The Legacy of ECP Software Efforts, Realized and to Come

EXASCALE COMPUTING PROJECT

Michael A. Heroux, Sandia National Laboratories
Director of Software Technology, US Exascale Computing Project
PI, PESO Project

14th Workshop on Latest Advances in Scalable Algorithms for Large-Scale Heterogeneous Systems (ScalAH’23)

November 13, 2023

NNSA National Nuclear Security Administration

U.S. DEPARTMENT OF ENERGY | Office of Science

Outline

- ECP, Briefly
- ECP Libraries and Tools, A Sample of Products
- Establishing software ecosystems
- 100X Opportunities and Recipes
- What comes next

ECP, Briefly

ECP's holistic approach uses co-design and integration to achieve exascale computing

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

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

6 Co-Design Centers
Machine learning, graph analytics, mesh refinement, PDE discretization, particles, online data analytics

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 run times,
math libraries,
data and visualization

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

EXASCALE COMPUTING PROJECT

Exascale Systems – Primary targets for ECP Software Teams

Exascale Systems

FRONTIER

ORNL
HPE/AMD

ANL
Intel/HPE

LLNL
HPE/AMD

- ECP libraries & tools migrating to GPU platforms
- Target AMD, Intel and Nvidia (Perlmutter) devices
- Growing support for Arm/SVE in the same stack
- Mature MPI/CPU stack also robust and evolving
- Eye toward specialized devices, e.g., dataflow
- Legacy:
  - A stack to support application portability
  - Across many different distributed systems with
  - Multiple kinds of devices (GPUs, CPUs, etc)

ECP's KPPs: Quantified with Explicit Targets

KPP ID	Description of Scope	Threshold KPP	Objective KPP	Verification Action/Evidence
KPP-1	11 selected applications demonstrate performance improvement for mission-critical problems	✓ 6 of 11 applications demonstrate Figure of Merit improvement ≥50 on their base challenge problem	All 11 selected applications demonstrate their stretch challenge problem	Independent assessment of measured FOM results and base challenge problem demonstration evidence
KPP-2	14 selected applications broaden the reach of exascale science and mission capability	5 of 10 DOE Science and Applied Energy applications *and* 2 of 4 NNSA applications demonstrate their base challenge problem	All 14 selected applications demonstrate their stretch challenge problem	Independent assessment of base challenge problem demonstration evidence
KPP-3	76 software products selected to meet an aggregate capability integration score	Software products achieve an aggregate capability integration score of at least 34 out of a possible score of 68 points	Software products achieve the maximum aggregate capability integration score of 68 points	Independent assessment of each software product's capability integration score
KPP-4	Delivery of 267 vendor baselined milestones in the PathForward element	✓ Vendors meet 214 out of the total possible 267 PathForward milestones	✓ Vendors meet all 267 possible PathForward milestones	Independent review of the PathForward milestones to assure they meet the contract requirements; evidence is the final milestone deliverable

KPP: Key Performance Parameters, used to official assess success by reach threshold

Similar to KPIs used in domains

EXASCALE COMPUTING PROJECT

KPP-3 Status

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

KPP-3 Status Summary – 2023/11/08

KPP-3 threshold achieved Sep 15, 2023!

Predicted final KPP-3 status:
- Expect 67.5/68 points
  - One product (FFTX) merged into other effort
- Might be as low as 64 points
  - 3 products – working up to the end
  - 3.5 weighted points
- Well above the 34 threshold, close to 68 objective

- Changes in past week:
  - Fully Approved: STRUMPACK
  - Received SME approval: PDT
  - Moved to In flight: VeloC
  - Submissions from CODAR (+3) and ExaGraph (+1) (now fully submitted)
  - 281 (up from 276; out of 292 total) Integrations ‘In Flight’
  - 13 products under active SME review by 16 reviewers; 6 products with 1 approval
- Forecast
  - 64 ST/CD points in flight, i.e., passing # of KPP3 integrations in review, SME approved or confirmed
  - 41.5 confirmed points
  - Other products at passing number of runs complete: PaRSEC and HDF-VOL representing another 1.5 weighted points; working with teams on submissions

*SME ~ Subject Matter Expert from outside ECP community

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

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

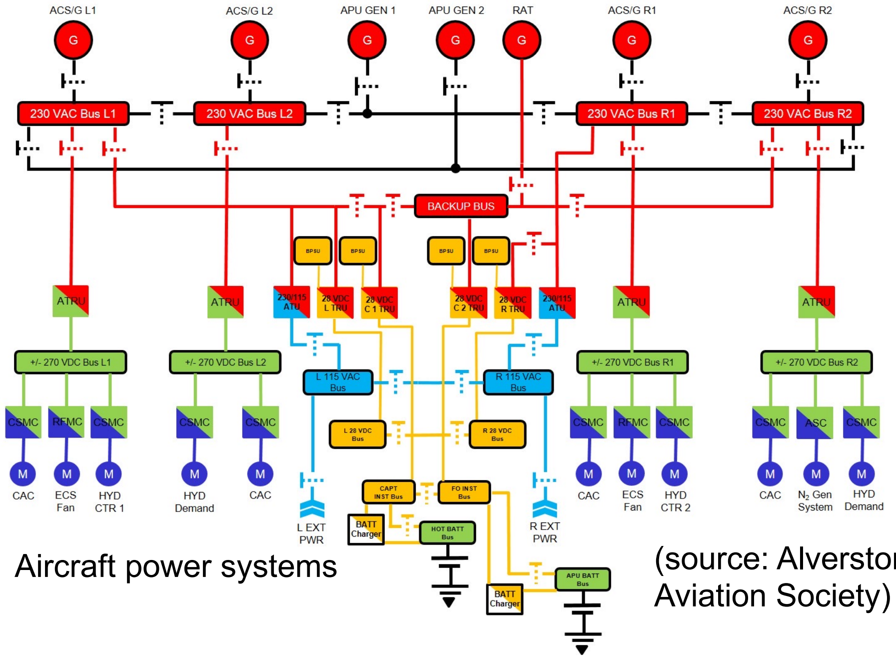
Software categories:
- **Next generation established products:** Widely used HPC products (e.g., MPICH, OpenMPI, PETSc)
- **Robust ~~emerging~~ products:** Address key new requirements (e.g., Kokkos, RAJA, Spack)
- **New products:** Enable exploration of emerging HPC requirements (e.g., zfp, Variorum)

