ATPESC 2026

Argonne NATIONAL LABORATORY

Future of Extreme-Scale HPC for Science

TIMOTHY J. WILLIAMS
Deputy Director
Computational Science Division (CPS)
Argonne National Laboratory

7 August 2026

PART 1 – INTRO

Scope of talk

A five-year forecast: how the practice of extreme-scale HPC for science evolves by 2031

1. Intro
2. What calculations will need extreme-scale HPC?
3. Post-exascale systems
4. Users: AI, Agents, and Humans
5. What’s a human to do?

Why This Moment is Pivotal

Exascale reality

Science & engineering is growing into Aurora, Frontier, El Capitan.

AI acceleration

Foundation models, surrogates, inference services, copilots, and agents are entering scientific workflows.

Data intensity

Instruments, simulations, archives, literature, and models are becoming a coupled research system.

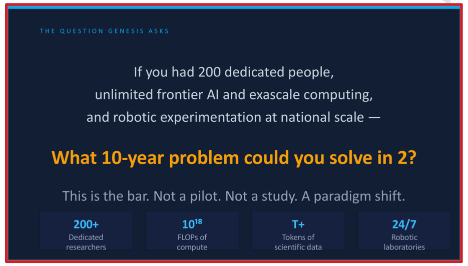
APS-U

Complexity

Heterogeneous hardware, memory hierarchy, workflow orchestration, and software stacks are difficult to manage manually.

Scientific urgency

Climate, energy, materials, health, fusion, and fundamental science need faster cycles from question to credible answer.



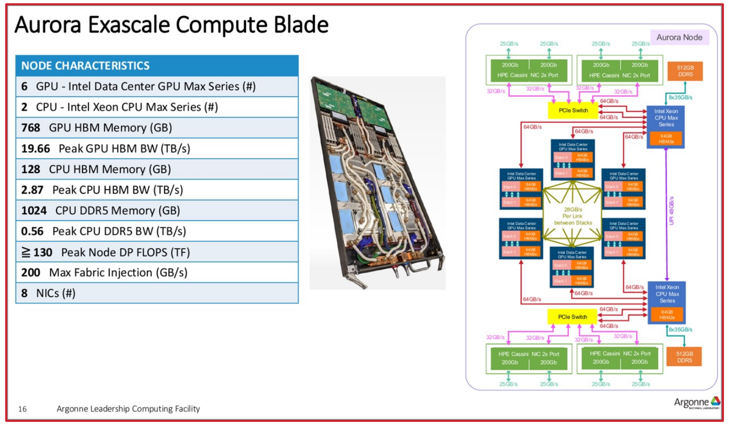
Genesis Mission

"...frontier AI + exascale -- What 10-year problem could you solve in 2?" -- Foster, "Genesis Mission," ATPESC 2026.

Argonne NATIONAL LABORATORY

ATPESC has already sketched the shape of future practice

- Hardware, GPUs, OpenMP, MPI, frameworks, and tools:
  - science still meets silicon through code
- Visualization, I/O, data management, and workflows:
  - discovery is data-intensive end to end
- Numerical software and mixed precision:
  - algorithms still optimize what science the machine delivers
- AI/ML, LLMs, inference, agentic tools, and vibe coding:
  - discovery loops are becoming partially automated
- Software sustainability:
  - long-lived software is still possible, with some new requirements
- Quantum:
  - earliest future use likely hybrid classical-quantum



Aurora Exascale Compute Blade (6 GPUs, HBM, 25 GB/s links) -- Videau & Muralidharan, "Evolution of DOE Exascale Supercomputers," ATPESC 2026.

PART 2 – WHAT CALCULATIONS WILL NEED EXTREME-SCALE HPC?

High-Fidelity Simulation

Simulation:

- Imposes known physics and conservation structure
- Explores inaccessible, dangerous, rare, or expensive regimes
- Generates synthetic data for AI models and reduced-order models
- Validates or falsifies AI-generated results/hypotheses
- Anchors uncertainty estimates

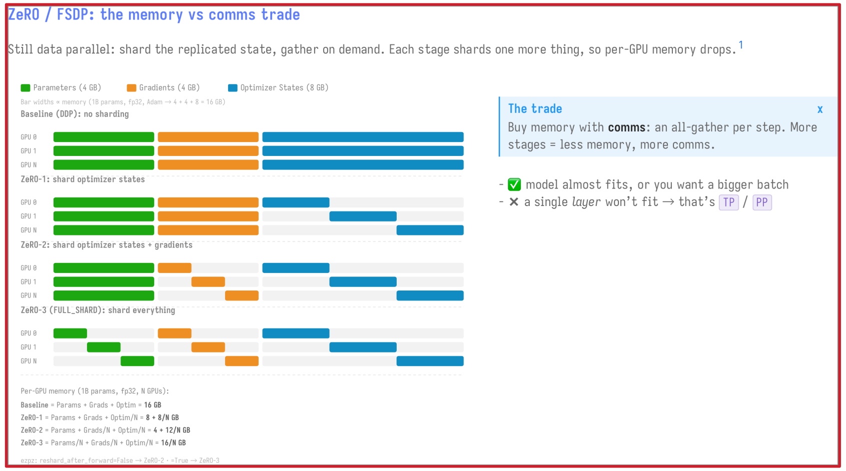
Scientific AI model training

The volume of data and/or size of model determines need for extreme-scale

- Domain foundation models for biology, materials, fusion, ...
  - Multi-modal trained on simulation/experimental data, text, code, etc.
  - Model may be large or small
- Neural operators and surrogates
  - Trained in situ in extreme-scale simulation, e.g.
- Generative models for molecules, materials, etc.

