ECP Update and Future Sustainability Efforts

Michael A. Heroux
Director of Software Technology, Exascale Computing Project
PI, PESO, Post-ECP Software-Ecosystem Sustainment Project
Senior Scientist, Sandia National Laboratories

HPC User Forum
Tucson, AZ
September 6, 2023

ECP Status

KPP-1 Definition

- KPP-1 is based on a Figure of Merit (FOM) defined individually for each project to capture the relevant **scientific work rate** for an application.
- Goal of KPP-1 is to measure the overall impact of ECP project, including both hardware-driven and algorithmic improvement.
- Each application measured a **baseline FOM value** at the inception of ECP.

KPP-1 Threshold

50% of KPP-1 applications have a Figure of Merit improvement ≥50

KPP-1 Objective

100% of KPP-1 applications have a Figure of Merit improvement ≥50

- KPP-1 is calculated as the ratio of the FOM **on the exascale challenge problem** to the baseline

$$KPP\text{-}1 = \frac{FOM_{exascale}}{FOM_{baseline}}$$

- The FOM ratio is measured throughout the project to track progress.
- KPP-1 success is determined by an external SME review at end of project.

ECP EXASCALE COMPUTING PROJECT

Status of KPP-1 applications on Frontier

Threshold: 6/11

KPP in progress	KPP completed	KPP submitted	KPP signed (SME)	KPP signed (FPD)
NWChemEx		LatticeQCD	WarpX	ExaSMR
QMCPACK		EXAALT		WDMApp
		NEW E3SM-MMF		ExaSky
		CANDLE		EQSIM

EXASCALE COMPUTING PROJECT

KPP-2 Definition

- KPP-2 is based on developing new mission-critical capabilities at exascale. Unlike KPP-1 applications, a well-defined baseline was not available at the inception of ECP.
- To meet KPP-2 an application must successfully execute a capability demonstration of the challenge problem on an exascale platform.
- All KPP-2 challenge problems were externally reviewed and determined to require exascale-level compute resources to execute.

KPP-2 Threshold
50% of KPP-2 applications can execute their exascale challenge problem

KPP-2 Objective
100% of KPP-2 applications can execute their exascale challenge problem

- KPP-2 success is determined by an external SME review at end of project. KPP-2 projects must
  - Demonstrate all new capability in place to meet challenge problem specification and utilize full exascale machine
  - Demonstrate reasonably efficient port to exascale machine (uses all accelerator nodes, etc.)
  - Execute demonstration calculation on target exascale platform.

Status of KPP-2 applications on Frontier

Threshold: 5/10

KPP in progress	KPP completed	KPP submitted	KPP signed (SME)	KPP signed (FPD)
ExaWind		GAMESS	ExaBiome	Combustion-PELE
ExaStar		ExaAM		MFIX-Exa
Subsurface		ExaSGD (NEW)		
ExaFEL				

ECP EXASCALE COMPUTING PROJECT

KPP-3 Definition

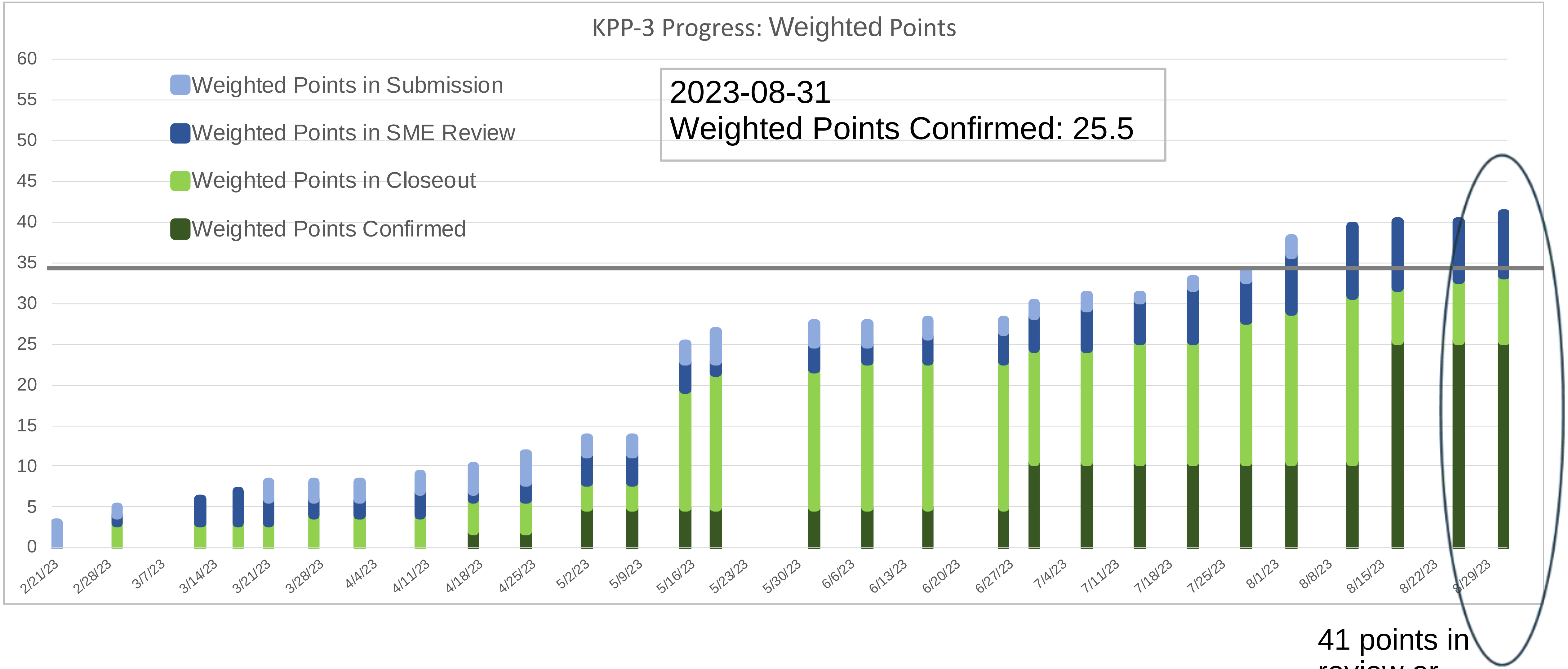
- KPP-3 is based on **integrations:**
  - developing a **significant new feature** that is
  - demonstrated in the **exascale environment** and
  - **sustainably integrated** for future use
- All KPP-3 integrations are externally reviewed

KPP-3 Threshold
34 of 68 possible points achieved

KPP-3 Objective
68 of 68 possible points achieved

- KPP-3 progress is determined by external SME reviews as integrations are achieved
  - A products accrues an unweighted **KPP-3 point** by demonstrating 4 (in a few cases 8) integrations
  - 70 libraries and tools were tracked with weights of 0.5, 1.0 and 2.0 depending on impact
  - Total unweighted points possible: 70 (number of products)
  - Total weighted points possible: 68

KPP3 Integration Progress Update – 2023/08/31



41 points in review or approved

ECP KPP Status Summary (Aug 31, 2023)

KPP ID	Approved	In Review	Total Approved or In Review		Needed for Threshold	Needed for Objective
KPP-1	4	5	9	≥	6	11
KPP-2	2	4	6	≥	6	10
KPP-3	25.5	15.5	41	≥	34	68
KPP-4	267	0	267	≥	214	267

- KPP-4 was completed some time ago
- Assuming successful review of submitted artifacts (reasonable):
  - KPP-1, KPP-2, KPP-3 will reach threshold
  - **ECP will achieve all KPPs, succeed as a project**

ECP Impact – Portable Libraries and Tools for Accelerators

ECP Software Technology works on products that apps need now and in the future

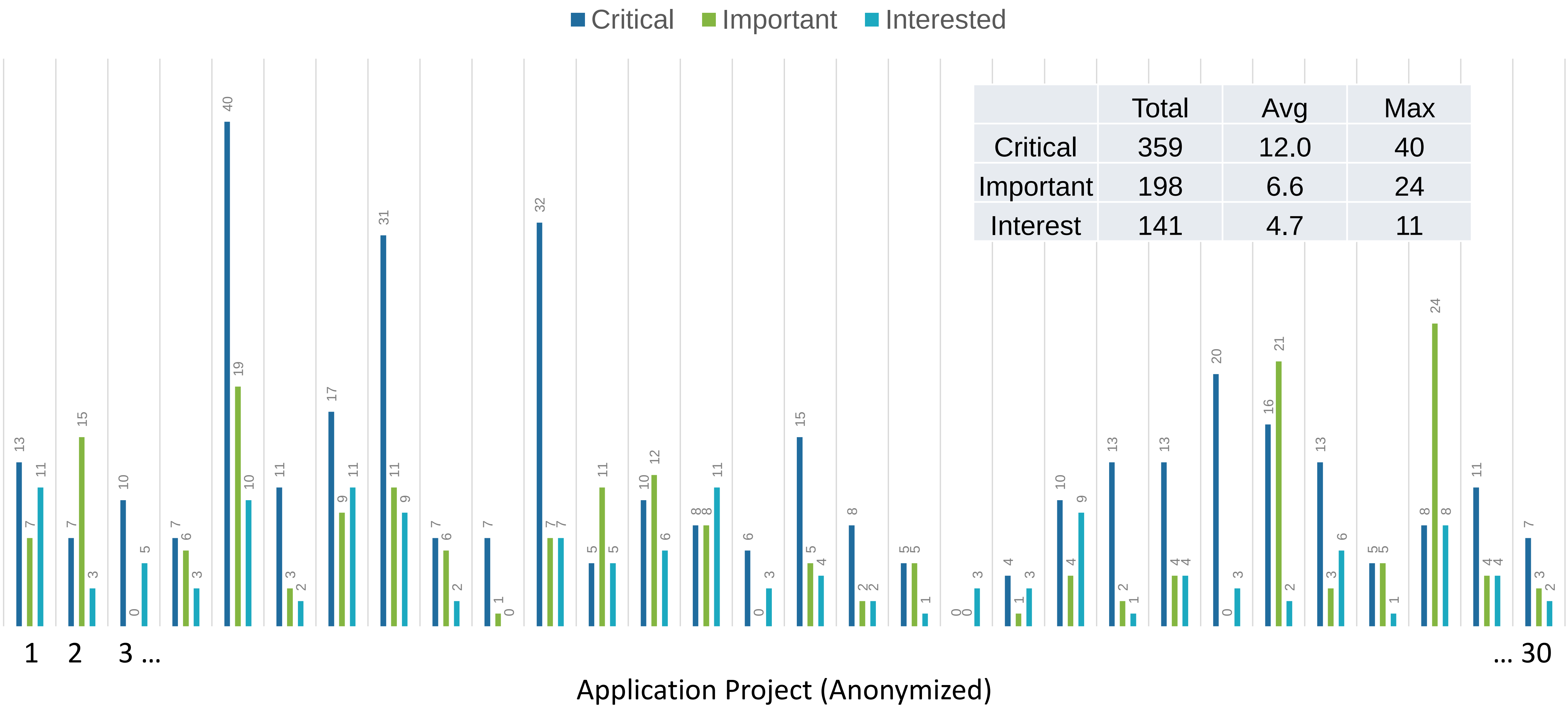
Key themes:
- Focus: GPU node architectures and advanced memory & storage technologies
- Create: New high-concurrency, latency tolerant algorithms
- Develop: New portable (Nvidia, Intel, AMD GPUs) software product
- Enable: Access and use via standard APIs

Software categories:
- **Next generation established products:** Widely used HPC products (e.g., MPICH, OpenMPI, PETSc)
- **Robust ~~emerging~~ products:** Address key new requirements (e.g., Kokkos, RAJA, Spack)
- **New products:** Enable exploration of emerging HPC requirements (e.g., zfp, Variorum)