2023 R&D 100 WINNER

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

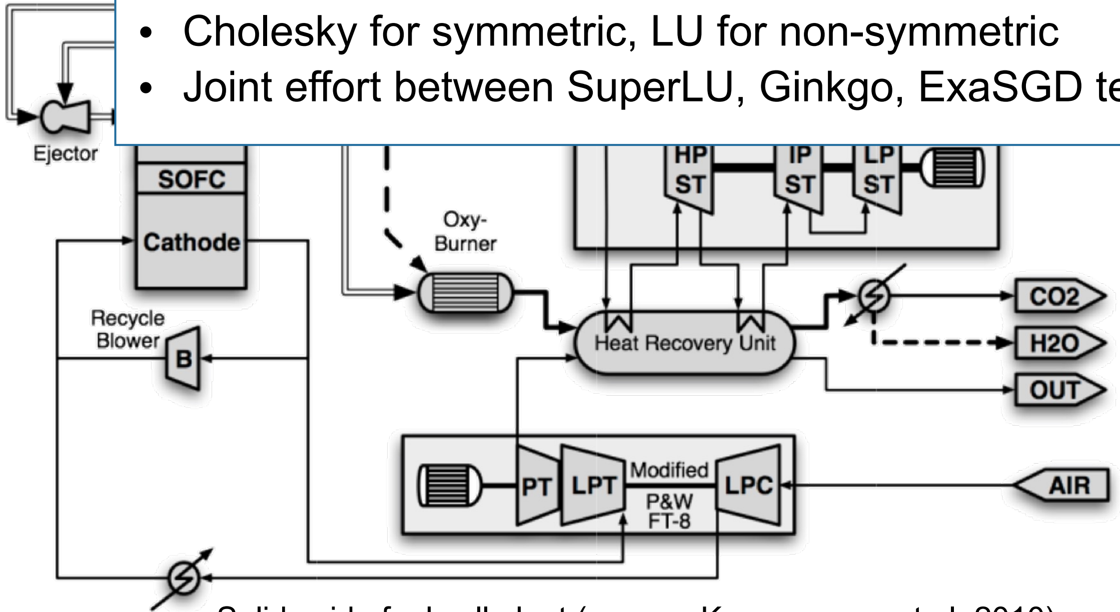
Systems Engineering Domain

- 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



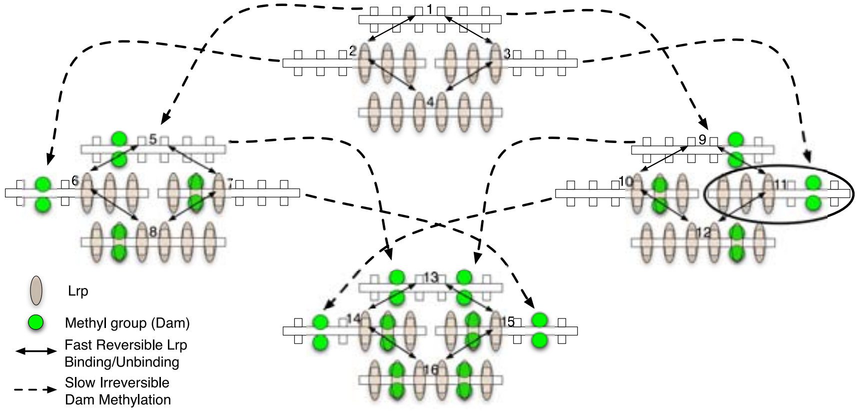
Aircraft power systems

(source: Alverstone Aviation Society)

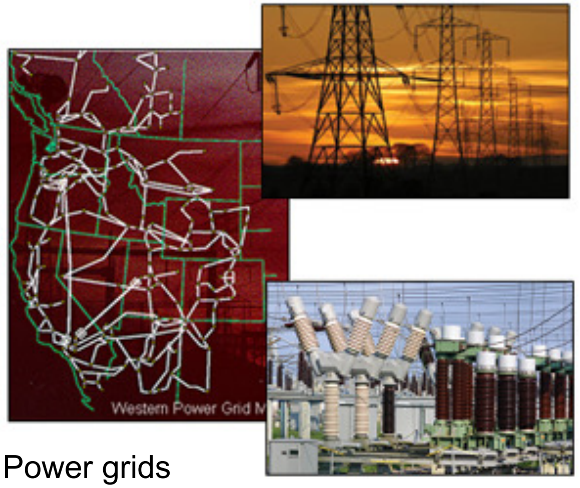


Solid oxide fuel cell plant (source: Kameswaran at al. 2010)

Buildings (source: EEB Hub, B661 2014)



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



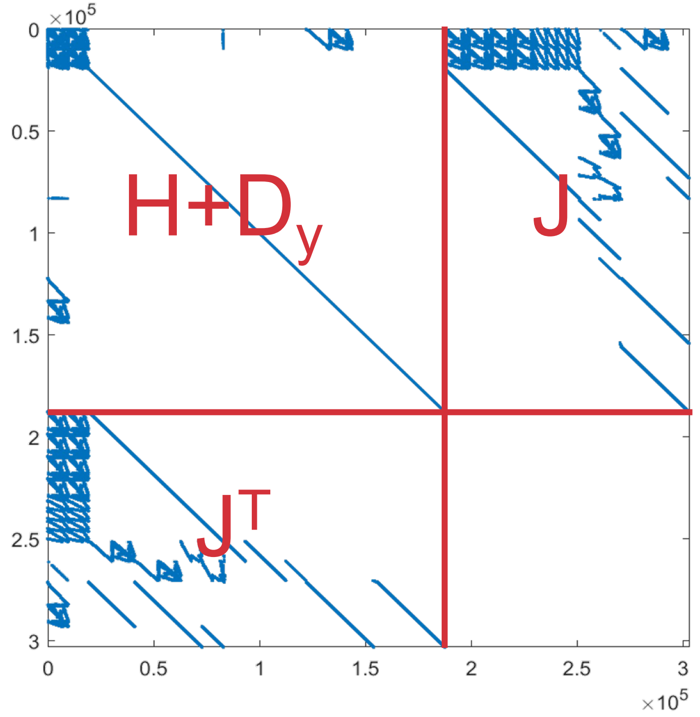
Power grids (source: PNNL)

