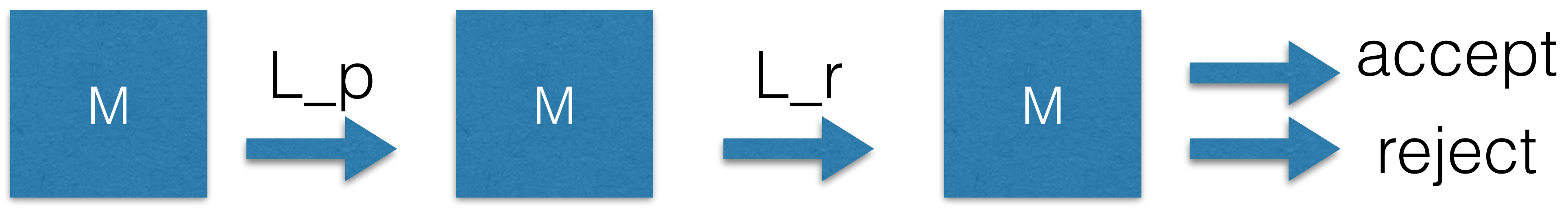
HPC Discussion / Issues

Argonne National Laboratory, November 14-16, US DOE SC NP NSAC Town Hall meeting

Kenneth J. Roche
Pacific Northwest National Laboratory, HPC group
University of Washington, Department of Physics

- Context (not a NP research talk)
- Large scale systems / simulations
- Disruptive hardware and methods
- Workflows
- (no summary … just food for thought)

Context: Classical comput(ers)ing



Programming implies control of machine state evolution

- machine can exist in finite, possibly very large, number of states
- states have representation in basis (instructions -> gates)
- transitions between states are well defined by transition function
- executable requires finite resources

software developer's challenge?

- Moore's Law persists
- sidestepped by massive increase in concurrency
- introduces challenges in all components of computing
- parallelism and concurrency are very poorly utilized in general
- programming model connected to machine design explicitly - no free lunch
  - distributed memory - message passing / remote memory operations
  - shared memory - thread control
  - hybrid (target combined CPU + GPU)
  - abstractions - PGAS (ie provide virtual global address space composed of aggregated resources); implementor pays the price
- implementation efficiency is dismal -nature is way more efficient

on supercomputers performance limited by …

1) **System power** -primary constraint (PUI, facility / total)
2) **Memory** bandwidth and capacity are not keeping pace
3) **Concurrency** 1000X increase in-node
4) **Processor** open question
5) **Programming model** compilers will not hide this
6) **Algorithms** need to minimize data movement, not flops
7) **I/O bandwidth** not on pace with machine speed
8) **Reliability and resiliency**
9) **Bisection bandwidth** limited by cost and energy

Machine construct

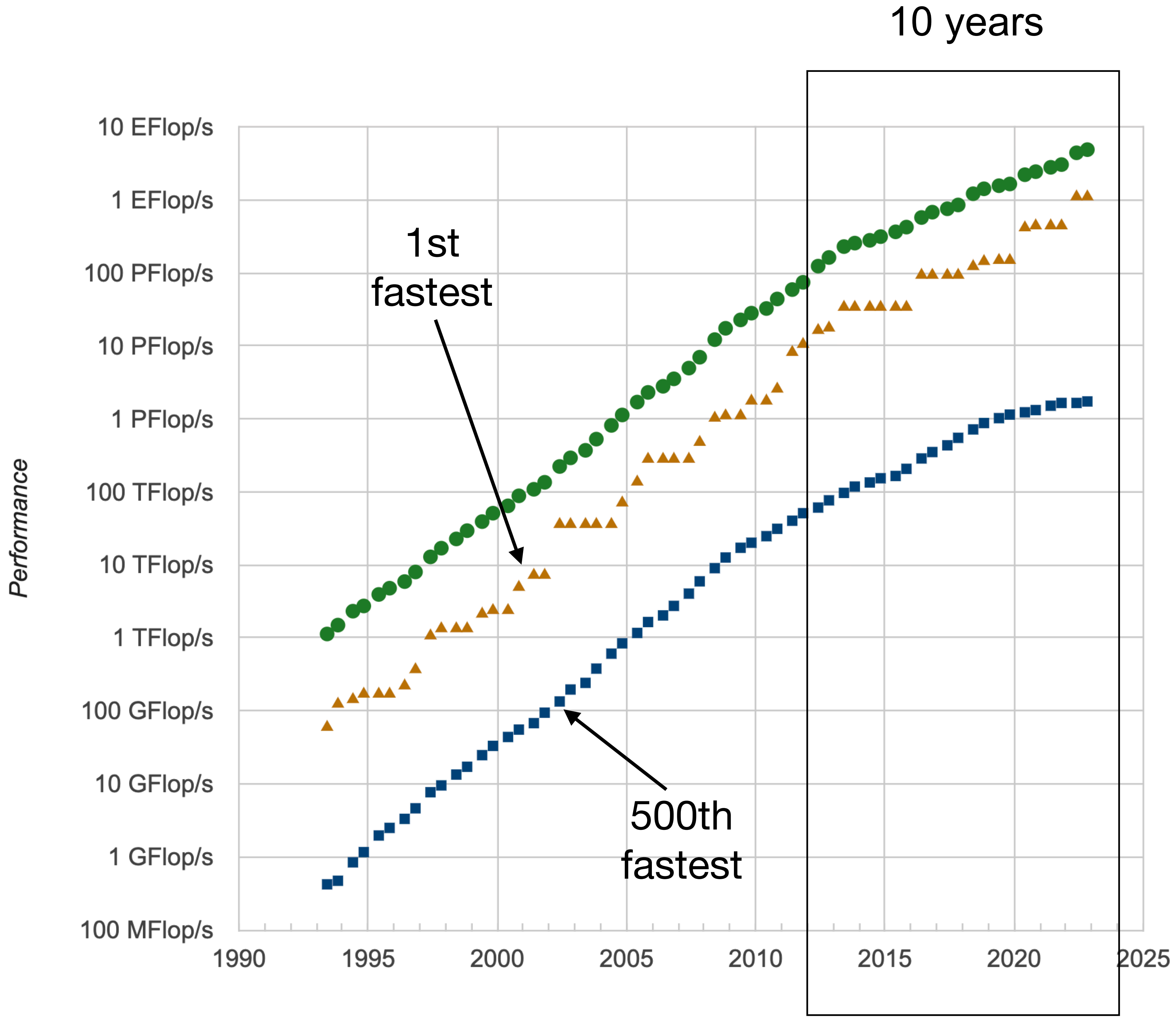
- storage and processing accomplished by switches called transistors

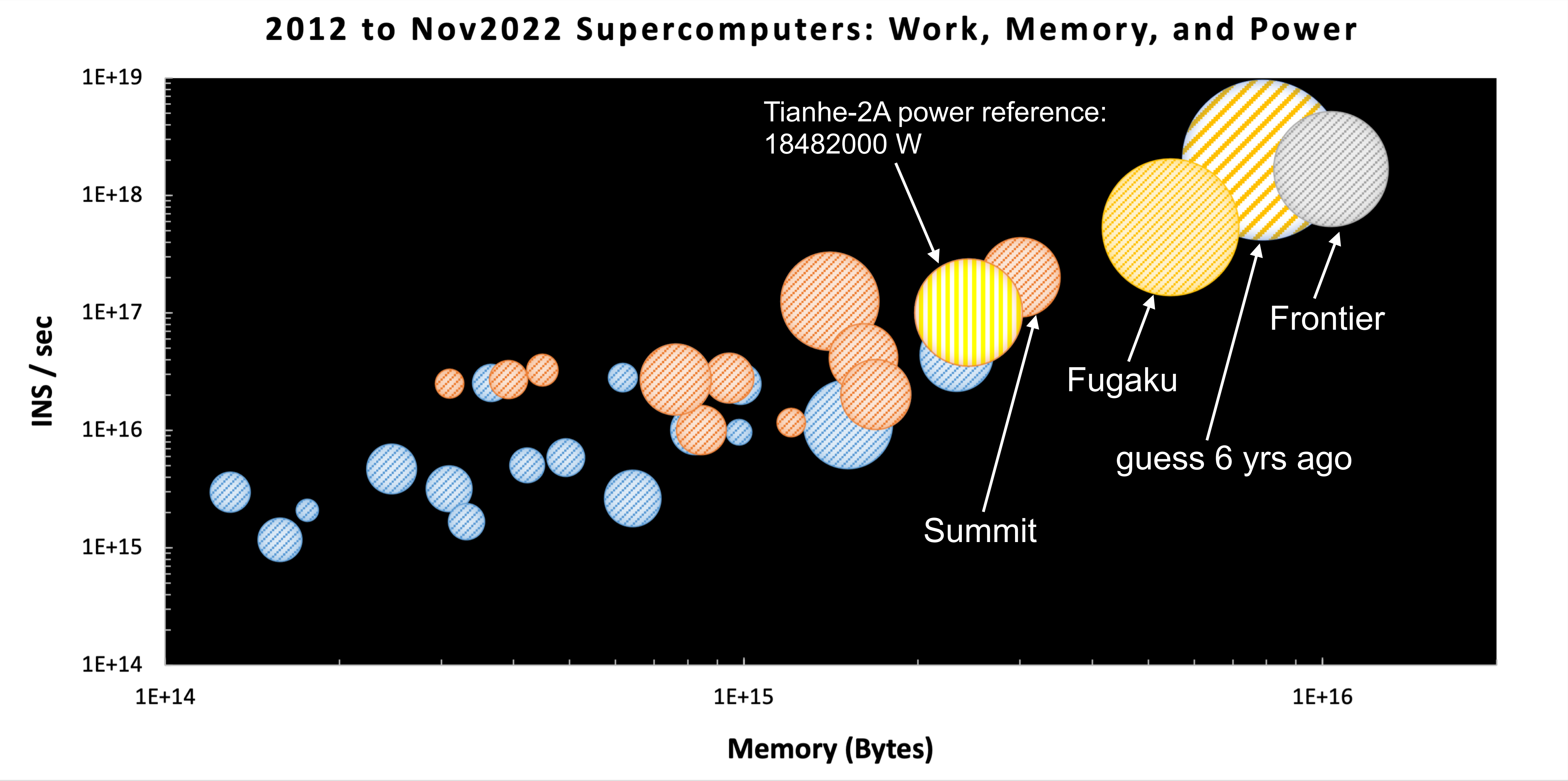
- calculate by using circuits composed of logic gates
  - made from a number of transistors connected together
  - operate predefined action on patterns of bits stored in temporary memories called registers
  - output is new patterns of bits

- algorithm that performs a particular calculation takes the form of an electric circuit made from a number of logic gates, with the output from one gate feeding in as the input to the next

Kenneth J. Roche, Pacific Northwest National Laboratory, HPC group, University of Washington, Department of Physics
Argonne National Laboratory, November 14-16, US DOE SC NP NSAC Town Hall meeting

Quick look at some features of the US DOE's (and the world's) fastest supercomputers





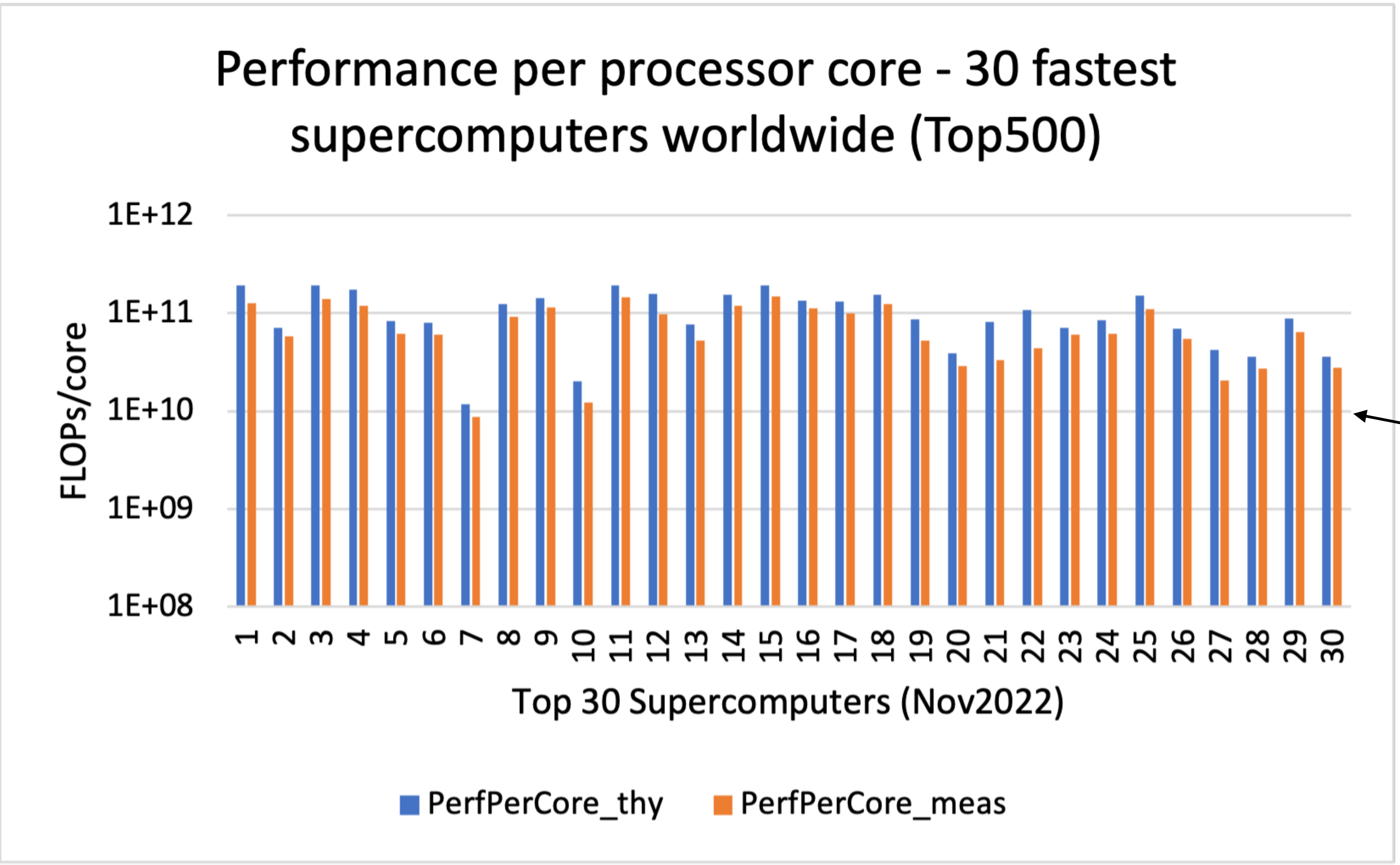
**NB: only the top 5 machines broke the 100PF layer, 500th fastest is < 2PF as of Nov2022
**flops to byte ratio looks difficult to achieve for most applications … let's explore this further

