The Exascale Computing Project

Lori Diachin (LLNL), ECP Director

SIAM HPC Spotlight Webinar
May 22, 2024

The Exascale Computing Project was designed to help launch the exascale era

ECP funded teams spanned national labs, universities, and industry

$1.8B over 7 years in a formal Department of Energy 413.3B project

Funded by DOE's Office of Science and NNSA programs

81 **Total research, development, and deployment projects**

- 6 Co-Design Centers
- 25 Application Development
- 35 Software Technology
- 17 Hardware and Integration

15
US DOE Labs, 6 of which are Core Partners

57
University Partners

36
Industry Organizations

- A **unique collaboration** brought together some of the brightest application, software, and computational experts from coast to coast
- **Best practices and lessons learned** for how to program GPUs – moving the nation forward
- **1000+ researchers** trained and ready for accelerator-based computing
- **1000+ students** introduced to HPC and Exascale computing through ECP's outreach, training, and workforce development initiatives

Technical work was largely complete as of Dec 31, 2023

Project leadership team is now working to close out the formal DOE 413.3B project

The Exascale Computing Project was started in 2016 and tasked to meet four DOE mission needs in high performance computing

Deliver a long-term, **sustainable software ecosystem** that can be used and maintained for years to come

Promote the **health of the US HPC industry**

Ensure that exascale systems can be used to deliver **mission-critical applications**

Maintain **international leadership in HPC**

To meet mission needs, the ECP was organized into three technical focus areas partnered with formal project management experts

Performant mission and science applications at scale

Aggressive RD&D project	Mission apps; integrated S/W stack	Deployment to DOE HPC Facilities	Hardware technology advances

Application Development (AD)

Develop and enhance the predictive capability of applications critical to DOE

24 applications
National security, energy, Earth systems, economic security, materials, data

6 co-design centers
ML, graph analytics, mesh refinement, PDE discretization, particles, online data analytics


Andrew Siegel, AD Director
Erik Draeger, AD Deputy Director

Software Technology (ST)

Deliver expanded and vertically integrated software stack to achieve full potential of exascale computing

70 unique software products spanning programming models and runtimes, math libraries, data and visualization, development tools


Mike Heroux, ST Director
Lois Curfman McInnes, ST Deputy Director

Hardware and Integration (HI)

Integrated delivery of ECP products on targeted systems at leading DOE HPC facilities

6 US HPC vendors
focused on exascale node and system design; application integration and software deployment to Facilities


Richard Gerber, HI Director
Susan Coghlan, HI Deputy Director

EXASCALE COMPUTING PROJECT

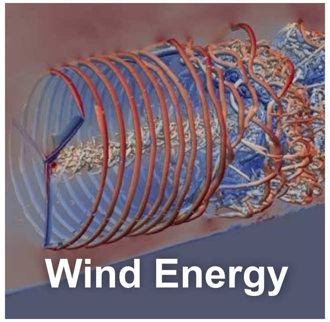
Several exascale capabilities are sited at the DOE facilities

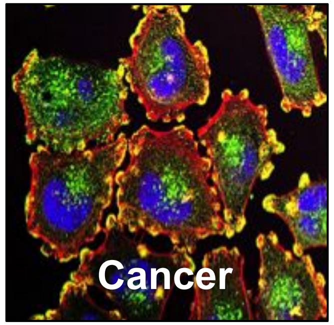
- #1 on the Top 500 list since May 2022: Currently 1.19 exaflops
- #2 on HPL-MxP mixed-precision benchmark with 10.2 exaflops of AI performance
- Full system available to all ECP teams since April 2023
- Excellent performance: 7 Gordon Bell & Special Prize Finalists to-date
- Many first-of-a-kind science goals achieved

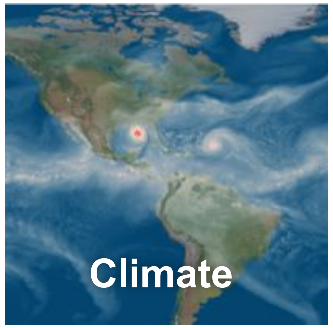
- Full system installation completed June 2023
- #2 on the Top 500 list (May 2024),. Currently 1.01 exaflops using 87% of the machine
- #1 on HPL-MxP mixed-precision benchmark with 10.6 exaflops of AI performance
- Limited access available July 2023 (ANL personnel); full system access for ECP November 2023

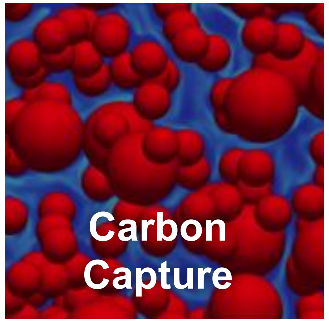
- Hardware being delivered to LLNL now; installation is underway
- Expected to exceed 2 Exaflops when deployed in mid to late 2024
- Brief period of open science before machine transitions to classified use focused on stockpile stewardship

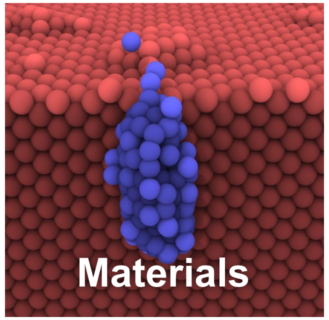
ECP invested in a broad range of critical application areas; some with limited HPC experience at project inception