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



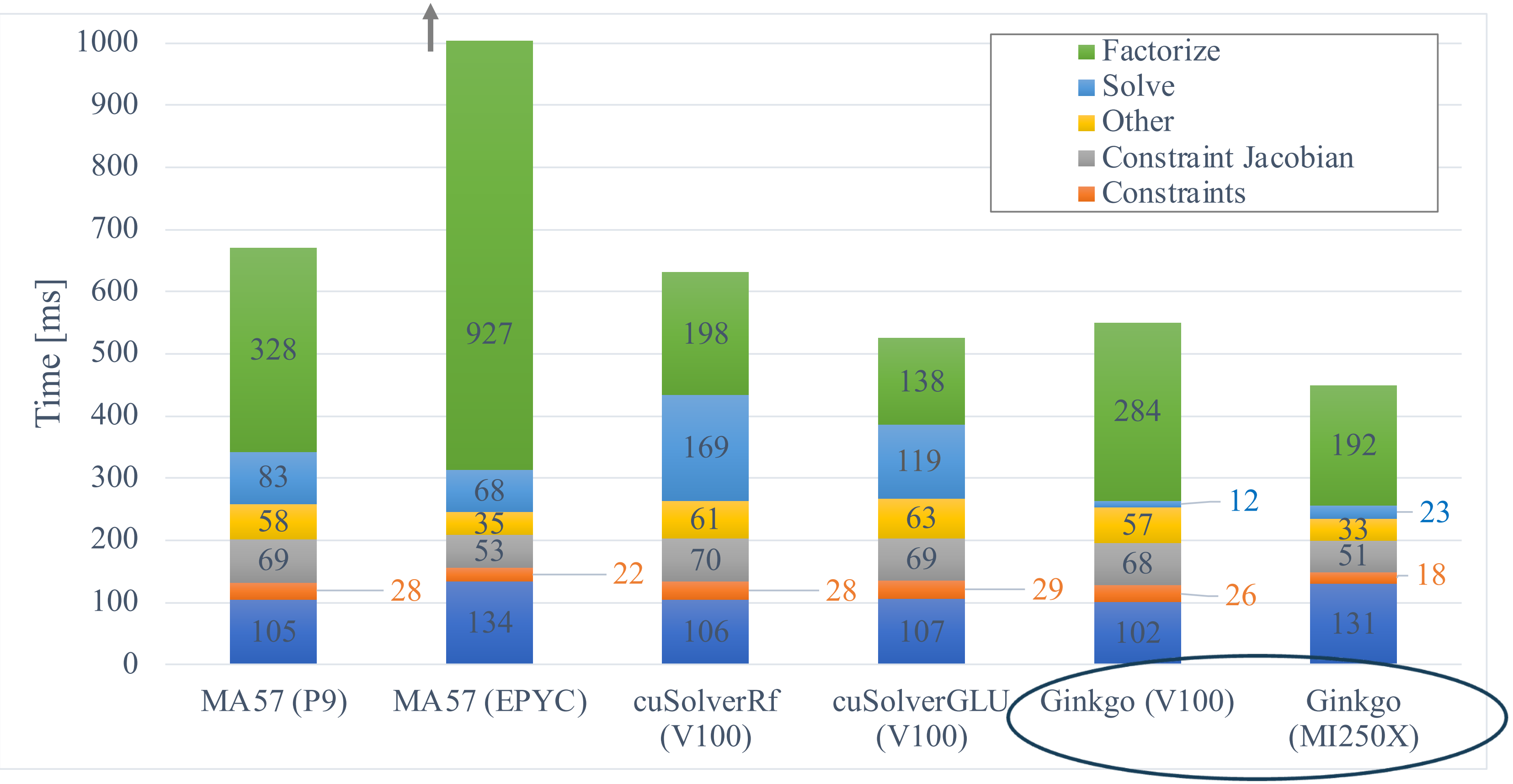
Typical sparsity pattern of optimal power flow matrices: No obvious structure that can be used by linear solver.

- 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)

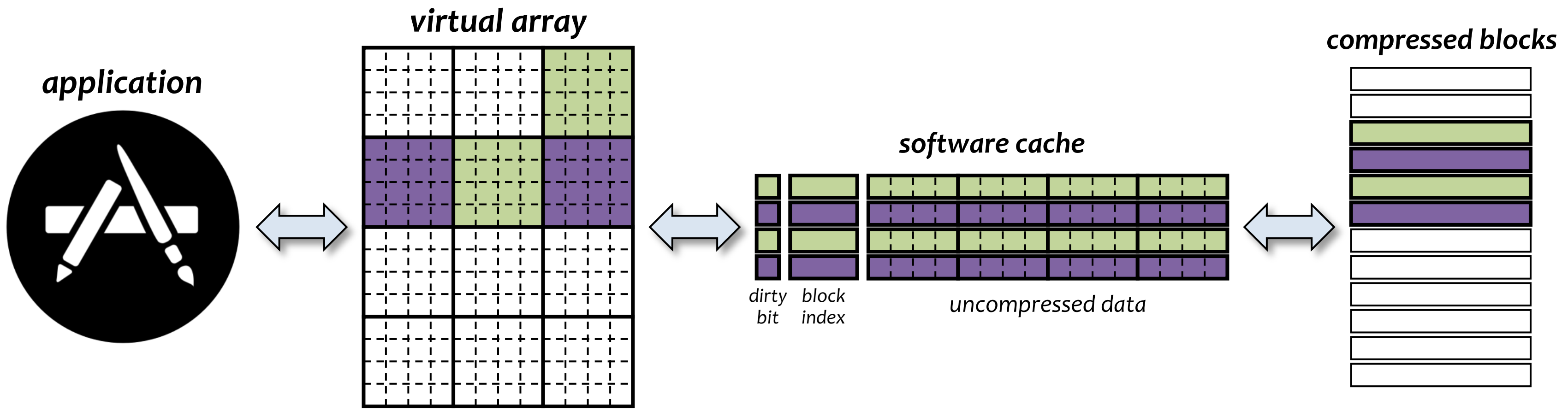
- Each GPU solution outperforms all CPU baselines
- Ginkgo performance improves on a better GPU
- Iterative refinement configuration affects linear solver performance and optimization solver convergence