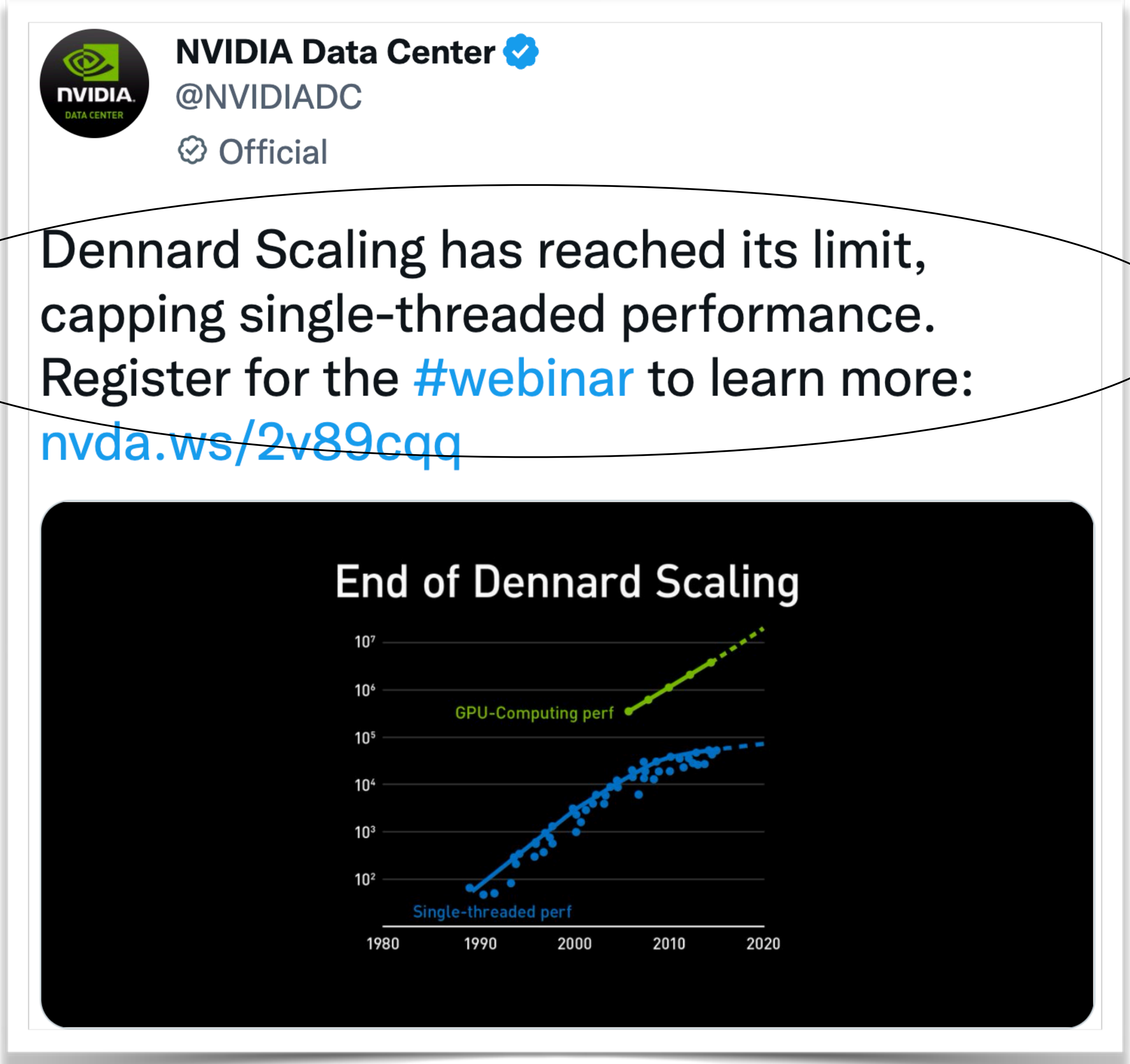
“Power Wall” has constrained practical processor frequency to around 4 GHz since 2006

Dennard scaling: one can continue to decrease the transistor feature size and voltage while keeping the power density constant

Power = a * CFV^2
- a – percent time switched
- C = capacitance
- F = frequency
- V = voltage

“leakage current” and “threshold voltage” cause practical power per transistor limit
consequence: power density increases for smaller transistors because these don’t scale with size

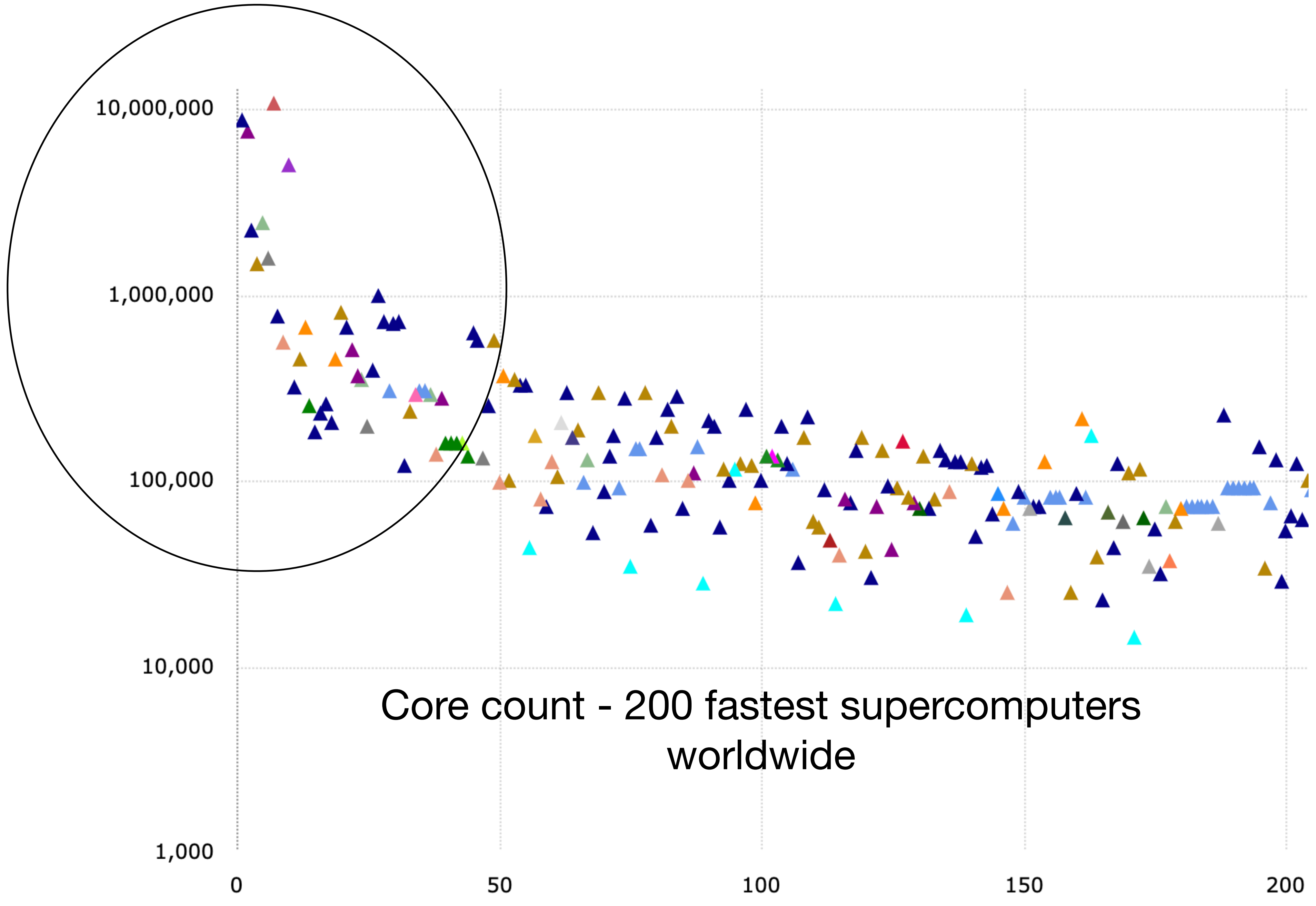


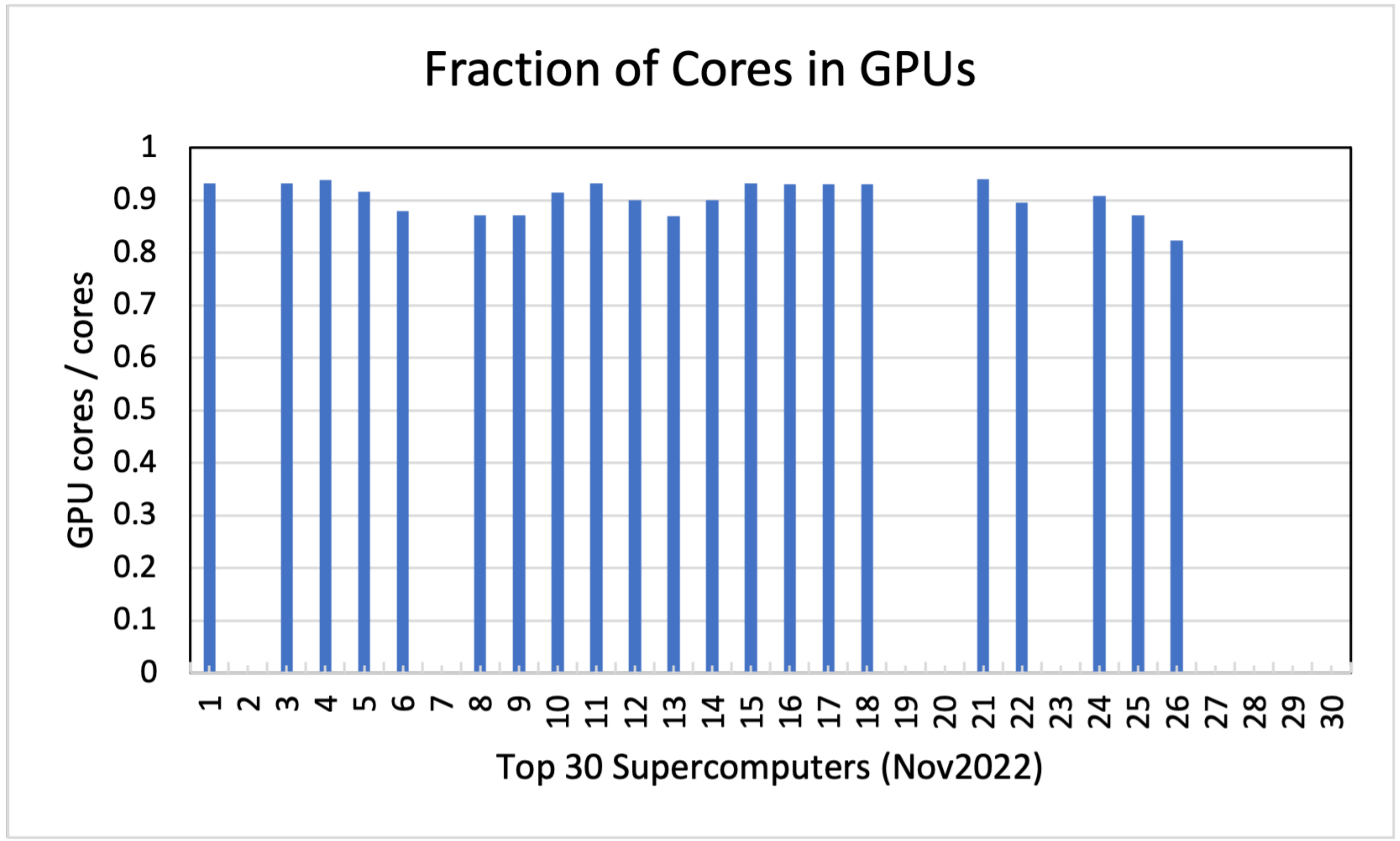


Kenneth J. Roche, Pacific Northwest National Laboratory, HPC group, University of Washington, Department of Physics
Argonne National Laboratory, November 14-16, US DOE SC NP NSAC Town Hall meeting

Factors of 100x come from increased core counts / special instructions / GPU processing

Kenneth J. Roche, Pacific Northwest National Laboratory, HPC group, University of Washington, Department of Physics
Argonne National Laboratory, November 14-16, US DOE SC NP NSAC Town Hall meeting