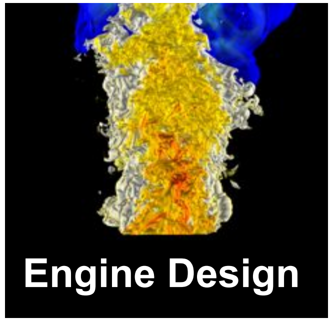


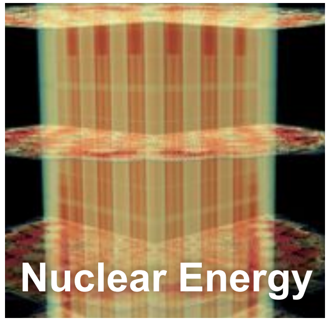


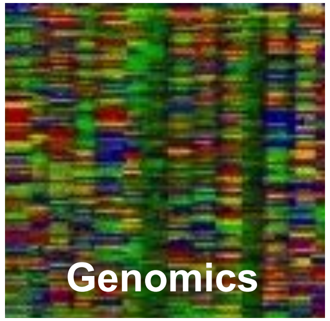


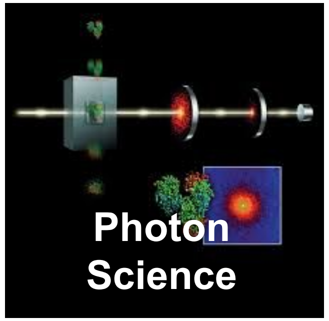


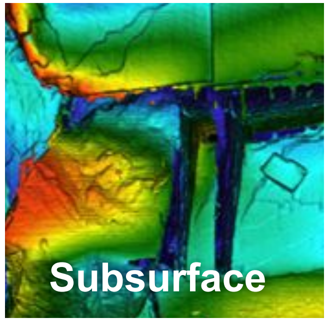


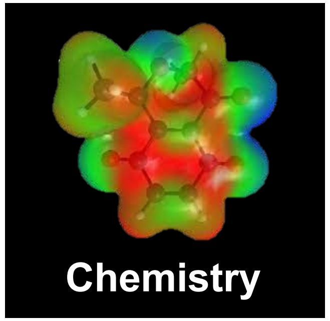


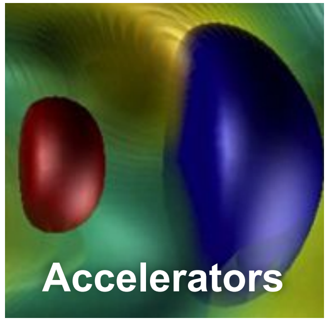


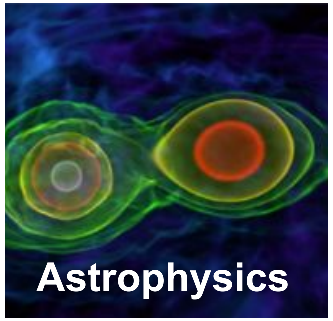


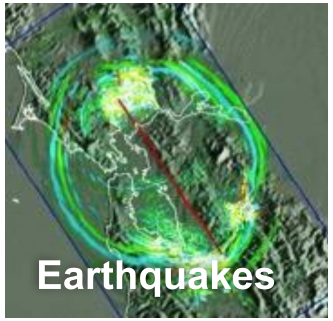


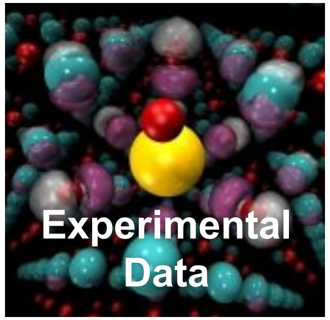


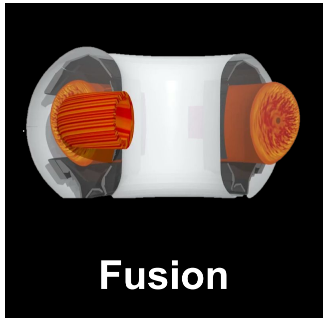


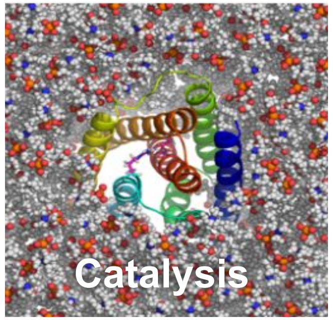








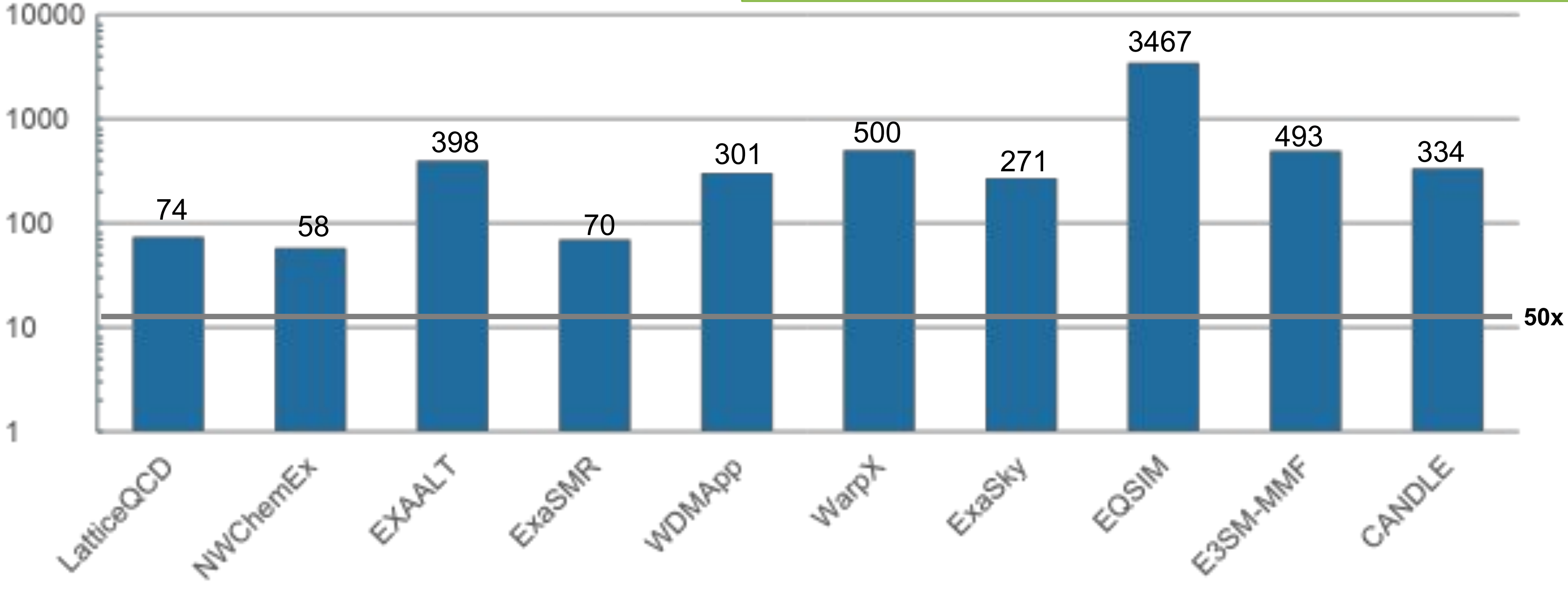




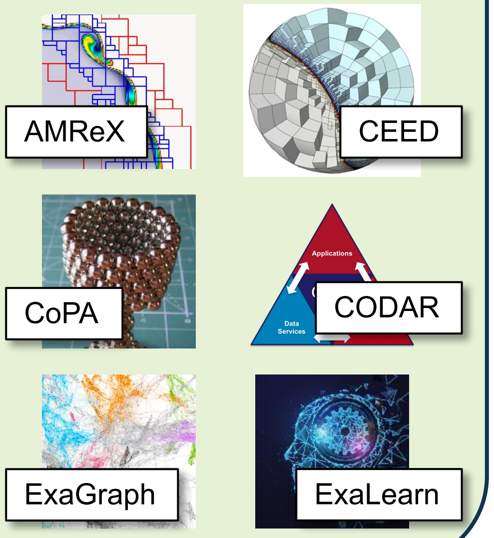
ECP EXASCALE COMPUTING PROJECT

ECP application results exceeded expectations

10 out of 11 projects surpassed an ambitious 50x performance target

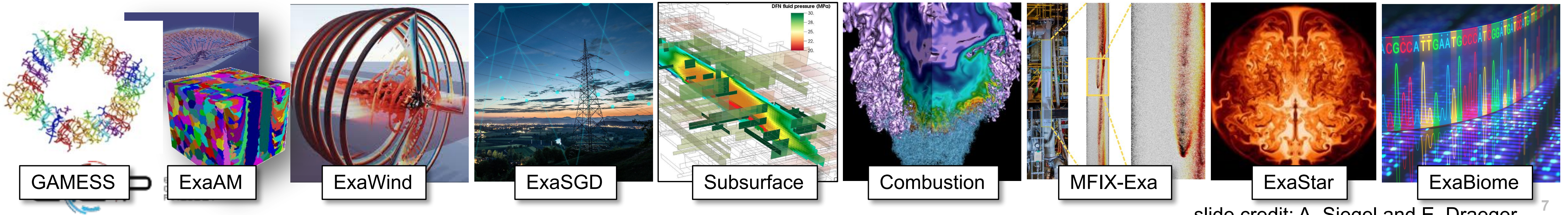


Co-design played a critical role



9 out of 10 new HPC science projects completed exascale capability demonstrations

3 out of 4 NNSA applications demonstrated exascale readiness



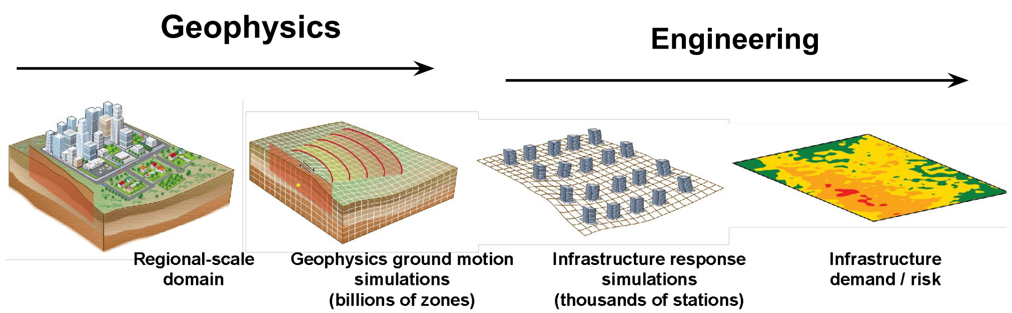
slide credit: A. Siegel and E. Draeger

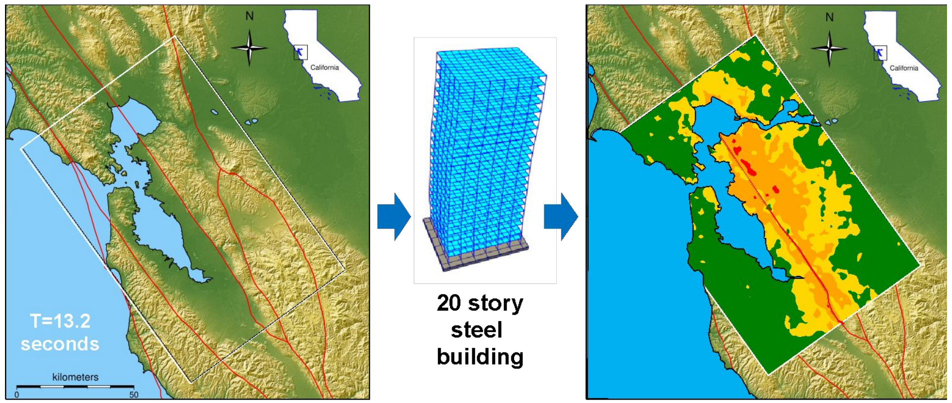
One example EQSIM: exascale-capable code redesign had massive impact

EQSIM PI: Dave McCallen, LBNL

- **Starting point**: CPU-only code written in C by physicists. Code had complex nested loops (fourth-order finite difference stencil) that the compiler struggled to optimize, find SIMD.

- **ECP accomplishments**
  - Algorithmic improvements using curvilinear mesh refinement improved scientific work-rate by a factor of 2.85
  - Rewrote code in C++ with RAJA, with ZFP data compression to save sufficient data to maintain adequate precision in stored data
  - Infrastructure simulations now include strong coupling with OpenSees soil/building modeling and using in soil-structure interaction models; help gain insight into areas of maximum risk.



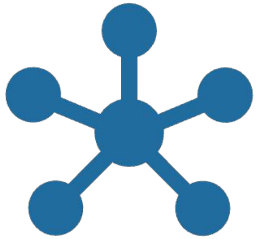


Achieved a **3500X improvement in computational performance** compared to initial baseline on Cori (~30 PF KNL system); Simulation of regional-scale ground motions at frequencies of engineering interest (5-10 Hz) now within reach.

