Frontier was the first exaflop system with 1.67×10^{18} floating point operations per second.
 - Cost \$600 million
 - Built by HPE Cray and Advanced Micro-Devices (Hewlett Packard bought Cray in 2019)

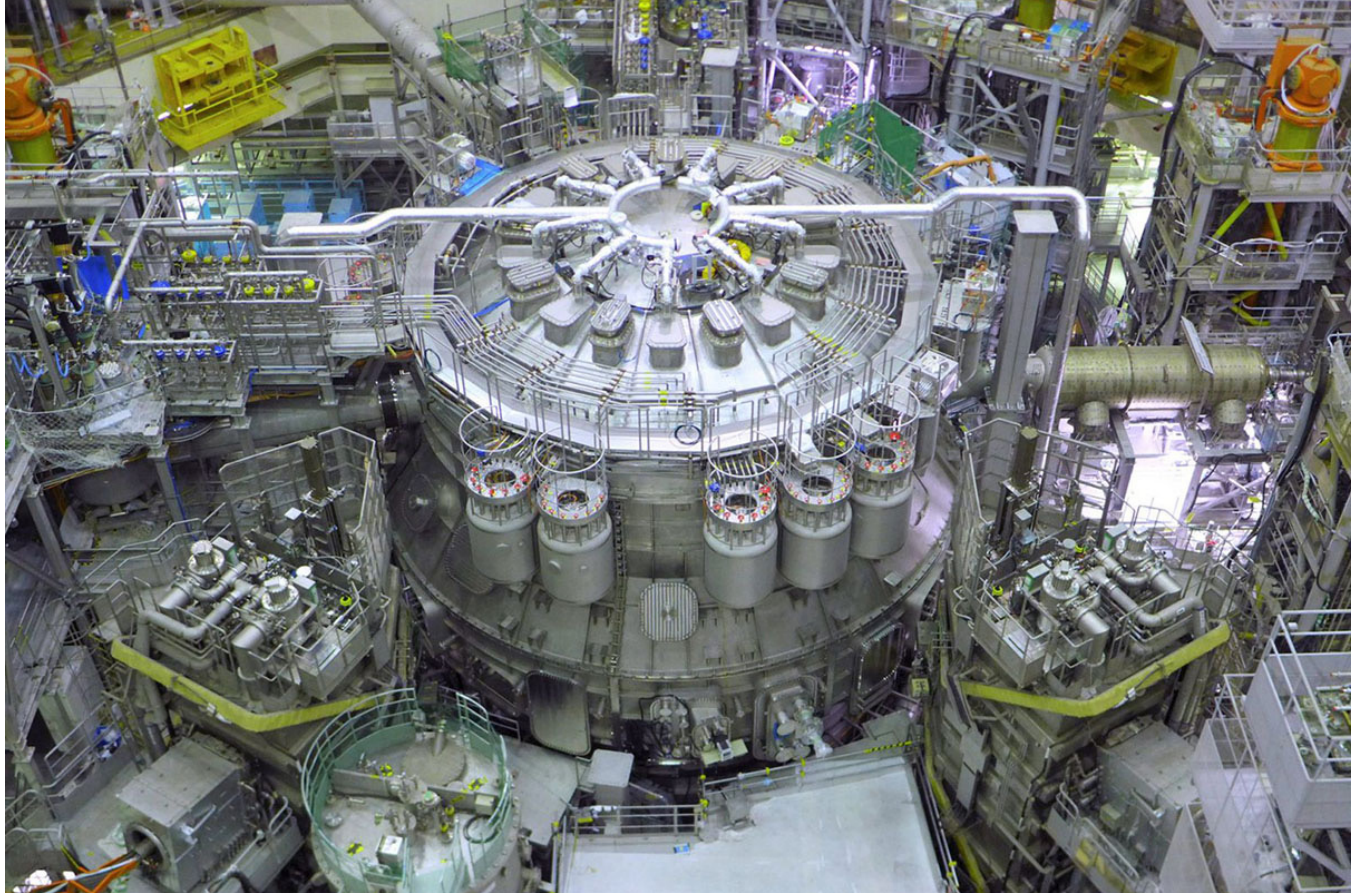
Cray is back under HPE



Frontier and Aurora

- Aurora is still being deployed but is already the second fastest super computer in the world at 500 petaflops.
- When complete it will be a 2 exaflop supercomputer as measured by the High Performance LINPACK (HPL) Benchmark.
- Argonne National Labs
- Frontier is at Oak Ridge National Labs
 - 1.1 million Intel Sapphire Rapids cores, 63,744 Intel GPUs (8,335,360 CUDA cores).
 - 60 Mwatt system
 - Cost \$500 million
 - Built by HPE Cray and Intel





Frontier and Aurora

- Both will primarily be used for simulating nuclear fusion and fission reactors.
- They will also be used for everything from modeling virus evolution to windfarms.

Measuring Performance

- I've mentioned FLOPS a lot so far along with LINPACK.
- Supercomputers have historically been defined by the number of floating point operations they can perform per second.
- The standard benchmark for measuring those operations is High Performance LINPACK.
- LINPACK is a very commonly used linear algebra package.
- Jack Dongarra developed LINPACK and HPL while he was a CS graduate student here at UNM.