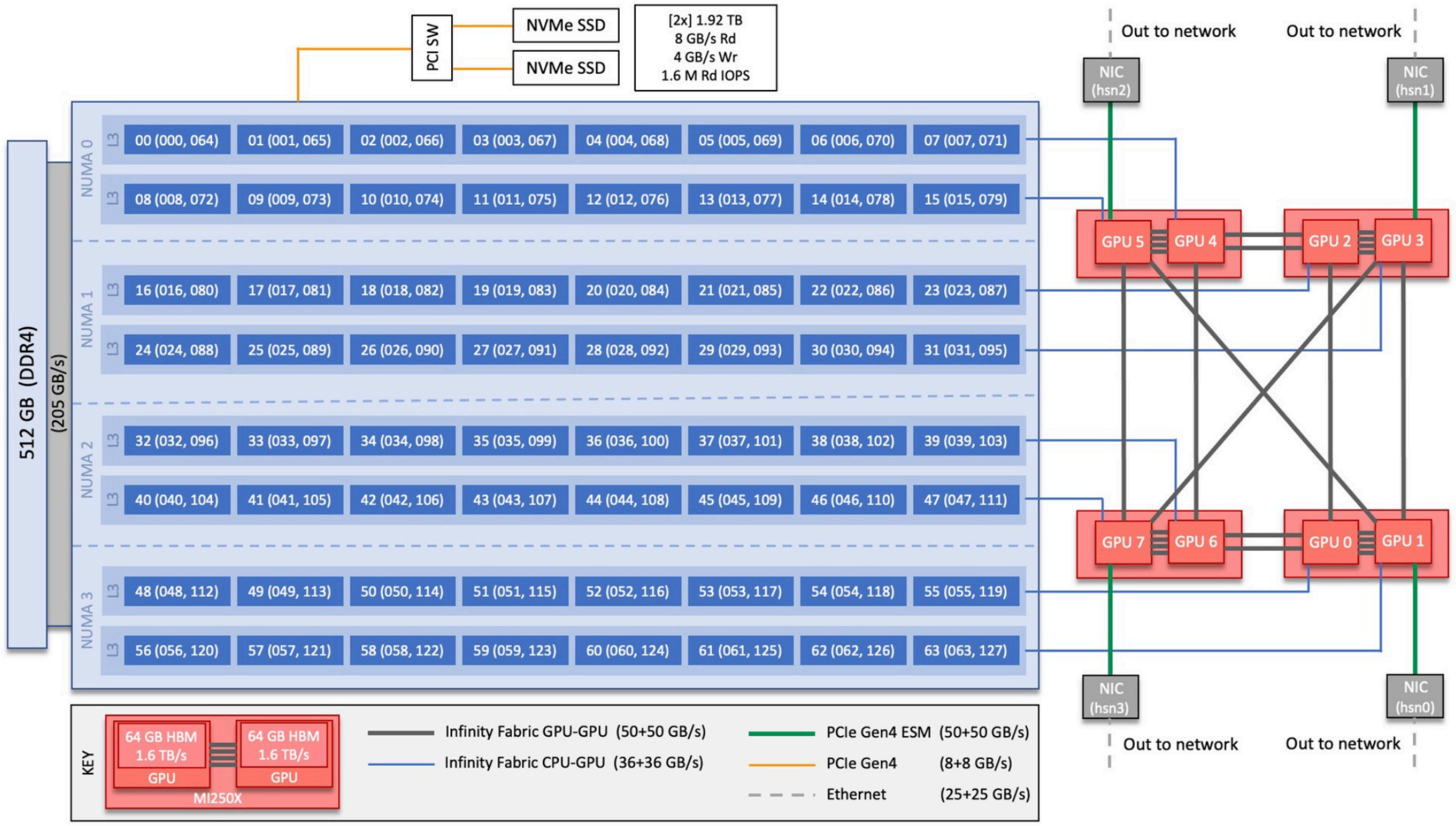


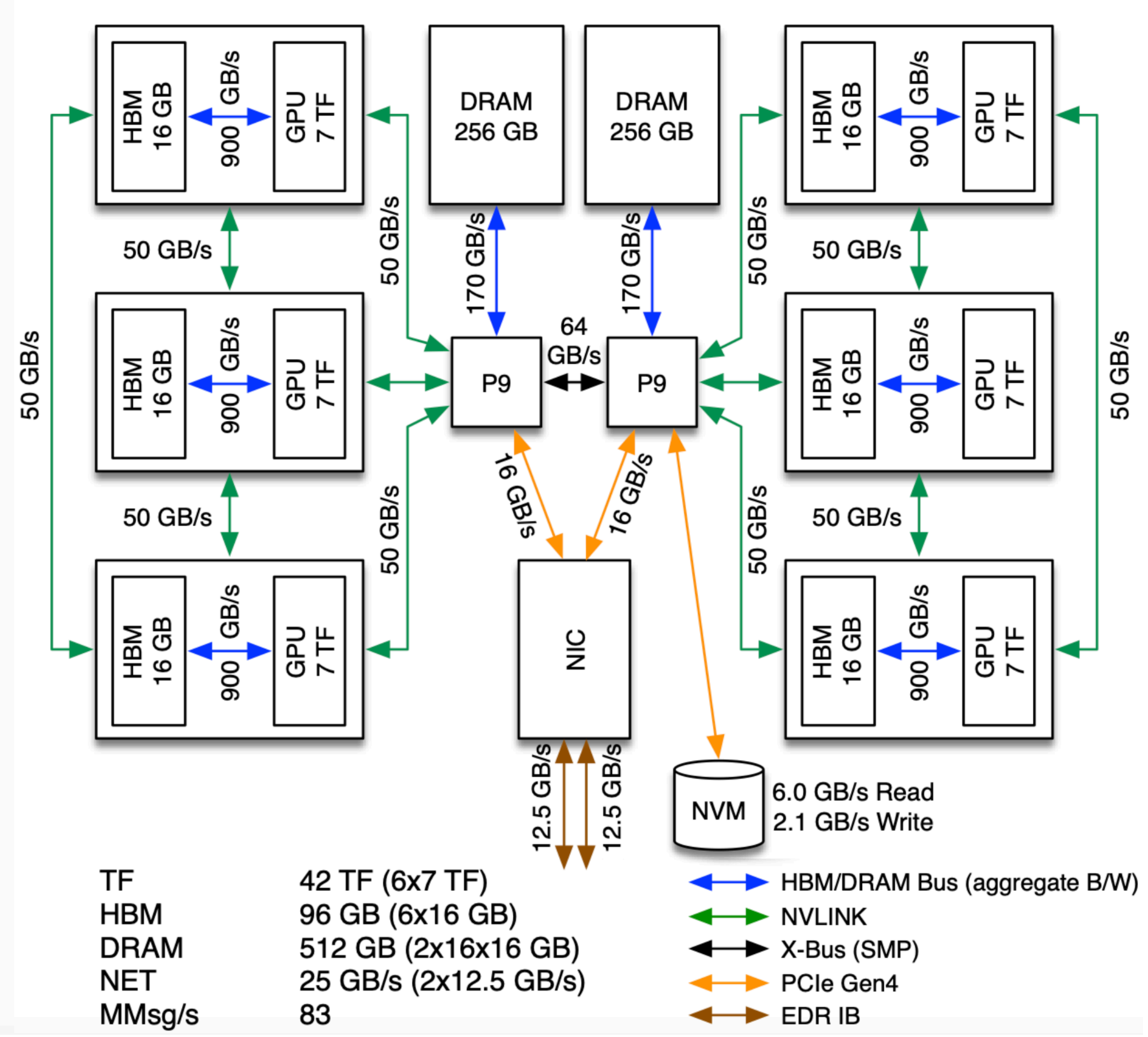


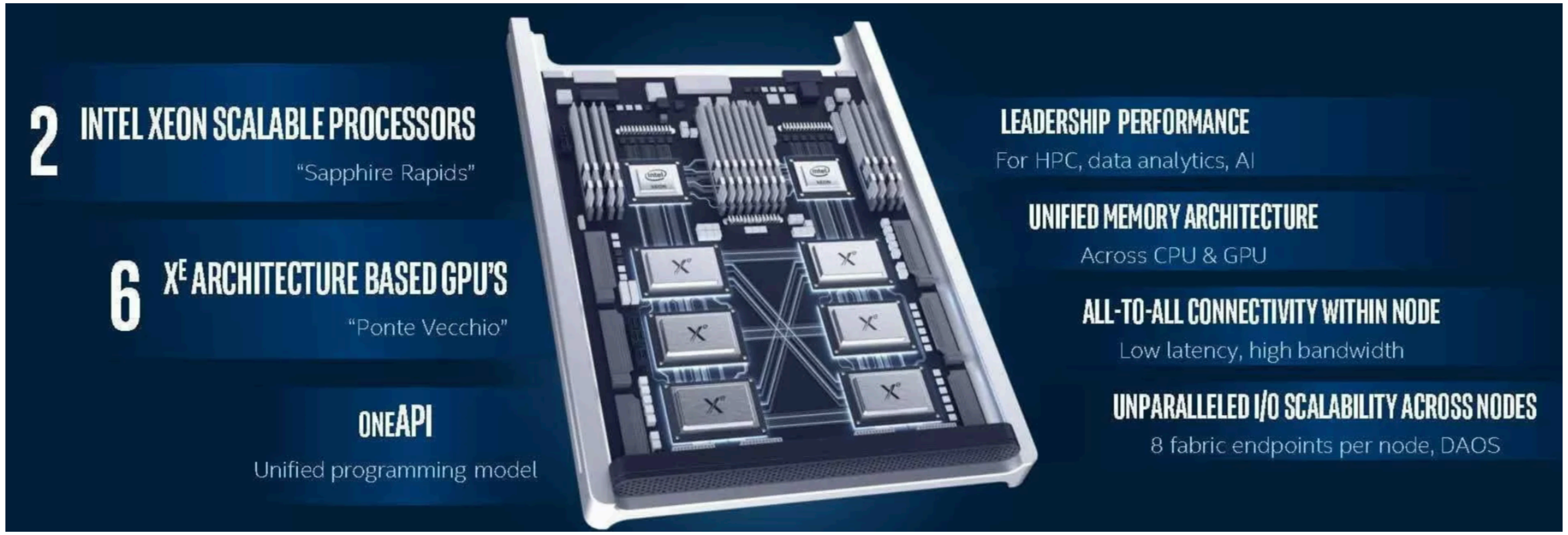
Requirement for increased concurrency to exploit performance and scaling potential of supercomputers forces programmers to design algorithms for their problems that expose instruction-level (ILP) and use thread-level parallelism (TLP)

- Problem dissection is tricky and not all problems are amenable
- Most the heavy lifting of our codes needs to execute efficiently on GPUs /accelerators
- What NP problems fit this picture? Would be useful to have a list …

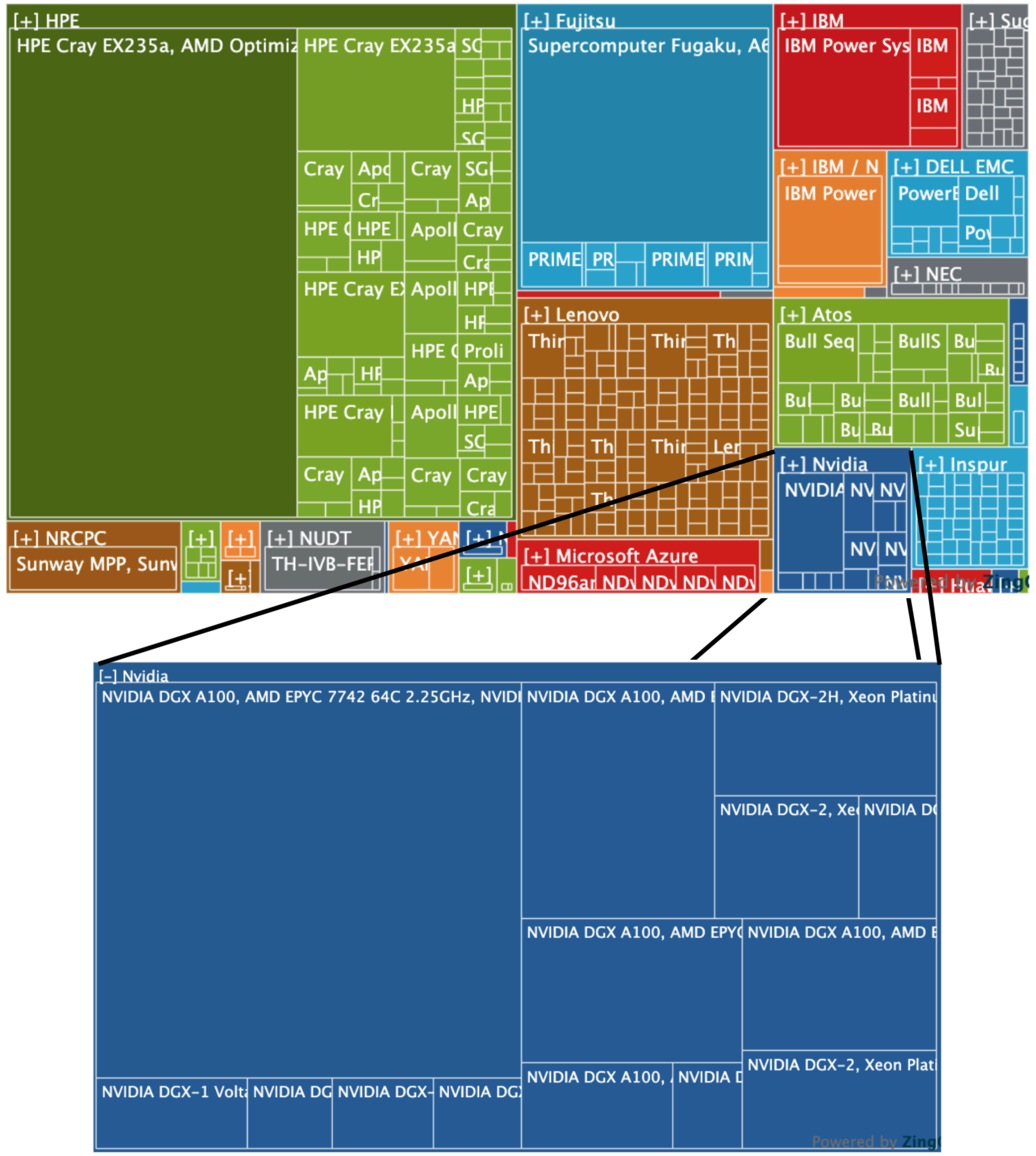
Aurora, Frontier, Summit : built on 3 GPU families by 3 Vendors with 3 Programming APIs