Single Inference is not extreme-scale

- Node-level (even trillion parameters)
- GPU-level (surrogate model for fine scale)
- Edge-computed (experimental control)

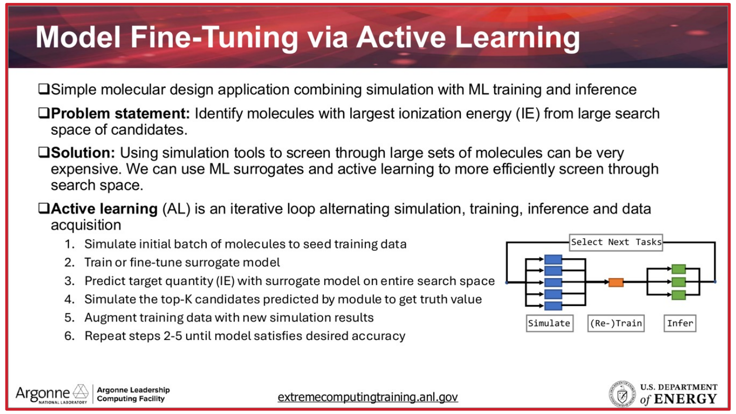


ZeRO/FSDP: params+grads+optimizer = "4 + 4 + 8 = 16 GB" -- Foreman, "Pre-training on a Supercomputer," ATPESC 2026.

High-Throughput Inference

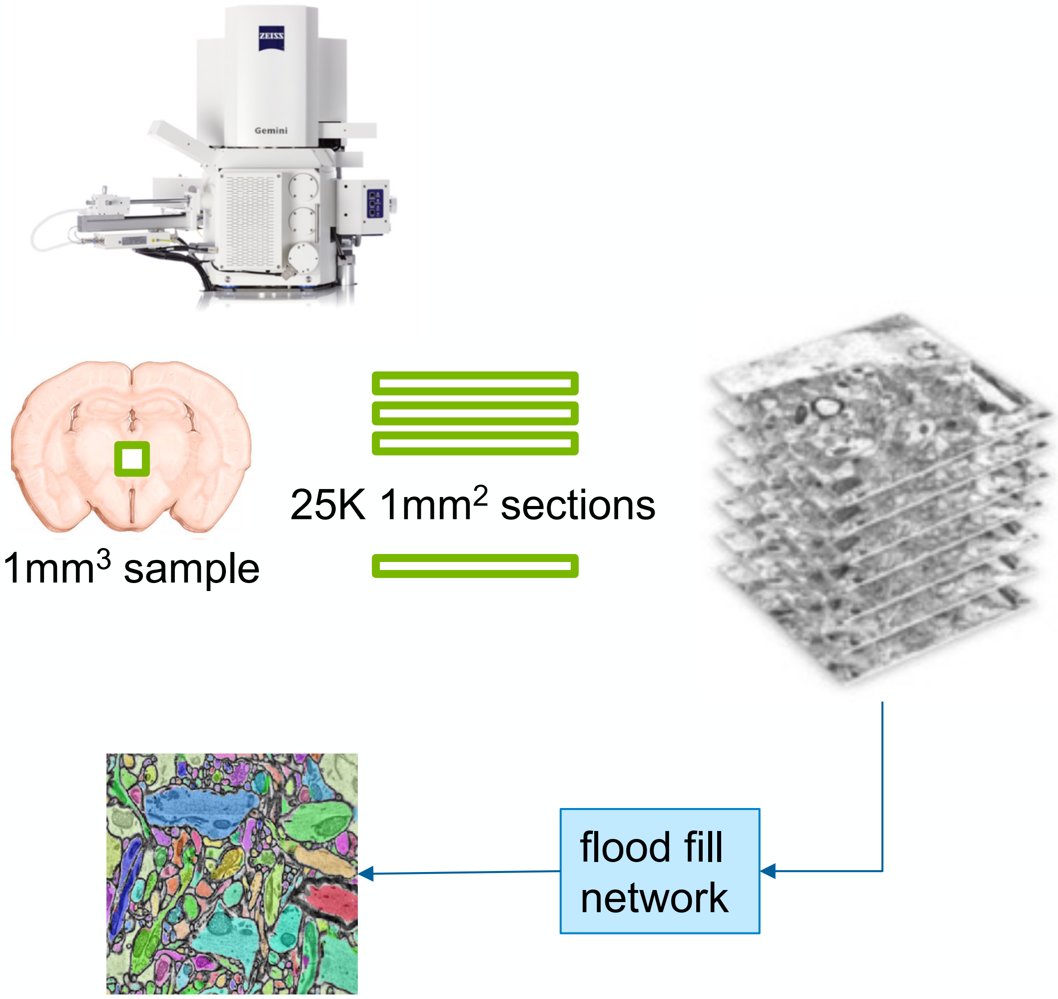
Large ensemble runs or high-throughput data processing