HPCG

- The Top500 list is based on LINPACK and HPL
- High Performance Conjugate Gradients (HPCG) benchmark is a newer benchmark
- HPCG is supposed to have access patterns that are more representative of other applications
- Benchmarks drive cluster development

What is Modern Supercomputing?



1 processor with 6 cores.
Can run 6 programs at a time.



1 processor 2 cores

What is Supercomputing?



1 processor with 6 cores.
Can run 6 programs at a time.



1 processor with 8 cores.
Can run 8 programs at a time.



1 processor 2 cores



Current Era – Corporations Driving Supercomputing not Government Labs

- **Shift in Supercomputing Leadership**
- Historically, the world's largest supercomputers were built and operated by **national laboratories** (e.g., DOE labs in the U.S., RIKEN in Japan).
- Today, **industry data centers** (Microsoft, Google, Amazon, Meta) host systems with *greater aggregate computing power* than government-funded facilities.
- This marks a **structural change**: the frontier of large-scale computing has moved from publicly funded labs to private corporations.

Current Era –
Corporations Driving
Supercomputing not
Government Labs
Systems like Aurora
and Frontier are
Exexascale

Zetascale is 1000x
more powerful



What do you think of
this headline knowing
what you know from
Wednesday



Oracle Announces Zettascale Cloud Supercluster with 131,000 Blackwells

September 11, 2024 by [staff](#)



Oracle today announced what it said is the first zettascale cloud HPC cluster, powered by Nvidia's forthcoming Blackwell GPUs, scheduled for shipment in the first half of 2025. Oracle Cloud Infrastructure will be available with up to 131,072 Blackwells delivering 2.4 zettaFLOPS of peak performance, according to Oracle.

The company said this offers more than three times as many GPUs as the Frontier supercomputer, currently the world's most powerful system on the TOP500 list, and more than six times that of



Amazon's Project Rainier Sets New Standard for AI Supercomputing at Scale

AWS's Project Rainier is building one of the world's largest AI supercomputing clusters, powered by custom Trainium2 UltraServers. More than just raw compute, Rainier reflects Amazon's push to vertically integrate its AI infrastructure, redefine data center scale, and challenge Nvidia's dominance in the hyperscale AI arms race.