… designs look similar, but programmed differently … frustrating



NVIDIA: CUDA Compute Capabilities from 3.X to 9.X

Software

Aurora Software Introduction

- oneAPI
  - oneAPI Overview
  - Intel oneAPI Materials
    - Intel oneAPI DevCloud
    - Intel oneAPI Documentation
    - Intel oneAPI Programming Guide
    - Intel oneAPI Specification Site

Programming Models

- SYCL/DPC++
  - SYCL and DPC++ for Aurora
  - Related Training Materials
    - DPC++ Open Source Github
    - Data Parallel C++Book Chapters
    - A Roadmap for SYCL/DPC++ on Aurora
- OpenMP
  - OpenMP Programming Model
  - Related Training Materials
    - Overview of OpenMP 4.5 and 5.0 Features
- Kokkos
  - Kokkos
- RAJA
  - RAJA

Data Science and Workflows

- Related Training Materials
  - Machine Learning with TensorFlow, Horovod, and PyTorch on HPC
  - Effective Use of Python

Performance Tools

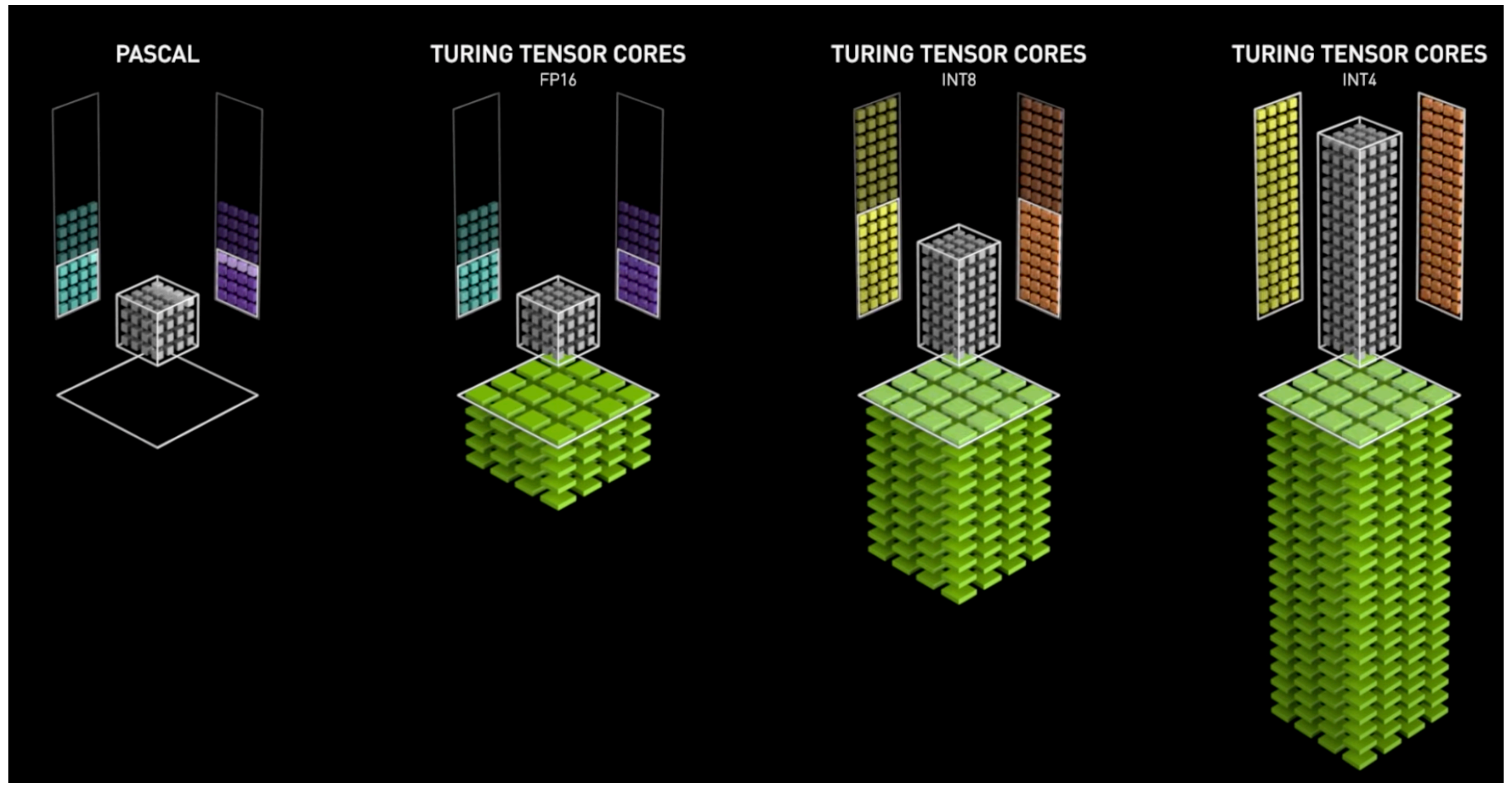
- Related Training Materials
  - Performance Tuning Using Intel Advisor and VTune Amplifier

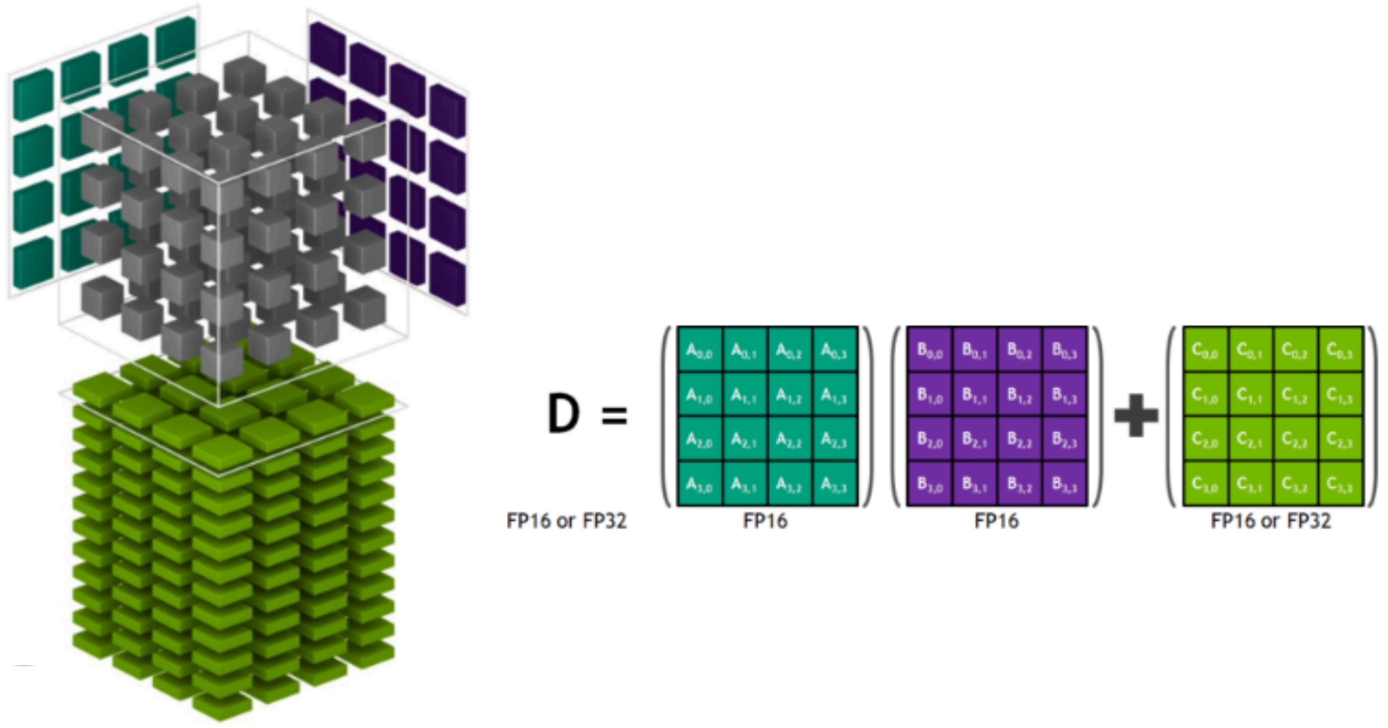
Massive programming efforts to refactor codes to target new exascale platforms

AMD	NVIDIA
Work-items or Threads	Threads
Workgroup	Block
Wavefront	Warp
Grid	Grid

Heterogeneous Interface for Portability (HIP) AMD's GPU programming environment

Disruptive special hardware



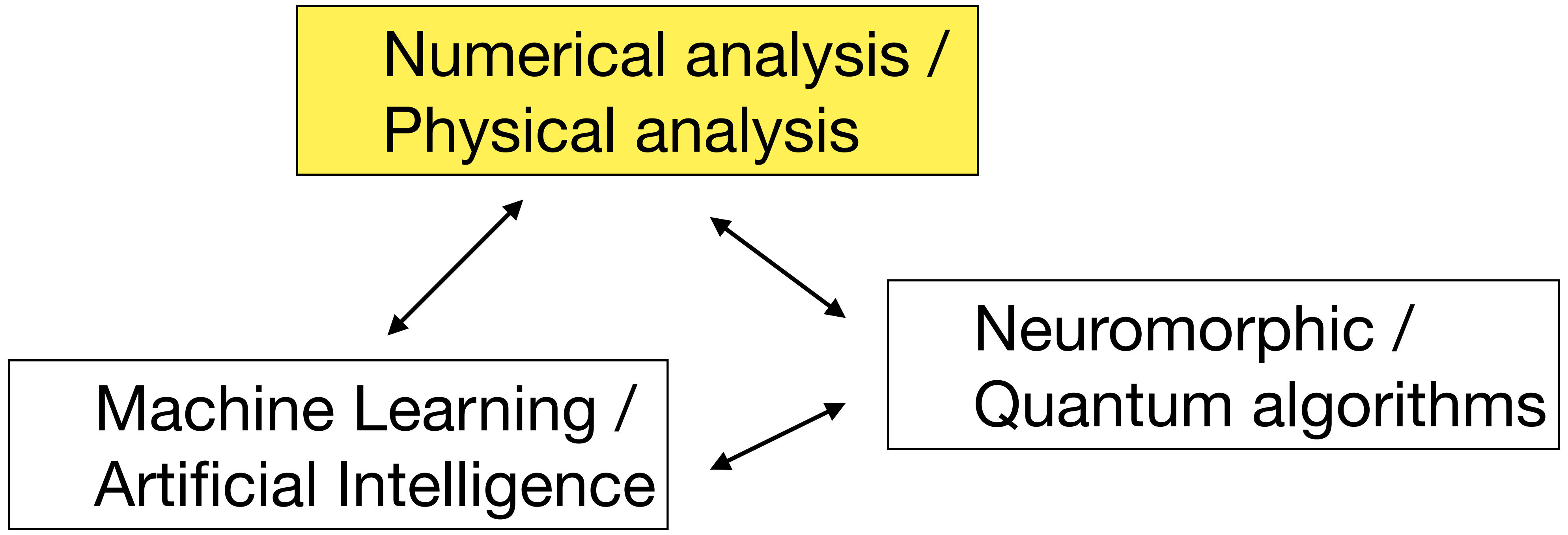


- i.e. NVIDIA Tensor Core AMD Matrix Core Unit technologies
- under-utilized until recently in simulations due to FP8 and FP16 constraints
- reduced bit and mixed precision
- exploit on-chip memory
- HPDA and DLNNs in particular can exploit this tech (made for it)
  - leverage model sparsity by spatially mapping the neural networks to computing tiles
  - remove fetch-decode-execute overheads through dataflow and/or systolic computation
  - FAST!

But how to align our current simulation algorithms with these units?

- need to research impact of reduced and mixed-precision computations on nuclear physics codes
- develop methods that can deliver high precision numerical evaluations from reduced-bit operations using physics

Embrace and Test Disruptive methods



- NP researchers need to investigate solving PDEs with machine learned techniques …
  - can these replace or improve quality AND performance of current approaches derived from operator theory and discrete numerical analysis?