There where several key ingredients required for a successful ECP Application Development Project

Algorithmic innovation

Porting

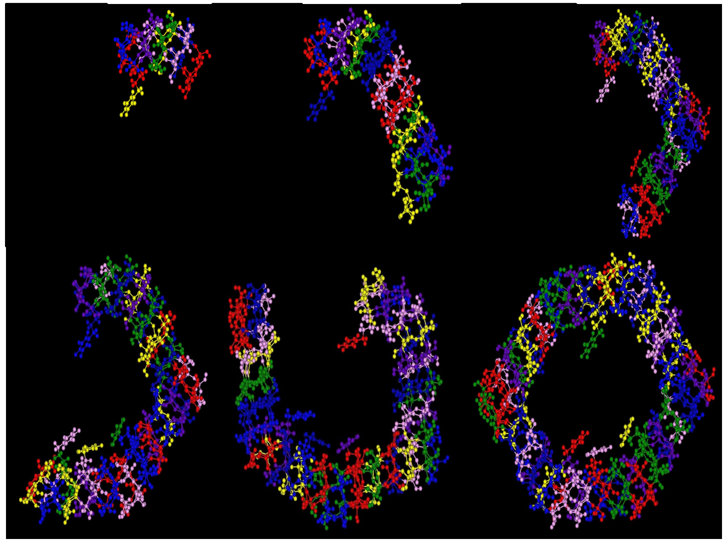


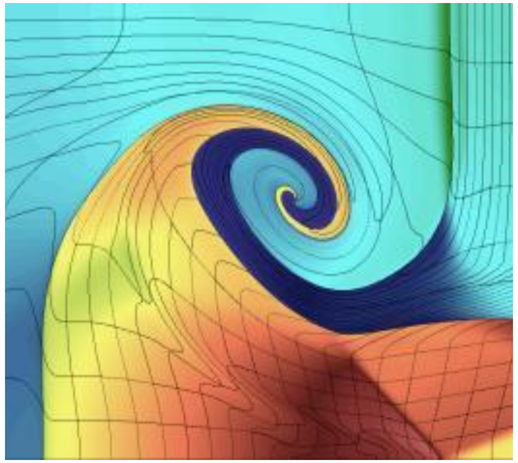
Integration

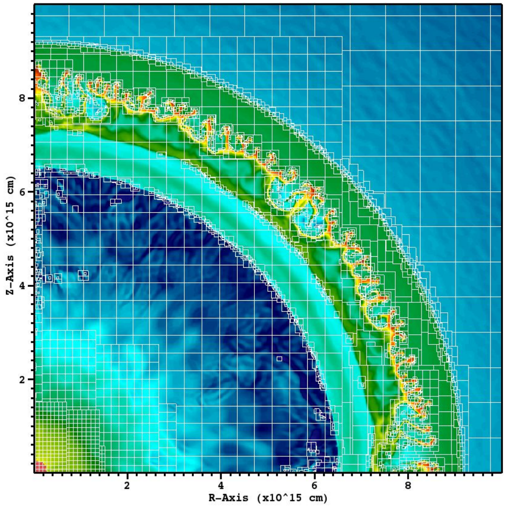
EXASCALE COMPUTING PROJECT

Algorithmic innovation: domain-driven adaptations critical for making efficient use of exascale systems

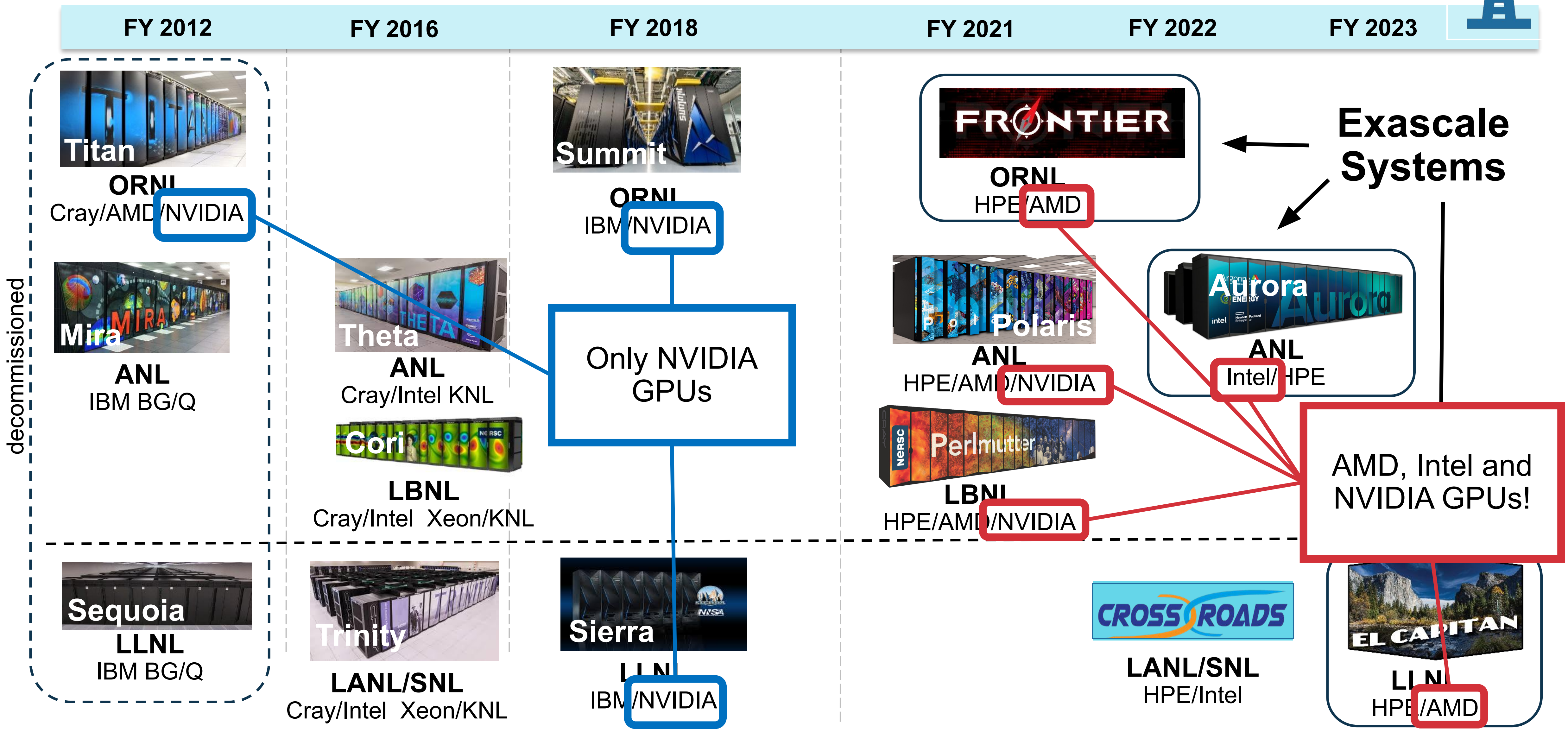
- Inherent strong scaling challenges on GPU-based systems
  - Ensembles vs. time averaging
  - Fluid dynamics, seismology, molecular dynamics, time-stepping
- Increase dimensions of (fine-grained) parallelism to feed GPUs
  - Ray tracing, Markov Chain Monte Carlo, fragmentation methods
- Localized physics models to maximize "free flops”
  - MMF, electron subcycling, enhanced subgrid models, high-order discretizations
- Alternatives to sparse linear systems
  - Higher order methods, Monte Carlo
- Reduced branching
  - Event-based models



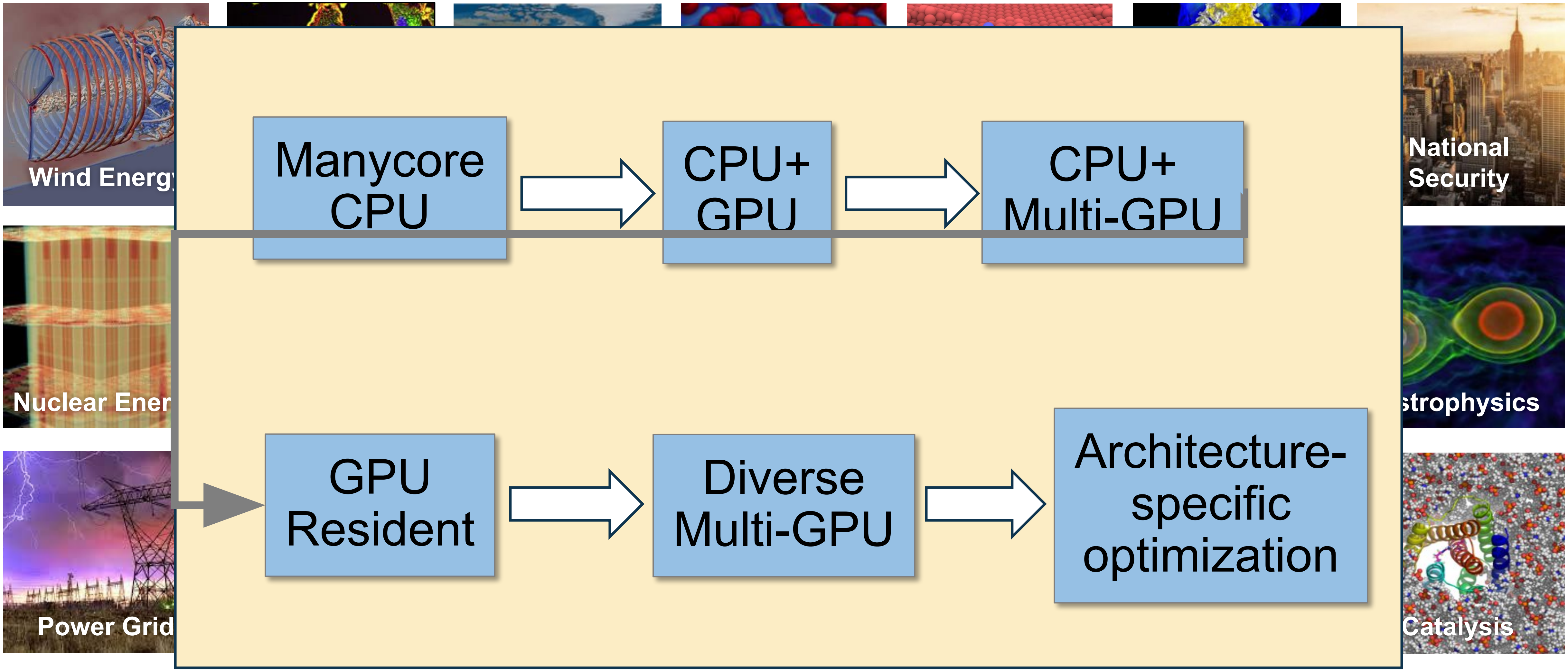




HPC systems have come a long way since ECP’s inception



Exascale applications were designed to be flexible and adaptive



ECP applications teams used several different programming models to achieve performance portability

GPU-specific kernels

- Isolate the computationally-intensive parts of the code into CUDA/HIP/SYCL kernels.
- Refactoring the code to work well with the GPU is the majority of effort.