David Chernicoff, Matt Vincent

July 8, 2025 ▪ 8 min read

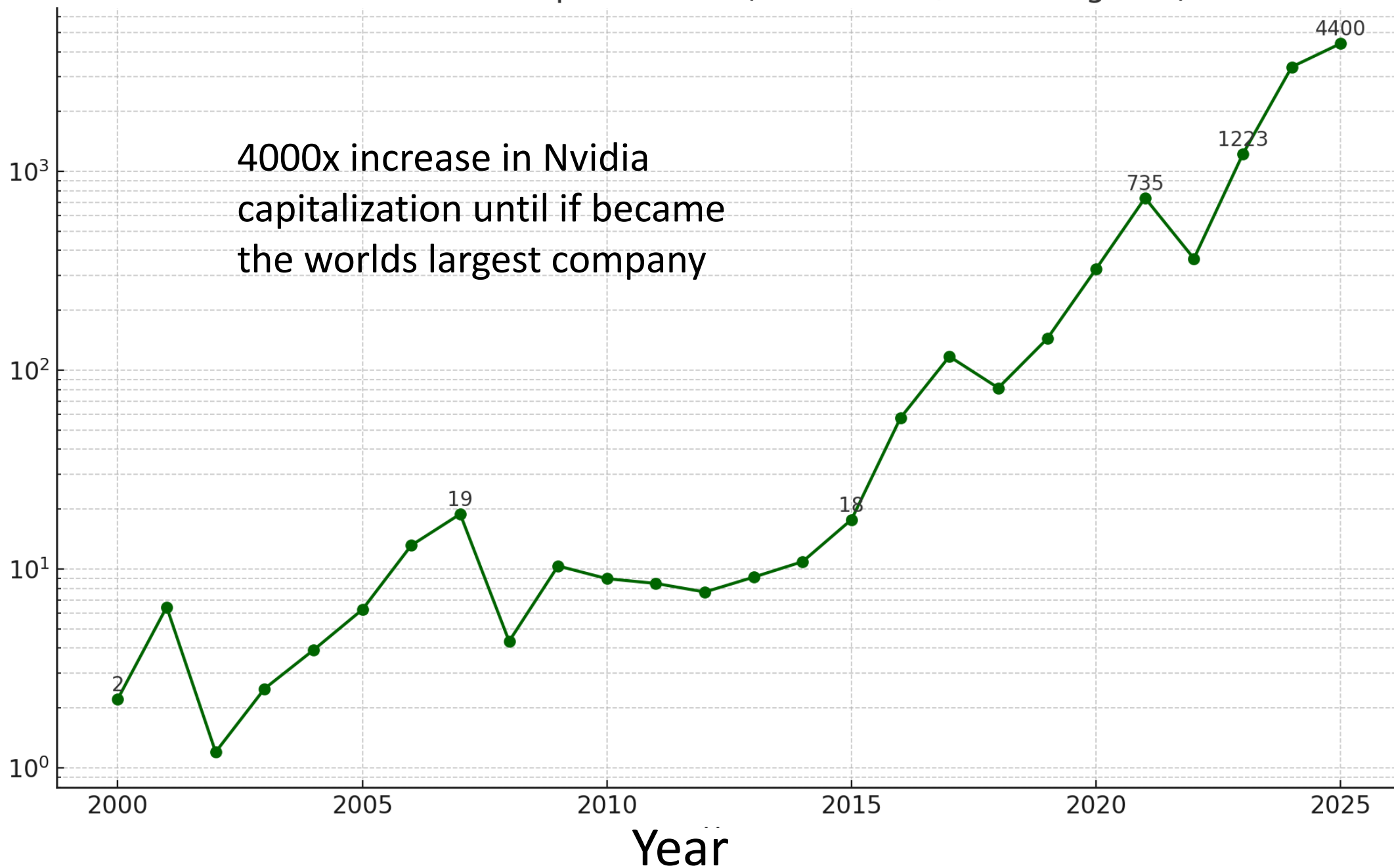


Project Rainier Server and Cable Installation

Current Era – Corporations Driving Supercomputing not Government Labs

NVIDIA Market Capitalization (2000-2025, Semi-Log Plot)

Nvidia Market Capitalisation
US\$ Billions (log-scale)



Artificial Intelligence Driving Supercomputing

-  **Early AI optimism** (1956–1973, Dartmouth conference → symbolic AI)
-  **First AI Winter** (1974–1980, funding cuts after the Lighthill Report)
-  **Expert systems revival** (1981–1986, commercial boom)
-  **Second AI Winter** (1987–1993, collapse of expert systems)
-  **Statistical ML growth** (1994–2011, support vector machines, probabilistic models, large datasets)
-  **Deep Learning boom** (2012–present, ImageNet breakthrough, GPU acceleration, generative AI)

Cutting Edge

Homework 1: sign up for <https://hoti.org> (Aug 20-23), attend any of the talks in the first two days and write a half page report summarizing the talk.

Email me that report to Alex by Monday.

- **Chiplets** – many chips integrated into one processor package, each with specialised tasks. CPUs and GPUs can be combined on-package for performance and efficiency.
- **Specialised hardware beyond GPUs** – HPC GPUs are designed for AI/ML and scientific compute, not graphics (no video outputs).
- **High-Bandwidth Memory (HBM)** – stacked memory directly on or beside compute dies, delivering terabytes per second of bandwidth critical for AI + simulation workloads.
- **Advanced Interconnects** – technologies like NVLink, CXL, and InfiniBand NDR/XDR that make multiple chiplets, accelerators, and nodes behave like a single large system.



Computational Science

Massive computation
allows us to simulate
all kinds of things that
we could never
observe directly.

George Luger called
this the
“computational
telescope”.

Four Eras

- The first thirty years of supercomputing was dominated by companies like Cray Research (Seymore Cray), Control Data Corp (CDC) and International Business Machines (IBM) (American companies).
- During this time Japan entered the supercomputing business with Hitachi, Fujitsu, and NEC.
- Until 1995 (What happened in 1995?)
- From 1995 until the present scaling up (monocomputers) became less important than scaling out (multicomputers).
- GPU Era – 2000s to now

Astronomy

A large radio telescope dish is silhouetted against a dark night sky. The dish is a complex metal lattice structure mounted on a pedestal. In the background, the Milky Way galaxy is visible as a bright, hazy band of light stretching across the sky, filled with numerous stars. The foreground shows the dark silhouette of a desert landscape with low hills and a fence line. A few other smaller telescope dishes are visible in the distance on the right side.

hopper.alliance.unm.edu

workshops/radio_astronomy/ casa_example2 - Jupyter Notebook

Trusted Python [conda env: ...]

View Insert Cell Kernel Widgets Help

running tclean, may take a bit...

0%....10....20....30....40....50....60....70....80....90....100

0%....10....20....30....40....50....60....70....80....90....100

Convert the output image into a png file...

```
1 imview('first_image.image', out='test.png')
```

Display the image...

```
1 Image(filename="test.png")
```

hopper.alliance.unm.edu

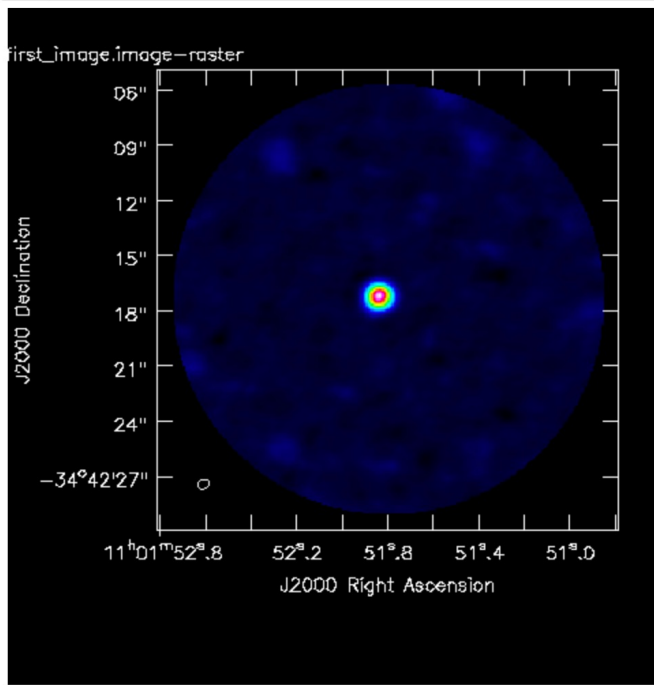
workshops/radio_astronomy/ casa_example2 - Jupyter Notebook

terhub casa_example2 Last Checkpoint: 2 hours ago (autosaved) Logout

dit View Insert Cell Kernel Widgets Help Trusted Python [conda env: conda]

```
1 Image(filename="test.png")
```

at[10]:

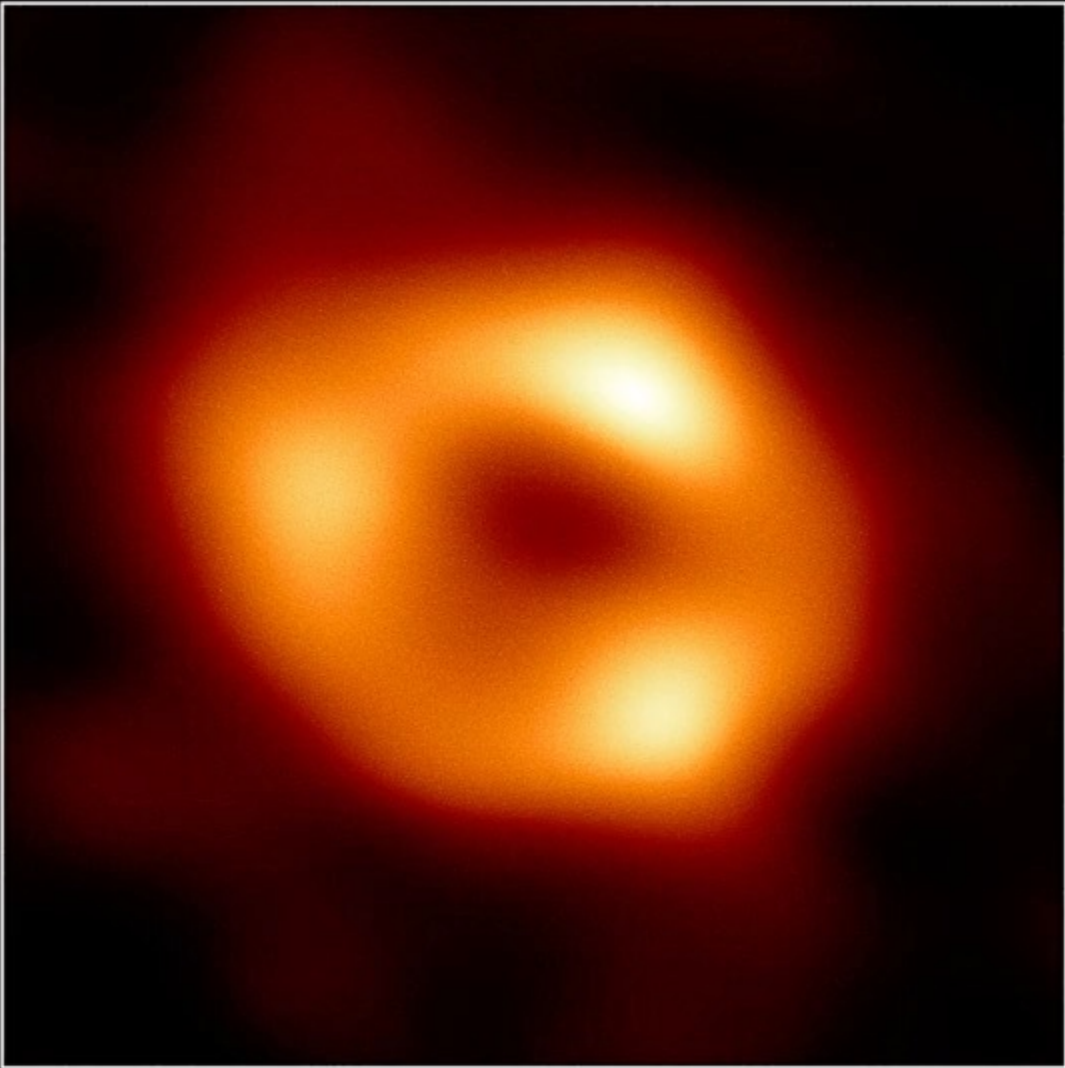


first_image.image - raster

J2000 Declination

J2000 Right Ascension

Texas Advanced Computing Center 448,448 cores (Frontiera Cluster)