Legacy: A stack that enables performance portable application development on leadership platforms

Example Products	Engagement
MPI – Backbone of HPC apps	Explore/develop MPICH and OpenMPI new features & standards
OpenMP/OpenACC –On-node parallelism	Explore/develop new features and standards
Performance Portability Libraries	Lightweight APIs for compile-time polymorphisms
LLVM/Vendor compilers	Injecting HPC features, testing/feedback to vendors
Perf Tools - PAPI, TAU, HPCToolkit	Explore/develop new features
Math Libraries: BLAS, sparse solvers, etc.	Scalable algorithms and software, critical enabling technologies
IO: HDF5, MPI-IO, ADIOS	Standard and next-gen IO, leveraging non-volatile storage
Viz/Data Analysis	ParaView-related product development, node concurrency

THE NUMBER OF ECP SOFTWARE TECHNOLOGY PROJECT DEPENDENCIES FOR EACH ECP APPLICATION PROJECT (ANONYMIZED)



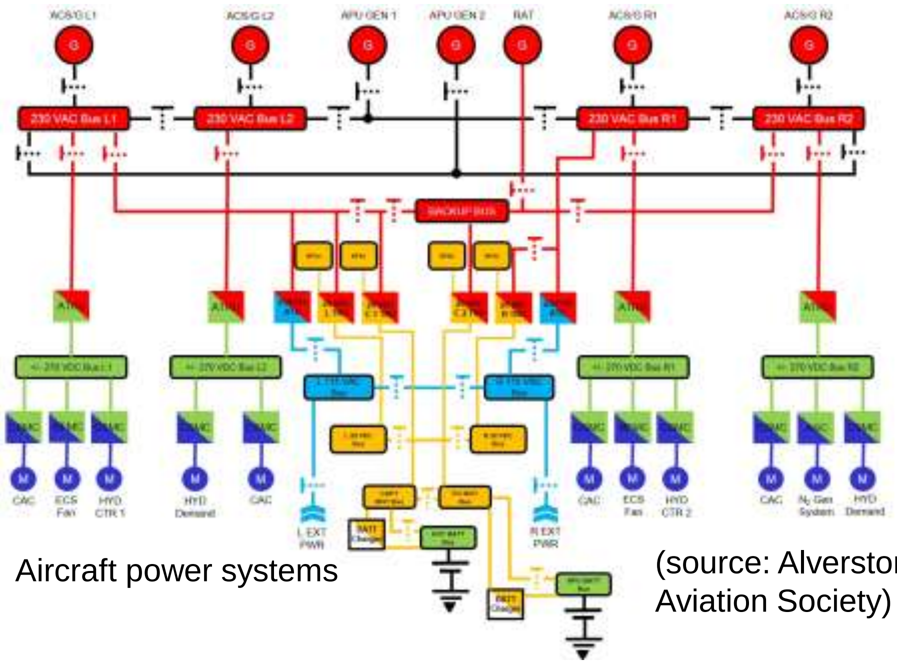
	Total	Avg	Max
Critical	359	12.0	40
Important	198	6.6	24
Interest	141	4.7	11

DOE apps (and many apps outside of DOE) rely on ECP-developed libraries and tools for GPUs

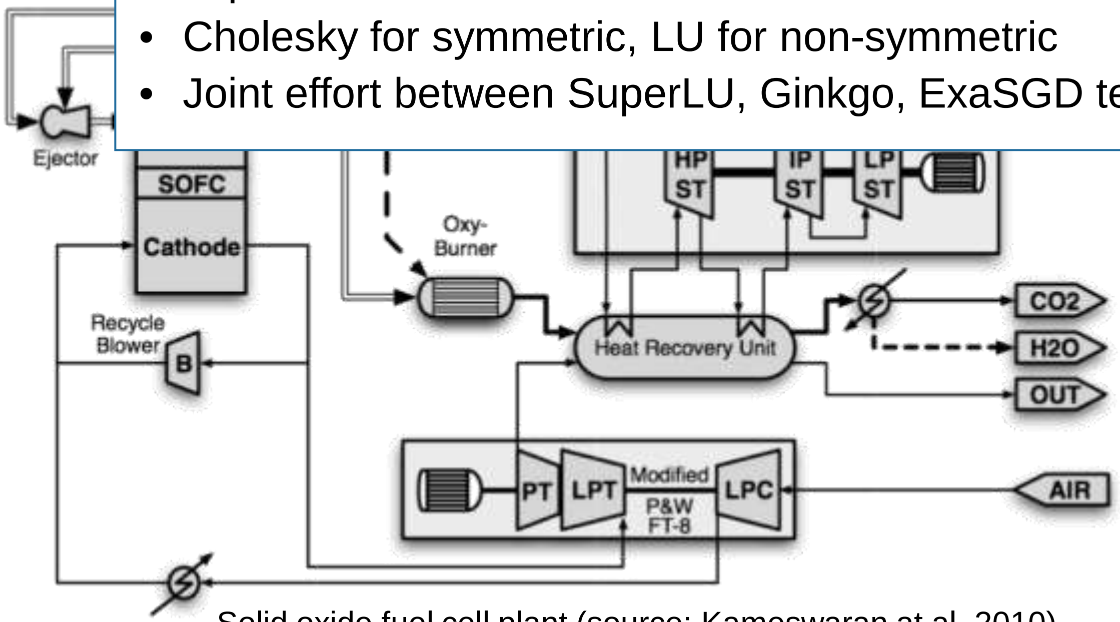
EXASCALE COMPUTING PROJECT

Example: Systems Engineering

- ExaSGD addresses systems engineering problems
- Produced new direct sparse solvers using non-supernodal structures, for GPUs
- Cholesky for symmetric, LU for non-symmetric
- Joint effort between SuperLU, Ginkgo, ExaSGD teams

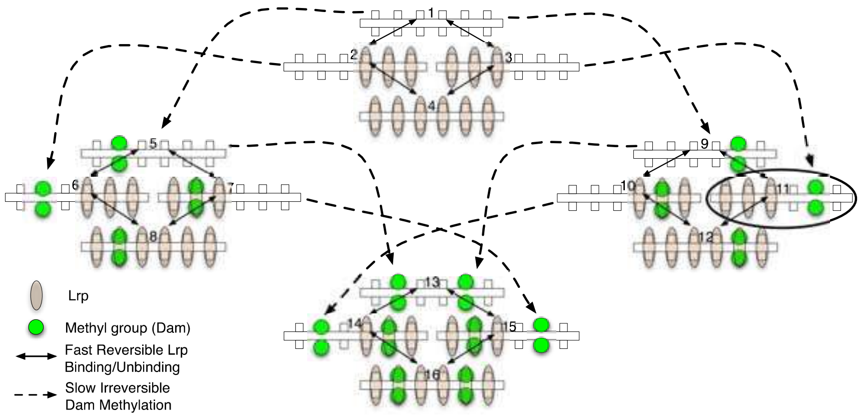


Aircraft power systems (source: Alverstone Aviation Society)

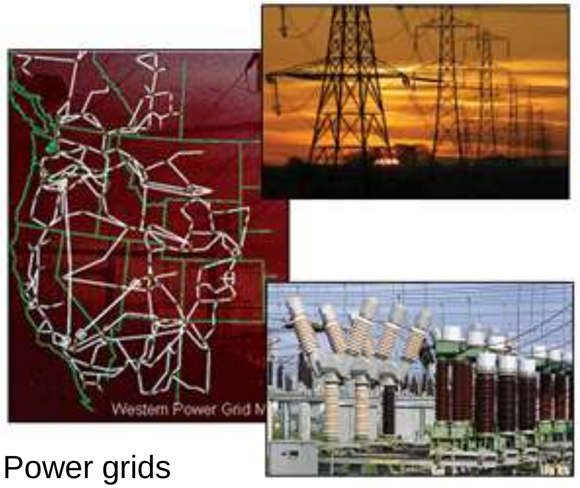


Solid oxide fuel cell plant (source: Kameswaran at al. 2010)

Buildings (source: EEB Hub, B661 2014)



Gene regulatory networks (source: Peles et al. 2006)



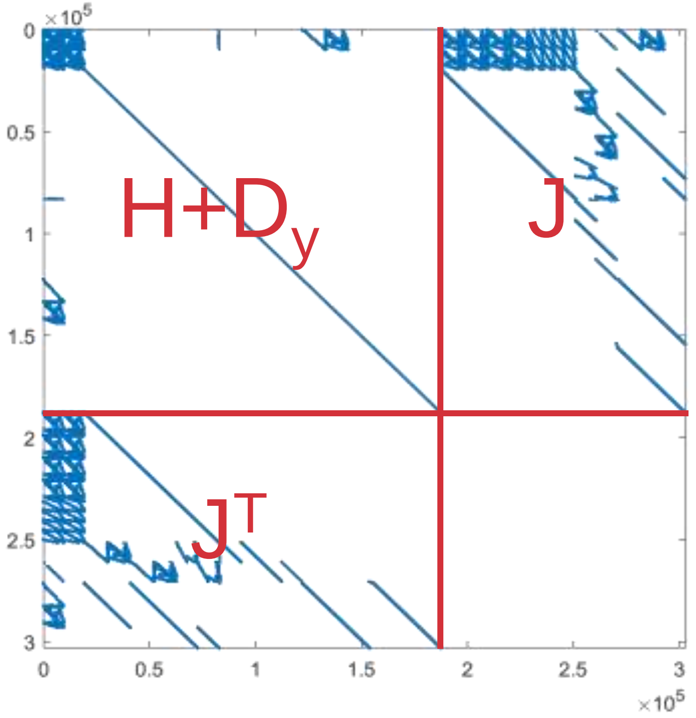
Power grids (source: PNNL)

Underlying KKT Linear System Properties

- Security constrained optimal power flow analysis
- The interior method strategy leads to symmetric indefinite linear systems

$$\overbrace{\begin{bmatrix} H + D_y & J \\ J^T & 0 \end{bmatrix}}^{K_k} \overbrace{\begin{bmatrix} \Delta y \\ \Delta \lambda \end{bmatrix}}^{\Delta x_k} = \overbrace{\begin{bmatrix} r_y \\ r_\lambda \end{bmatrix}}^{r_k},$$

  - J – sparse constraints Jacobian,
  - H – sparse Hessian,
  - D_y – arises from log-barrier function