- most downloaded paper in JCP for a long time introduces method that eschews numerical analysis in favor of combining the physical rules governing the PDE system with data and applying deep learning neural networks

  - PINNs have the massive advantage of industry DNN library stacks for immediate use

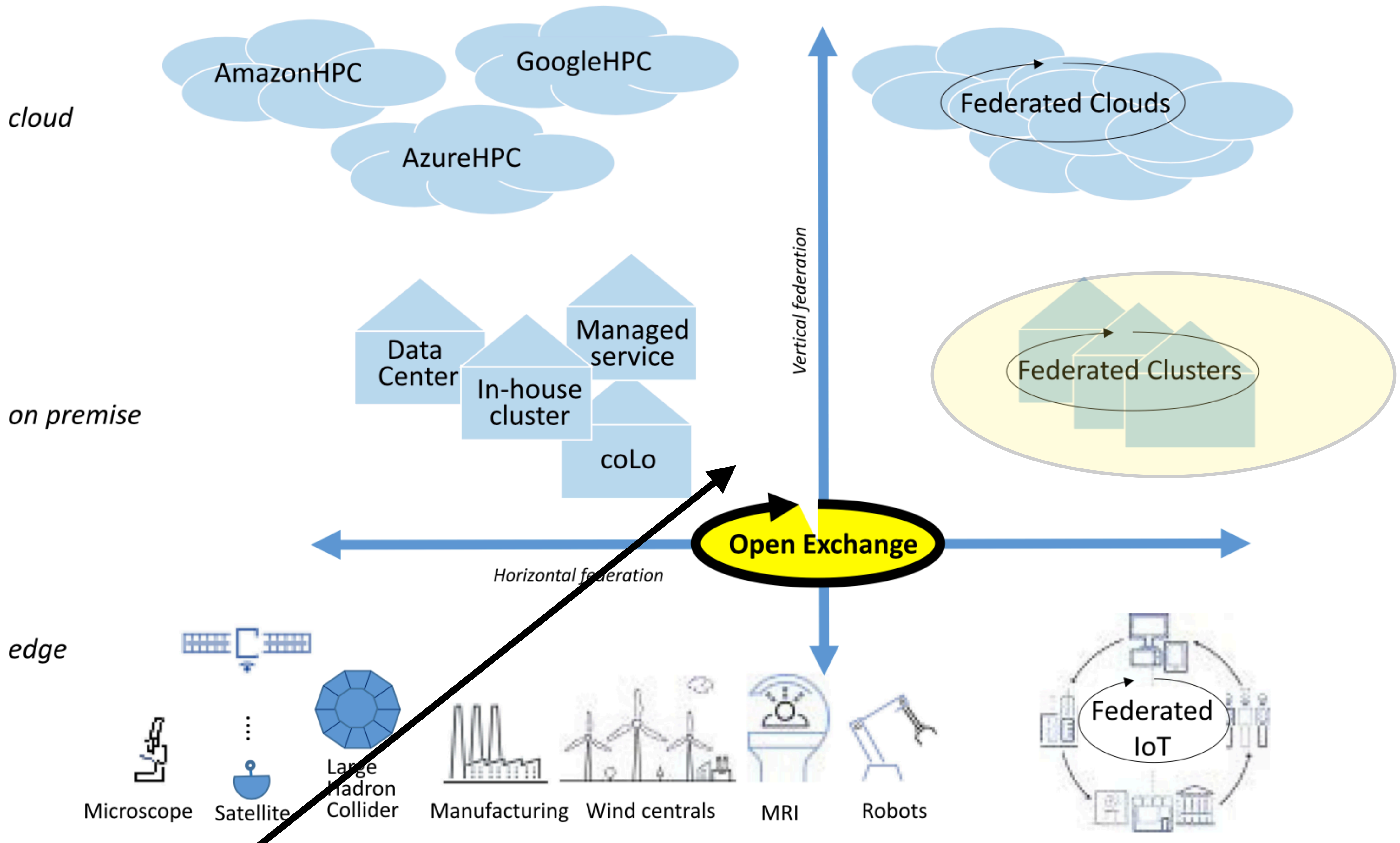
  - effectively utilize the tensor core technology previously mentioned

M. Raissi, P. Perdikaris, G.E. Karniadakis,
Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations,
Journal of Computational Physics,
Volume 378,
2019,
Pages 686-707,
ISSN 0021-9991,
https://doi.org/10.1016/j.jcp.2018.10.045.
(https://www.sciencedirect.com/science/article/pii/S0021999118307125)
Abstract: We introduce physics-informed neural networks – neural networks that are trained to solve supervised learning tasks while respecting any given laws of physics described by general nonlinear partial differential equations. In this work, we present our developments in the context of solving two main classes of problems: data-driven solution and data-driven discovery of partial differential equations. Depending on the nature and arrangement of the available data, we devise two distinct types of algorithms, namely continuous time and discrete time models. The first type of models forms a new family of data-efficient spatio-temporal function approximators, while the latter type allows the use of arbitrarily accurate implicit Runge–Kutta time stepping schemes with unlimited number of stages. The effectiveness of the proposed framework is demonstrated through a collection of classical problems in fluids, quantum mechanics, reaction–diffusion systems, and the propagation of nonlinear shallow-water waves.

- Workflows

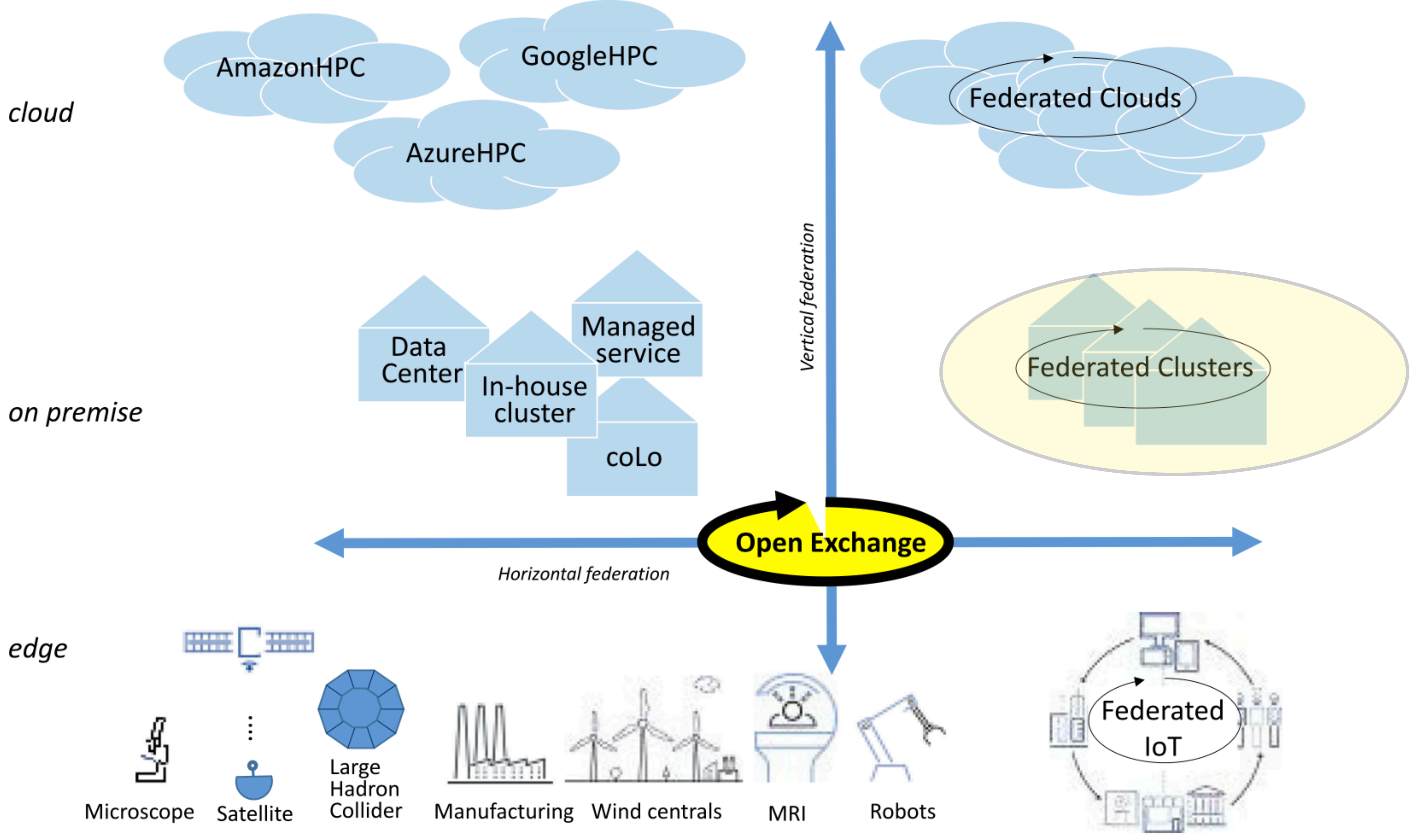
Workflows

- Facilities (beams, accelerators, supercomputers, etc.)
    - collect / generate / curate data
        - indexing and meta-data are critical
    - HPDA may happen elsewhere
        - data is usually moved, processed and transformed many times
        - enable access to the data to external users, edge devices, and clouds
- Developer productivity and performance portability are priority in heterogeneous workflows



Copied from N. Dube, et al, IEEE Internet Computing, vol. 25, no. 05, pp. 26-34, 2021

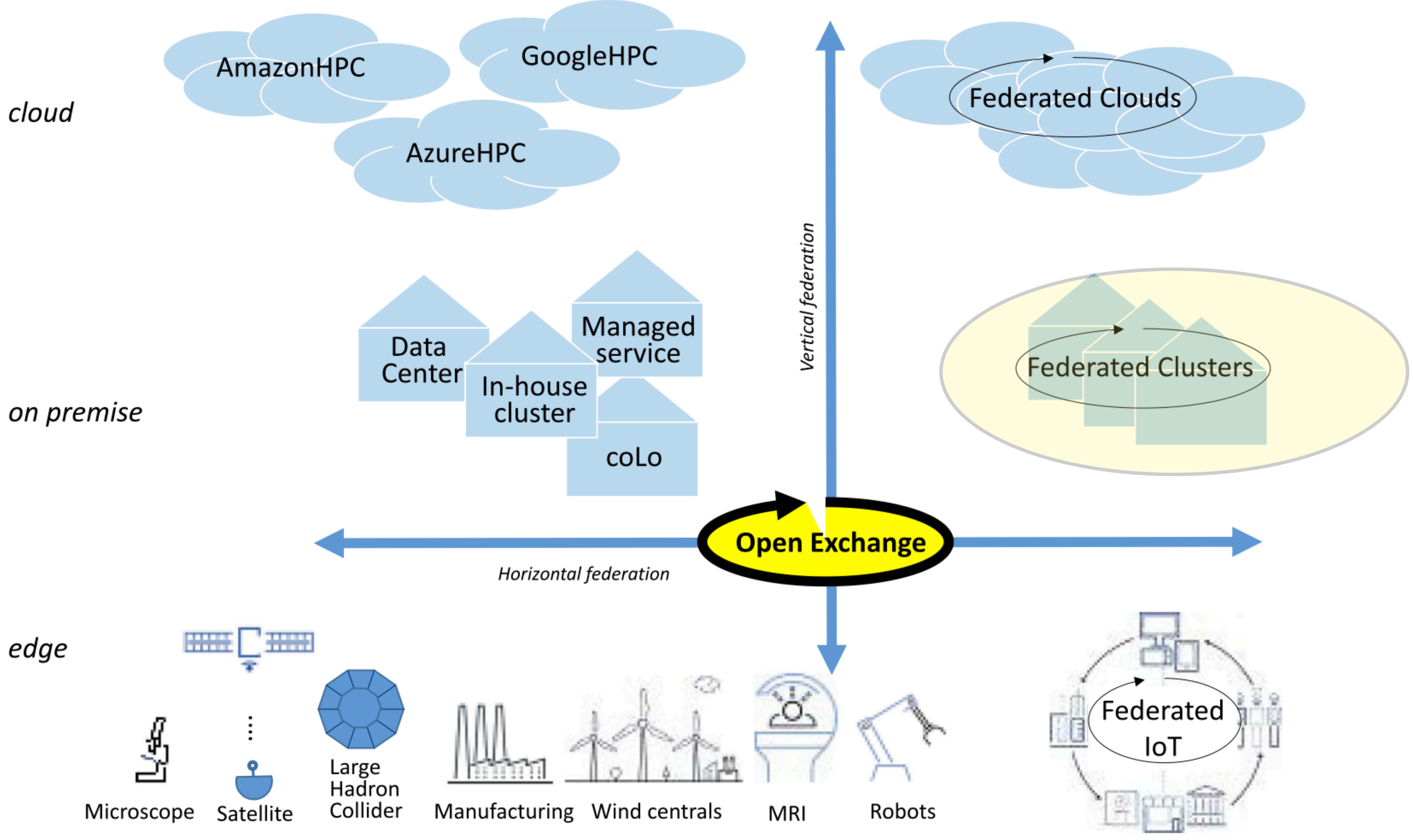
heterogeneous infrastructure



Copied from N. Dube, et al, IEEE Internet Computing, vol. 25, no. 05, pp. 26-34, 2021

Decentralized storage architecture

- note: data held in a cloud service provider is tied to the CSP and their cost model
- need a secure, reliable, and economical store for the long term -maybe one for NP even
- should commit to data format standards to facilitate data exchange -too many hacked formats in use



Copied from N. Dube, et al, IEEE Internet Computing, vol. 25, no. 05, pp. 26-34, 2021

Compose nuclear workflows

- use existing workloads
- use data deployed across multiple organizations
- benefits users, developers, and resources providers
- improves efficiency, portability, productivity, resource utilization

A word of warning:

- Workflows will succumb to industry software stacks

- NSA just issued a warning you should read:
    - https://media.defense.gov/2022/Nov/10/2003112742/-1/-1/0/CSI_SOFTWARE_MEMORY_SAFETY.PDF

- Based largely on OSS - Open Source Software
    - See https://cpb-us-w2.wpmucdn.com/sites.gatech.edu/dist/a/2878/files/2022/10/OSSI-Final-Report.pdf
    - punchline: transition to Memory-Safe Programming Languages
        - focus is on increasing adoption in OSS
        - because ~ 70% of all software vulnerabilities continue to be memory safety problems that arise from the use of memory unsafe languages such as C or C++
    - (gradually) encourage the transition software developers to use memory safe languages

- Enable the democratization of software development, rapid evolution, de-duplication of effort on an unprecedented scale, and broad transparency
    - (on the other hand) diffuse structure and large scale make it infeasible to specify or enforce minimum standards for tools and development practices. Combined with the large volume of already-written (legacy) OSS code, this poses unique challenges when it comes to security

- Suggests use of Rust
    - See https://www.rust-lang.org/

- aside on typical candidate NP problems

- NP researchers are leaders at mapping problems to needed computer resources (re)Assess status of nuclear physics computing challenge examples

- did we make it?

nuclear forces / cold QCD

- calculations of the spectrum and properties of excited states of mesons (ie w/ gluonic DOFs), GlueX experiment at JLab
- origin of mass, spin, charge, currents in protons from QCD
- nucleon interaction from QCD
- calculations of parity-violating nuclear forces, time-reversal violating observables from weak and strong interactions from QCD
- finite volume EFTs matched to LQCD to interface nuclear structure calculations

estimates from previous workshops

- >20PF years
  - baryon-baryon, meson-baryon interactions, spectrum of excited nucleons
- >100PF years
  - nucleon transition form factors, spectrum and photo-couplings of iso vector mesons, axial charge of nucleon, nucleon form factors, axial charge of deuteron and electroweak interactions
- >1EF year
  - spectrum of mesons, gluon contribution to hadron structure, nnn, spectrum of alpha, parity-violating nuclear force

(re)Assess status of nuclear physics computing challenge examples

- did we make it?