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

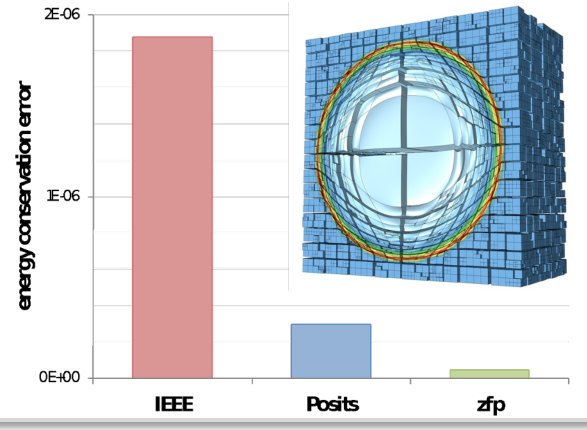
ECP EXASCALE COMPUTING PROJECT

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

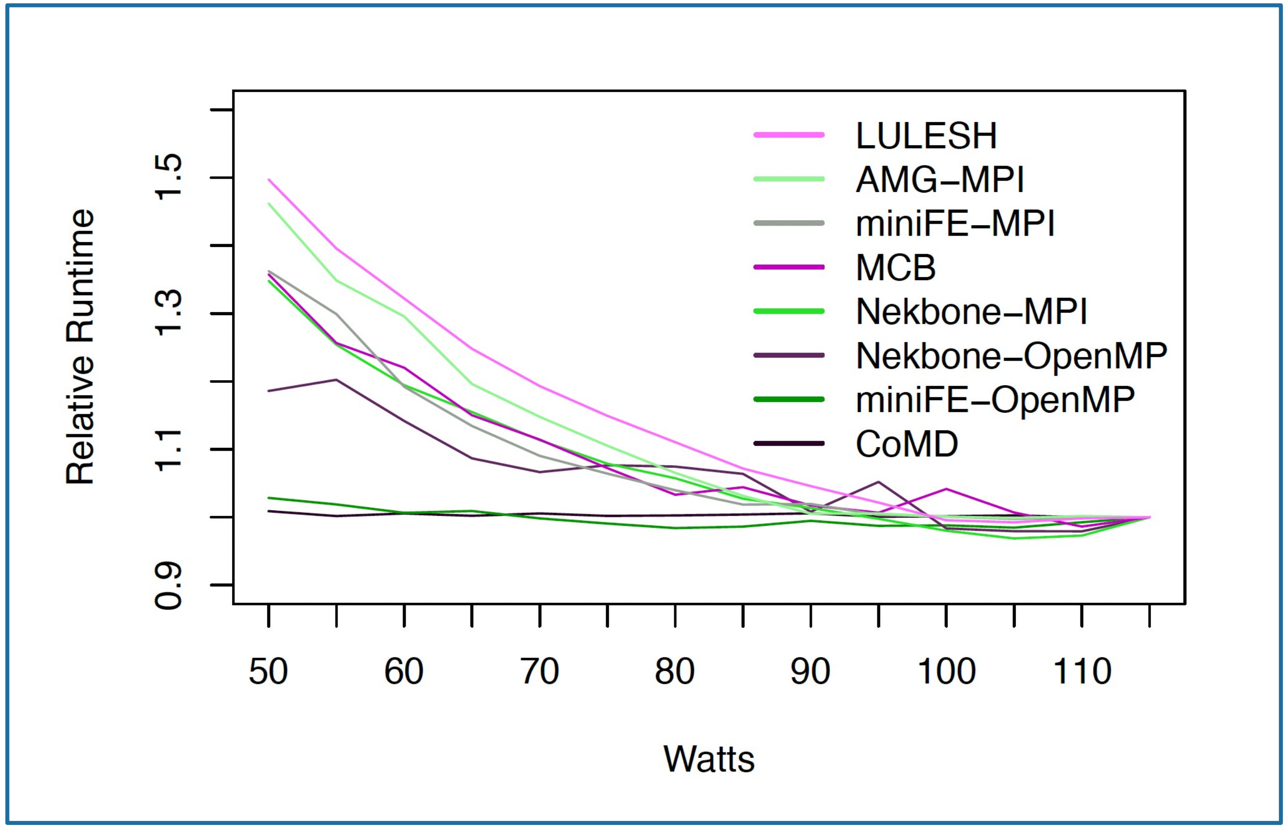


- Fixed-length compressed blocks enable fine-grained read & write **random access**
  - C++ compressed-array classes hide complexity of compression & caching from user
  - User specifies per-array storage footprint in bits/value
- Absolute and relative **error tolerances** supported for offline storage,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

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



Current HPC Systems under-utilize power significantly, and this trend is expected to worsen at scale with GPU-based systems