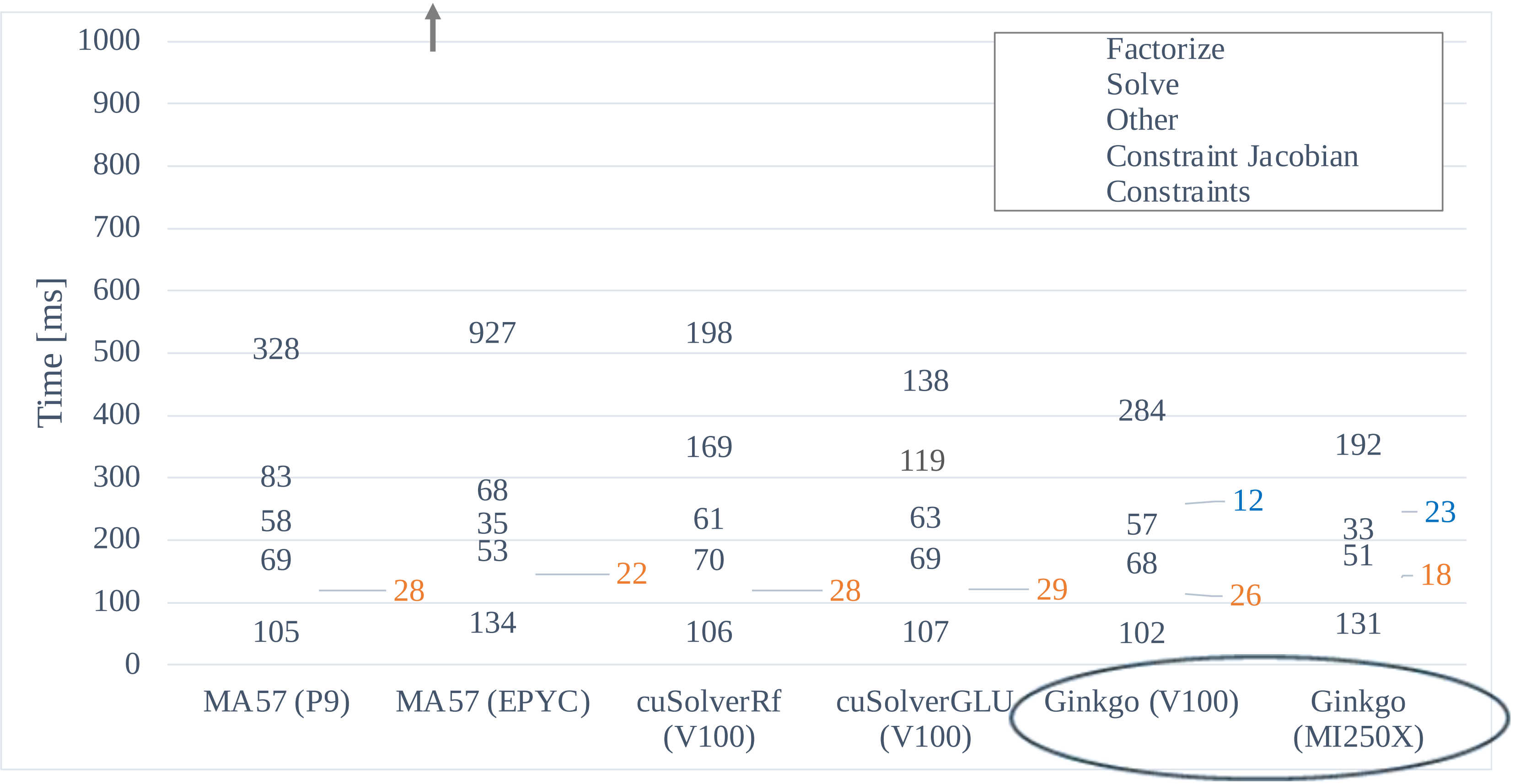
Typical sparsity pattern of optimal power flow matrices: No obvious structure that can be used by linear solver.

- The challenge: we need to solve a long sequences of such systems

Linear Solver Performance within Optimization Algorithm

Average per iteration times (including first iteration on CPU)

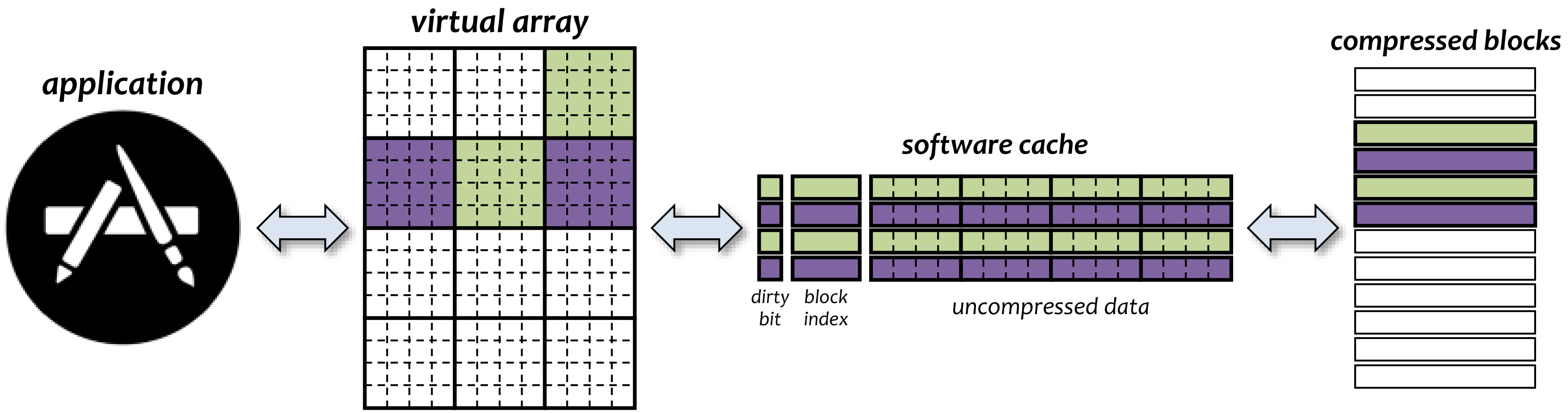
- Each GPU solution outperforms all CPU baselines
- Ginkgo performance improves on a better GPU
- Iterative refinement configuration affects linear solver performance and optimization solver convergence



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

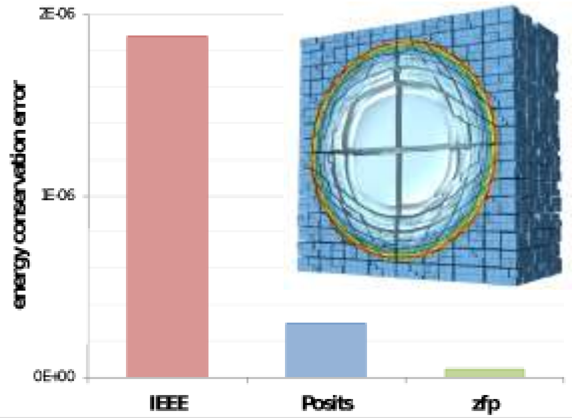
EXASCALE COMPUTING PROJECT

Example: Addressing growing gap of ops vs bw vs memory
ZFP compressed multidimensional array primitive



- Fixed-length compressed blocks enable fine-grained read & write **random access**
  - C++ compressed-array classes hide complexity of compression & caching from user
  - User specifies per-array storage footprint in bits/value
- Absolute and relative **error tolerances** supported for offline storage, sequential access
- Fast, hardware friendly, and parallelizable: **150 GB/s throughput** on NVIDIA Volta
- **HPC tool support**: ADIOS, CEED, CONDUIT, HDF, intel, VTK-m

Inline compression boosts solution accuracy by 40x over IEEE in CEED code



And Many More…

- ECP generated a
  - Collection of portable GPU-capable libraries and tools for AMD, Intel, and NVIDIA devices
  - Designed for future adaptation to next-generation highly-concurrent node architectures
  - Foundation for others who will make the transition from CPU to GPU and beyond
- Appendix of 11 slides are leave-behind as a sample of the 70 products ECP has contributed to

ECP Libraries and Tools – as a portfolio

ST's Extreme-scale Scientific Software Stack (E4S) is a key ECP product to sustain and evolve

- 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**
- Leverages and enhances SDK interoperability thrust
- Not a commercial product – an open resource for all
- Growing functionality: August 2023: E4S 23.08 – 100+ full release products

Community Policies Commitment to SW quality	**Curated collection** The end of dependency hell
Turnkey stack A new user experience	**DocPortal** Single portal to all E4S product info
Quarterly releases Release 22.2 – February	https://e4s.io
Portfolio testing Especially leadership platforms	**Build caches** 10X build time improvement
Post-ECP Strategy LSSw, ASCR Task Force	

https://spack.io

Spack lead: Todd Gamblin (LLNL)

https://e4s.io

E4S lead: Sameer Shende (U Oregon)

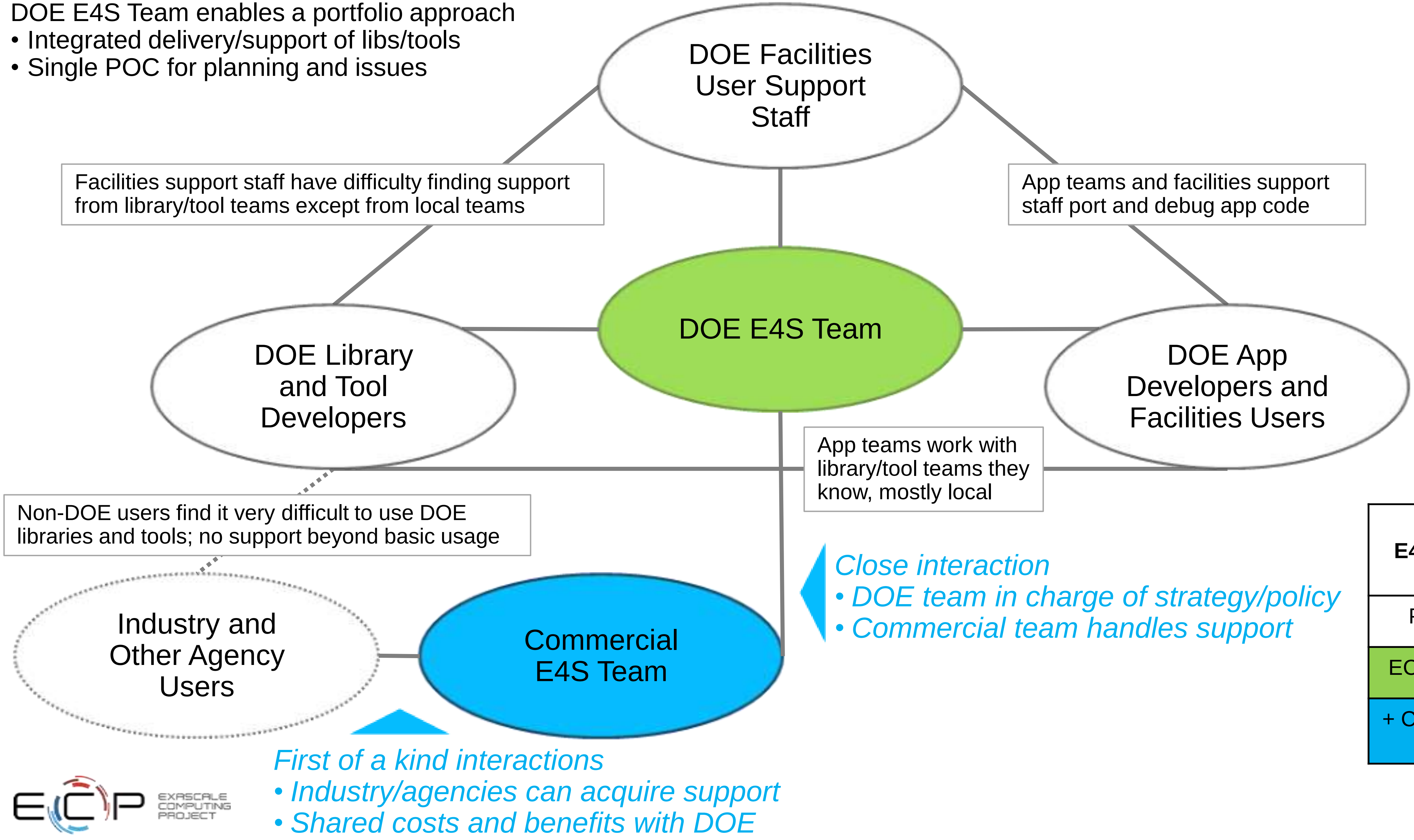
Also includes other products, e.g.,
AI: PyTorch, TensorFlow, Horovod
Co-Design: AMReX, Cabana, MFEM