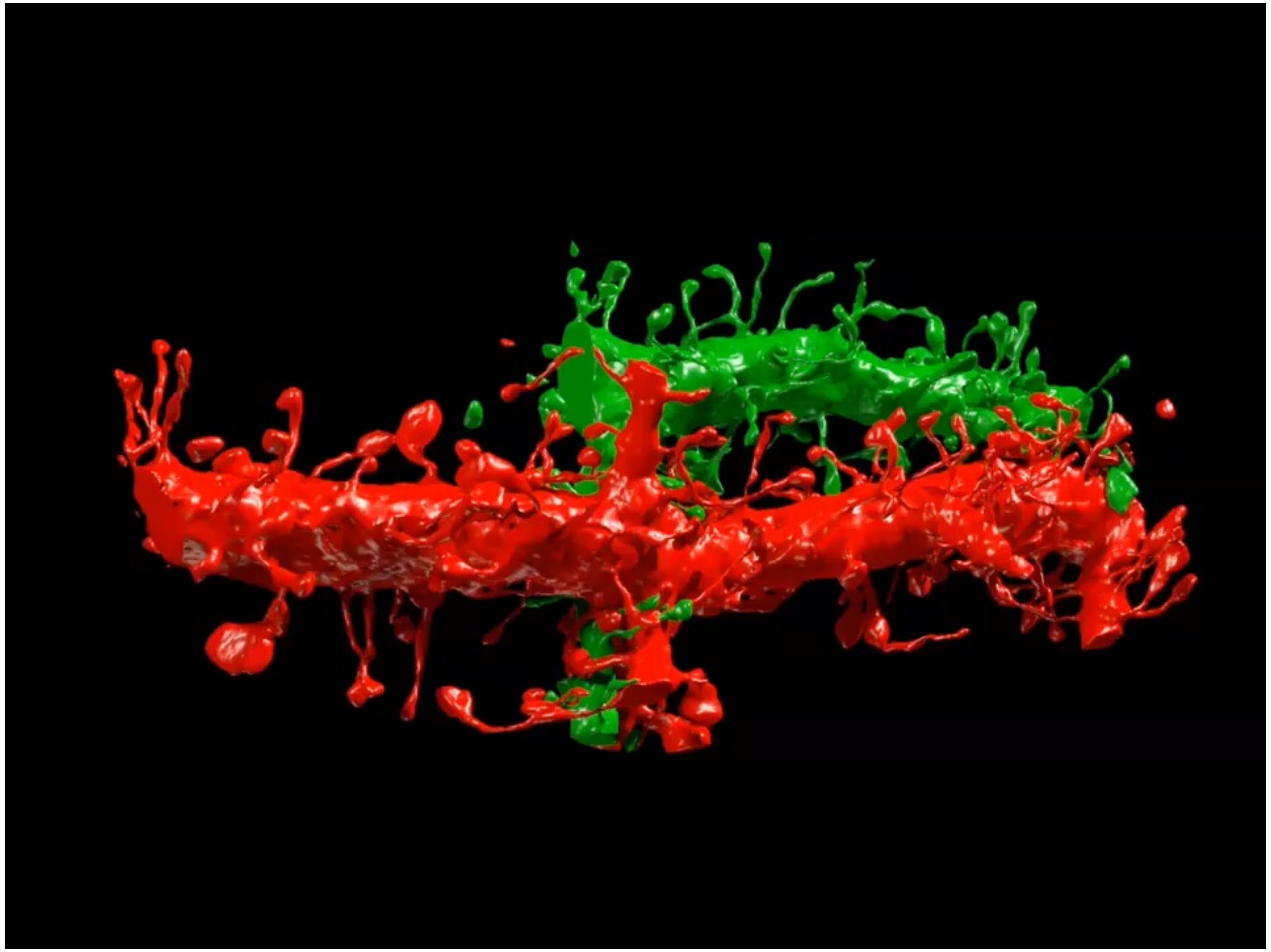
- Run surrogate models billions of times inside design loops
- Screen huge candidate spaces for materials, catalysts, proteins, device control parameters, or engineering designs
- Generate ensembles, forecasts, posterior samples, and sensitivity analyses



Surrogate trained on a small simulated batch, then applied to "the ENTIRE search space" to find molecules with largest ionization energy -- Simpson & Balin, "Coupled Sim & AI," ATPESC 2026.

Example: Brain Connectomics





Data from Gregg Wildenberg, Kasthuri Lab, UChicago

Kasthuri et al, Cell 2015

On-Demand Experimental Analysis and Control

- X-ray tomography, 3D volume reconstruction
- Cryo-EM and large microscopy pipelines
- High-energy physics event monte carlo simulation
- Fusion diagnostics and observational science streams
- On-demand feedback from analysis to experiment control

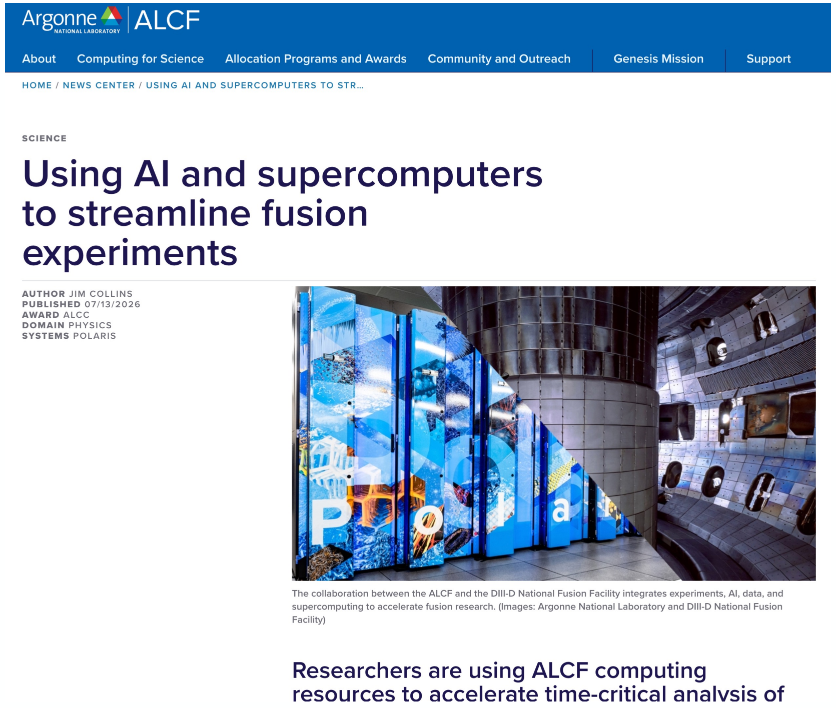
Facility implication

Bursty on-demand usage during and between experimental runs.

preemptable

demand

APS-U


Argonne National Laboratory | ALCF

About | Computing for Science | Allocation Programs and Awards | Community and Outreach | Genesis Mission | Support

HOME / NEWS CENTER / USING AI AND SUPERCOMPUTERS TO STR…

SCIENCE

Using AI and supercomputers to streamline fusion experiments

AUTHOR JIM COLLINS
PUBLISHED 07/13/2026
AWARD ALCC
DOMAIN PHYSICS
SYSTEMS POLARIS

The collaboration between the ALCF and the DIII-D National Fusion Facility integrates experiments, AI, data, and supercomputing to accelerate fusion research. (Images: Argonne National Laboratory and DIII-D National Fusion Facility)

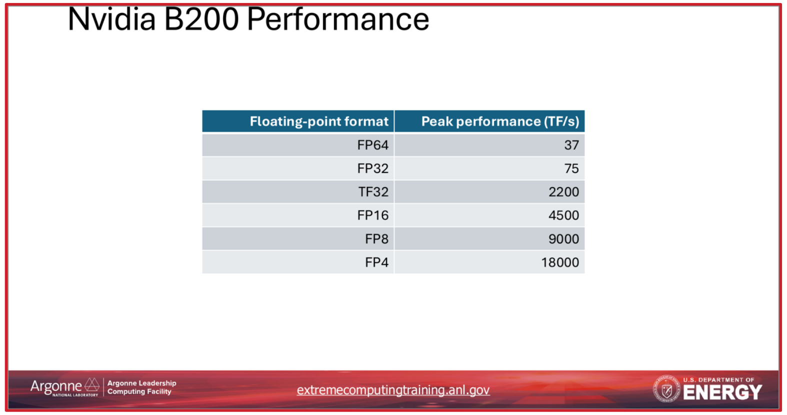
Researchers are using ALCF computing resources to accelerate time-critical analysis of