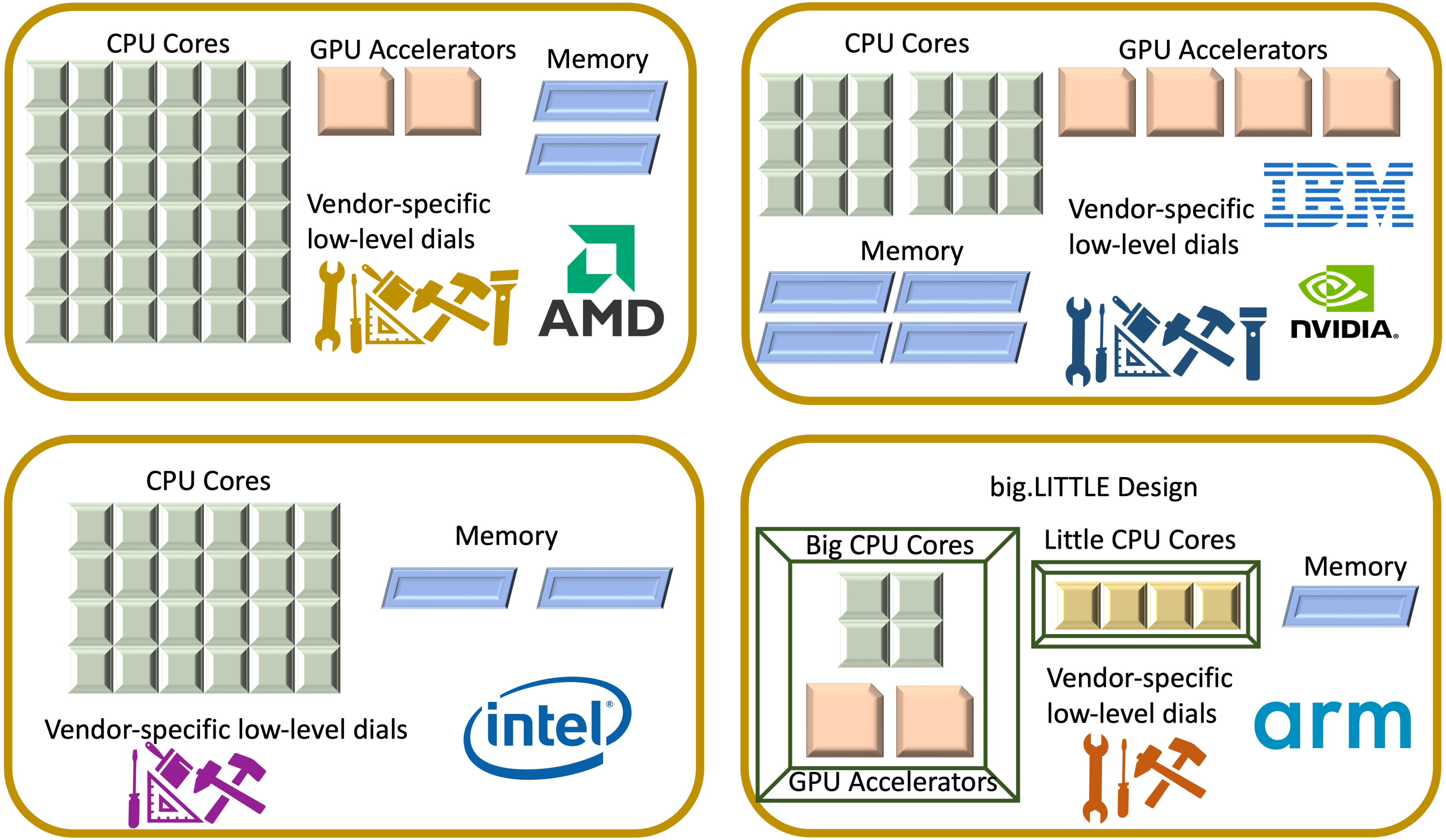
Source: Daniel A Ellsworth, Allen D Malony, Barry Rountree, and Martin Schulz. 2015. Dynamic power sharing for higher job throughput. In Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis. ACM, 80

Variorum

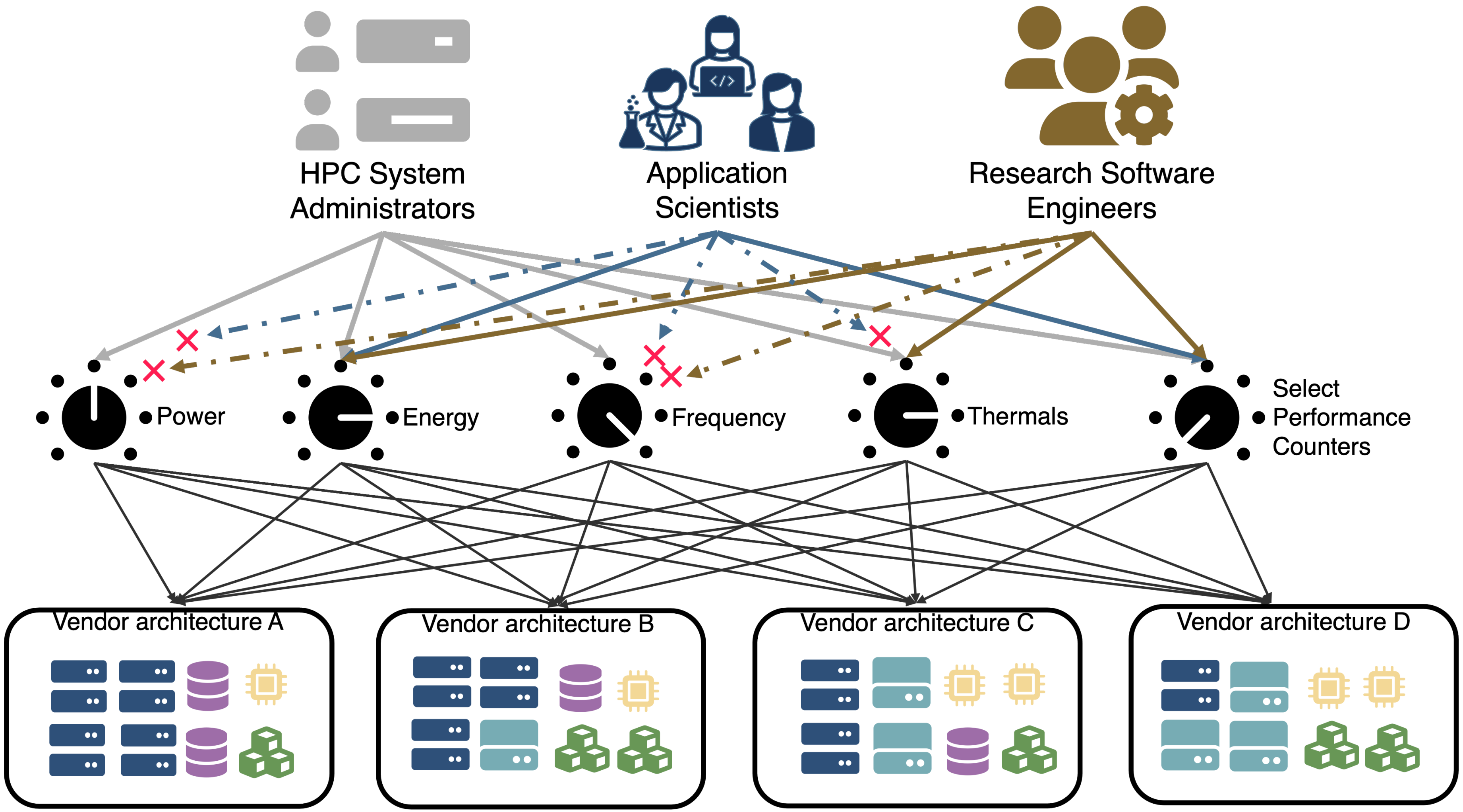
- Power-performance curves for HPC applications vary significantly due to their CPU, GPU and memory characteristics. Many applications do not utilize the amount of power that is procured to them. Performance of such applications is not impacted by giving them less power, as can be seen on the graph, where some applications continue to have the same runtime despite power being reduced by half (115 W to 51 W).
- Allocating power to where it is needed most results in optimal performance, energy savings, and higher scientific throughput, requiring a power stack which allows for intelligent power scheduling.