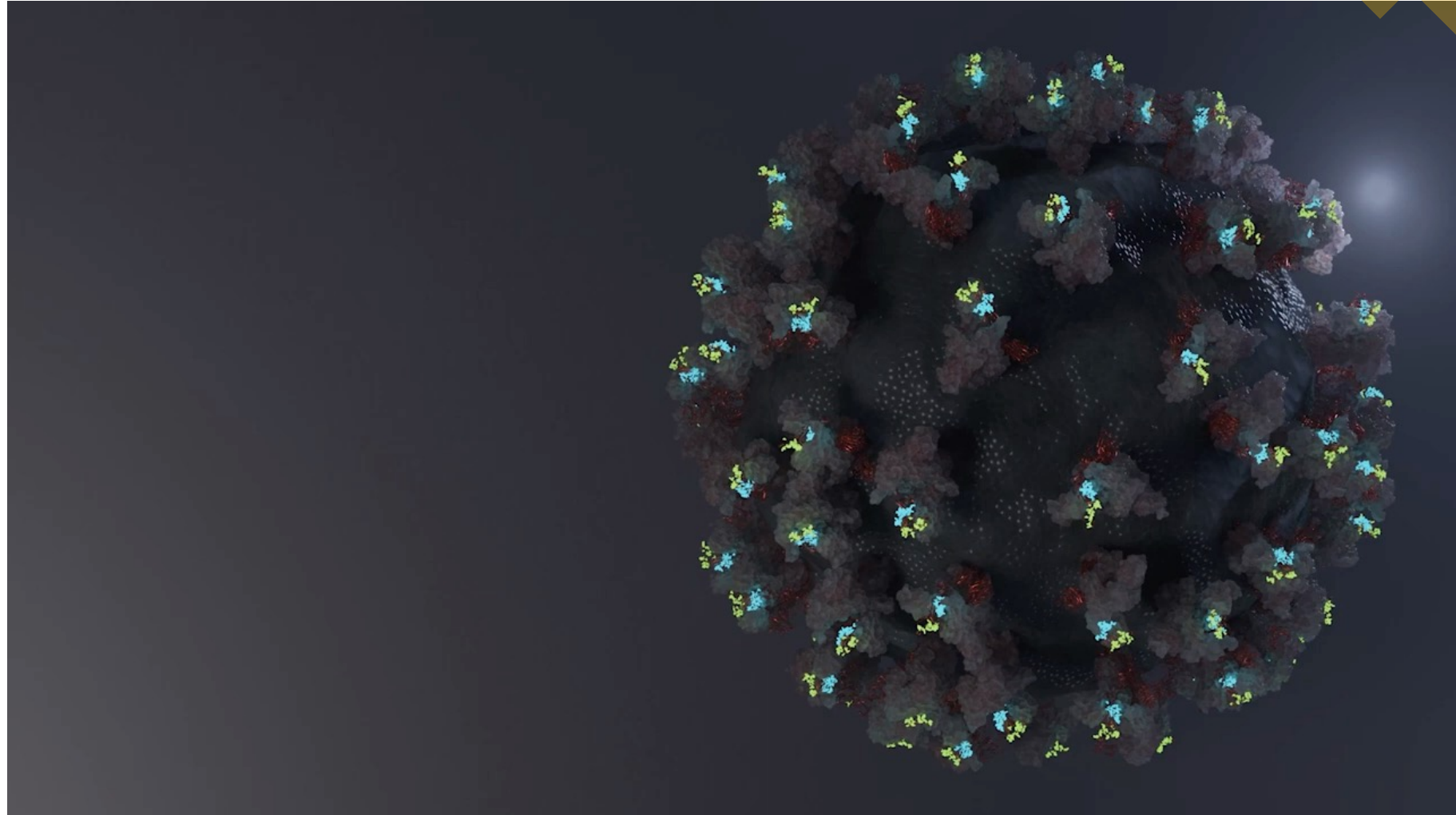


DiRAC High-Performance Computing facility