Reactions
- GFMC (Green's function MC), NCSM (no core shell model)
  - light nuclei from realistic n-n interactions, ab-initio reactions; 12C on 30K cores for 24 hours
  - calculate examples where precision data exists at NNSA labs, tune the 3-nucleon interaction
  - 1000X more difficult ~ those w/ weakly bound initial or final states

Fission (dynamics, cross-sections, fragment properties, prompt neutrons & gammas)
- constrained HFB (Hartree-Fock-Bogolyubov)
  - adiabatic computation of PES in space of collective coordinates -dynamics requires evaluation of inertia tensor in p-h and p-p channels
- ATDHFB
  - search for optimum collective that minimizes collective action in space defined by at least elongation, mass asymmetry, necking, triaxiality DOFs, evaluate the penetration probability by integrating action along this path; 20-30 CPU years for analysis of a small number of isotopes
- stochastic extension of TD DFT
  - estimates from previous workshops
    - >20PF years
      - ATDHFB description of fission, partial implementation of stochastic TDSLDA, static properties of neutron star crust
    - >100PF years
      - fission in hot nuclei, full stochastic TDSLDA
    - >1EF year
      - fission for odd nuclei 235U, dynamic properties of neutron start crust

In short, NP Problems definitely lead to massive computer simulations !!!

Example(shameless): Time-dependent Superfluid Local Density Approximation (TDSLDA) applied to cold atoms, neutron stars and nuclei

238U in a 50^2 x 90 fm^3 volume
16,760 GPUs - 4190 Sierra nodes
10 wall hours
evolve a system of 3,600,000 TDPDEs 37,695 time steps
3x10^{-5} relative accuracy

238U in a 48^2 x 120 fm^3 lattice volume
27,648 GPUs - 4608 Summit nodes
10 hours
conserving system energy to less than 100 KeV
particle numbers to better than 10^{-6}

**from NUMERICALLY SOLVING
50,000 to 5,000,000 PDEs coupled
3D+1 nonlinear PDEs

This is impressive, but these codes ignore multiple important artifacts of both the programming capability supported in CUDA and the NVIDIA hardware stack

NEED sustained funding for code development, maintenance, refactoring, and optimization!

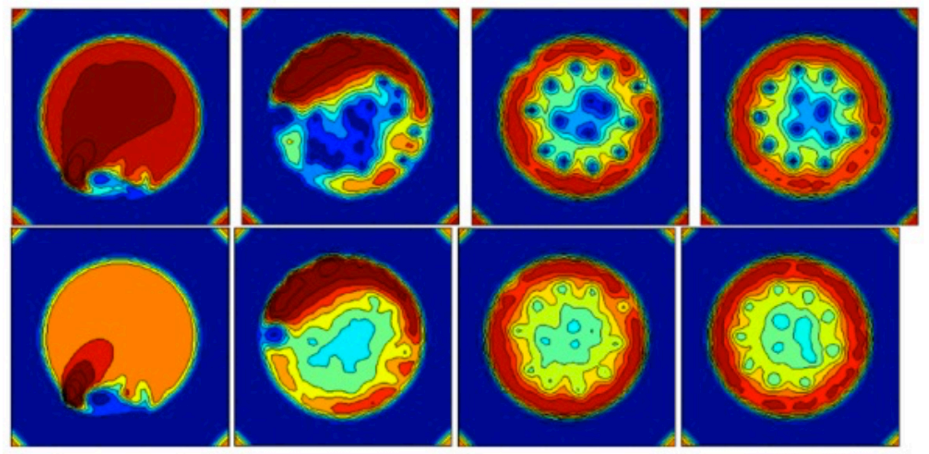
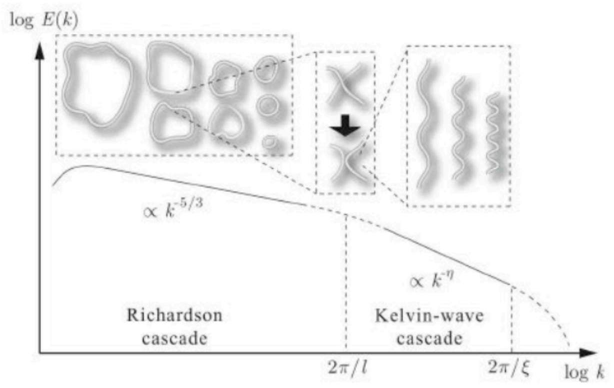
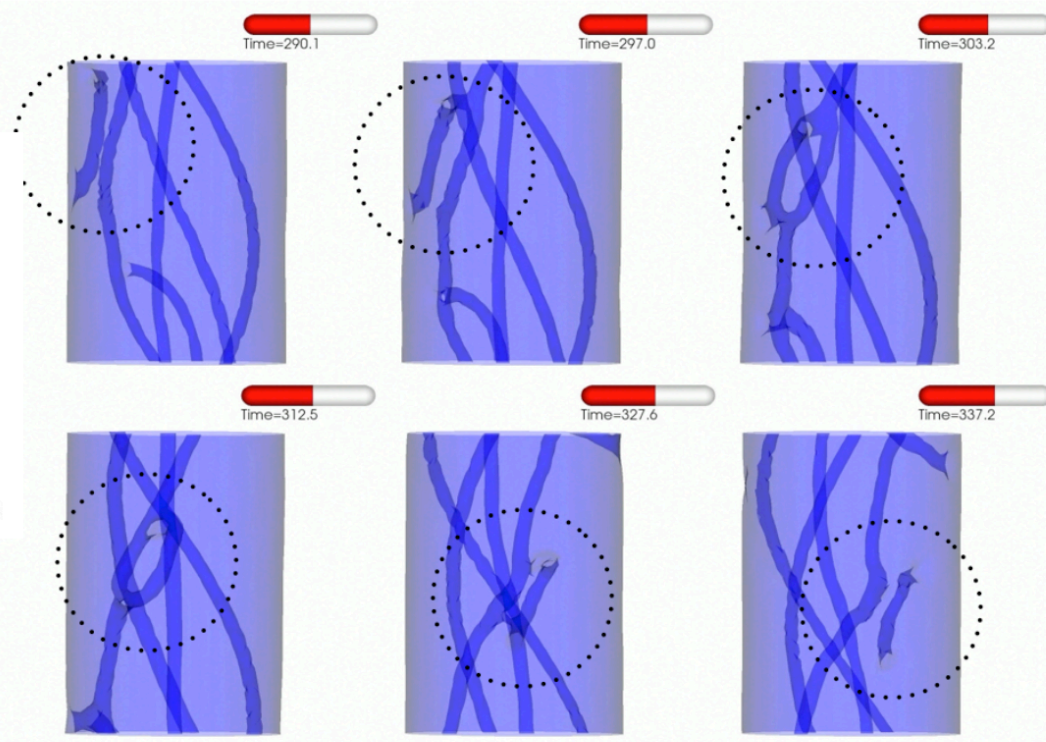
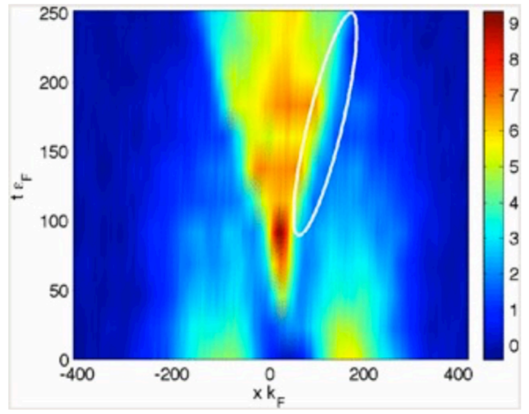


Figure 2: The magnitude of the pairing field (Δ, top row) and the corresponding density (n), bottom row) for a UFG system composed of 1800 particles in a 48^3 lattice stirred at supercritical velocity $1.216 v_c$. Here thirteen vortices are formed once the stirring concludes.



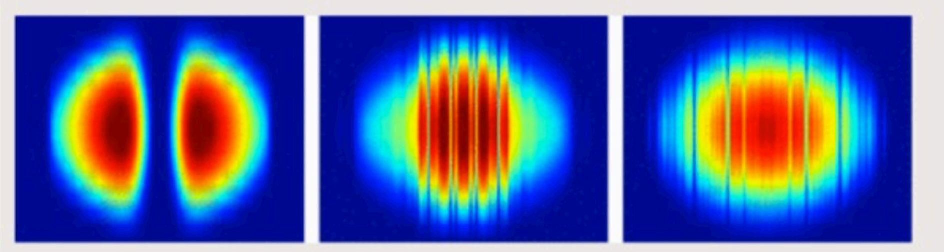


Quantum Shock Waves / Domain Walls w/in TDSLDA

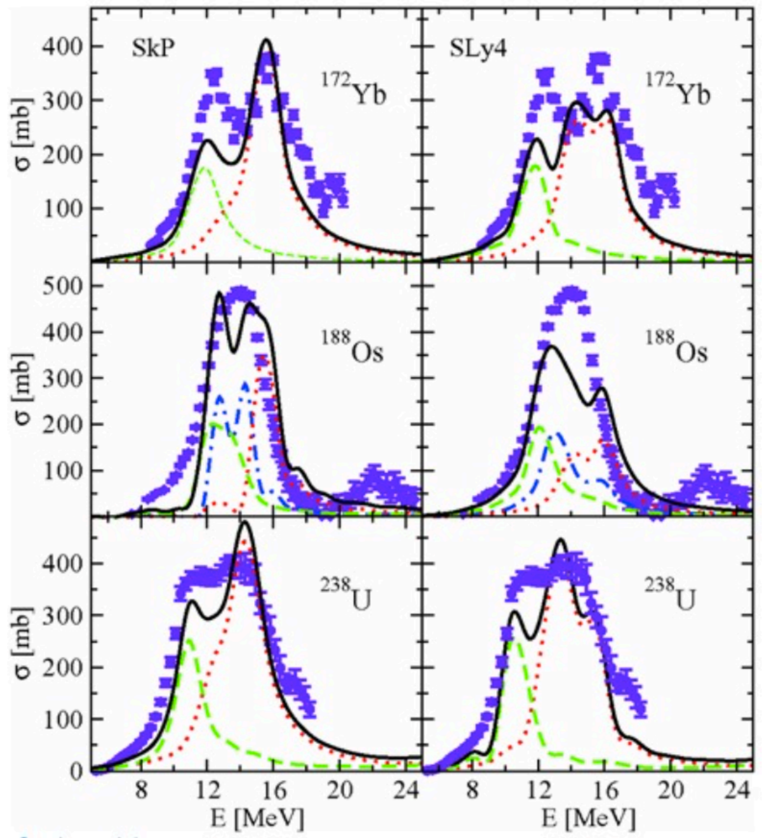


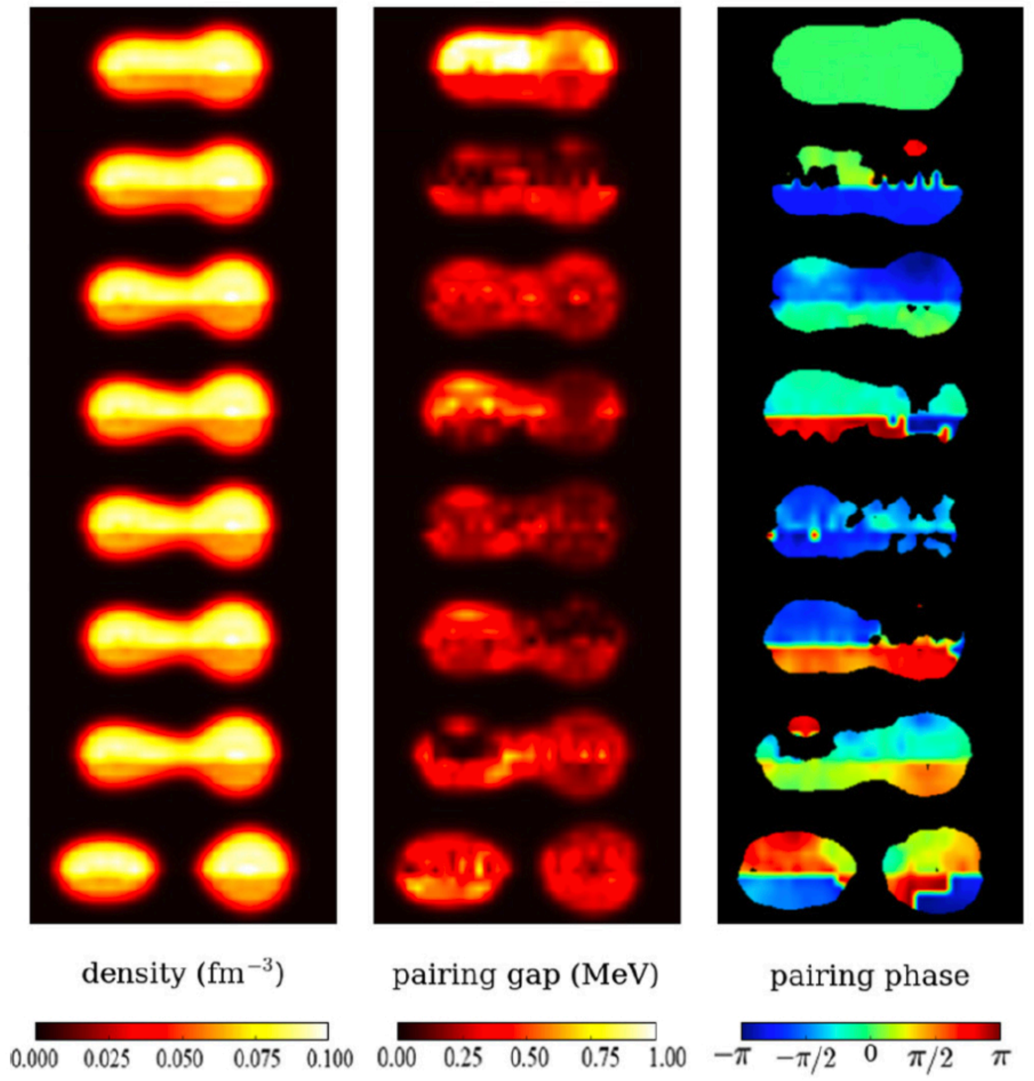
J. A. Joseph, J. E. Thomas, M. Kulkarni, and A. G. Abanov, Phys. Rev. Lett. 106, 150401