EXASCALE COMPUTING PROJECT

Low-level dials and interfaces for power and performance management vary significantly across vendors



Accessing vendor-specific dials is unwieldy for domain scientists as well as experts, due to lack of uniform APIs, privileged access, and limited documentation

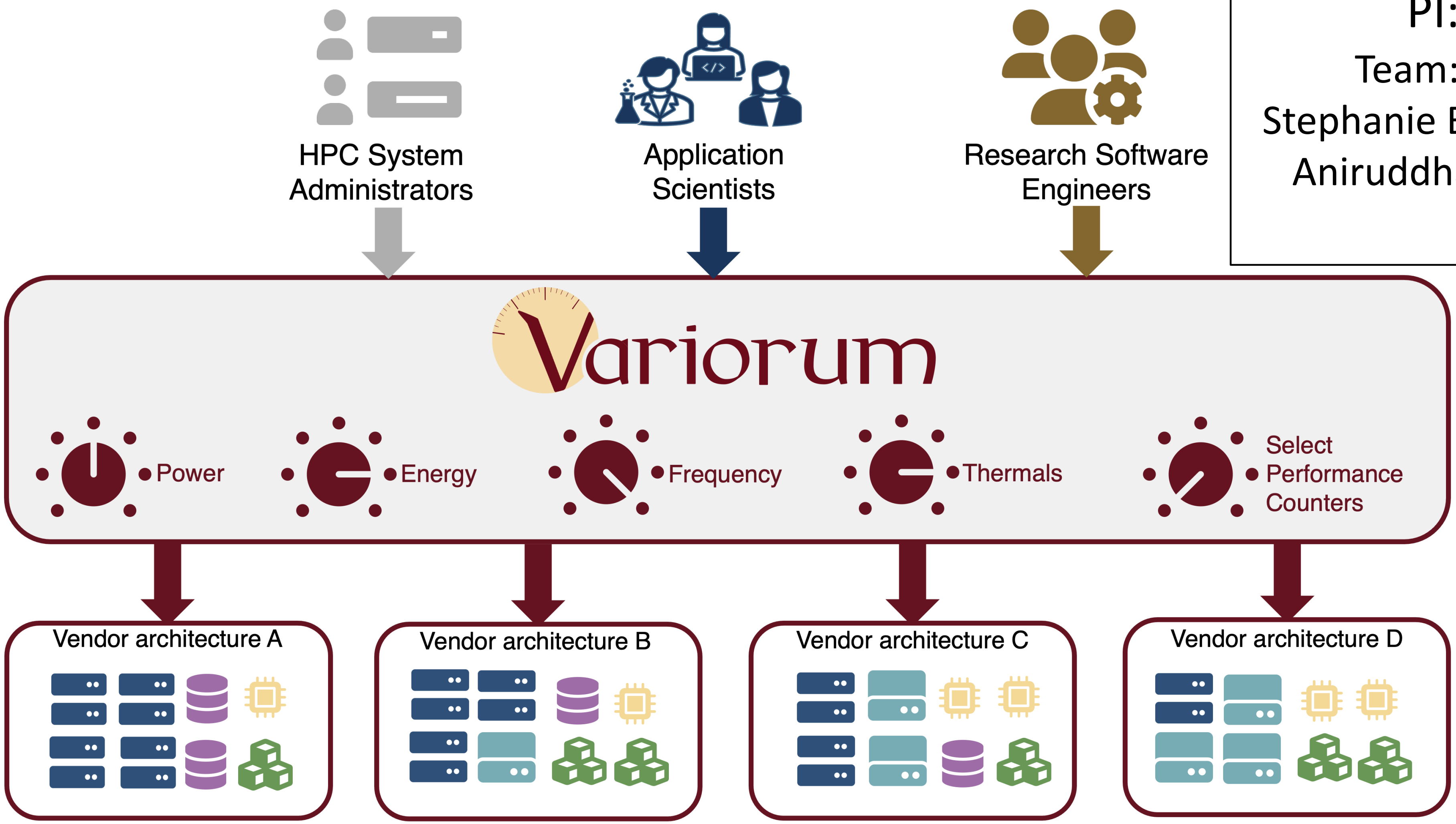


EXASCALE COMPUTING PROJECT

Variorum provides safe, user-space, vendor neutral access for all users: administrators, application scientists and RSEs

2023 R&D 100 WINNER

PI: Tapasya Patki
Team: Kathleen Shoga, Stephanie Brink, Eric Green, Aniruddha Marathe, Barry Rountree



Variorum: Vendor-neutral user space library for power management

- Power management capabilities (and their interfaces, domains, latency, capabilities) widely differ from one vendor to the next, needing common interfaces
- Variorum: Platform-agnostic vendor-neutral, simple front-facing APIs
  - Evolved from *libmsr*, and designed to target several platforms and architectures
  - Abstract away tedious and chaotic details of low-level knobs
  - Implemented in C, with function pointers to specific target architecture
  - Integration with higher-level power management software through JSON
- Integrated with Flux, GEOPM, LDMS, Kokkos, Caliper and PowerAPI to enable a PowerStack
- Supported on all upcoming exascale systems (Aurora, Frontier, El Capitan) and several other supercomputers: architecture support includes CPU support for ARM, AMD, Intel, IBM; and GPU support for NVIDIA, AMD and Intel.