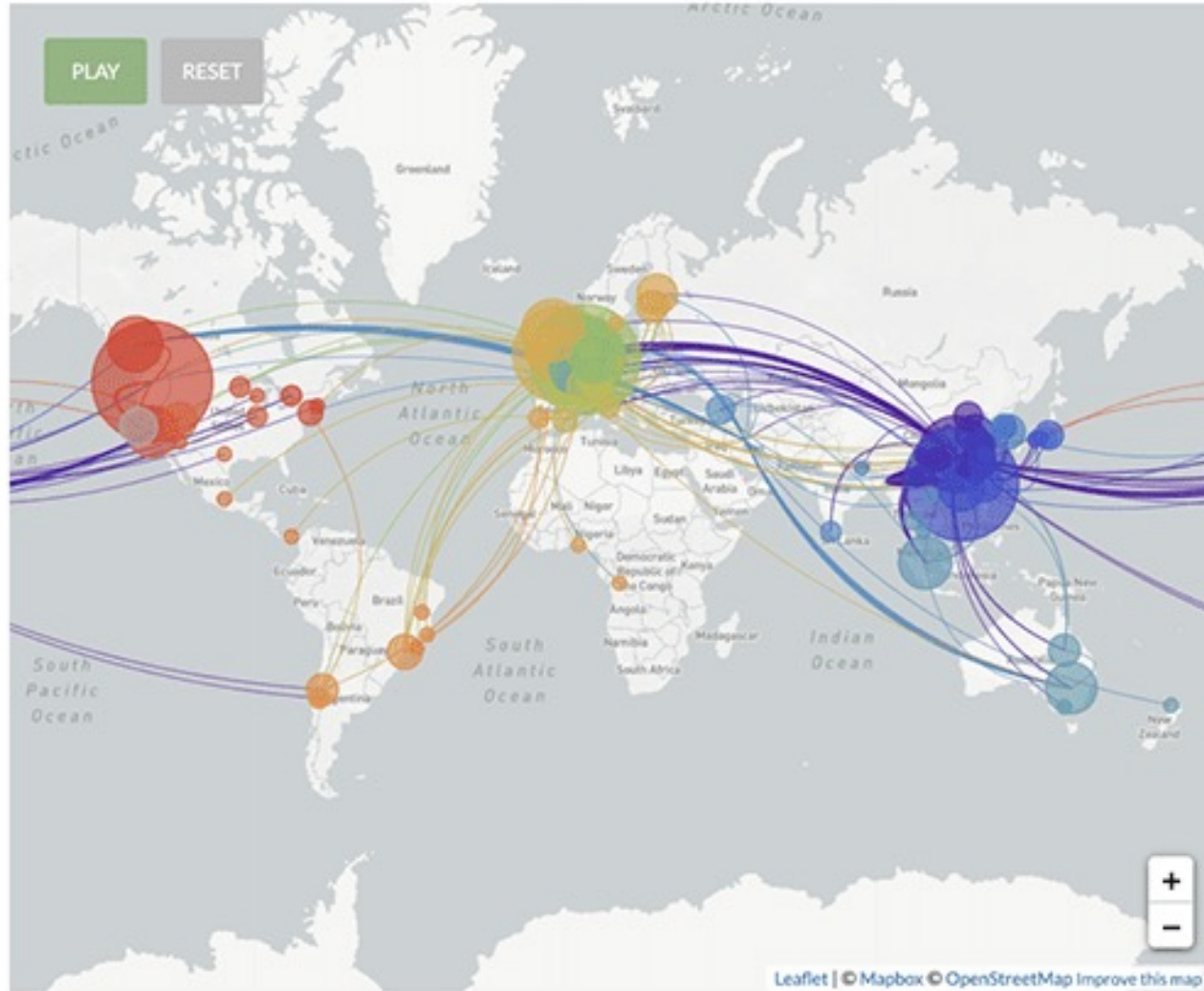
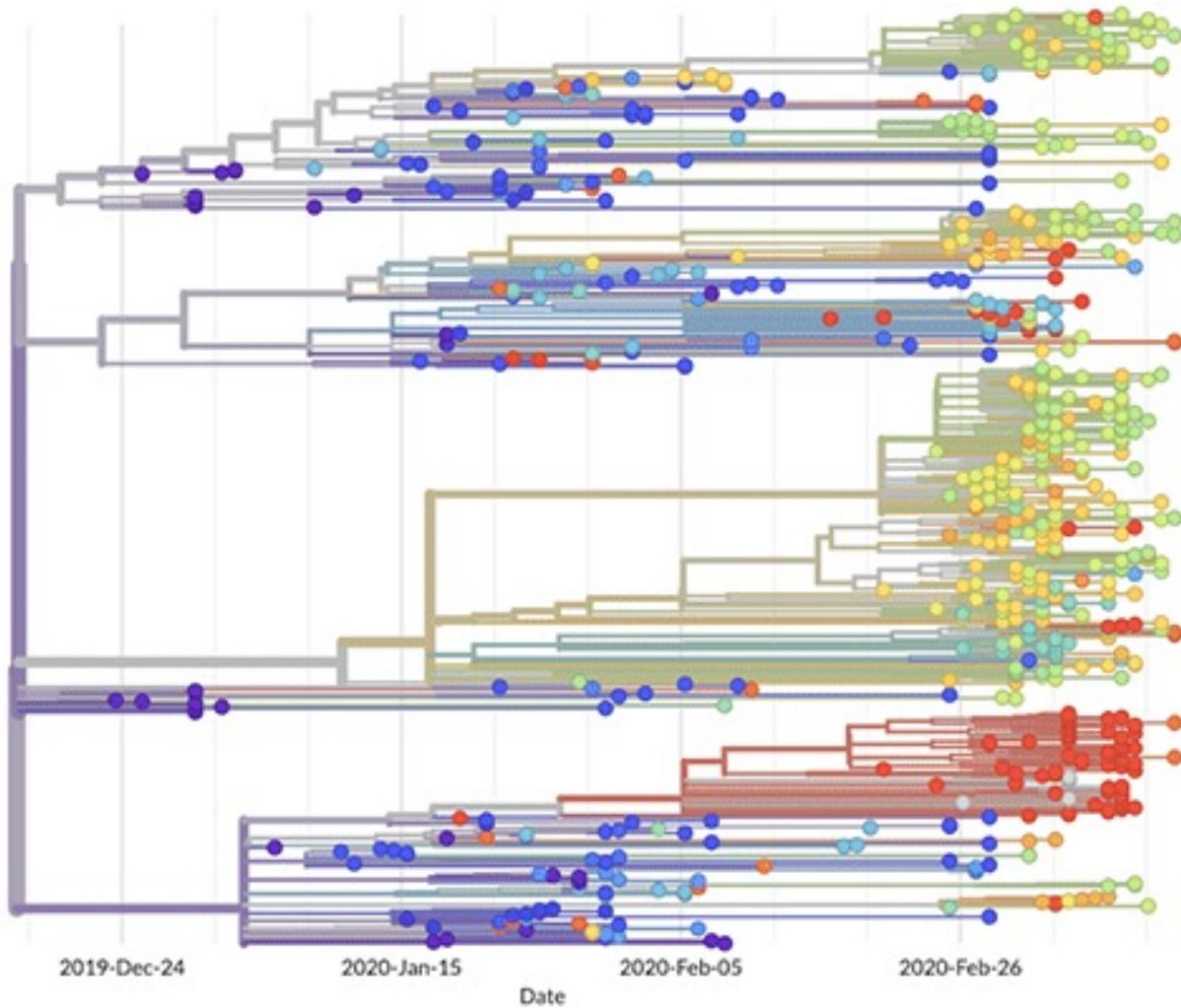