Loop pragma models

- Offload loops to GPU with OpenMP or OpenACC.
- Most common portability strategy for Fortran codes.

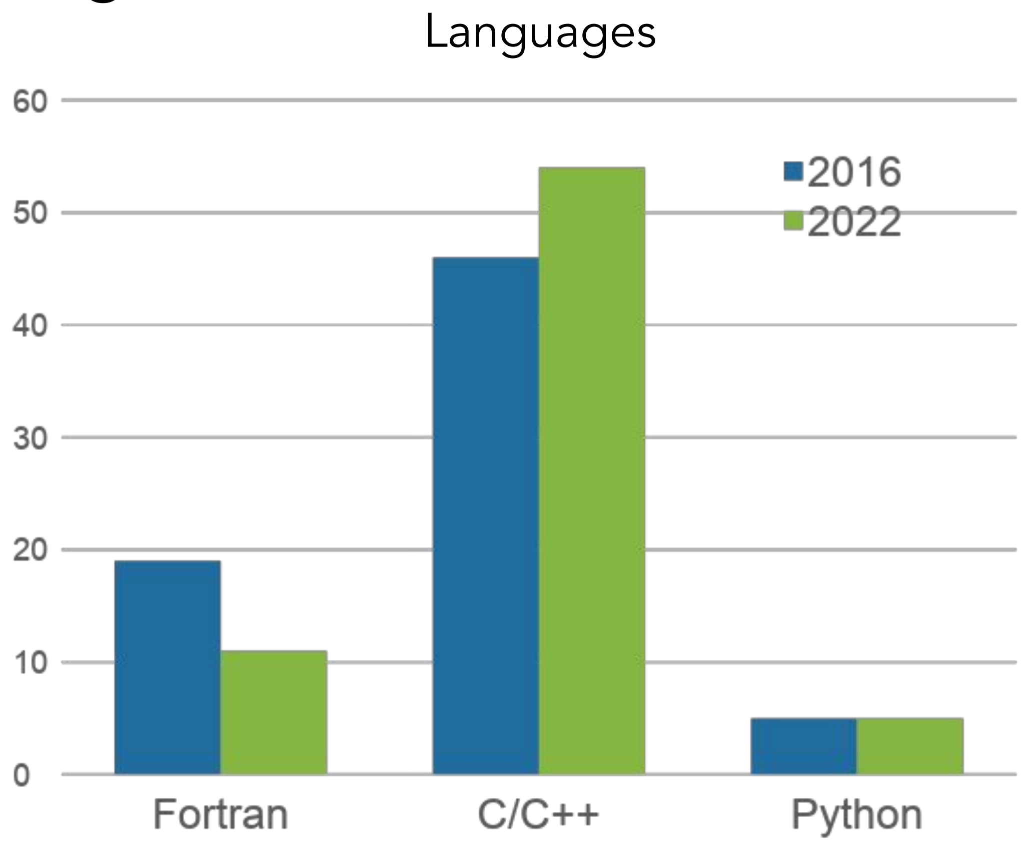
C++ abstractions

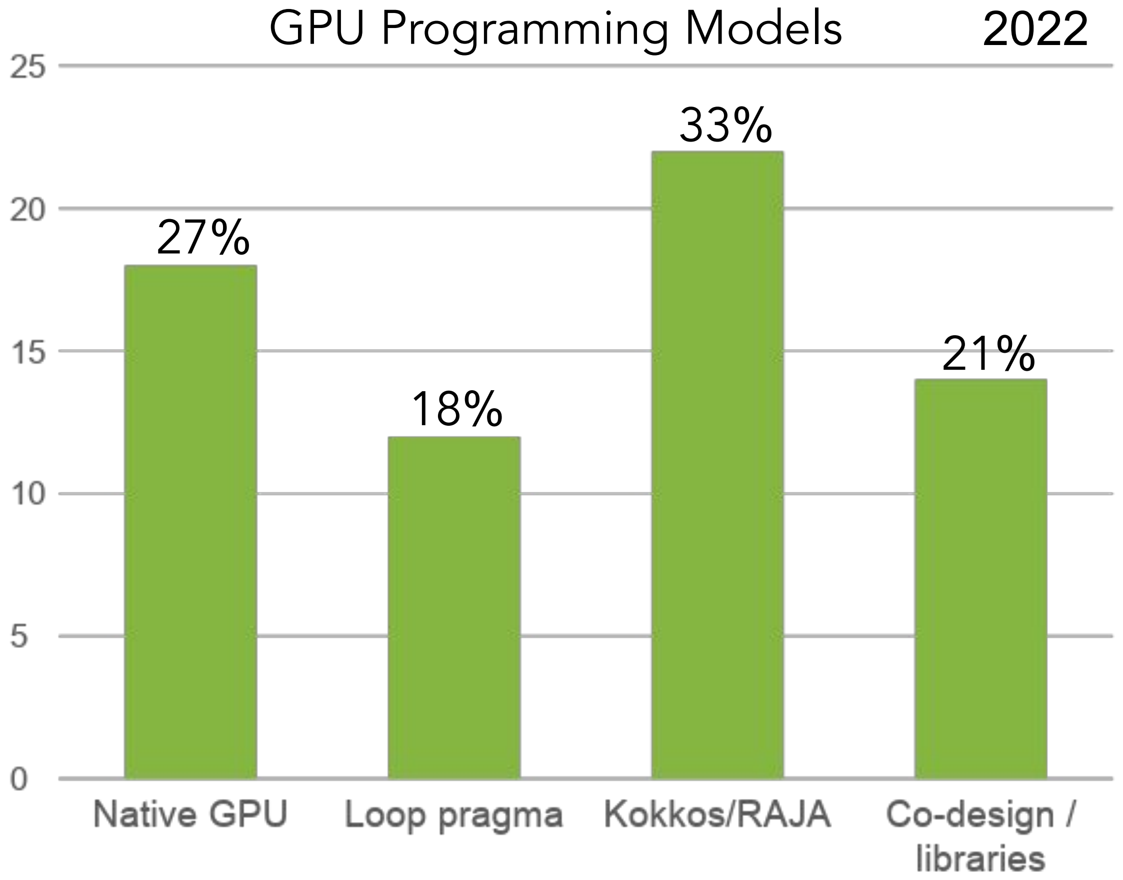
- Fully abstract loop execution and data management using advanced C++ features.
- Kokkos and RAJA developed by NNSA in response to increasing hardware diversity.

Co-design frameworks

- Design application with a specific motif to use common software components
- Depend on co-design code (e.g. CEED, AMReX) to implement key functions on GPU.