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 10 slides are leave-behind as a sample of the 70 products ECP has contributed to

E4S: A Software Stack for ECP and Beyond

EXASCALE COMPUTING PROJECT

Extreme-scale Scientific Software Stack (E4S)

- 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: Nov 2023: E4S 23.1 – 120+ 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 Release 23.11 – November	Build caches 10X build time improvement
Turnkey stack A new user experience	https://e4s.io	Post-ECP Strategy LSSw.io, PESO

E4S

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

https://spack.io

Spack lead: Todd Gamblin (LLNL)

- E4S uses the Spack package manager for software delivery
- Spack used by a large collection of software tools and libraries
- Supports achieving and maintaining interoperability between ST software packages
- Has increasing support for many ecosystem features (build caches, testing, etc)
- When ECP libraries and tools decided on Spack in 2016:
  - Success was not guaranteed
  - Now we are thrilled with the choice ☺

E4S: Extreme-scale Scientific Software Stack

- E4S is a community effort to provide open-source software packages for developing, deploying and running scientific applications on HPC platforms.
- E4S has built a comprehensive, coherent software stack that enables application developers to productively develop highly parallel applications that effectively target diverse exascale architectures.
- E4S provides a curated, Spack based software distribution of 120+ HPC, EDA (e.g., Xyce), and AI/ML packages (e.g., TensorFlow, PyTorch, JAX, Horovod, and LBANN).
- With E4S Spack binary build caches, E4S supports both bare-metal and containerized deployment for GPU based platforms.
  - X86_64, ppc64le (IBM Power 10), aarch64 (ARM64) with support for GPUs from NVIDIA, AMD, and Intel
  - HPC and AI/ML packages are optimized for GPUs and CPUs.
- Container images on DockerHub and E4S website of pre-built binaries of ECP ST products.
- Base images and full featured containers (with GPU support) and DOE LLVM containers.
- Commercial support for E4S through ParaTools, Inc. for installation, maintaining an issue tracker, and ECP AD engagement.
  - https://dashboard.e4s.io https://e4s.io/talks/E4S_Support_Oct23.pdf
- E4S for commercial cloud platforms: AWS image supports MPI implementations and containers with remote desktop (DCV).
  - Intel MPI, NVHPC, MVAPICH2, MPICH, MPC, OpenMPI
- e4s-cl container launch tool allows binary distribution of applications by substituting MPI in the containerized app with the system MPI.
- Quarterly releases: E4S 23.11 released on Nov 9, 2023: https://e4s.io/talks/E4S_23.11.pdf

E4S 23.11: What's New?

- E4S includes 120+ HPC packages on ARM, x86_64, and ppc64le platforms, 110K+ binaries in E4S Spack Build Cache
- E4S includes new AI/ML packages: JAX, PyTorch, TensorFlow, Horovod, and LBANN. Updated Python tools including Jupyter notebook.
- E4S includes new applications: ExaGO and previously supported Xyce, Quantum Espresso, LAMMPS, WARPX, Dealii, and OpenFOAM
- E4S includes support for Intel oneAPI 2023.2.1 software (BaseKit and HPCToolkit) in containers on x86_64 with support for HPC packages built with Intel compilers
- New solvers that support SYCL and Intel GPUs for the first time: PETSc, SUNDIALS, and SLATE.
- GPU support: ARM64 (aarch64) with H100 (90) with CUDA 12.1 and NVHPC 23.9
- E4S includes support for CUDA architectures
  - 80 (A100), and 90 (H100) under x86_64
  - 70 under ppc64le (IBM Power 10)
  - 75, 80, and 90 (H100) under aarch64
- Updated E4S tools: Release 1.0 of e4s-alc (à la carte) customizes container images, e4s-cl (container launch) replaces MPI at runtime!
- New AWS E4S 23.11 image [ami-08c2daa0fb4864b90 in US-West-2 OR] with support for 50+ EDA tools with DCV and UPC++, CAF, and Chapel
- Adaptive Computing's ODDC platform for launching E4S images to AWS on multiple EFA enabled nodes using MVAPICH through a web browser
  - https://youtu.be/kudLzNGE9sU

Steady Stream of E4S to ALL Facilities!

ST Development Team

GitHub

ST Development Team works any issues reported

PHASE I

Product establishes:

1. Spack-based package and build
2. Validation testing in E4S testsuite
3. Documentation for install and use
4. Accessible public repository

E4S community policies

E4S

PHASE II

- E4S establishes install at facilities
- E4S packages get tested and validated in facility environment
- New E4S releases automatically tested through ECP CI infrastructure