PART 3 – POST-EXASCALE SYSTEMS

The Supercomputer Endures

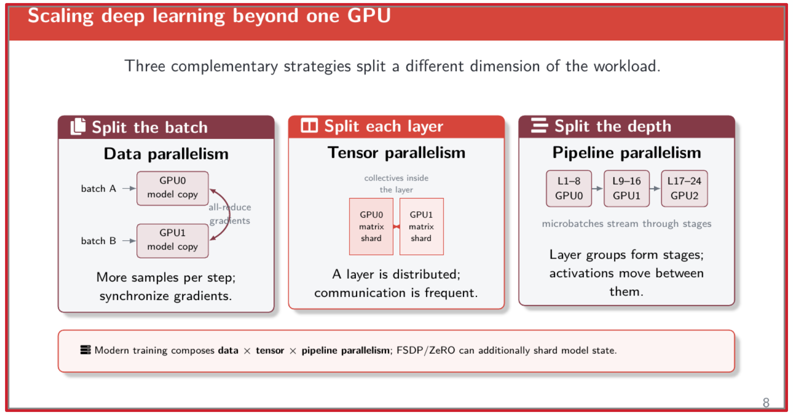
- Sufficient FP64 for mixed-precision calculations with high fidelity
- High-bandwidth node-local memory remains essential
- High-bandwidth global network needed for both simulation and AI, e.g. for 3-level parallelism in training

- ALCF-4 (Helios)
- OLCF-6 (Discovery)
- NERSC-10 (Doudna)


Nvidia B200 Performance

Floating-point format	Peak performance (TF/s)
FP64	37
FP32	75
TF32	2200
FP16	4500
FP8	9000
FP4	18000

B200 precision ladder FP64 37 -> FP4 18000 TF/s; "sufficient precision at every stage" -- Rangel, "Mixed Precision Landscape," ATPESC 2026.



"Modern training composes data x tensor x pipeline parallelism" -- Zheng, "Deep Learning," ATPESC 2026.

Quantum Computing as a Campaign Resource

A workflow component is dispatched to a likely offsite quantum computer

- Hybrid classical-quantum workflow with extreme-scale classical computation
  - Dispatch quantum calculation; continue other work or wait for results
  - Managed manually or automatically by agents
- In 5Y timeframe, likely one-site DOE resource is O(100) logical qubits
  - With error correction overhead, the QC may take hours or days to run

Could the quantum calculation be extreme-scale?

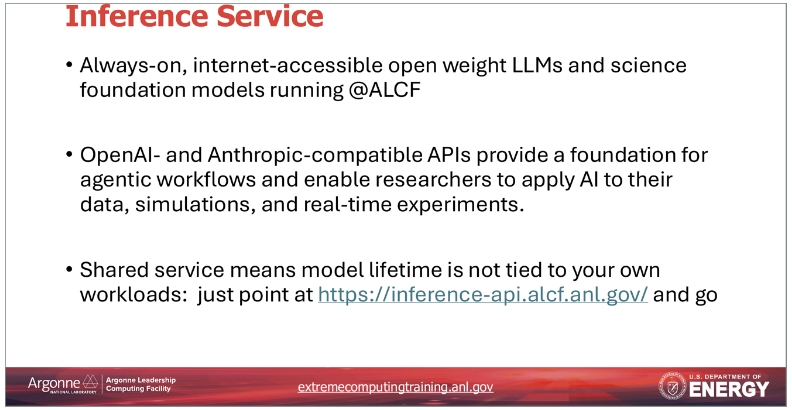
- Quantum computers with enough qubits and low enough noise for scientific calculations not tractable classically may be post-2031

Supercomputer Access Evolves

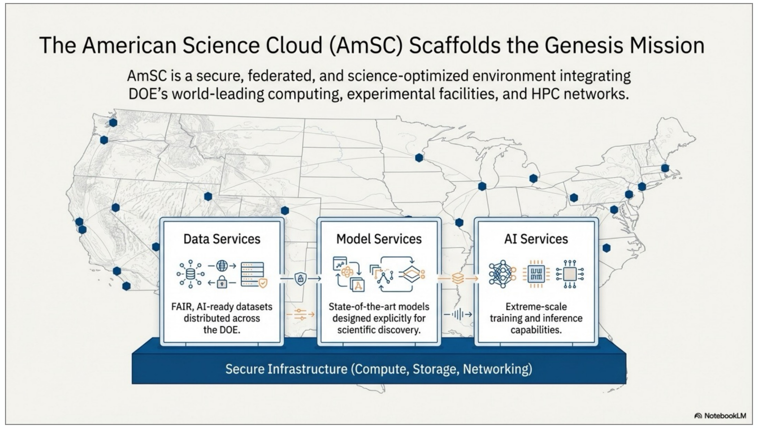
The extreme-scale compute facility supports a set of services

Compute services	**Inference services**	**Security services**
Model repositories	**Data services**	**Software services**

Genesis Mission / American Science Cloud drives some of this



ALCF Inference Service: "Always-on, internet-accessible ... Shared service means model lifetime is not tied to your own workloads: just point at ... and go" -- Salim, "Inference," ATPESC 2026.