Spack

- E4S uses the Spack package manager for software delivery
- Spack provides the ability to specify versions of software packages that are and are not interoperable
- Spack is a build layer for not only E4S software, but also a large collection of software tools and libraries outside of ECP ST
- Spack supports achieving and maintaining interoperability between ST software packages
- https://spack.io

E4S Business Model: Optimize Cost & Benefit Sharing

DOE E4S Team enables a portfolio approach
- Integrated delivery/support of libs/tools
- Single POC for planning and issues



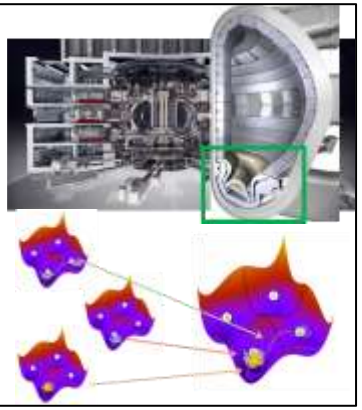
E4S Phase	Cost & Benefit Scope
Pre-E4S	Local Facility
ECP support	DOE complex
+ Commercial support	Universal

Leveraging the Potential of ECP Investments

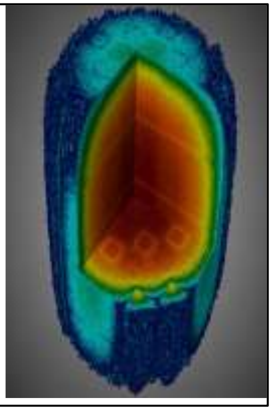
100X

100X Demonstrated: ECP-sponsored application FOMs

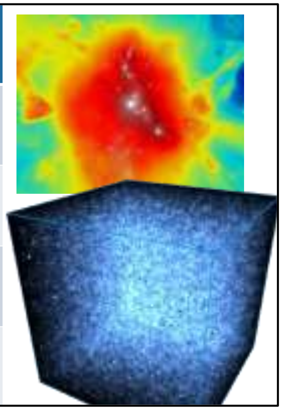
Project/PI	EXAALT: Molecular Dynamics Danny Perez
Challenge Problem	Damaged surface of Tungsten in conditions relevant to plasma facing materials in fusion reactors • 100,000 atoms • T=1200K
FOM Speedup	398.5
Nodes Used	7000
ST/CD Tools	Used in KPP Demo: **Kokkos, CoPa**



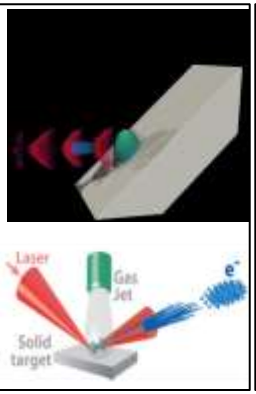
Project/PI	ExaSMR: Small Modular Reactors Steve Hamilton
Challenge Problem	NuScale-style Small Module Reactor (SMR) with depleted fuel and natural circulation • 213,860 Monte Carlo tally cells/6 reactions • 5.12×10^{12} particle histories/cycle, 40 cycles • 1098×10^{6} CFD spatial elements • 376×10^{9} CFD degrees of freedom • 1500 CFD timesteps
FOM Speedup	70
Nodes Used	6400
ST/CD Tools	Used in KPP Demo: **CEED** Additional: Trilinos



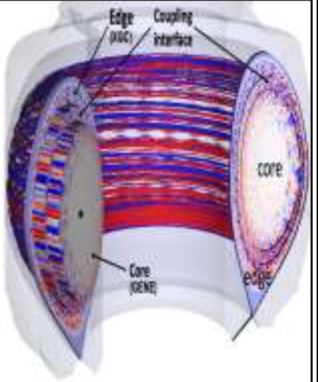
Project/PI	ExaSky: Cosmology Salman Habib
Challenge Problem	Two large cosmology simulations • gravity-only • hydrodynamics
FOM Speedup	271.65
Nodes Used	8192
ST/CD Tools	Used in KPP demo: **none** Additional: CoPa, VTK-m, CINEMA, HDF5.0



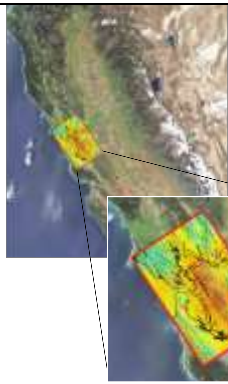
Project/PI	WarpX: Plasma Wakefield Accelerators Jean-Luc Vay
Challenge Problem	Wakefield plasma accelerator with a 1PW laser drive • 6.9×10^{12} grid cells • 14×10^{12} macroparticles • 1000 timesteps/1 stage
FOM Speedup	500
Nodes Used	8576
ST/CD Tools	Used in KPP Demo: **AMReX**, **libEnsemble** Additional: ADIOS, HDF5, VTK-m, ALPINE



Project/PI	WDMApp: Fusion Tokamaks Amitava Bhatacharjee
Challenge Problem	Gyrokinetic simulation of the full ITER plasma to predict the height and width of the edge pedestal
FOM Speedup	150
Nodes Used	6156
ST/CD Tools	Used in KPP Demo: **CODAR**, **CoPA**, **PETSc**, **ADIOS** Additional: VTK-m



Project/PI	EQSIM: Earthquake Modeling and Risk Dave McCallen
Challenge Problem	Impacts of Mag 7 rupture on the Hayward Fault on the bay area.
FOM Speedup	3467
Nodes Used	5088
ST/CD Tools	Used in KPP Demo: **RAJA, HDF5**



ECP investments enabled a 100X improvement in capabilities

- **7 years** building an **accelerated**, **cloud-ready** software ecosystem
- Positioned to utilize **accelerators from multiple vendors** that others cannot
- **Emphasized software quality**: testing, documentation, design, and more
- Prioritized **community engagement**: Webinars, BOFs, tutorials, and more
- DOE **portability layers** are the credible way to
  - Build codes that are sustainable **across multiple GPUs** and
  - **Avoid vendor lock-in**
  - **Avoid growing divergence** and hand tuning in your code base
- ECP software can **lower costs** and **increase performance** for **accelerated** platforms
- Outside of AI, industry has not caught up
  - DOE enables an entirely different class of applications and capabilities to use accelerated nodes
  - In addition to AI
- **ECP legacy: A path and software foundation for others to leverage**

Efficiently utilizing GPUs goes far beyond typical code porting

Port Code

- Rewrite, profile, and optimize
- Memory coalescing
- Loop ordering
- Kernel flattening

Adapt Numerics

- Reduced synchronization
- Reduced precision
- Communication avoiding

Adapt Models

- Mathematical representation
- “On the fly” recomputing vs. lookup tables
- Prioritization of new physical models

ECP efforts have de-risked (but not removed) these activities for others

EXASCALE COMPUTING PROJECT

Opportunities to realize 100X leveraging ECP investments

- Port to full use of GPUs:
  - Hotspot use of GPUs is a start but not sufficient.
  - Scalability very limited and capped for future GPU devices
- Utilize Spack ecosystem:
  - Opens ready access to hundreds of curated libraries and tools
  - Makes your code easy to consume if you publish Spack recipes for your code
  - Utilize Spack build caches (10X speedup in rebuild times)
- Utilize E4S
  - Curated libraries, tools, documentation, build caches, and more
  - Commercial support via ParaTools
  - Pre-built containers, binaries,
  - Cloud instances for AWS, Google – Permit elastic expansion, neutral collaboration for cross-agency work
- Leverage ECP team experience:
  - Engage with DOE HPC staff

100X Recipe

- Ingredients
  - A compelling science impact story
  - $$ - $$$
  - Staff
  - Computing resources, training
  - The deliverables and experience from DOE/ECP
  - Delivered via post-ECP organizations like PESO
  - And more…
- Steps
  - Translate science story to strategy and plan – leverage experience from ECP, others
  - ID node-level parallelization strategy – CUDA, HIP, DPC++, Kokkos, RAJA, OpenMP, others
  - Survey existing libraries and tools – Vendors, E4S, others
  - Explore available platforms – DOE Facilities, cloud, others
  - Leverage existing software ecosystem – containers, Spack, others
  - Leverage software communities – Product communities, communities of practice, others
  - Construct new codes within the broader ecosystem
  - Produce new science results

More than one way to leverage 100X

- 100X can be realized as exciting new science capabilities at the high end
  - Fundamental new science on leadership platform
  - New opportunities on affordable machines that fit in current data centers
- **But can also reduce costs**
- Migration to accelerated platforms can be used to
  - Migrate a problem from an HPC cluster to a deskside or laptop systems
  - Lower your AWS monthly charges – E4S is available for container/cloud
  - Keep energy costs in check while still growing computing capabilities
- **Biggest ECP impact will be accelerating GPU transition – at all levels**

Post-ECP Software Efforts

Six seed projects

- COLABS: Collaboration for Better Software (for Science)
  – Lead Anshu Dubey, research software engineer (RSE) resources
- PESO: Toward a Post-ECP Software-Sustainability Organization
  – Lead Mike Heroux, cross-cutting hub to assist product lifecycle across aggregate ecosystem
- STEP: Sustainable Tools Ecosystem Project
  – Lead Terry Jones, performance tools ecosystem including DOE, non-DOE products
- SWAS: Center for Sustaining Workflows and Application Services
  – Lead Rafael Ferreira da Silva, workflows and app services ecosystem
- S4PST: Sustainability for Node Level Programming Systems and Tools
  – Lead Keita Teranishi, node-level programming ecosystem
- SRSI: Sustainable Research Software Institute
  – Lead Greg Watson, explore the role of software foundations to address sustainability

Resources for all projects available at Leadership Scientific Software portal: https://lssw.io

Toward a Post-ECP Software Sustainability Organization (PESO)

- Michael Heroux (Sandia National Laboratories; PI)
- James Ahrens (Los Alamos National Laboratory)
- Todd Gamblin (Lawrence Livermore National Laboratory)
- Timothy Germann (Los Alamos National Laboratory)
- Xiaoye Sherry Li (Lawrence Berkeley National Laboratory)
- Lois Curfman McInnes (Argonne National Laboratory)
- Kathryn Mohror (Lawrence Livermore National Laboratory)
- Todd Munson (Argonne National Laboratory)
- Sameer Shende (University of Oregon)
- Rajeev Thakur (Argonne National Laboratory)
- Jeffrey Vetter (Oak Ridge National Laboratory)
- James Willenbring (Sandia National Laboratories)

PESO Draft Report: https://pesoproject.org/files/2023-08-31-PESOCommunityReport-Draft-V0.1.pdf

PESO Project Brief Description

PESO will

- **Serve as a hub** for software-ecosystem sustainment efforts for DOE's open-source libraries and tools for advanced scientific computing
- **Work with**
  - **software product communities (SPCs) and**
  - **communities of practice (COPs)**
- **Provide cross-cutting services and support** that are broadly needed by developers, users, and stakeholders
- **Realize the potential of DOE library and tool investments**:
  - By taking a broad, strategic view
  - Through project growth, improved software quality and availability,
  - Through sustainable delivery, deployment, and support
  - Making real the 100X potential enabled by ECP investments, and beyond

PESO != ECP

- ECP
  - Hierarchy
  - Centralized finance org
  - Fixed set of apps, scope
  - Heavyweight reporting
  - Justified by size and design
- PESO
  - Peer collaboration – Hub
  - Leverage institutional finance orgs
  - Dynamic, adaptive scope targets
  - Tunable reporting strategy
  - Lighter weight approach