Biology and Chemistry – CoVID Virus Model

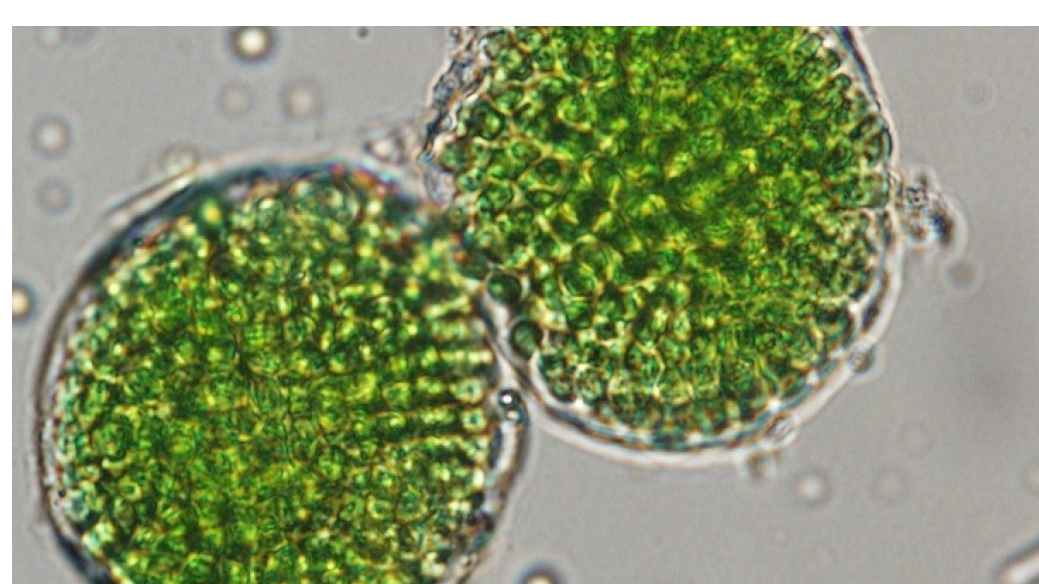
- VMD – Visual Molecular Dynamics
- This model shows how the CoVID spike works (It's the part in yellow)
- That's how CoVID gets into our cells
- This simulation used 200,000 GPUs



Biology and Chemistry – CoVID Evolution



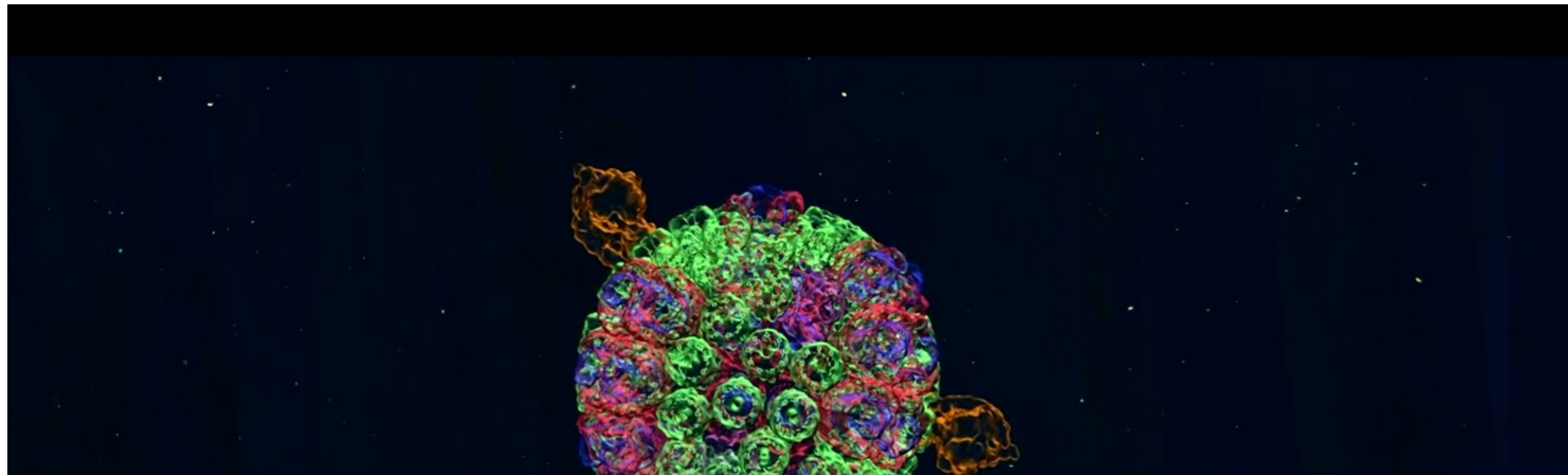
CoVID 19's family tree. CARC systems are responsible for sequencing CoVID variants for 4 states.