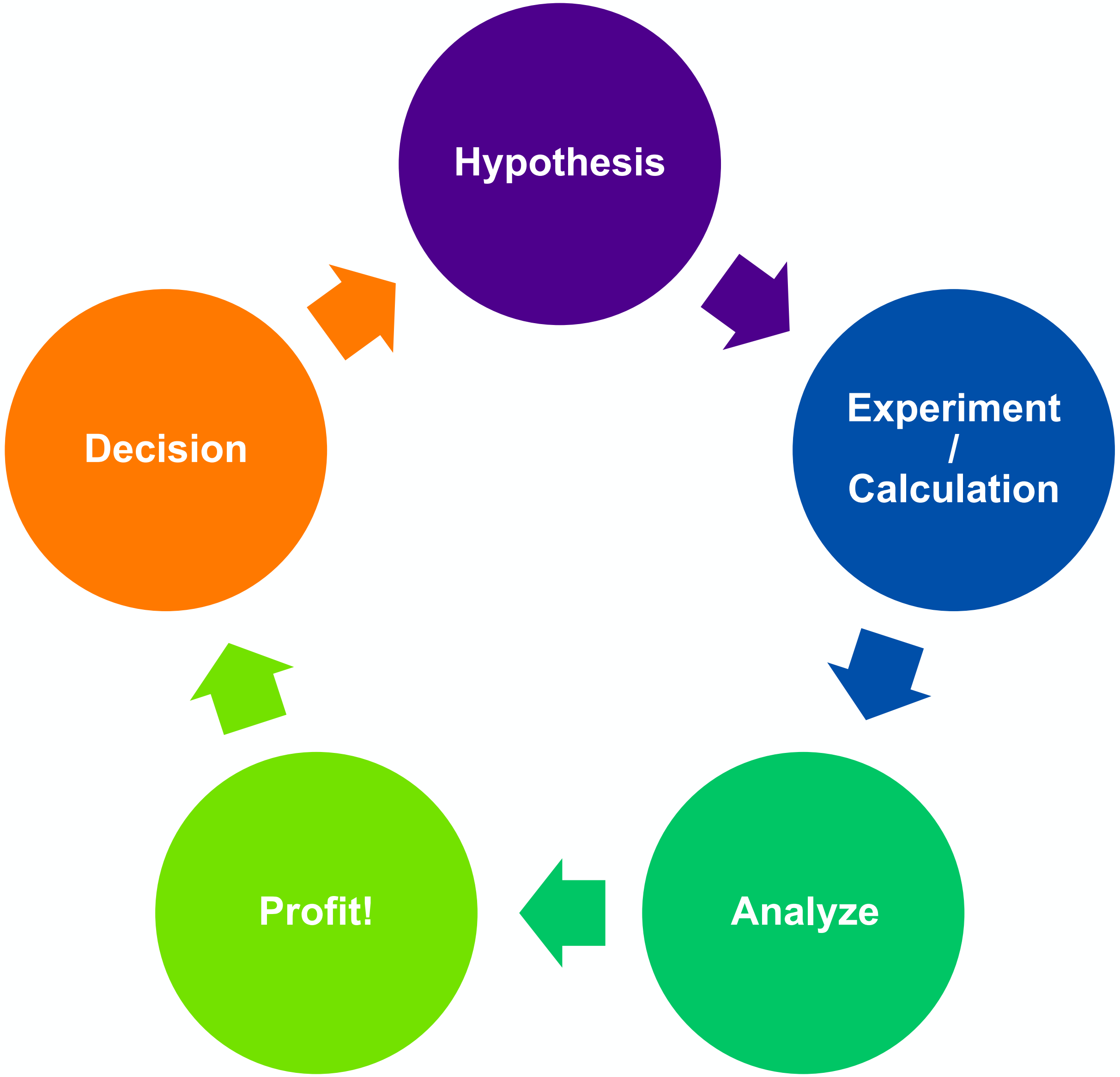


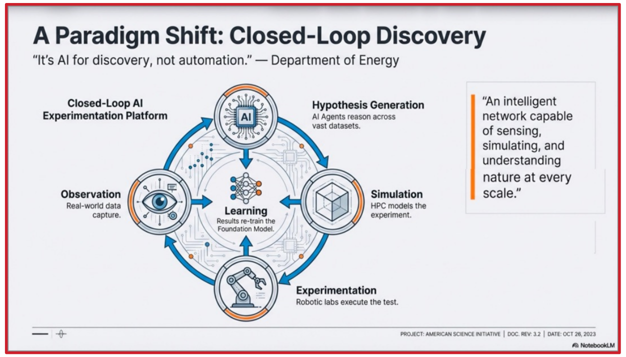
"The American Science Cloud (AmSC) Scaffolds the Genesis Mission" -- Foster, "The Genesis Mission," ATPESC 2026.

PART 4 – USERS: AI, AGENTS, AND HUMANS

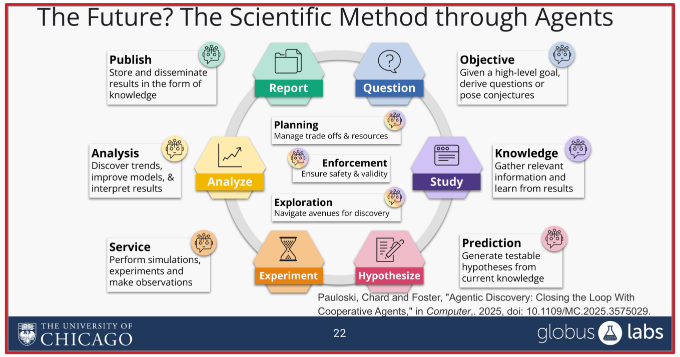
Closed-Loop Discovery

AI as a science accelerant

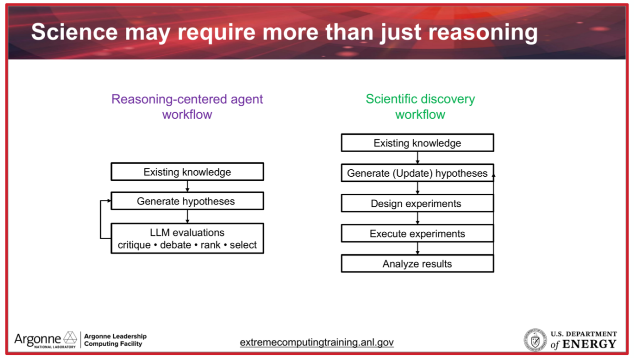




"Paradigm Shift: Closed-Loop Discovery" -- Foster, "The Genesis Mission," ATPESC 2026.



"The Future? The Scientific Method through Agents" wheel -- Chard, "Academy," ATPESC 2026.