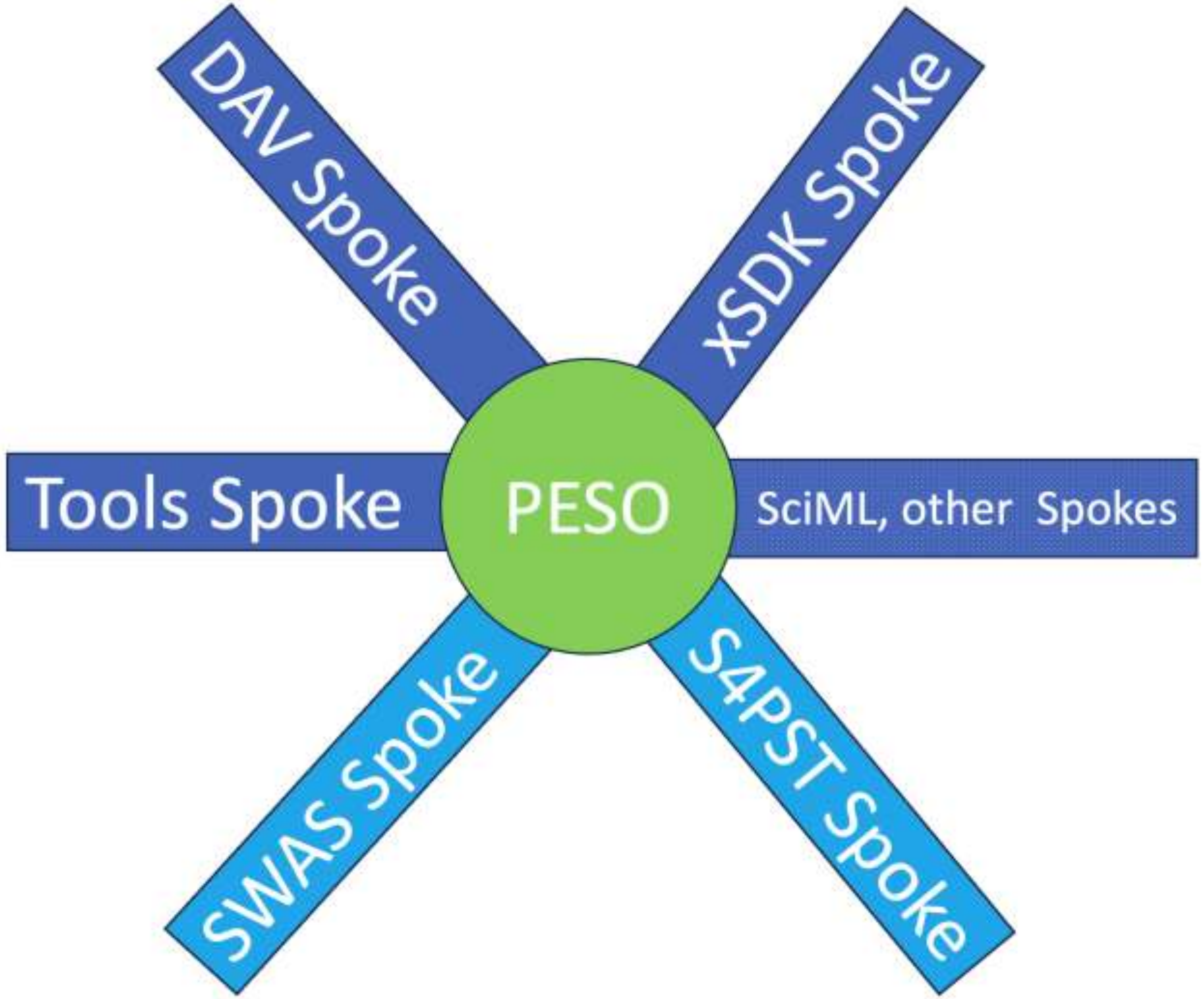
PESO > Spack+E4S

- Spack+E4S
  - Major PESO deliverable
  - Important product delivery conduits
  - Platforms for agency, industry collaboration
  - Keys for testing on new and diverse platforms & software environments

- PESO = Spack+E4S+More:
  - Impact on science via 100X efforts
  - Collaborative planning, executing tracking and reporting
  - HPC community engagement: apps, facilities, vendors, agencies
  - Cross-cutting training, community engagement, sustainability efforts
  - More

PESO Hub-and-Spoke Status

- **PESO (Hub)** – Funded seed for cross-community capabilities, engagements, services that are common to most or all software product communities
- **SWAS, S4PST, STEP** – Funded seed projects for workflows, programming systems
- **Tools, xSDK, Data and Viz (DAV)** – Existing ECP SDKs that fit the PESO approach
- **SciML** – Unmanaged product community that would fit the PESO approach
- **Others** – Some products are missing – need to ID



PESO Proposed Organization Strategy

A lot of work is best done at the product team level:

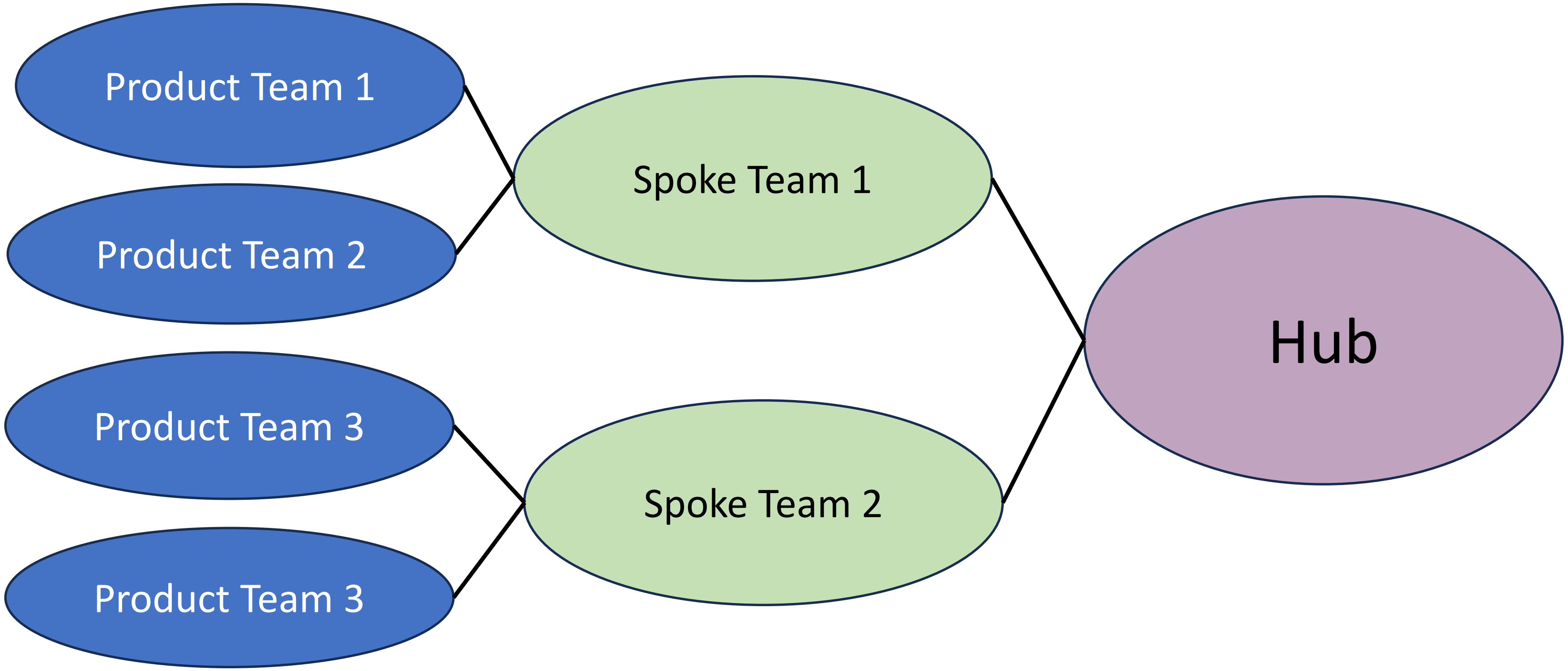
- Everyday development work
- Testing & product improvement

Some work is best done at the spoke level:

- Portfolio planning, coordination
- Holistic tutorial delivery
- Design space exploration for next-gen platforms

Some work is best done at the hub level:

- Software stack (E4S)
- Specialized CI testing
- Community-wide engagement: foundations, SQA improvements



Key goals:

- Put budget and work at the level where it can be done better, faster, and cheaper than elsewhere
- Coordinate across levels with the goal to serve product teams, users, sponsors
- Deliver a trustworthy software ecosystem

Spoke Characteristics

- Tight domain compatibility
  - Current “spokes”: Data and Viz, Math Libs, Programming systems, Tools, Workflows
  - Others as they emerge:
    - The hub-and-spoke model is scalable and malleable
    - A product might belong to more than one
  - The goal is high cohesion within spoke, loose coupling across
- Products sponsored by DOE **and** products not:
  - Commercial vendors (HPE, NVIDIA, Intel, AMD), SW companies (Kitware, ParaTools)
  - Other community products (NSF, international)
  - **These teams bring their own funding – Participate because of in-kind business value**
- Deep community collaboration
  - Leadership representation from all major institutions
  - Exploring next-gen features, computing environments, etc.
- Holistic community engagement in
  - Requirements gathering,
  - Tutorials, etc.

Core hub-and-spoke responsibilities

- Fundamental responsibilities of hub-and-spoke leadership
  - Set a vision for success for DOE-sponsored scientific libraries and tools for leadership computing
  - Define and provide capabilities for the most important DOE scientific mission applications
  - Determine cost, scope, and schedule of project activities
  - Monitor, assess, and manage cost and scope progress over time

- Scope (work description)
  - Funders need to know what work will be done with their money
  - We need a process for setting expectations, accepting proposals, selecting projects
  - PETA: Plan, execute, track, and assess
- Budget
  - Tightly coupled with scope
  - Must be adaptable because of funding uncertainty
- Schedule
  - Span of time for planning and executing work
  - Iteratively refining scope is effective

Other PESO roles: What a software sustainability organization should do to be a successful steward of DOE software

- **Funding and Stability:** Ensure a steady level of funding to support and sustain software products over the long term. It will help in retaining the talented team members and attracting new ones to participate and contribute.
- **Visibility and Engagement:** Increase visibility of software products and carefully choose engagement levels to avoid overstretching resources. Facilitate interactions between different groups developing software and help developers connect with users to understand their needs. Reach out to application developers and users actively.
- **Sustainability and Long-term Planning:** Acknowledge the fear of products not being sustained in the HPC community. Address this by explicitly sustaining products and assuring users that products will be maintained in the long run.
- **User Support and Usability:** Strive for a balance between user support and technological advancement. Ensure software is robust, has repeatable build and install instructions, and provides a clear description of usage expectations. Prioritize excellent user support in developing research libraries or tools.
- **Documentation and Best Practices:** Enforce good documentation practices, CI/CD practices, and circulate best practices across software packages. Advocate for best practices and provide conventions or standards for a uniform experience for application developers.
- **Innovation and Avoiding Monoculture:** Encourage innovation and avoid a situation where only a single dominant package/product is supported in any given category. Promote open-source software and ensure flexibility.
- **Adaptation to Technological Changes:** Adapt the software to changes in operating systems and GPU vendor software stacks. Improve software reliability, scalability, and performance based on user needs.
- **Outreach and Collaboration:** Foster community development, facilitate outreach to grow user base beyond traditional DOE HPC users, and establish mechanisms for meaningful collaborations.
- **Portability and Composability:** Ensure portability on future systems (including emerging architectures and programming models/languages), ensure composability with third-party tools.
- **Integrated Approach:** The organization should encompass various libraries and tools in an integrated manner so that end users and system administrators can deploy the stack easier.
- **Future Orientation:** Have a clear vision of the scientific communities' needs in the next decade and invest in highly performant sustainable software ecosystems.
- **Education and Awareness:** Raise awareness of the need for dedicated support and guidelines for best practices. The organization could also drive sustainability and adoption at the university education level.
- **Quality Assurance and Simplified Installation:** Provide as much quality assurance as possible and maintain simplified installation

Summary of Responses from PESO Project Request for Input

What about?