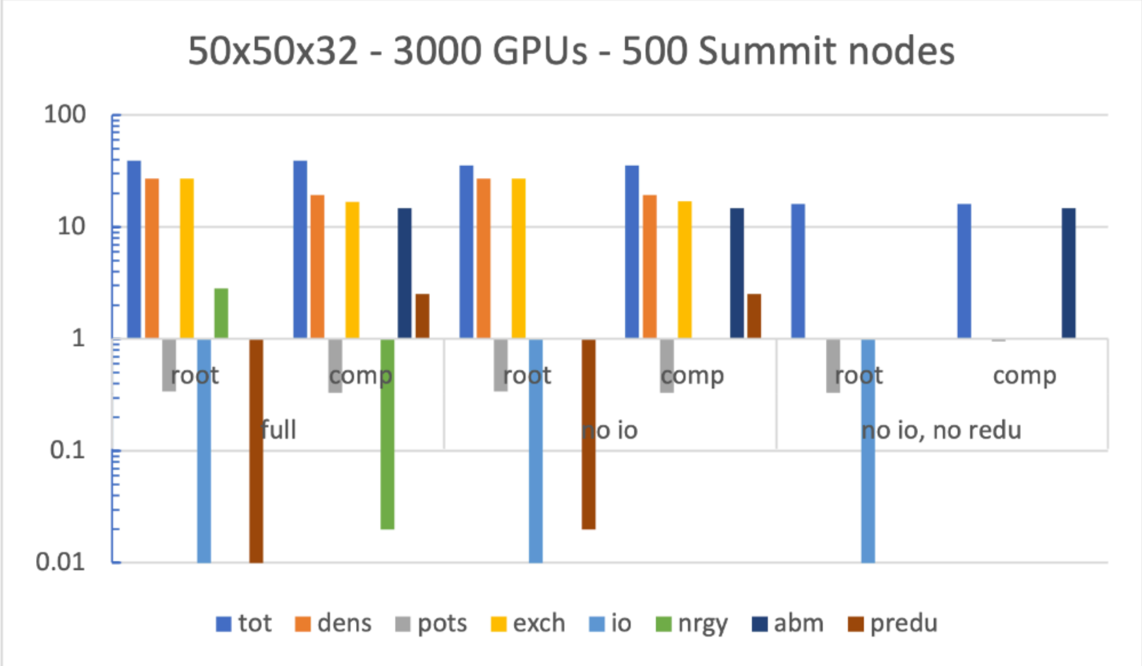
Phys Rev Lett, 108, 150401 (2012) , A. Bulgac, Y.-L. Luo, and K.J. Roche

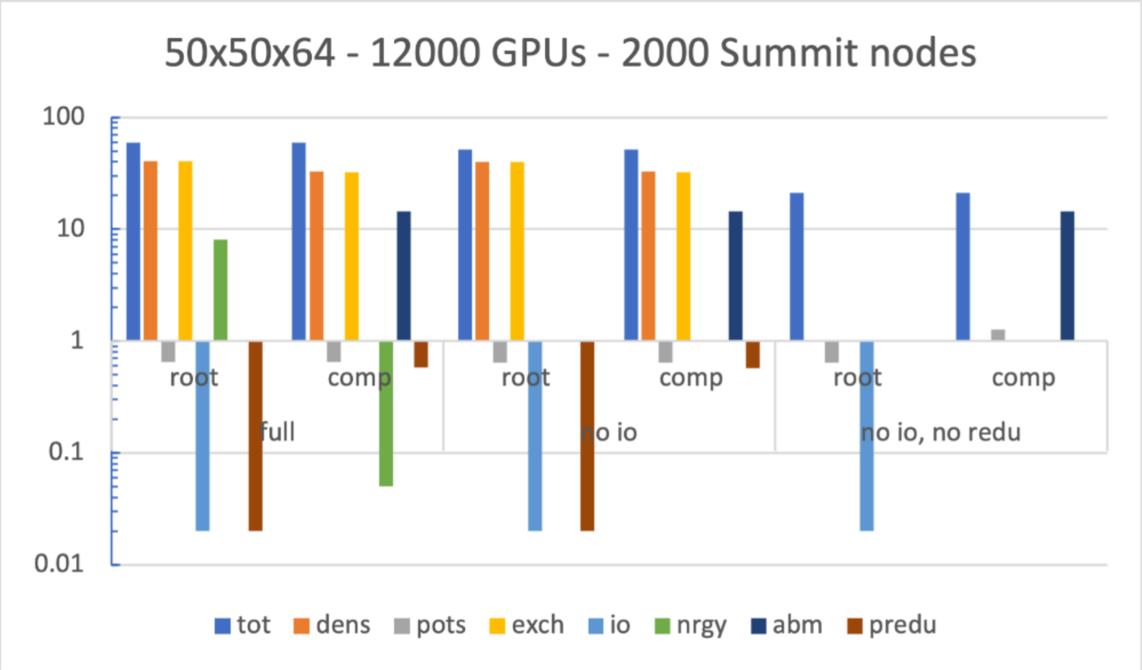


Magnitude of pairing field at t ef ~ 30, 350, 690. DWs appear as planar number density depletions with a width comparable to the diameter of a quantum vortex

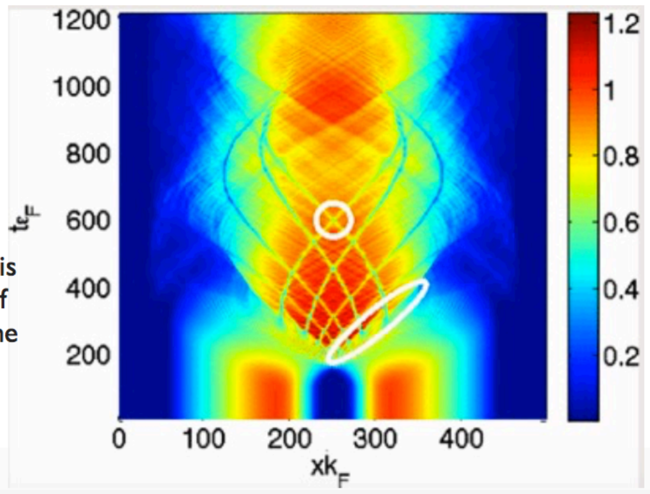




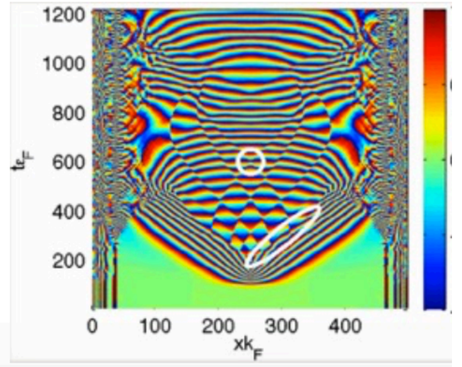


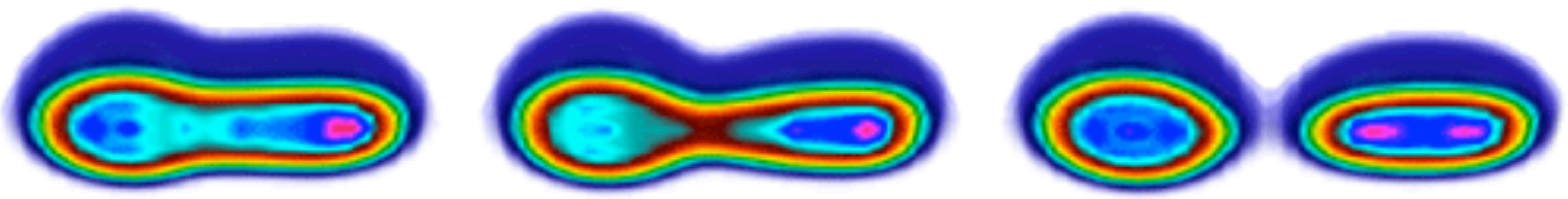


n of the number
ong the collision axis
n of the collision of
f and .5 kf ^2 are the
i wave vector and
clouds. The region
on of two DWs
white circle.