Molecule level model of a photosynthesizing organelle. The green objects are chloroplasts.

Specialized for low level light.

NASA was able to grow these bacteria under the same light conditions that you find around red dwarf stars. They are a lot more common than our own type of star.



Artificial Intelligence

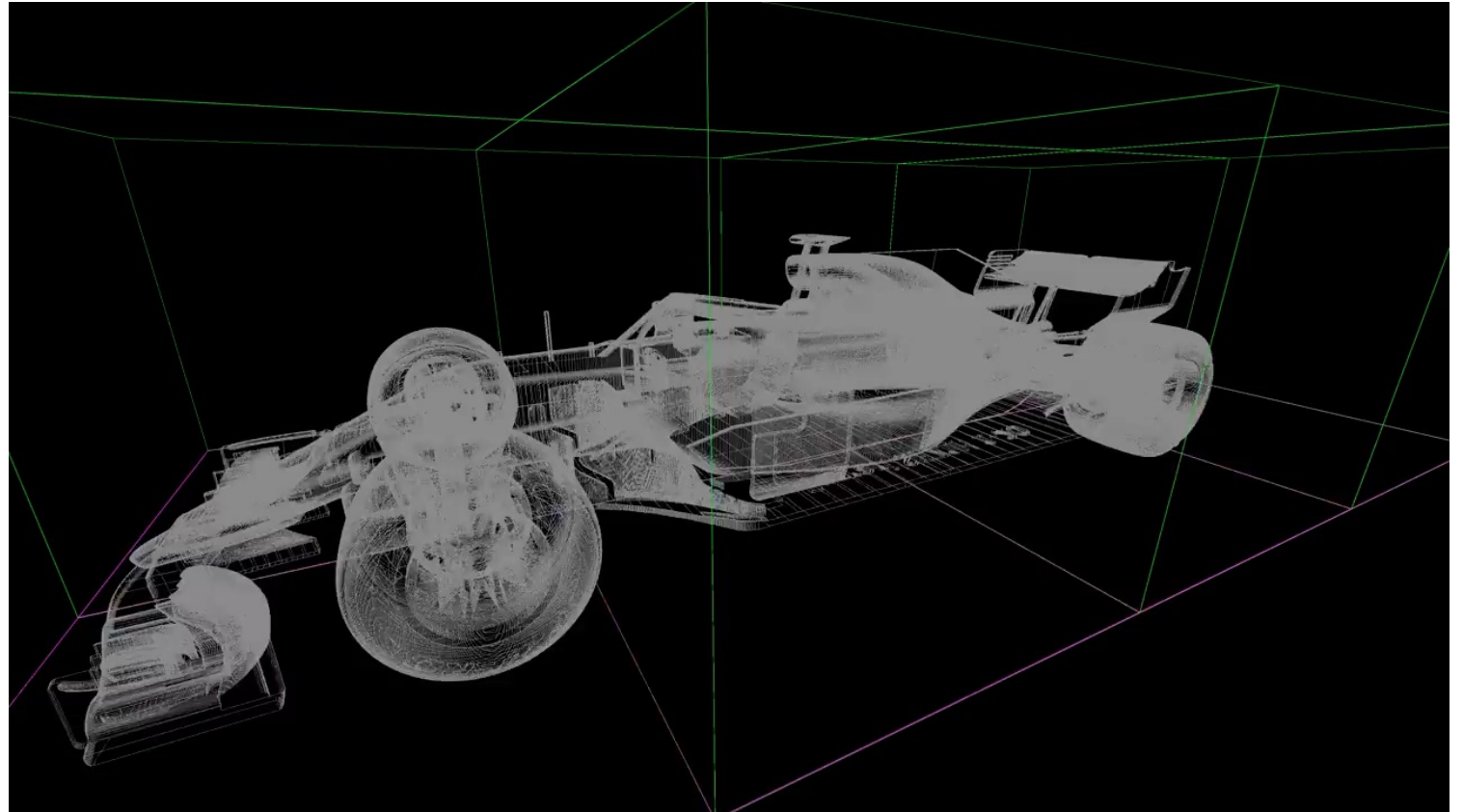
- Machine Learning is everywhere, needs lots of processing power which supercomputers provide.
- Cars (self-driving)
- Banking (money laundering detection)
- Medicine (cancer diagnosis, brain hemorrhage detection)
- Chihuahua vs muffins
- Volcano mapping



Engineering

Computational Fluid Dynamics

The UNM Racecar Team uses these tools
To design their cars.



Engineering – NASA Mars Lander Rockets

- Computational Fluid Dynamics
- Modelling the rockets that will allow a lander to deliver people safely to Mars.
- Run on the Oakridge National Labs Summit Cluster
- 202,752 cores