- AI/ML
  - E4S already builds AI/ML products: PyTorch, TensorFlow, Horovod
  - Opportunity: Curate and support additional stacks
    - Many scientific teams rely on their own ad hoc fragile stack, often generations behind latest
    - DOE teams are working on their own AI/ML capabilities, need integration and support
  - The “Frank” system sponsored by DOE includes key AI target devices
  - Bottom line:
    - **Extension of ecosystem efforts to AI should require marginal change to our approach**
    - Certainly, better than establishing a different stack
    - For science, M&S and AI/ML software are used in combination – **a single stack makes sense**
- Cloud
  - E4S is already available in containers, on AWS, and Google Cloud
  - We use these resources for testing, and so do the cloud providers
- Quantum
  - Most people I know in this field are physicists
  - We don’t know enough to say what is needed
  - Even so, these devices will be hosted - a lot of what we know about HPC software can apply

Next Steps and Opportunities

ECP has been very active with our Industry and Agency Council

Date	ECP Events
October 26-27, 2022	IAC Meeting at ORNL
October 31, 2022	Cloud Computing Workshop (IAC) (virtual)
November 2022	Fortran Workshop (IAC) (virtual)
November 2022	SC22 in Dallas
January 17-20, 2023	2023 ECP Annual Meeting
January 27, 2023	Winter Quarterly IAC (virtual)
February 2023	ECP community BOF days and tutorials
April 11, July 18	NASA Workshops
October 2023	Continuous Integration Workshop
October 24, 2023	IAC Meeting at ORNL
Ongoing	Much more

ECP is very active in agency outreach with many conversations around use of E4S

NOAA

- NOAA deep dive meeting on July 20 was very successful. Discussed NOAA goals and shared lessons learned.
- NOAA experimenting with Spack build caches to significantly reduce compile times and, using E4S, build their code AM4 for the first time on AWS cloud.
- Working on ideas for collaboration projects post-ECP.

NSF

- Planning an exascale system; very interested in E4S software stack. Exploring deployment of E4S on NFS commodity clusters.
- Joint NSF-DOE workshops on E4S.
- Shared lessons learned in ECP project management for portfolios of applications and software technologies.
- Led a plenary panel at the 2023 ECP Annual Meeting with the other agencies.

NASA

- ST presentation at the NASA Science Mission Directorate Open Source Science Initiative Data and Compute Architecture Study.
- Technical deep dive on applications April 11, 2023. Looking for opportunities for targeted engagement.
- Technical deep dive on ST July 18.

DoD

- Deployment and evaluation of E4S on DoD Narwal HPC system planned (Navy).
- Planning technical deep dive; requested topics of interest.

E4S

https://e4s.io

E4S lead: Sameer Shende (U Oregon)

Spack

https://spack.io

Spack lead: Todd Gamblin (LLNL)

ECP EXASCALE COMPUTING PROJECT

Post-ECP and Final Remarks

- DOE has **learned a lot about producing software contributions** to the HPC community:
  - Improved planning, executing, tracking, assessing, integrating, and delivering
  - Improved interactions with the broader HPC software and hardware community
  - Direct engagement with industry, US agencies, and international collaborators
- In post-ECP efforts **we propose to continue and expand these efforts:**
  - Further engage with commercial partners to provide a rich, robust software ecosystem
  - Evolve a stable, sustainable business model for engaging with agencies and industry
  - Engage with cloud providers, software foundations, and others to optimize cost & benefit sharing
  - Further the ECP strategy for direct industry and agency engagement
- We intend to **realize the potential of the ECP legacy across the HPC community:**
  - **Realize the "100X" potential** by transferring scientific computations to accelerated architectures
  - **Increase the trustworthiness, sustainability, and cost effectiveness** of our software in the future
- **We want to work with the HPC User Forum community to realize the legacy of ECP, and beyond**

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: Doug Kothe
ECP Deputy Director: Lori Diachin

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.**

Appendix: ECP software product sampler

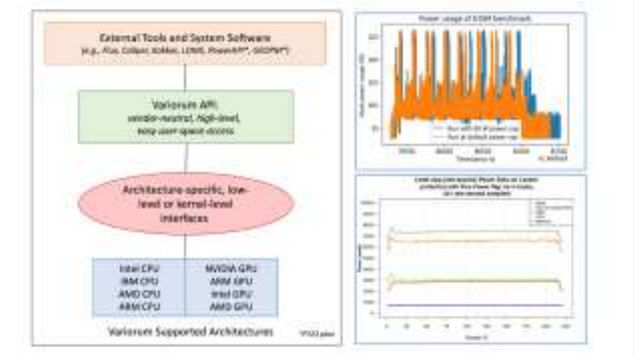
ECP Argo PowerStack: Vendor-Neutral Power Management at Exascale

ECP Team and Funding

- ECP Stakeholders: **Facilities (El Capitan, Aurora), ST (Flux, Caliper, Kokkos, PowerAPI, E3SM Scream)**
- Core team members

LLNL:	Stephanie Brink, Aniruddha Marathe, Tapasya Patki, Eric Green, Kathleen Shoga, Barry Rountree
U. Arizona:	David Lowenthal and his student.

Key Milestone



Variorum v0.6, released in Sept 2022, allows for vendor-neutral power management for more than ten platforms, including El Capitan and Aurora. Right figure shows results of PowerStack from LDMS and Flux on two clusters (Intel and IBM) with E3SM and LBANN workflows at scale, including CPU/memory/GPU data.

Product Description

Dynamic power management is critical in the exascale era, both in terms of not exceeding the overall available power budget and in terms of utilizing the available power to make the most application progress. In the Argo project, we employ hierarchical power management that includes system-global, runtime-level, and node-local mechanisms and policies – called *PowerStack*.

New capabilities	• Variorum multi-node support for El Capitan's and Aurora's architectures, including CPU, memory & GPU power • Integration of Variorum with Sandia's OVIS LDMS for large-scale vendor-neutral monitoring of clusters • Initial integration of Variorum with Flux and testing on Tioga for resource management towards El Capitan
What users can do because of ECP work	Vendor-neutral power management, on diverse HPC architectures for HPC applications and science workflows
Community impact	Essential for utilizing available power efficiently as well as mitigating power swings in HPC applications. HPC PowerStack is a newly established charter of this project, that actively collaborates with Power API and EEHPC WG, showing wide applicability.
Key Software Dependencies	`msr-safe`, `hwmon`, IBM OPAL, AMD HSMP, Energy Driver and E-SMI, Intel OneAPI, AMD ROCm, NVIDIA NVML

Post-ECP Funding for Argo PowerStack

Needed	Total FY24 $750K for ECP-level status quo
Secured	$0K/year
Pending	$0K/year
Targeted	$100K/year for one year through ISCP
Gap	$750K/year

Expected Future Impact from PowerStack

- With LDMS integration, PowerStack features can be used for large scale monitoring of power and energy on HPC systems with diverse architectures
- With Flux, Caliper and Kokkos integrations, HPC applications and scientific workflows can benefit from identifying critical path in terms of power consumption, including CPU, memory and GPU components, enabling code optimizations.
- HPC PowerStack (https://hpcpowerstack.github.io) has emerged as a new community-wide effort for power management in HPC systems. Bringing together our partners from industry, academia, and other research labs across the United States, Europe, and Asia, and collaborating with PowerAPI and EEHPCWG allows for development of standardization of power knobs.
- Variorum library can be easily extended to various upcoming architectures for exascale and beyond. With a vendor-neutral and uniform API framework, integration into system software such as resource managers, runtime tools as well as monitoring frameworks is efficient.

2023 R&D 100 WINNER

Variorum

https://web.cels.anl.gov/projects/argo/overview/#power-management
https://variorum.readthedocs.io/en/latest/

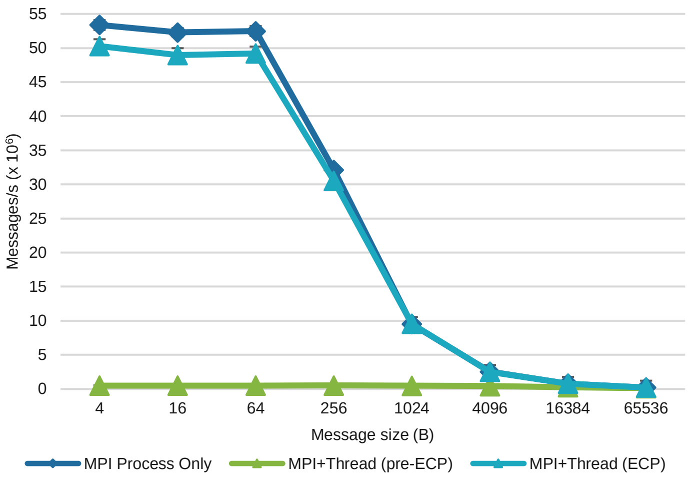
MPICH: Exascale MPI

ECP Team and Funding

- ECP Stakeholders: **AD, ST and HI (many applications, libraries, and all 3 exascale systems)**
- Core team members

ANL	Yanfei Guo, Ken Raffenetti, Hui Zhou, Rajeev Thakur, Xiaodong Yu, Sudheer Chunduri, Rob Latham, Thomas Gillis

Key Milestone



Message Rate of Multi-Threaded MPI Communication(16 threads/node)

Significant performance improvement in MPI+Threads communication. Benefits most multi-threaded applications.

Product description

The Exascale MPI project aims to making MPICH exascale-ready by incorporating new design and approaches to improve communication performance and efficiency and to improve support for hybrid and heterogeneous programming models including MPI+GPUs and MPI+Threads. The project also benefits the broader MPI community due to its wide adoption in vendor MPI libraries (**Intel MPI, Cray MPI, etc**.).

New capabilities	• Extremely low communication latency for high-performance networks • Support MPI+GPU programming model • Improved MPI+Thread performance • Optimized MPI collective algorithm and runtime performance • Improved large-scale MPI application launch time
What users can do because of ECP work	Scalable MPI communications, Hybrid programming models (MPI+GPU, MPI+Thread)
User impact	Majority of applications and libraries; MPI on Frontier, Aurora, El Capitan
Key Software Dependencies	UCX, OFI, CUDA, ROCm, Level Zero, hwloc

Expected future impact from MPICH

- Optimized support for high-performance hybrid programming with accelerators
- Machine-learning-based tuning and optimization for MPI communication
- Integration of compression for MPI communication
- Efficient asynchronous communication by leveraging emerging hardware
- Optimization for interoperability with external tools and libraries
- Sustained development and optimization of MPICH
- Support for next-generation, post-exascale systems
- Vendor collaboration and future MPI standardization

EXASCALE COMPUTING PROJECT

https://www.mpich.org/

MPICH

Kokkos : Performance Portability Programming Model

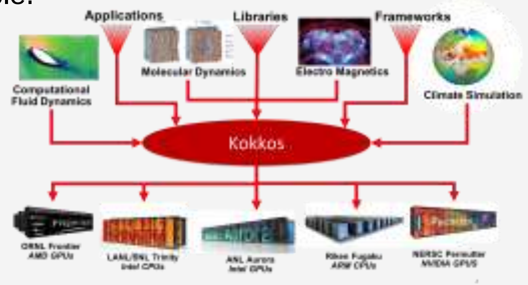
ECP Team and Funding

- ECP Stakeholders: **20 ECP Projects list Kokkos as critical dependency; OLCF, ALCF, NERSC require support for Kokkos for non-ECP facility users;**
- Core team members

[SNL]	Christian Trott
[ORNL]	Damien Lebrun-Grandie
[LBNL]	Rahul Gayatri
[LANL]	Galen Shipman
[ANL]	Nevin Liber

Key Milestone

Develop mature support for the Frontier and Aurora exascale machines, allowing codes which were developed on pre-exascale platforms such as Summit to execute efficiently on the new systems as soon as they are available.