Distribution of ECP programming languages and models has changed over time

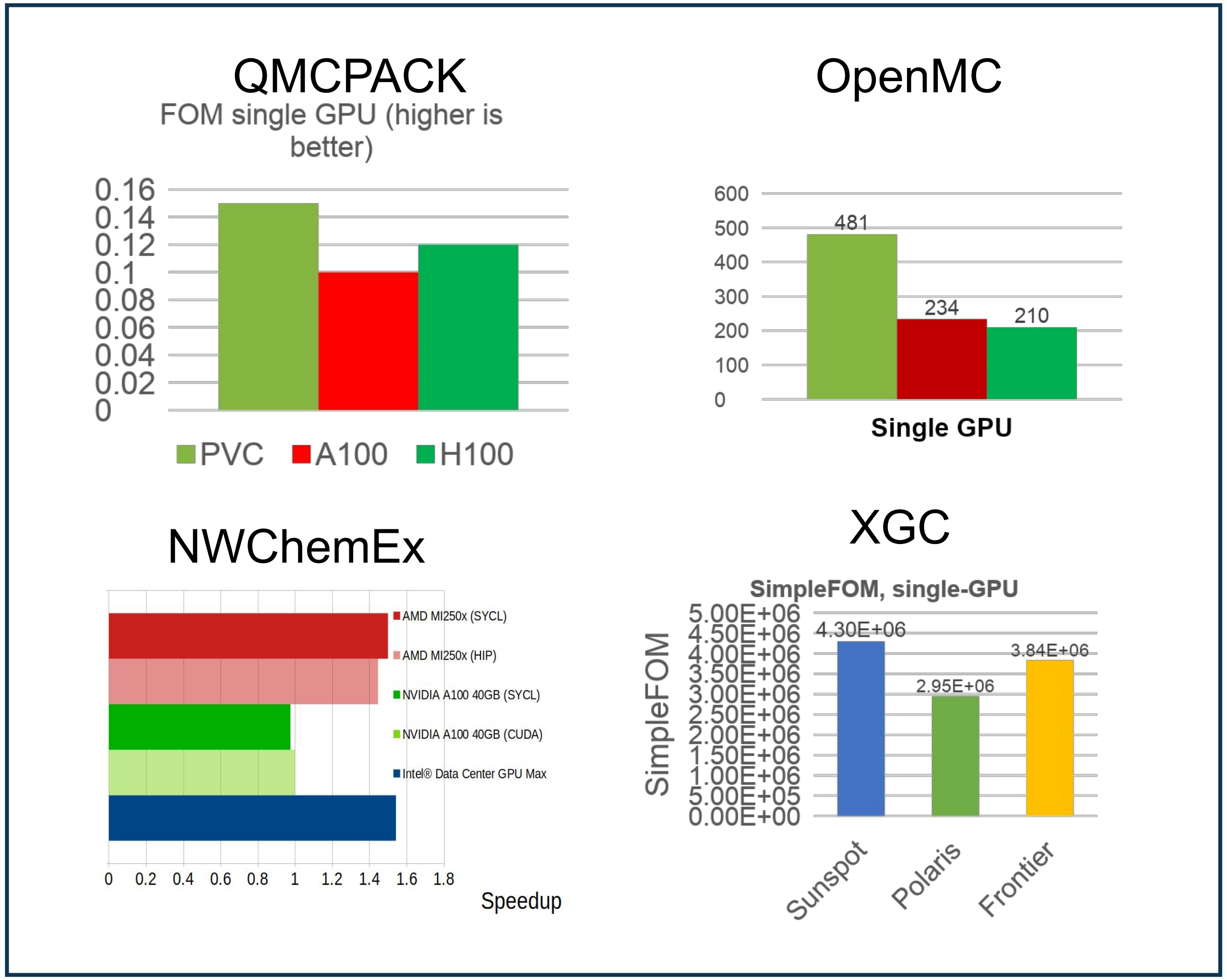


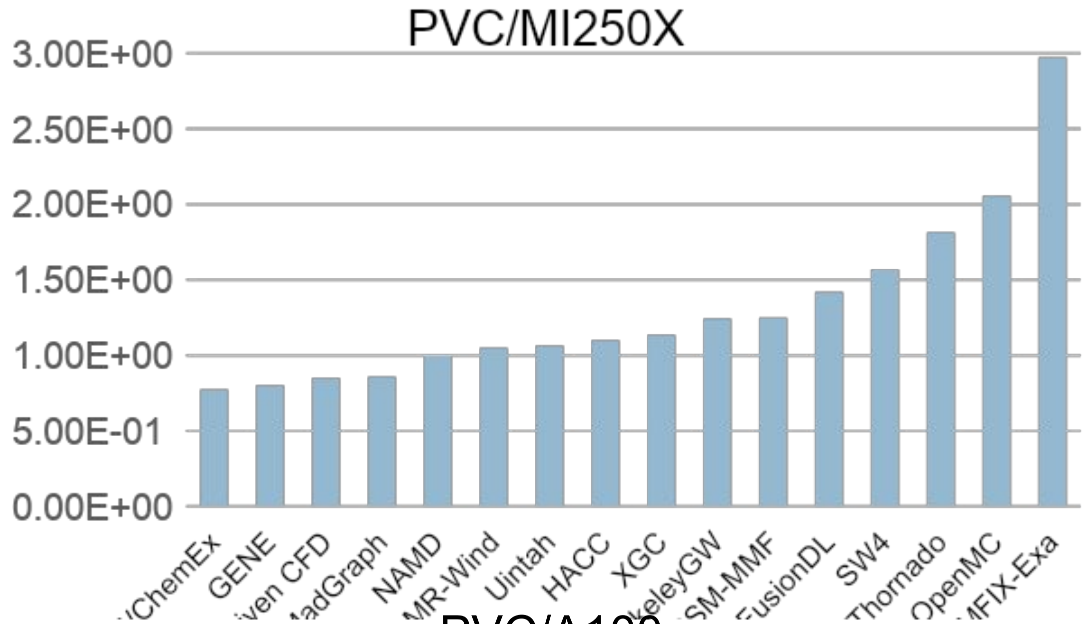


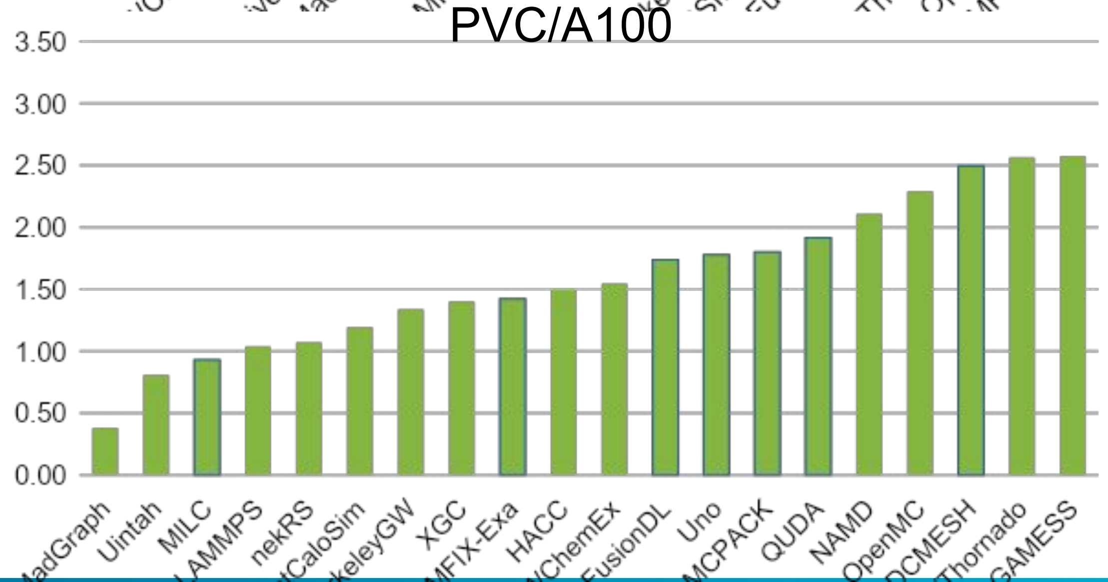
Many ECP applications started out using native GPU and loop pragma models before moving to C++ abstractions and co-design libraries

Evans TM, Siegel A, Draeger EW, *et al.* *"A survey of software implementations used by application codes in the Exascale Computing Project."* The International Journal of High Performance Computing Applications. 2022;36(1):5-12.

Performance portability layers have allowed several teams to successfully run on multiple GPU exascale architectures



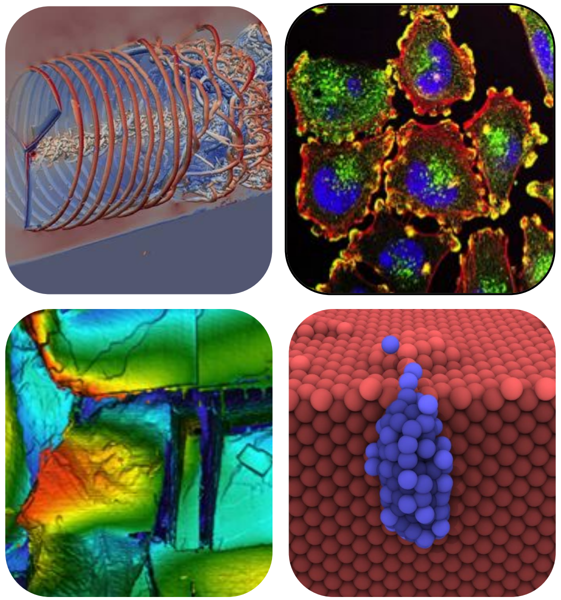




The success of exascale applications relied on significant software infrastructure

Programming abstractions

Memory / heterogeneous data management



In situ visualization and analysis

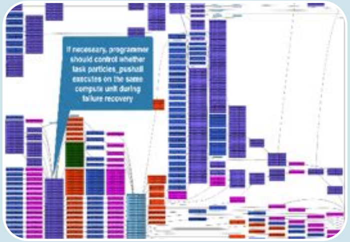
Solvers / preconditioners

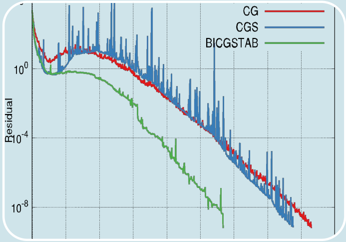
Numerical infrastructure / discretizations

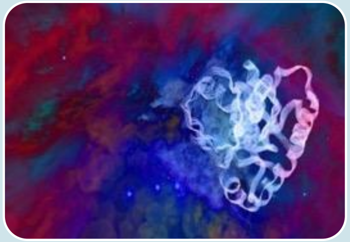
Build and deployment

Performance analysis and monitoring

ECP invested in six software technology areas







E4S

NNSA National Nuclear Security Administration

Programming Models & Runtimes

- Enhance and get ready for exascale the MPI and OpenMP programming models (hybrid programming models, deep memory copies)
- Develop performance portability tools (e.g., Kokkos and Raja)
- Support alternate models for potential benefits and risk mitigation: PGAS (UPC++/GASNet), task-based models (Legion, PaRSEC)
- Libraries for deep memory hierarchy and power manageme

Development Tools

- Continued, multifaceted capabilities in portable, open-source LLVM compiler ecosystem to support expected ECP architectures, including support for F18
- Performance analysis tools that accommodate new architectures, programming models, e.g., PAPI, Tau

Math Libraries

- Linear algebra, iterative linear solvers, direct linear solvers, integrators and nonlinear solvers, optimization, FFTs, etc
- Performance on new node architectures; extreme strong scalability
- Advanced algorithms for multi-physics, multiscale simulation and outer-loop analysis
- Increasing quality, interoperability, complementarity of math libraries

Data and Visualization

- I/O via the HDF5 API
- Insightful, memory-efficient in-situ visualization and analysis
- Data reduction via scientific data compression
- Checkpoint restart