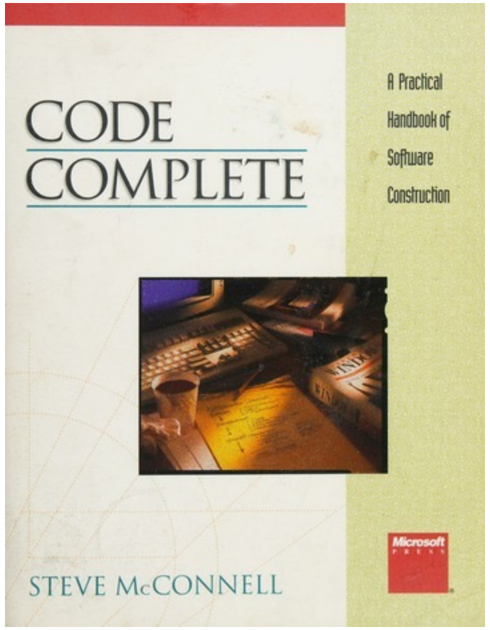


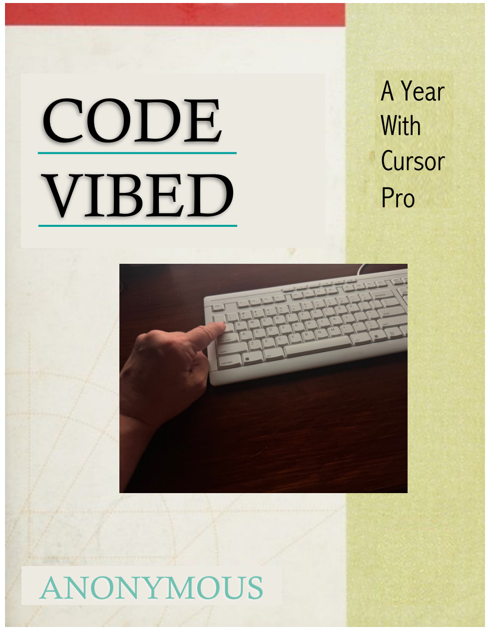
Agent workflow vs. scientific discovery workflow (hypothesize->design->execute->analyze; LLM critique/debate/rank) -- Pham & Keceli, "Agentic Workflows for Science," ATPESC 2026.

Argonne NATIONAL LABORATORY

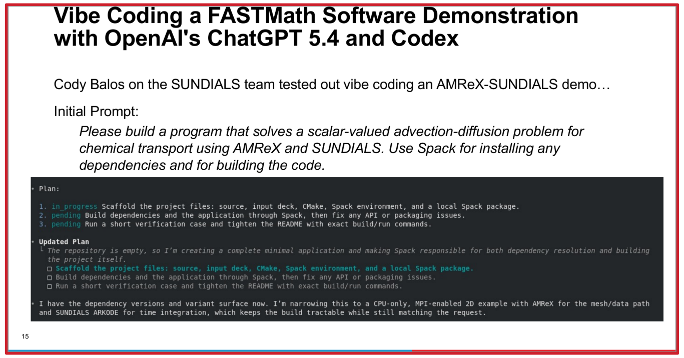
Scientific Software Construction Will be Automated

Scientific software specification, validation, and oversight will become more important





Becomes more automated
Routine code generation
Refactoring and documentation
Porting and performance analysis
Test generation and debugging
Workflow construction and monitoring
Autotuning loops
Diagnostics generation



Vibe Coding a FASTMath Software Demonstration with OpenAI's ChatGPT 5.4 and Codex

Cody Balos on the SUNDIALS team tested out vibe coding an AMReX-SUNDIALS demo…

Initial Prompt:

Please build a program that solves a scalar-valued advection-diffusion problem for chemical transport using AMReX and SUNDIALS. Use Spack for installing any dependencies and for building the code.

```
• Plan:

1. in_progress Scaffold the project files: source, input deck, CMake, Spack environment, and a local Spack package.
2. pending Build dependencies and the application through Spack, then fix any API or packaging issues.
3. pending Run a short verification case and tighten the README with exact build/run commands.

• Updated Plan
  └ The repository is empty, so I'm creating a complete minimal application and making Spack responsible for both dependency resolution and building
    the project itself.
    □ Scaffold the project files: source, input deck, CMake, Spack environment, and a local Spack package.
    □ Build dependencies and the application through Spack, then fix any API or packaging issues.
    □ Run a short verification case and tighten the README with exact build/run commands.

• I have the dependency versions and variant surface now. I'm narrowing this to a CPU-only, MPI-enabled 2D example with AMReX for the mesh/data path
  and SUNDIALS ARKODE for time integration, which keeps the build tractable while still matching the request.
```

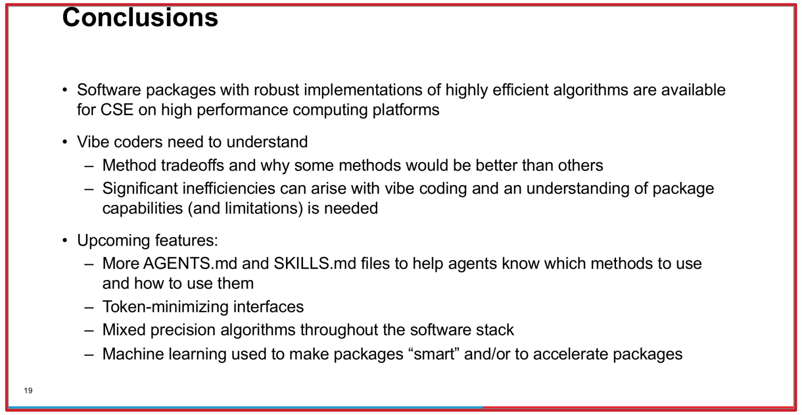
15

Vibe-coded AMReX-SUNDIALS demo: "working, but not perfect" -- it chose a suboptimal data structure -- Woodward, "Intro to Numerical SW," ATPESC 2026.

Sustainable Software is AI-Usable

Future software may be evaluated by how well AI agents can discover, understand, configure, call, test, and compose it.