Product Description

Kokkos provides a C++ Programming Model for Performance Portability for science and engineering codes. It leverages platform specific backends such as CUDA, HIP and SYCL to map its semantics and APIs to diverse hardware architectures. Next to OpenMP, Kokkos is now arguably the most widely used multi-vendor programming model in HPC.

New capabilities	• AMD GPU support via a HIP backend • Intel GPU support via a SYCL backend • Multi instance execution spaces allow for concurrent multi-kernel execution on the same device. • ISO C++23 mdspan multi-dimensional arrays, transition Kokkos capabilities into an ISO standard
What users can do because of ECP work	Kokkos allows developers to implement their code in a single version, and execute it efficiently on all DOE exascale era architectures.
Community impact	Kokkos is now used by more than 140 organizations, providing a performance portability solution to the world-wide HPC community, well beyond DOE. Furthermore, some key Kokkos innovations are now part of the ISO C++ standard, providing capabilities for developers in gaming, finance, and AI.
Key Software Dependencies	Standard C++, cmake and optionally on vendor specific backends.

Expected Future Impact from Kokkos

- Kokkos adoption is still spreading fast beyond DOE confines and may become the de facto standard for writing performance portable code.
- Kokkos also plays an important role in developing APIs and semantics for features later proposed and adopted in the C++ standard. As such it prepares the way for enabling performance portability for non-HPC application areas such as gaming, finance, AI, data-analytics and even embedded computing for sensor analysis.

EXASCALE COMPUTING PROJECT

kokkos

https://github.com/kokkos/kokkos

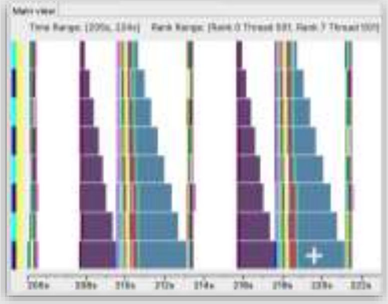
HPCToolkit : Performance Analysis Tools for Exascale

ECP Team and Funding

- ECP Stakeholders: AD and ST Teams; ANL, LLNL, ORNL
- Core team members

[Rice University]	John Mellor-Crummey Mark Krentel Laksono Adhianto Wileam Phan	Scott Warren Jonathon Anderson Yumeng Liu

Key Milestone



Measurement and attribution of profiles and traces of GPU-accelerated ECP applications within and across nodes on Crusher - Frontier's test and delivery system

A screenshot of HPCToolkit's `hpcviewer` *displaying two iterations of a trace of ECP GAMESS on Crusher, filtered to show only GPU computations of several MPI ranks. This figure revealed that load imbalance is severe for the most costly GPU kernels in GAMESS, which compute over triangular iteration spaces. This insight led the GAMESS team to redesign their strategy for partitioning work across MPI ranks.*

Product Description

HPCToolkit is designed to measure and analyze the performance of applications, libraries, and frameworks within and across the compute nodes of GPU-accelerated platforms. HPCToolkit helps developers identify bottlenecks and inefficiencies that keep codes from achieving exascale performance. HPCToolkit summarizes code performance in profiles, traces, and graphs.

New capabilities	• Heterogeneous call path profiles and call path traces that include instruction-level measurements of GPU activity • Detailed measurement and attribution of performance to function calls, inlined code, and loops within kernels for AMD, Intel, and NVIDIA GPUs • Highly-scalable post-mortem performance analysis that employs both shared and distributed-memory parallelism
What users can do because of ECP work	HPCToolkit helps developers measure and analyze the performance of software on extreme-scale GPU-accelerated supercomputers within and across compute nodes
Community impact	Application, library, framework, and tool developers can pinpoint causes of performance bottlenecks and scalability losses in their software to identify opportunities for improvement
Key Software Dependencies	Dyninst, Elfutils, MPI and OpenMP Vendor SW: ROCm, CUDA, Level0, GT-Pin

Expected Future Impact from HPCToolkit

- Measure and analyze software performance at extreme-scale on Frontier, Aurora, and El Capitan
- Provide a more scalable solution for performance measurement and analysis than vendor tools
- Use a combination of binary instrumentation and hardware capabilities for instruction-level measurement on AMD, Intel, and NVIDIA GPUs to assess performance losses within GPU kernels and understand their causes
- Integrate information from traces, hardware counters, and instruction-level measurements to assemble a wholistic view of performance
- Transform measurement data into insight by automatically identifying root causes of performance losses and suggesting optimizations that address them
- Assess opportunities for improving mechanisms used by template-based programming models such as Raja and Kokkos as well as frameworks such as AMReX
- Identify needs for improved GPU compiler capabilities
- Extend capabilities to analyze AI and ML workloads
- Provide support for emerging programming models

http://hpctoolkit.org

OpenMP

ECP Team and Funding

- ECP Stakeholders: **AD List, Facilities List, ST List**

Core team members

[ANL]	Michael Kruse, Jose Diaz
[BNL]	Sunita Chandrasekaran, Abid Malik, Dossay Oryspayev
[LLNL]	Bronis de Supinski, Tom Scogland, Johannes Doerfert
[ORNL]	David Bernholdt, Verónica G. Melesse Vergara, Swaroop Pophale, Seyong Lee
[SBU]	Shilei Tian
[GA Tech]	Vivek Sarkar, Lechen Yu
[UDEL]	Felipe Cabarcas

Key Milestone

OpenMP 5.0 was released in November 2018, 5.1 released in 2020, 5.2 in 2021.

- Full support for heterogeneous systems
- Broadly support on-node performant, portable parallelism
- Support for interoperability with other GPU APIs
- Addition of loop transformation directives
- Compiler-agnostic built-in assume
- Adds full support for C11, C++11, C++14, C++17, C++20 and Fortran 2008 and partial support for Fortran 2018

Product Description

The product, OpenMP, is a widely popular programming model that has been rapidly evolving in the past several years to fully support accelerator devices. Major vendors and open source compilers have implemented parts of the OpenMP 5 specification and beyond in their products. LLVM, GCC, AMD, Intel, HPE, NVIDIA, Mentor Graphics to name just a few.

New capabilities	• Full support for accelerator devices • Improvements in accelerator device interactions • Support for the latest versions of C, C++, and Fortran • Multilevel memory systems • Enhanced portability • Improved debugging and performance analysis • Various new combined constructs
What users can do because of ECP work	Availability of a standard-based programming model for the community to use
Community impact	Seamless migration of applications from one platform to another by using a directive-based programming model
Key Software Dependencies	None

Expected Future Impact from OpenMP

- SOLLVE drives the widely popular and broadly used OpenMP standard. The standard has been rapidly evolving ratifying several critical features that covers a broad spectrum of heterogeneous architectures.

- SOLLVE aims to create a performant yet portable software using the OpenMP programming model for legacy applications to be seamlessly ported across different types of architecture. To that end, developing such a capability pushes this product to explore novel compiler techniques and implementations that can facilitate an application developer with portable software so that they worry less about the software and more about the science.

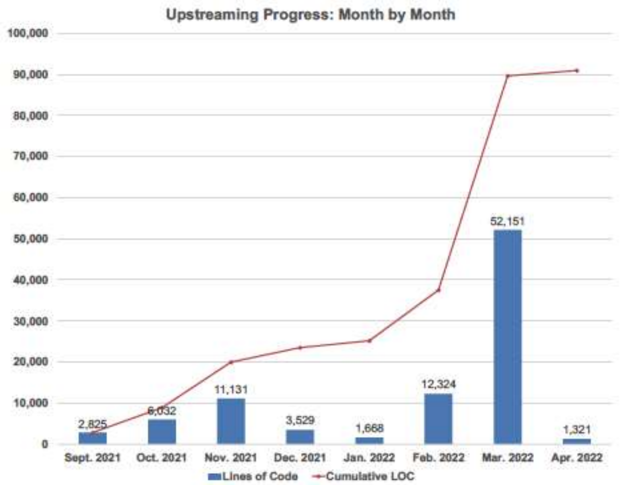
Flang: Fortran front-end for LLVM

ECP Team and Funding

- ECP Stakeholders: **LLVM**
- Core team members

[LANL]	[P. McCormick, A. Perry-Holby, T. Prabhu]
[NVIDIA]	[S. Scalpone]
[ORNL]	[D. Bernholdt, A. Cabrera]
[ANL]	[M. Kruse]
[LBNL]	[B. Friesen, D. Rouson]

Key Milestone



Total Lines of Code merged upstream monthly

Completed merge of code into LLVM main repository. Eliminated bifurcated workflow, maximized ability to collaborate with other teams, and streamlined release of new features to users.

Product Description

The Flang project is developing an open-source, standard-compliant Fortran compiler front-end for the LLVM Compiler Infrastructure.

New capabilities	• Full Fortran 77 and Fortran 95 support • Completion of new driver • Upstreaming of FIR lowering completed => compiler can now generate executables without the aid of a secondary compiler for supported programs
What users can do because of ECP work	Fortran code teams now have access to a modern, open-source compiler that leverages LLVM, an industry standard toolchain used by most vendors.
Community impact	Flang is now part of the greater LLVM community efforts, enabling collaborations and sustainability of the Flang code base beyond the scope of ECP.
Key Software Dependencies	LLVM

Expected Future Impact from Flang

- Fortran support within the LLVM community, including features from the most recent standards and greater breadth of testing
- Fortran-centric performance optimizations within the compiler
- Enabling the development of Fortran tools that leverage the LLVM infrastructure to assist developer productivity and code performance

Source Code: https://github.com/llvm/llvm-project/tree/main/flang
Software release via LLVM (version 11.0.0 and later): https://releases.llvm.org/

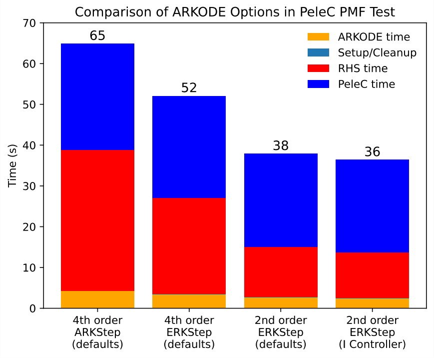
SUNDIALS: Time Integrators for High Performance Systems

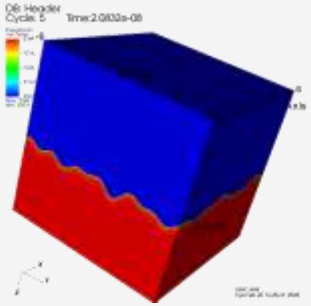
ECP Team and Funding

- ECP Stakeholders: **AD [Pele, ExaAM, ExaSky, CEED, AMReX], Facilities [NERSC, OLCF], ST [xSDK, E4S]**
- Core team members

[LLNL]	[C. Balos, D. Gardner, C. Woodward]
[SMU]	[D. Reynolds]

Key Milestone





SUNDIALS is integrated into the PeleC combustion code. Through use of flexible interfaces and a multitude of features available in SUNDIALS that were not before available to PeleC, several algorithmic choices were tested and new settings and methods were shown to give a 44% reduction in run time on a pre-mixed flame test on two Summit nodes (12 GPUs).