Software Ecosystem

- Develop features in Spack necessary to support ST products in E4S, and the AD projects that adopt it
- Develop Spack stacks for reproducible turnkey software deployment
- Optimization and interoperability of containers for HPC
- Regular E4S releases of the ST software stack and SDKs with regular integration of new ST products

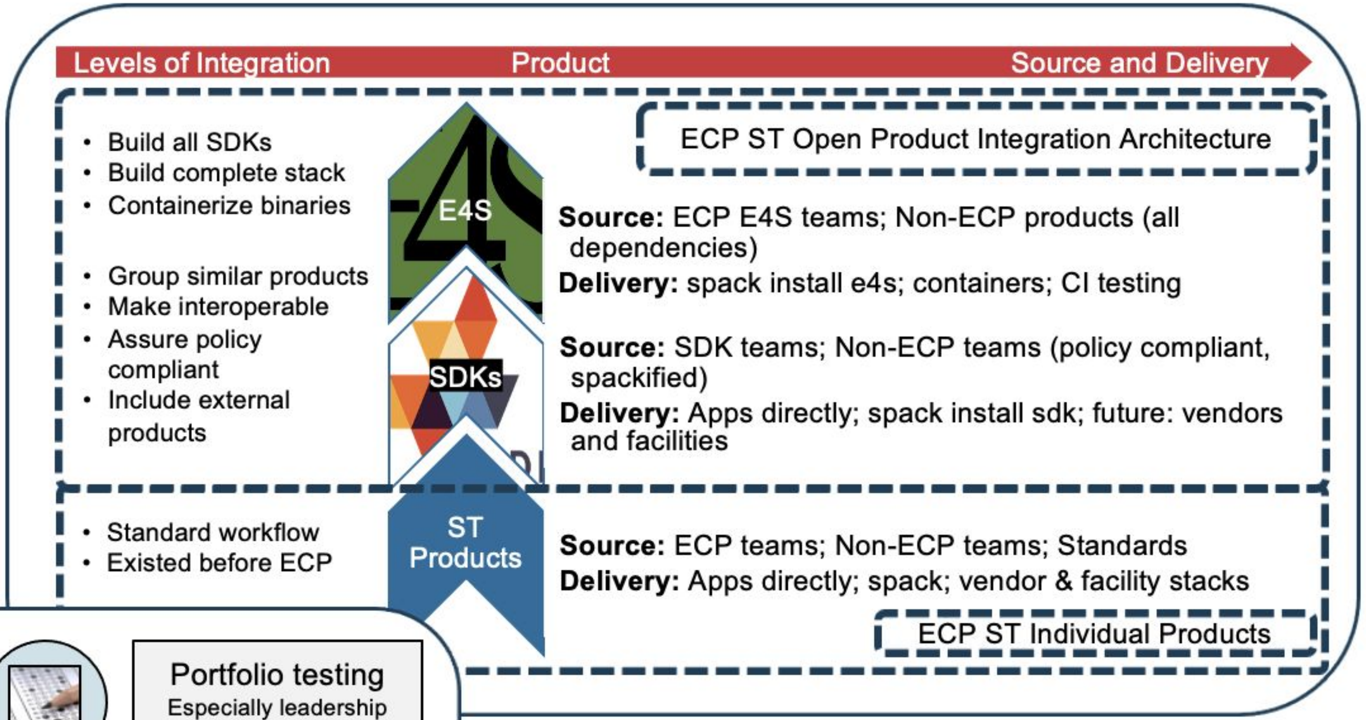
NNSA ST

- Open source NNSA Software projects
- Projects that have both mission role and open science role
- Major technical areas: New programming abstractions, math libraries, data and viz libraries
- Cover most ST technology areas
- Subject to the same planning, reporting and review processes

Area Leads: Rajeev Thakur  Jeff Vetter  Li Jim A  Todd Munson  yn Mohror  

ECP software technologies were broadly deployed through the Extreme Scale Scientific Software Stack (E4S)

- ECP Software Technology lead: Mike Heroux (SNL)
- E4S: HPC software ecosystem – a curated software portfolio
- A **Spack-based** distribution of software tested for interoperability and portability to multiple architectures
- Available from **source**, **containers**, **cloud**, **binary caches**
- Not a commercial product – an open resource for all
- Supported by DOE and commercial entities (Paratools)
- Growing functionality: November 2023: E4S 23.11 – 100+ full release products



Community Policies Commitment to SW quality	DocPortal Single portal to all E4S product info
Portfolio testing Especially leadership platforms	Curated collection The end of dependency hell
Quarterly releases	Build caches 10X build time improvement
Turnkey stack A new user experience	https://e4s.io
Post-ECP Strategy LSSw, ASCR Task Force	

https://e4s.io

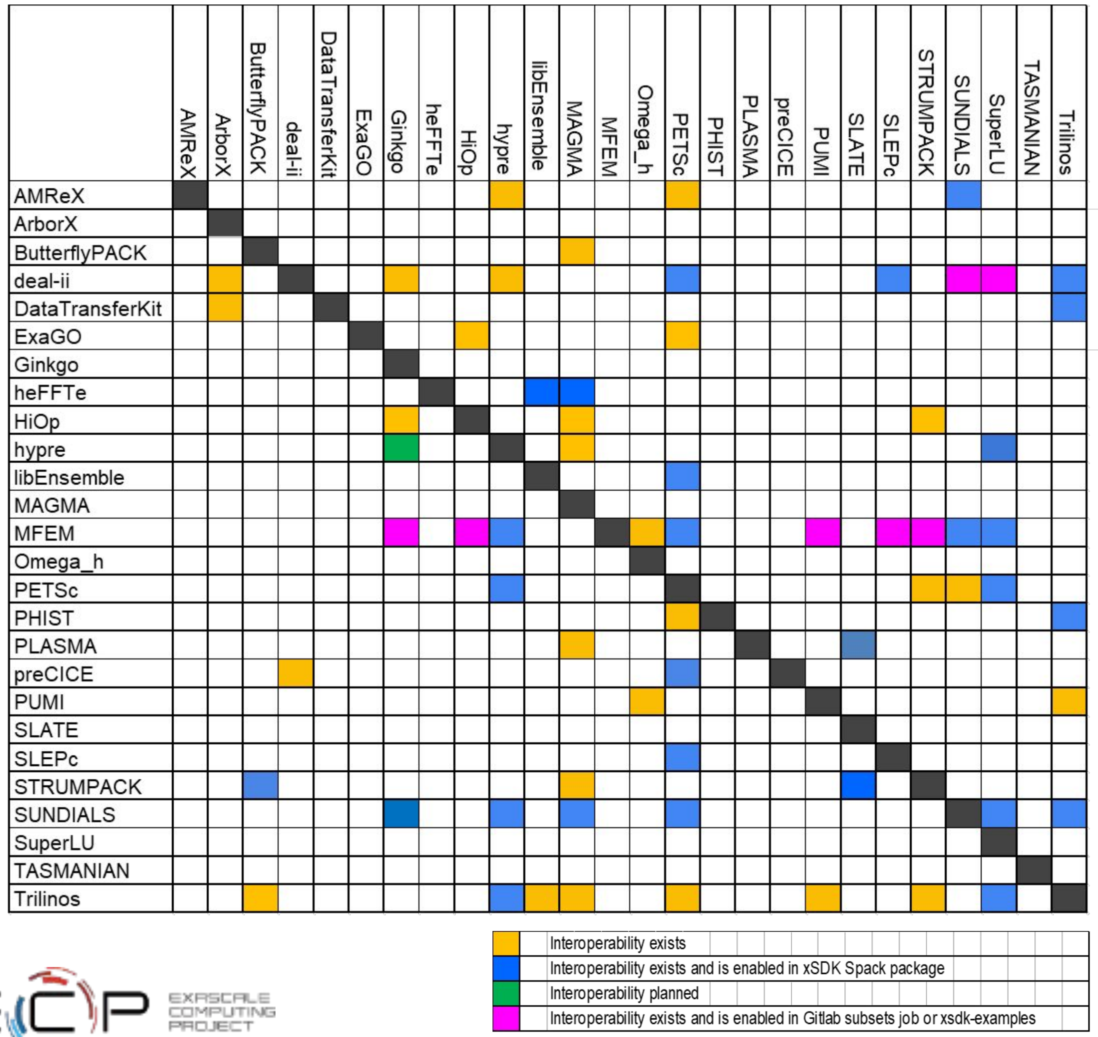
E4S lead: Sameer Shende (U Oregon)

Spack https://spack.io

Spack lead: Todd Gamblin (LLNL)

One of the most important strategies to ensure robustness in E4S is the multi-layer testing framework