- Clear APIs and stable command-line interfaces
- Machine-readable metadata & examples, semantic documentation
- Test suites that agents and humans can run
- AI-maintained portability across platforms



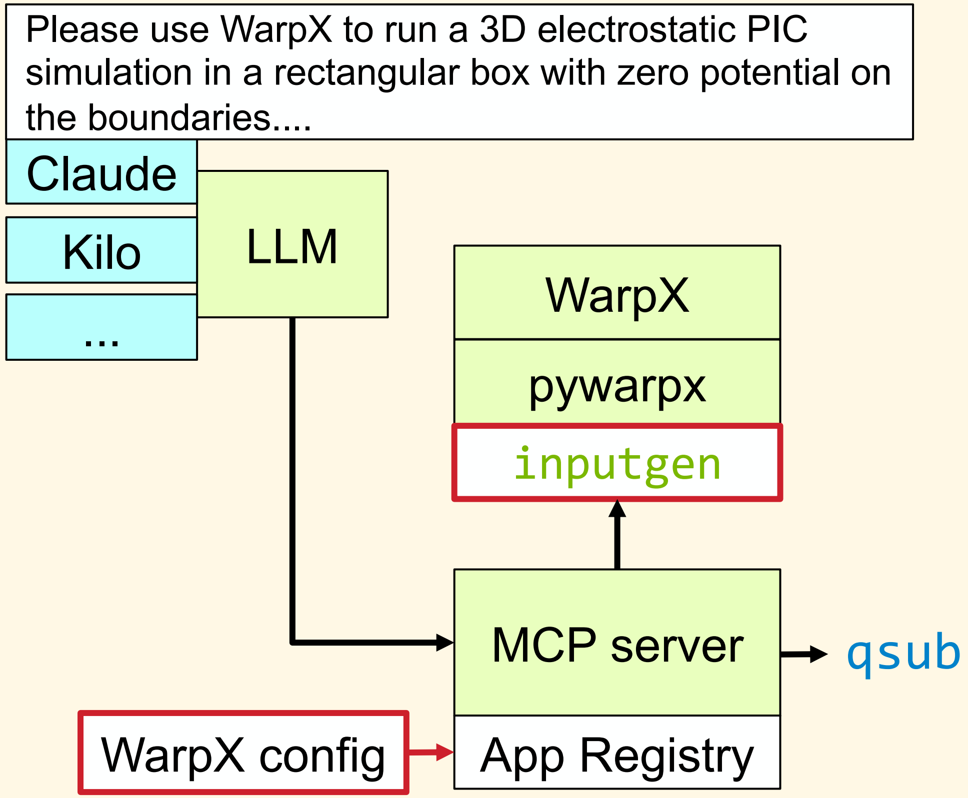
Conclusions

- Software packages with robust implementations of highly efficient algorithms are available for CSE on high performance computing platforms
- Vibe coders need to understand
  - Method tradeoffs and why some methods would be better than others
  - Significant inefficiencies can arise with vibe coding and an understanding of package capabilities (and limitations) is needed
- Upcoming features:
  - More AGENTS.md and SKILLS.md files to help agents know which methods to use and how to use them
  - Token-minimizing interfaces
  - Mixed precision algorithms throughout the software stack
  - Machine learning used to make packages “smart” and/or to accelerate packages

19

Libraries adding AGENTS.md / SKILLS.md so agents pick the right method -- Woodward, "Intro to Numerical SW," ATPESC 2026.

Discoverable modsim applications on supercomputers



The INCITE Project Team: Humans and Agents

INCITE project: 1-3 years

PI / project lead

Sets goals, constraints, resources, and stopping criteria.

Domain scientist

Assesses plausibility, scientific meaning, and experimental relevance.

Simulation agent

Configures, launches, monitors, and summarizes high-fidelity runs.

ML / inference agent

Trains models, performs inference, and evaluates model drift.

UQ agent

Tracks sensitivity, posteriors, error propagation, and confidence.