Product Description

The SUNDIALS library is a suite of packages providing efficient time integrators, some with sensitivity analysis, and nonlinear algebraic solvers. The integrators use highly efficient adaptive methods and are implemented to allow for easy use in application-specific contexts. Our newest integrators include methods for problems with multiple time scales.

New capabilities	• Multi-node GPU support for NVIDIA, AMD, and Intel GPUs • Support for many GPU-enabled linear algebra packages, including hypre, SuperLU_DIST, MAGMA, cuSolver, oneMKL, and Ginkgo • Performance profiling, extensive documentation, and automated testing on LLNL HPC systems
What users can do because of ECP work	SUNDIALS users can use state-of-the-art time integrators utilizing GPU accelerators and new, GPU-enabled solvers on a variety of platforms within applications.
Community impact	SUNDIALS is a key tool for highly efficient time-dependent scientific simulations, including multiphysics and multiscale simulations
Key Software Dependencies	Optional: MPI, OpenMP, CUDA, HIP, SYCL, RAJA, LAPACK, MAGMA, hypre, Trilinos, SuperLU_DIST, KLU, PETSc, Gingko, cuSparse, oneMKL, XBraid

Expected Future Impact from SUNDIALS

- The suite-wide restructuring conducted during the ECP provides SUNDIALS with a new and significantly higher level of flexibility that facilitates use on GPU-based architectures and architectures of the future.
- New flexibility provides the ability to add interfaces to a host of new solver packages and data structures and allows SUNDIALS to more easily be used by applications.
- Applications will see faster and more accurate solutions as a result of upgrading their time integrators from simple, first order methods to high order methods from SUNDIALS during the ECP.
- GPU-enabled new flexible and high-order multirate time integrators will allow for users to better map accurate and adaptive time integrators to their multiphysics applications on exascale systems.
- The new performance monitoring layer and logging capabilities enable enhanced debugging and performance evaluation for application users.
- Autotuning work can provide a systematic and semi-automated approach to algorithm selection and optimization of heuristic time-integrator parameters for particular applications.

EXASCALE COMPUTING PROJECT

sundials

https://computing.llnl.gov/sundials

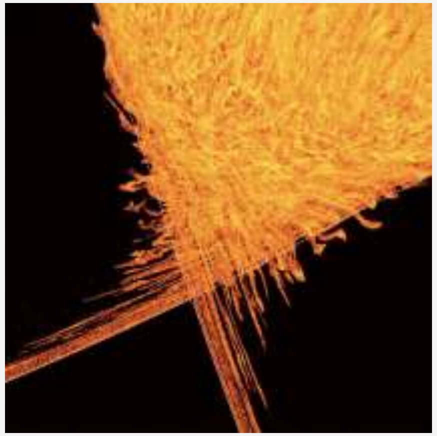
VTK-m: Visualization on Accelerators

ECP Team and Funding

- ECP Stakeholders: **AD [WDMApp, ExaSky, MFIX-Exa, WarpX , ExaWind], Facilities[Frontier, Aurora], ST [ALPINE, ADIOS]**
- Core team members

[ORNL]	[K. Moreland, D. Pugmire]
[LANL]	[L.-t. Lo, R. Bujack]
[SNL]	[M. Bolstad]
[UOregon]	[H. Childs]
[Kitware]	[B. Geveci, V. Bolea, S. Philips, A. Yenpure]

Key Milestone



Poincaré plots are an important tool for understanding the energy transport that occurs as energetic particles interact with components in the ITER reactor. GPU-enabled particle tracing from VTK-m reduced the time from 2 hours for a single plot down to 2 minutes.

Poincaré plot of the fluctuating magnetic field at the edge of ITER plasma highlighting turbulent homoclinic tangles.

Product Description

VTK-m delivers scientific visualization and analysis infrastructure and algorithms to ECP applications and software technology. VTK-m provides the linchpin technology of executing visualization algorithms on exascale accelerator processors for the ECP applications like ParaView, VisIt, and Ascent. Visualization for DOE exascale problems would not be possible without VTK-m.

New capabilities	• Core visualization capabilities for ECP processors (AMD Radeon and Intel X^e) heretofore not considered by DOE. • Improvements and optimization of core algorithms (e.g., contouring, rendering). • Implementation of multiple critical algorithms (e.g., particle advection, Lagrangian structures, contour trees, density estimation).
What users can do because of ECP work	Much larger scale visualization and analysis, and consequently understanding of exascale simulations, by leveraging the accelerators at leadership class facilities.
Community impact	Users can visualize and analyze their exascale simulations with their full detail using familiar tools like ParaView, Visit, and Ascent.
Key Software Dependencies	Kokkos

Expected Future Impact from VTK-m

- What is exciting about the future for this product?
  - As HPC hardware continues to evolve, VTK-m is ready to address the new challenges and opportunities to take full advantage of available computing power.
  - VTK-m will better leverage features unique to each processor (shared memory, vector processing, tensor cores).
  - VTK-m will continue to innovate new visualization algorithms to satisfy DOE science needs.
  - End user tools such as ParaView, VisIt, and Ascent will be updated to improve visualization capabilities with no further requirements from users.
- What is the expected client impact in the future?
  - VTK-m-enabled tools significantly reduce the time to scientific understanding.

EXASCALE COMPUTING PROJECT

https://m.vtk.org

AMReX: Block-structured Adaptive Mesh Refinement

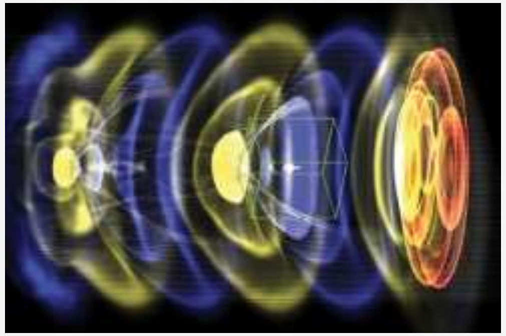
ECP Team and Funding

- ECP Stakeholders: **ExaSky, ExaStar, ExaWind, MFiX-Exa, Pele, WarpX**
- Core team members

[LBNL]	Ann Almgren, John Bell, Andrew Myers, Jean Sexton, Weiqun Zhang

Key Milestone

WarpX named 2022 Gordon-Bell FInalist



"[WarpX] enabled 3D simulations of laser-matter interactions on Fugaku and Summit, which have so far been out of the reach of standard codes," the description reads. "These simulations helped remove a major limitation of compact laser-based electron accelerators, which are promising candidates for next generation high-energy physics experiments and ultra-high dose rate FLASH radiotherapy."

Product Description

AMReX is a software framework to support the development of block-structured adaptive mesh refinement algorithms. AMReX provides support for both field and particle data and includes native multigrid solvers for linear systems. AMReX also incorporates support for embedded boundary discretizations for representing complex geometries.

New capabilities	• Abstraction layer for launching kernels that provide performance portability between NVIDIA, AMD and Intel GPUs and CPUs • Memory arenas for mesh and particle data to minimize memory allocation latency • Flexible design that provides different levels of abstraction that support a wide range of algorithms
What users can do because of ECP work	AMReX enables users to develop multiphysics PDE-based simulations (with particles) that effectively utilize GPU accelerators on exascale architectures
Community impact	AMReX provides a pathway for rapid development of scalable, high-performance multiphysics simulation codes that effectively utilize GPU and multicore architectures
Key Software Dependencies	Requires MPI and C++14 Provides optional interfaces to hypre, SUNDIALS, HDF5, ALPINE

Expected Future Impact from AMReX

- What is exciting about the future of AMReX
  - Performance portability abstraction layer provides a general approach to supporting new architectures with different types of accelerators
  - Framework is extensible to support hybrid algorithms that combine different types of physical models in different regions of the domain and more complex meshing strategies such as mapped multiblock
  - Provides a vehicle for basic algorithm research in discretization methodology, linear solvers and multiscale modeling
- What is the expected client impact in the future?
  - AMReX provides the framework for multiple ECP applications codes as well as other applications in areas as diverse as modeling microbial communities, atmospheric and ocean modeling, epidemiology, kinetic models of plasmas, granular materials, microelectronics, non-Newtonian flows
  - AMReX reduces development costs while providing performance portability, resulting in faster pathways to scientific discovery

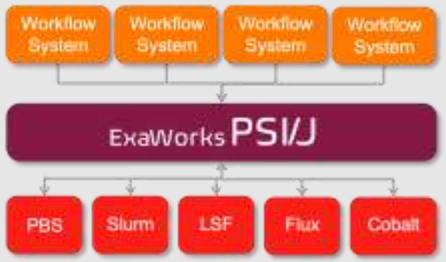
ExaWorks SDK: an SDK for Exascale Workflows

ECP Team and Funding

- ECP Stakeholders: CANDLE, ExaLearn, ExaAM, ORNL, LLNL, ANL, LBNL, E4S

[ANL]	K. Chard, Y. Babuji, B. Clifford, M. Hategan, A. Wilke
[BNL]	S. Jha, A. Alsaadi, A. Merzky, M. Titov, M. Turilli
[LLNL]	D. Laney, J. Corbett, P. Aschwanden
[ORNL]	R. Ferreira da Silva, K. Maheshwari

Key Milestone



Product-ready release of the Portable Submission Interface for Jobs (PSI/J) API and reference implementation will enable authors of workflows, including domain scientists writing bespoke infrastructure, to port and maintain their infrastructures on multiple systems and sites with less effort.

Product Description

ExaWorks is assembling a software development kit (SDK) for exascale-ready workflow technologies and instantiating an open source workflow community to enable the creation and adoption of shared API's and components for high performance workflow systems.

New capabilities	• First of a kind CI/CD infrastructure for multiple workflow management systems technologies with real-time test status dashboard (deployed to ORNL, ANL, LLNL, and ECP test) • PSI/J: a portable submission interface for jobs designed with the community
What users can do because of ECP work	Users can quickly explore a set of scalable workflow technologies and integrate them in their workflows.
Community impact	Workflow technologies are notoriously hard to adopt due to perceived lack of testing and documentation, ExaWorks is providing an easy on-ramp to well-tested tools with robust tutorial materials.
Key Software Dependencies	Parsl, RADICAL CyberTools, Swift/T, Flux, Spack

Expected Future Impact from ExaWorks

- The SDK will provide a well-tested (across multiple DOE facilities and exascale systems) set of open source workflow technologies with high quality documentation and tutorials to encourage adoption by domain scientists.
- Facilitating an active open source community that encourages the growth of the SDK with additional workflow technologies and active discussion on moving towards common API's for workflows.
- Propagation of PSI/J reference implementation into the community, encouraging adoption and support by facilities to make it easier to port and maintain workflow systems across multiple compute facilities and machines
- Community-based design of additional common API layers for aspects of workflow sytsems that exist across multiple workflow systems

2.3.6 NNSA Software Portfolio

Lab PI's are listed as the ASC ATDM execs for communication, but technical work is handled by multiple technical POCs

Project Short Name	PI Name	Short Description/Objective
LANL NNSA Software	Tim Randles	**Legion** (PM/R), **Kitsune LLVM** (Tools), **Cinema** (Data/Vis), and **BEE/CharlieCloud** (Ecosystem)
LLNL NNSA Software	Becky Springmeyer	**Spack**, **Flux** (Ecosystem), **RAJA**, **CHAI**, **Umpire** (PMR), **Debugging @ Scale**, **Flux/Power**, **Caliper** (Tools), **MFEM** (Math Libs)
SNL NNSA Software	Jim Stewart	**Kokkos** (PM/R), **KokkosKernels** (Math Libs), **VTK-m** (Data/Vis), **OS&ONR** (EcoSystem and PM/R)

A symbiotic relationship exists between DOE NNSA and Office of Science via these products