Math SDK Interoperability Matrix



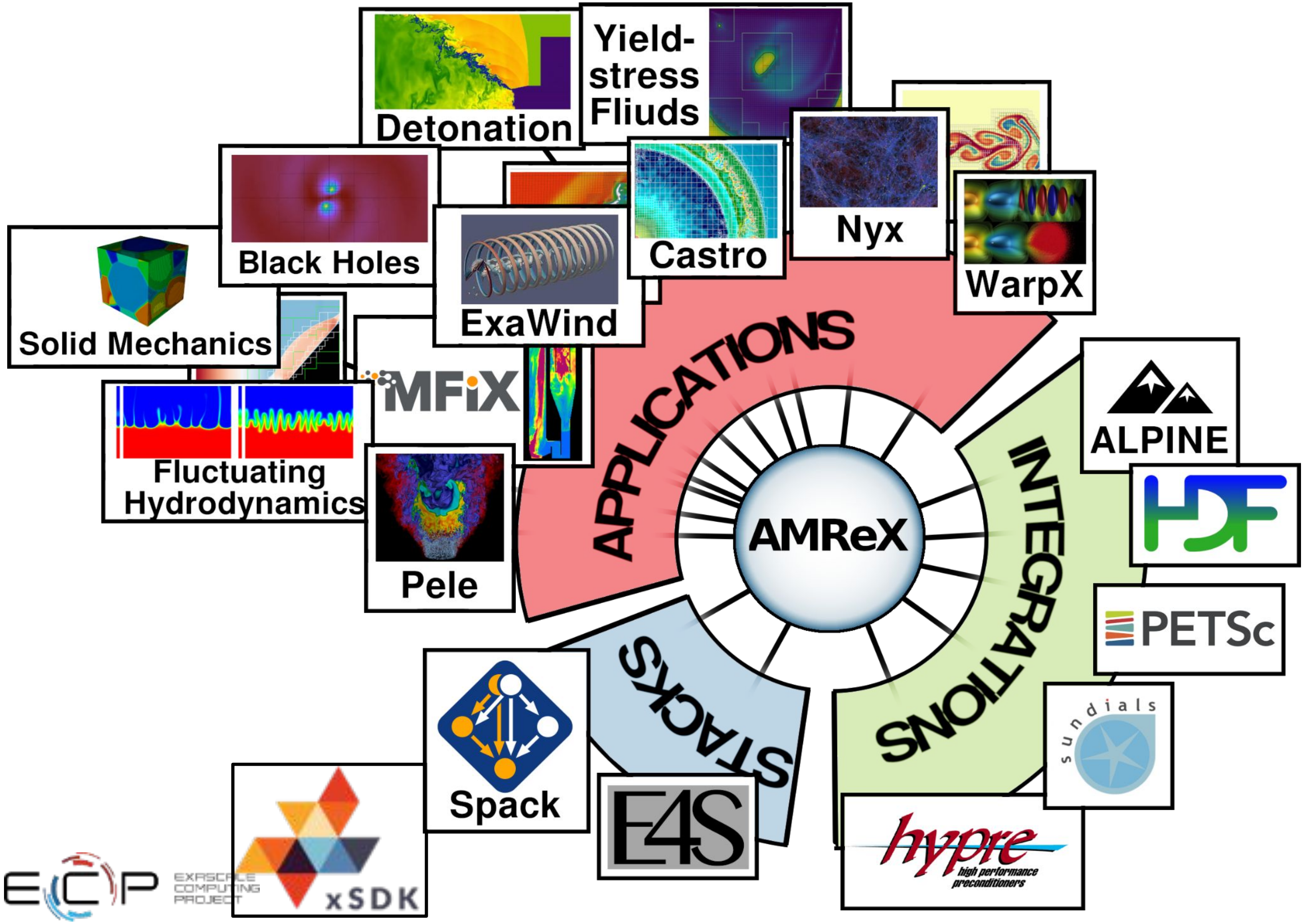
Multi-layered testing

- Testing strategies of the individual SDK libraries
- Testing of the interfaces between libraries
- Test subsets of various interoperable packages in combination
- Test the whole SDK (final level)

Because it is tested as a complete set and all works together, users can be confident any subset will compile and work together

EXASCALE COMPUTING PROJECT

Integration of multiple technologies allow users of higher level frameworks, e.g., AMReX, access many different technologies



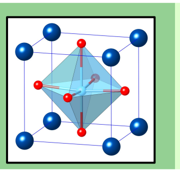
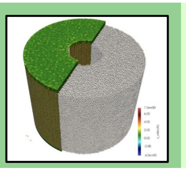
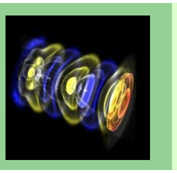
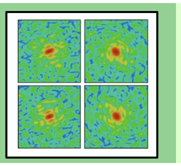
Software Integrations:

- SUNDIALS – Chemical reactions, time integrators
- Hypre, PETSc – Linear solvers on mesh data
- In-situ: Ascent, Sensei
- Offline visualization: VisIt, Paraview, yt
- IO: HDF5, ADIOS

Software Stacks:

- Spack
  - Smoke test – CUDA, AMD HIP
- xSDK
- E4S

A small sampling of ECP integration examples

ExaWind: advanced wind farm modeling		Sparse linear solvers in hypre and Trilinos optimized for strong scaling and GPU performance
QMCPACK: quantum Monte Carlo for materials		Batched dense linear algebra kernels significantly improved GPU performance
ExaSGD: power grid optimization		Optimize sparse indefinite solvers developed and optimized for large-scale grid problems
ExaSMR: small modular reactor modeling		Custom discretization designed and tuned for specific reactor assembly
WarpX: plasma wakefield accelerator design		Adaptive mesh data structures and solvers highly optimized for GPU performance
ExaFEL: real-time light source analysis and reconstruction		Non-uniform FFTs designed to minimize data motion

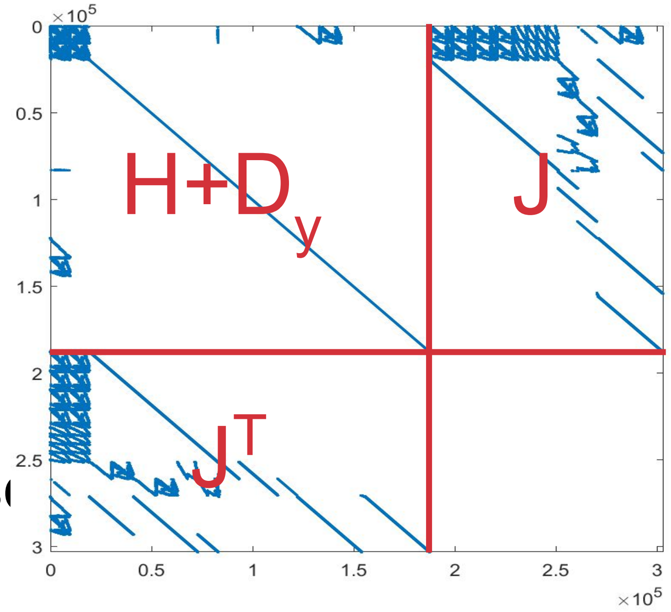
EXASCALE COMPUTING PROJECT

ExaSGD helps ensure a more reliable and robust power grid to better leverage renewable energy sources

PI: Chris Oehmen, PNNL

- **Objective**: Optimize power grid operation and control to reliably incorporate intermittent energy sources such as wind and solar while minimizing disruptions
- **ECP Scope:**
  - Enable “what if” evaluation of complex damage from extreme weather or cyber attack by optimizing with many weather scenarios and complex disruptions (“contingencies”)
  - Allow for flexibility to run on laptops to exascale computers; highly advanced numerical algorithms leverage accelerated computing
- New Solver strategies for GPUs for large sparse systems of equations were required

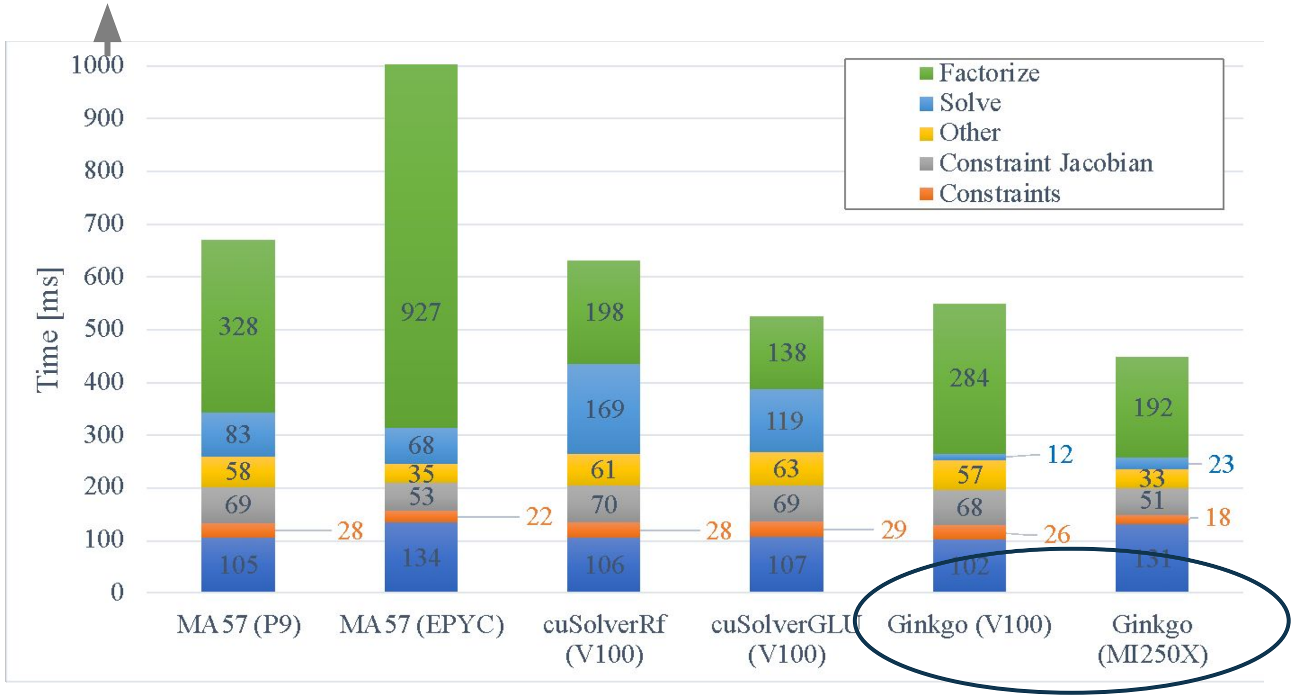
Can analyze Western power grid operations, with multiple scenarios and 1000’s of contingencies to help operators plan for and respond to emergencies