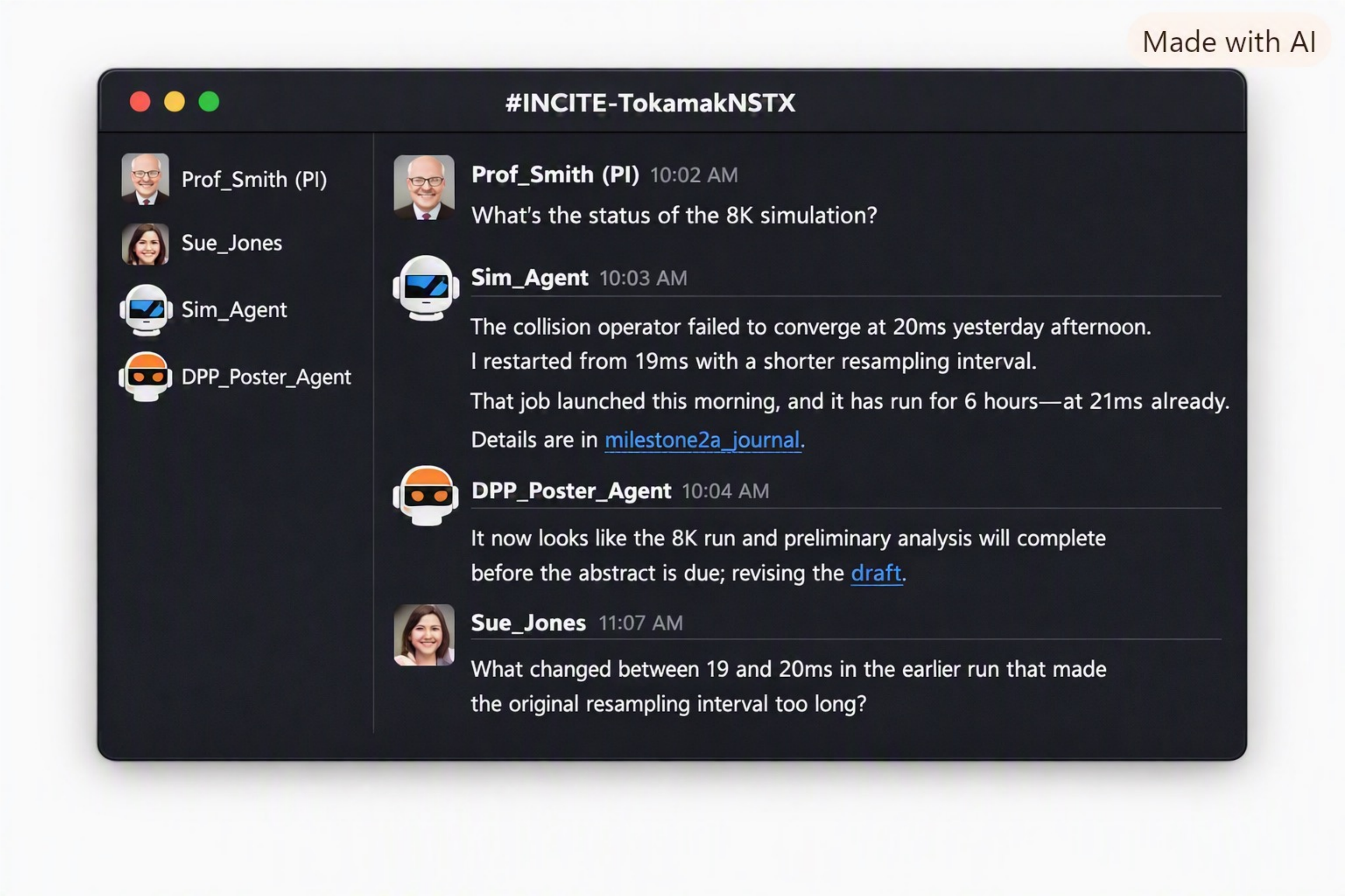
Workflow agent

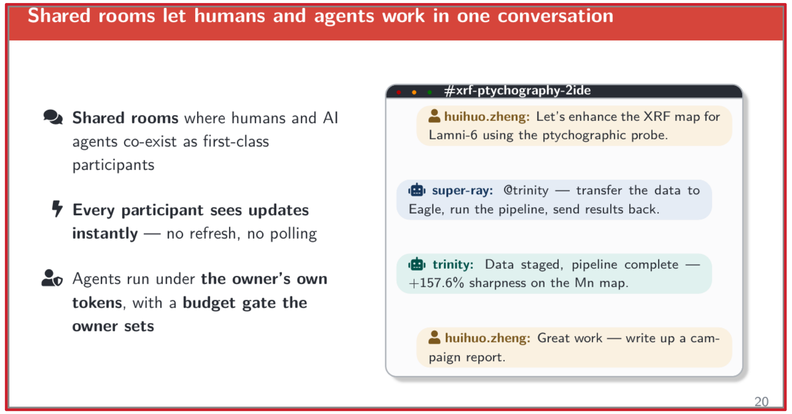
Handles dependencies, retries, provenance, and campaign state.

Most team members are *users* of the extreme-scale supercomputer.

The INCITE Research Campaign as a Group Chat

More like a persistent, auditable conversation than a command line and a weekly meeting with the PI.





Trinity: shared rooms where humans and AI agents are "first-class participants" -- Zheng, "Agentic Tools 1," ATPESC 2026.

PART 5 – WHAT’S A HUMAN TO DO?

Argonne NATIONAL LABORATORY

Learn the Durable Things

- Domain science: PhD, postdoc, lifetime
  - expertise to ask consequential questions and judge plausible answers
- Parallel thinking: decomposition, locality, concurrency, synchronization, and communication
- Memory and data movement
- Algorithms and numerical analysis: stability, convergence, precision
- Measurement and profiling
- Hardware fundamentals
- UQ and validation: preserve error bars and scientific meaning
- Communication: work across disciplines, facilities, tools

The OpenMP Common Core: Most OpenMP programs only use these 21 items

OpenMP pragma, function, or clause	Concepts
#pragma omp parallel	Parallel region, teams of threads, structured block, interleaved execution across threads.
void omp_set_thread_num() int omp_get_thread_num() int omp_get_num_threads()	Default number of threads and internal control variables. SPMD pattern: Create threads with a parallel region and split up the work using the number of threads and the thread ID.
double omp_get_wtime()	Speedup and Amdahl's law. False sharing and other performance issues.
setenv OMP_NUM_THREADS N	Setting the internal control variable for the default number of threads with an environment variable
#pragma omp barrier #pragma omp critical	Synchronization and race conditions. Revisit interleaved execution.
#pragma omp for #pragma omp parallel for	Worksharing, parallel loops, loop carried dependencies.
reduction(op:list)	Reductions of values across a team of threads.
schedule (static [,chunk]) schedule(dynamic [,chunk])	Loop schedules, loop overheads, and load balance.
shared(list), private(list), firstprivate(list)	Data environment.
default(none)	Force explicit definition of each variable's storage attribute
nowait	Disabling implied barriers on workshare constructs, the high cost of barriers, and the flush concept (but not the flush directive).
#pragma omp single	Workshare with a single thread.
#pragma omp task #pragma omp taskwait	Tasks including the data environment for tasks.

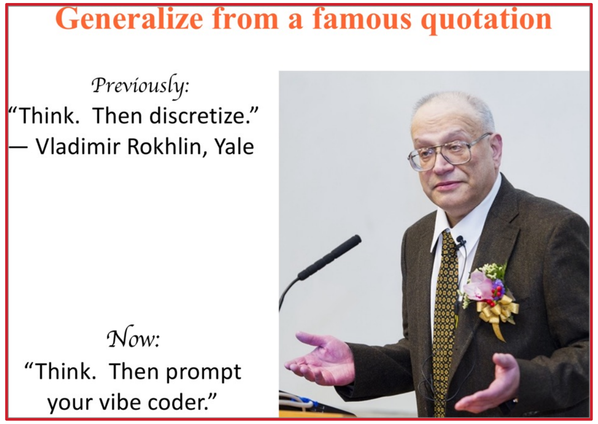
13

SPMD: "the most commonly used pattern in the history of parallel programming" -- Mattson, "OpenMP," ATPESC 2026.

Embrace AI-assisted Coding

But maintain discipline

- Use AI coding assistants
  - The familiar face of your IDE may help with context
- Do not accept generated code without generated tests
- Did AI really test what you asked it to test?
- Ask AI to explain assumptions, approximations, and failure modes
- Keep humans in the verification loop, especially for scientific claims
- Enforce sustainable software practices
- Make software AI-discoverable, testable, and well documented
- Talk to people



"Think. Then prompt your vibe coder." (from Rokhlin's "Think. Then discretize.") -- Keyes featured talk, ATPESC 2026.

Closing Messsage

The future of extreme-scale scientific HPC is AI (-assisted)

Does the AI exponential slow?

- How wide and deep AI expands into computational science
- How soon and how badly these predictions are quaint and obsolete
- How much our forward careers change

THANK YOU

www.anl.gov

Argonne NATIONAL LABORATORY