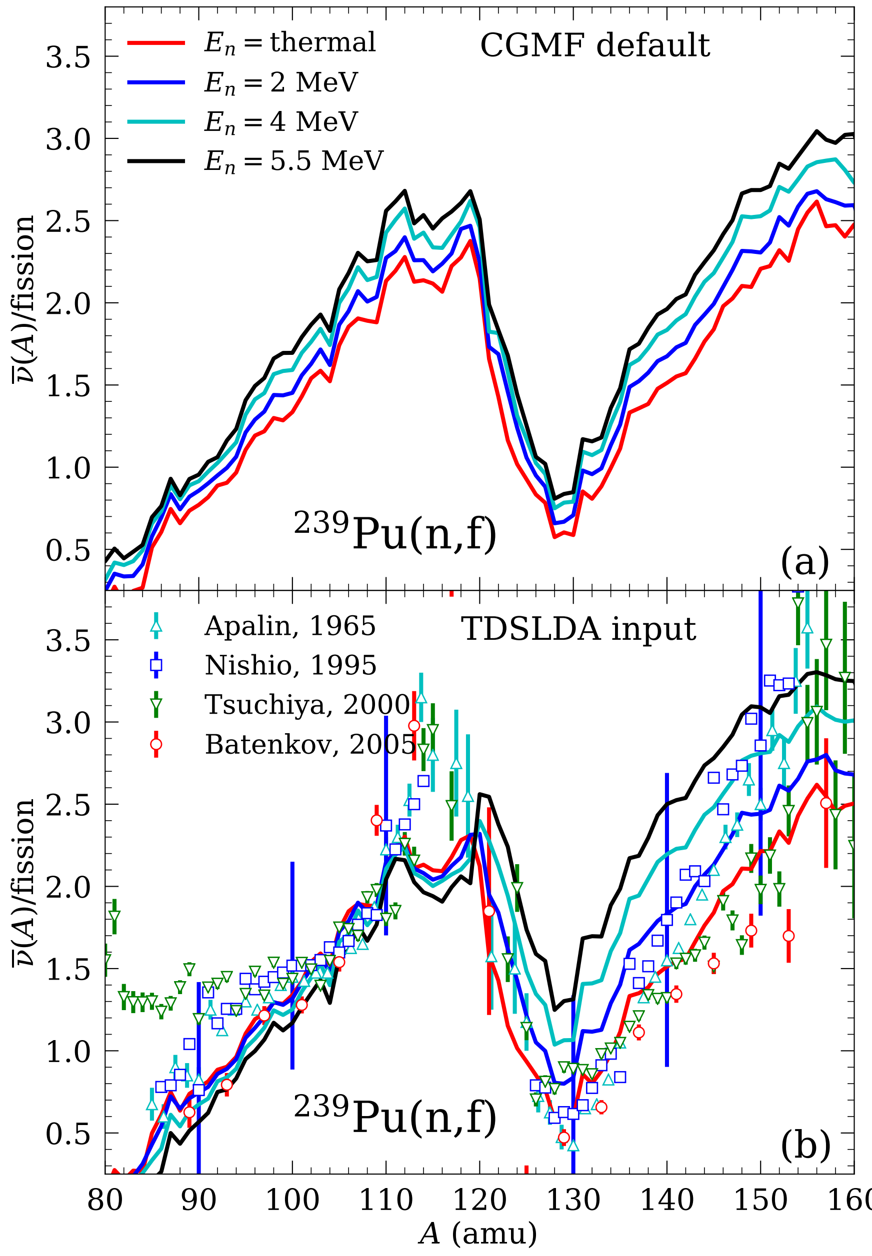


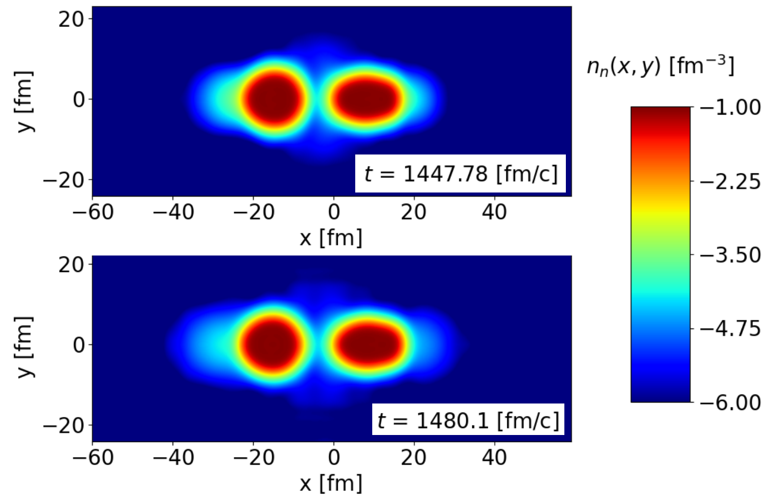
phase of pairing gap

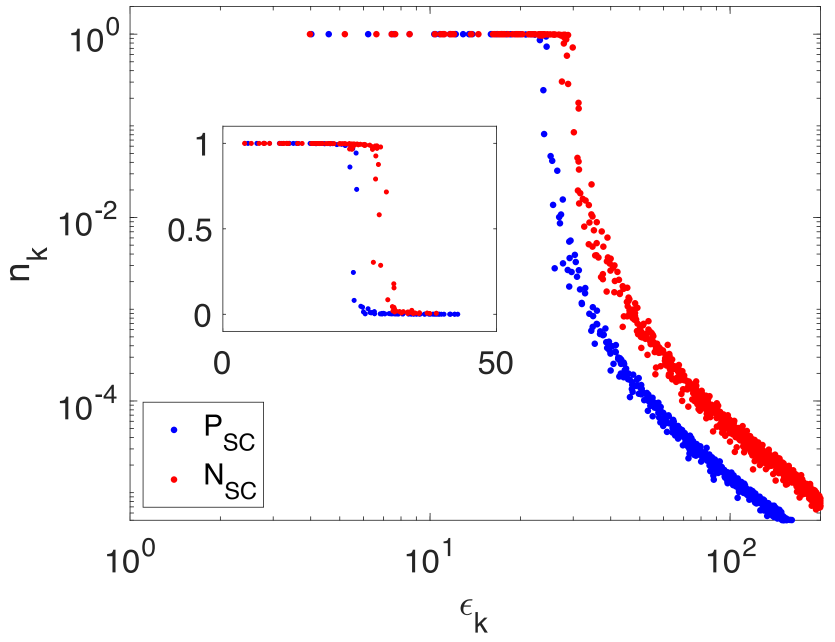


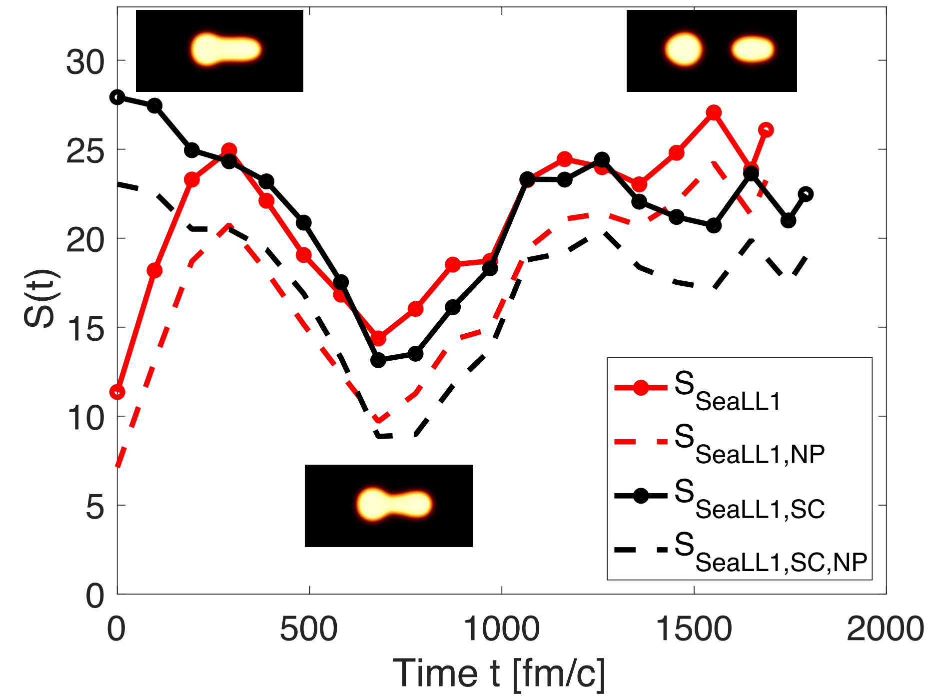


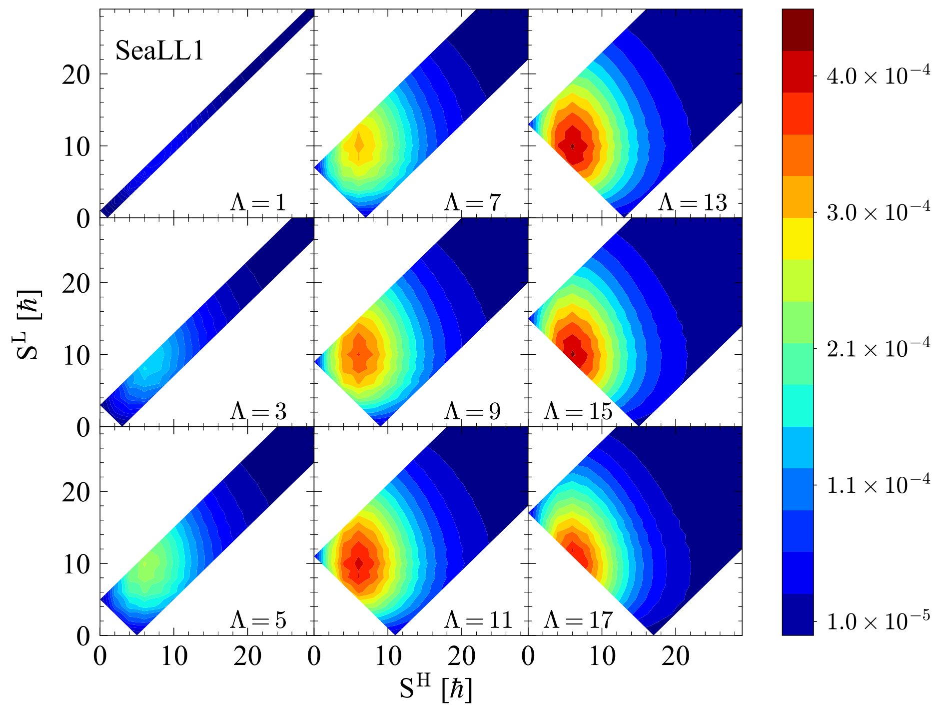
- Large Amplitude Dynamics of the Pairing Correlations in a Unitary Fermi Gas, Phys. Rev. Lett. 102, 085302 (2009)
- Induced P-wave superfluidity within full energy-momentum dependent Eliashberg approximation in asymmetric dilute Fermi gases, Phys. Rev. A 79, 053625 (2009)
- Real-Time Dynamics of Quantized Vortices in a Unitary Fermi Superfluid, Science, 332, 1288 (2011)
- Quantum Shock Waves and Domain Walls in Real-Time Dynamics of a Superfluid Unitary Fermi Gas, Phys. Rev. Lett. 108, 150401 (2012)
- Isovector Giant Dipole Resonance from 3D Time-Dependent Density Functional Theory for Superfluid Nuclei, Phys. Rev. C 84, 051309(R)
- Quantized Superfluid vortex ring in the unitary Fermi gas, Phys. Rev. Lett. 112, 025301
- Auxiliary-Field Quantum Monte Carlo Simulations of Neutron Matter in Chiral Effective Field Theory, Phys. Rev. Lett. 113, 182503 (2014)
- Relativistic Coulomb excitation within Time Dependent Superfluid Local Density Approximation, Phys. Rev. Lett. 114, 012701 (2015)
- Life cycle of superfluid vertices and quantum turbulence in a unitary Fermi gas, Phys. Rev. A 91, 063602(R)
- Vortex pinning and dynamics in the neutron star crust, Phys. Rev. Lett. 117, 232701 (2016)
- Induced fission of 240Pu within a real-time microscopic framework, Phys. Rev. Lett. 116, 122504
- Dynamics of Fragmented Condensates and Macroscopic Entanglement, Phys. Rev. Lett. 119, 052501 (2017)
- Fission Dynamics of 240Pu from saddle to scission and beyond, Phys. Rev. C 100, 034615 (2019)
- Nuclear Fission Dynamics: Past, Present, Needs, and Future, Frontiers in Physics, 8, 63
- Emergence of a pseudogap in the BCS-BEC crossover, Phys. Rev. Lett. 125, 060403 (2020)
- Fission fragments intrinsic spins and their correlations, Phys. Rev. Lett. 126, 142502 (2021)
- Fragment Intrinsic Spins and Fragments' Relative Orbital Angular Momentum in Nuclear Fission, Phys. Rev. Lett. 128, 022501 (2022)
- Pure quantum extension of the semiclassical Boltzmann-Uehling-Uhlenbeck equation, Phys. Rev. C, 105, L021601 (2022)

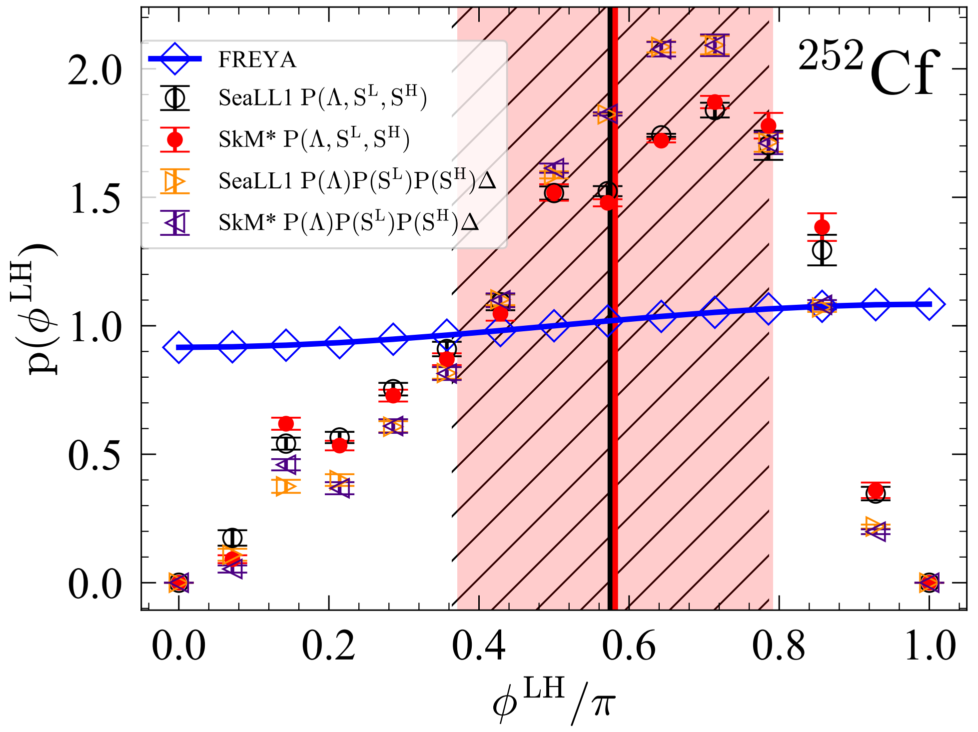


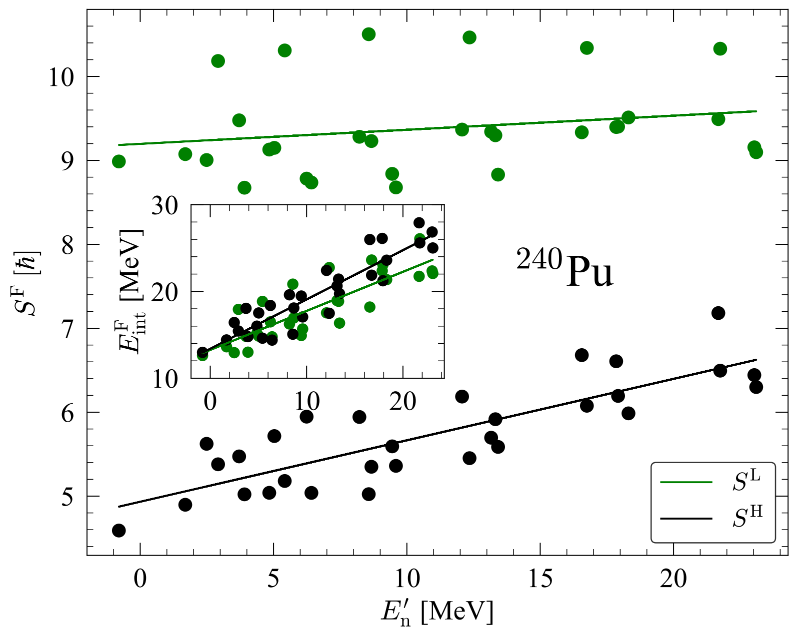












Kenneth J. Roche, Pacific Northwest National Laboratory, HPC group, University of Washington, Department of Physics

- *Are neuromorphic systems the future of high-performance computing?, MATHEMATICS AND COMPUTATION Blog, Physics World, IOP, March 2022;* *https://physicsworld.com/a/are-neuromorphic-systems-the-future-of-high-performance-computing/*

- *https://www.nsa.gov/Press-Room/News-Highlights/Article/Article/3215760/nsa-releases-guidance-on-how-to-protect-against-software-memory-safety-issues/*

- *https://cpb-us-w2.wpmucdn.com/sites.gatech.edu/dist/a/2878/files/2022/10/OSSI-Final-Report.pdf*

- *http://www.itrs2.net/*

- *https://www.top500.org/*

- *N. Dube, D. Roweth, P. Faraboschi and D. Milojicic, "Future of HPC: The Internet of Workflows" in IEEE Internet Computing, vol. 25, no. 05, pp. 26-34, 2021.*

NULL

- Solve math and physics problems that require (BIG) computer evaluations
  - Many-body stationary and time-dependent problems w/ Bulgac et al
  - 2 time INCITE award recipient for NT computing

- Led national development, benchmarking, optimization, and code porting efforts for US DOE SC
  - Exascale Computing Project (ECP) Application Assessment
    - Performance
      - FOM
      - Efficiency
    - Scaling
      - Weak (FOM)
      - Strong (efficiency)
    - Portability
      - Developed for System X, Demonstrated on System Y != X
    - Software Quality
      - Engineering practices
      - Open Source Software (OSS)
  - Joule Software Effectiveness metric (w/ D. Kothe)
    - Q2 performance baseline
    - Q4 demonstrate efficiency / scaling enhancements