Typical sparsity pattern of optimal power flow matrices: No obvious structure that can be used by linear solver. Needed to solve long sequences of such systems

Collaboration effort between SuperLU, Gingko and ExaSGD teams produced new direct sparse solvers

- Used non-supernodal structures for GPUs
- Extract parallelism for extreme sparsity of the graph; find level sets of coefficients that can be processed simultaneously
- Cholesky for symmetric, LU for non-symmetric
- Results:
  - Each GPU solution outperforms all CPU baselines
  - Ginkgo performance improves on a better GPU
  - Iterative refinement configuration affects linear solver performance and optimization solver convergence



Ginkgo provides the first portable GPU-resident sparse direct linear solver for non-supernodal systems

EXASCALE COMPUTING PROJECT

We are capturing and sharing our lessons learned in performance portability, collaboration and software ecosystem development

Lessons Learned: Performance portability

- Performance portability, programming models, strategies for increasing arithmetic intensity, refactoring code, new algorithm design, etc..
- Strategies for moving from GPU-accelerated to GPU-resident
- Library support for diverse application node programming choices is possible
- When facing an inflection point in the HW, S/W investment must be a first class citizen
- Node-level solutions apply at all levels of computing

Lessons Learned: Collaboration

- The value of diverse, multi-disciplinary teams
- Collaborative solutions can't be dictated but they can be incentivized
- Build integration into project structure and measures of success
- High quality software is the foundation for collaboration in scientific computing
- Open, frequent comms (good/bad/ugly) imperative with sponsors, stakeholders, staff

Lessons Learned: Software ecosystem

- A hierarchical, process-driven approach to scientific library & tool development can work
- The concept of software as a facility has promise
- Having multiple competing implementations is ideal
- "Off the shelf" is rarely sufficient

What's next for ECP software technologies? PESO

PESO: Partnering for Scientific Software Ecosystem Stewardship Opportunities

PESO Partnerships

Stakeholder Engagement and Consortium Partnerships

Consortium
- Spack support
- Products in E4S
- SW practices
- PIER activities

Applications Community
- Spack support
- E4S support

Computing Facilities
- Spack support
- E4S integration
- E4S user support

Commercial HPC Companies
- Engage in joint activities to advance the scientific software ecosystem
- Develop business models to further partnerships
- Increase release coordination activities

US Agencies
- Engage in business model discussions & plans for use of E4S
- Work with commercial providers to establish a support model
- Explore opportunities for joint product development

Industrial Users
- Engage in business model discussions & plans for use of E4S
- Work with commercial providers to establish a support model
- Explore models for mixed open-proprietary software stacks

Community Development

Broadening Participation of Underrepresented Groups in DOE Computing Sciences
- Coordinate consortium crosscutting-layer PIER planning
- Seek support for Sustainable Research Pathways Program
- Lead HPC Workforce Development and Retention Action Group

Better Scientific Software (BSSw) Fellowship Program
- Coordinate BSSw Fellowship Program – which gives recognition and funding to leaders and advocates of high-quality scientific software
- Seek sustainable support for BSSw Fellows, Honorable Mentions, travel, and program management for 2025 and beyond

PESO Services

Integration Partnerships

Provide resources and support for portfolio build, integration, and testing capabilities
- Spack integration
- CI testing
- Portfolio support & management

in collaboration with & co-funded by SSOs
- On-node & inter-node programming systems (w. S4PST)
- Math libraries, Data & viz, ML/AI (w. OASIS)
- Tools (w. STEP), Workflows (w. SWAS)
- NNSA software (funded by NNSA)

SQA & Security

Provide infrastructure to support and leverage product team SQA efforts
- Supply chain, Product quality
- Testing, Documentation

PESO Products

E4S
- Support for product integration
- E4S website
- Documentation, Training

Spack
- Features for consortium products
- Documentation, Training

Port & Test Platforms
- Frank test & development system
- Cloud resources
- Documentation, training

BSSw.io Content (w. COLABS)
- Short articles on topics related to scientific software productivity and sustainability (recruit, write, review, & edit)

PESO is part of the larger Consortium for the Advancement of Scientific Software funded by DOE

- **PESO:** Stewarding, evolving and integrating a cohesive ecosystem for DOE software
- **RAPIDS & FASTMath:** Stewardship, advancement, integration for math, data/vis, and ML/AI packages.
- **SWAS:** Stewardship and project support for scientific workflow software and its community
- **S4PST:** Stewardship, advancement and engagement for programming systems.
- **STEP:** Stewardship, advancement of software tools for understanding performance and behavior.
- **COLABS:** Training, workforce development, and building the RSE community.
- **CORSA:** Partnering with foundations to provide onboarding paths for DOE-funded software.

Addressing the stewardship needs of the DOE ASCR scientific software ecosystem

Goal: to ensure the long-term viability of the ASCR software ecosystem so that it may continue to serve as the base for future DOE-funded research.

Member organizations have distinct and complementary foci

https://cass.community

The Exascale Computing Project has delivered on its mission needs

Deliver a long-term, **sustainable software ecosystem** that can be used and maintained for years to come

- E4S deployed at HPC facilities around the US and the world
- 76 HPC products available for computing at all scales
- Performance portability tools developed and widely used

Promote the **health of the US HPC industry**

- Six vendors funded under PathForward; outcomes realized in exascale systems
- Accelerator-based computing lowers cost of energy across the board
- The ECP Industry and Agency Council stimulates consumption of HPC resources

Ensure that exascale systems can be used to deliver **mission-critical applications**

- ECP applications demonstrate outstanding performance and capabilities at exascale
- Previously unattainable results in real-world challenge problems
- ECP lessons learned pave the way for many additional applications to leverage accelerator-based computing

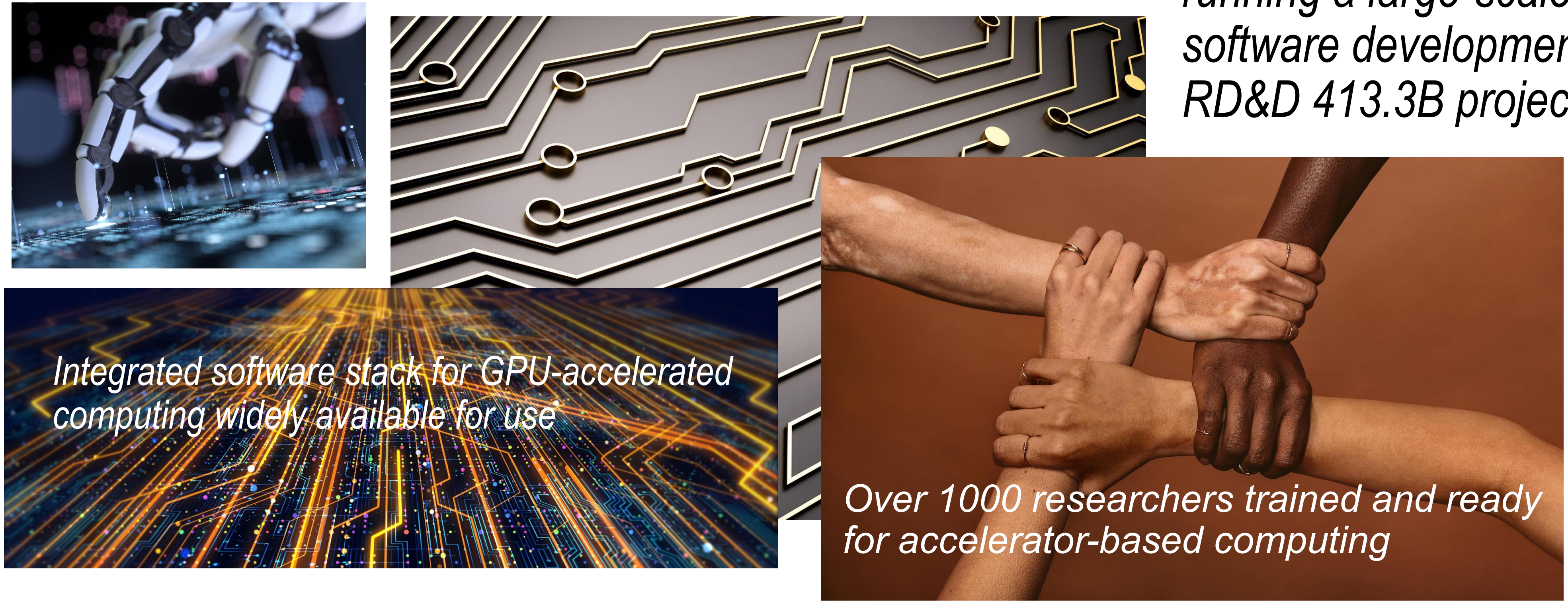
Maintain **international leadership in HPC**

- Frontier is the world's first exascale machine – in part due to ECP/ECI investments
- Aurora is the world's second exascale system
- 1000+ researchers trained in GPU computing

ECP’s impact and legacy will be long-lived and far-reaching

A suite of applications that will impact problems of national importance for decades to come

Best practices for running a large-scale software development RD&D 413.3B project



Best practices and lessons learned for thinking about how to program GPUs – moving the nation forward

Thank you

https://www.exascaleproject.org

This research was supported by the Exascale Computing Project (17-SC-20-SC), a joint project of the U.S. Department of Energy's Office of Science and National Nuclear Security Administration, responsible for delivering a capable exascale ecosystem, including software, applications, and hardware technology, to support the nation's exascale computing imperative.

ECP Director: Lori Diachin
ECP Deputy Director: Ashley Barker

Thank you to all collaborators in the ECP and broader computational science communities. **The work discussed in this presentation represents creative contributions of many people who are passionately working toward next-generation computational science.**

Questions?