OUTPUT

- **High-quality Spack recipes, for ECP products, ready for facility systems**

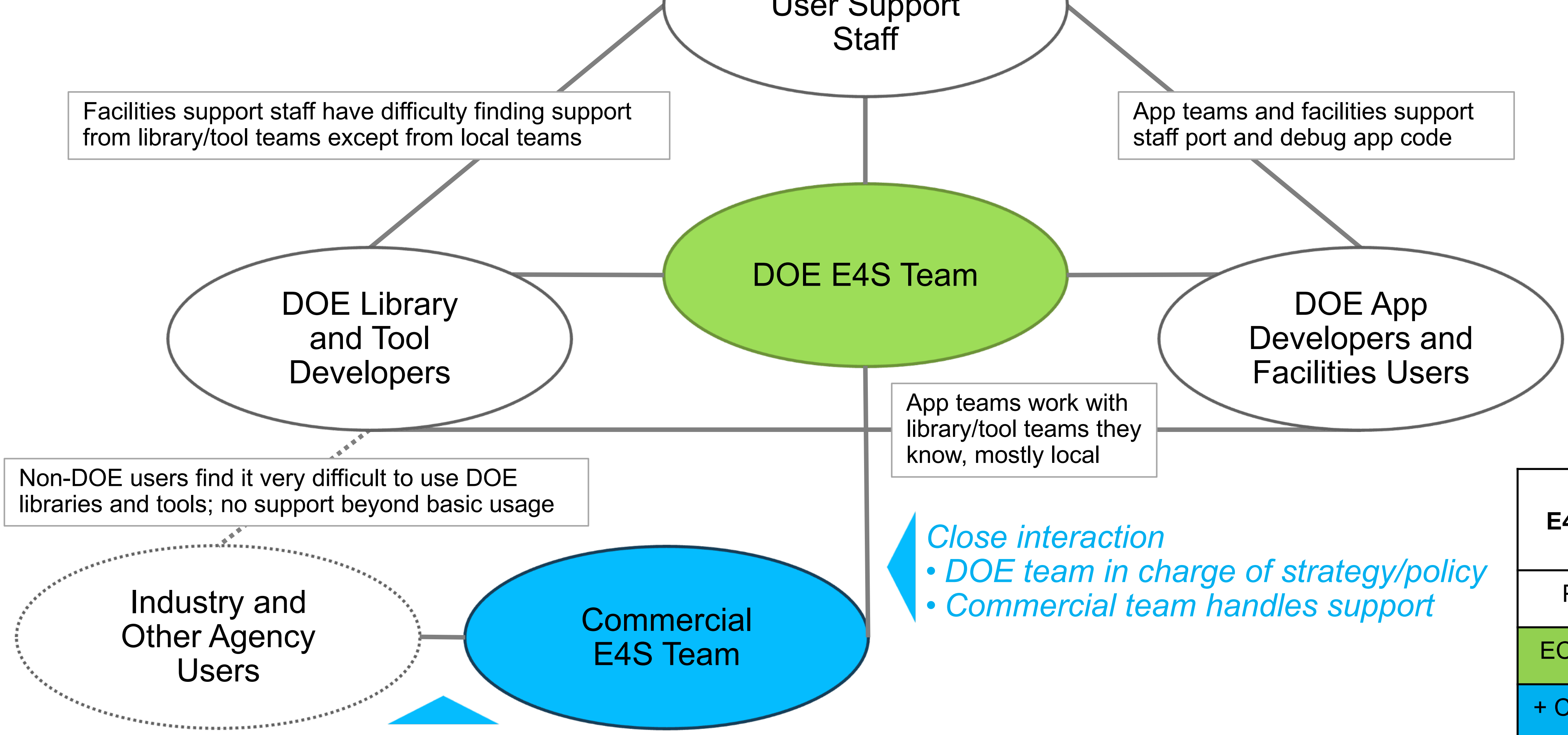
E4S

FEEDBACK PHASE

- **Software Integration team integrates packages into facility system**
- **New E4S release up-streamed and support requests from facility generated as needed**
- **Issues/Fixes/changes worked with developers as needed**

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

First of a kind interactions
- *Industry/agencies can acquire support*
- *Shared costs and benefits with DOE*

E4S Engagements: DOE, Other US Agencies

- DOE
  - NERSC, OLCF, ALCF – Active porting on leadership, exascale platforms
  - Multiple ECP apps: ExaWind, WDPApp, Cinema
  - Emerging Sandia effort: Xyce on E4S on AWS for a summer class
- NSF
  - E4S installed on Frontera, TACC; Bridges-2, PSC; BlueWaters, NCSA; Expanse, SDSC
  - SDSC: E4S Singularity containers available on Open Science Grid High Throughput Computing (https://OSG-HTC.org)
- NOAA
  - E4S base images being used in production on AWS and in custom containers
- DoD
  - Testing installation of E4S on Narwhal, Navy DSRC
- NASA
  - Singularity support for E4S on Pleiades
  - Custom E4S images exploration
  - Day-long workshop, July 18

E4S Engagements: International

- CEA, France: E4S engagement discussed with CEA
  - Workshop planned in July 2023 with ParaTools, SAS
- CSC, Finland: Lumi Supercomputer
  - E4S Workshop in March 2023
  - https://ssl.eventilla.com/event/WL761
  - E4S 23.02 installed on Lumi
- Pawsey Supercomputing Center, Perth, Western Australia
  - E4S workshop planned in April 2023
  - https://pawsey.org.au/event/evaluate-application-performance-using-tau-and-e4s-april-4-5/
  - E4S 23.02 installed on Setonix

> - E4S provides a large stack of reusable software libraries and tools
>   - Build from scratch using Spack, or use via containers, cloud, build caches
>   - Makes stack management easier, portable, lower cost
> - We expect E4S to be one of the most important legacies of ECP

IEEE Computing in Science and Engineering (CiSE) Special Issue on ECP Software

- **Issue Title: *Transforming Science through Software: Improving while delivering 100X***
  Editors: Steven Gottlieb (Managing), Guests: Richard Gerber, Michael Heroux, Lois Curfman McInnes

  This issue will focus on experiences with the practice and science of scientific software development, with emphasis on developing a coherent, portable, and sustainable software ecosystem for high-performance computing, as needed to support the needs of next-generation computational science. The work represented in this issue has been conducted by teams from the U.S. Department of Energy's Exascale Computing Project (ECP), …

- Papers:
  - *Scalable Delivery of Scalable Libraries and Tools: How ECP Delivered a Software Ecosystem for Exascale and Beyond*
  - *Providing a flexible and comprehensive software stack via Spack, E4S, and SDKs,* S Shende, J Willenbring, T Gamblin
  - *Community CI workflows,* Ryan Prout, Ryan Adamson, Shahzeb Siddiqui
  - *Deploying Optimized Scientific and Engineering Applications on Exascale Systems,* Scott Parker and Balint Joo
  - *Advancing scientific productivity through better scientific software,* Lois McInnes + IDEAS-ECP team
  - *Then and now: How our software has improved,* Axel Huebl, Sherry Li, Hartwig Anzt

E4S Summary: One of the most important ECP legacies

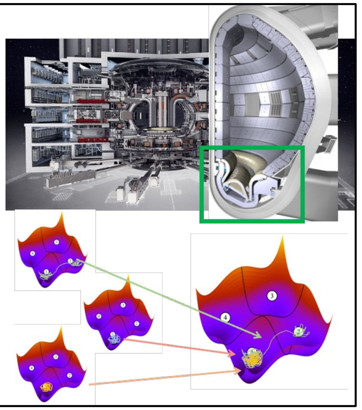
- Large and scalable curated product portfolio
- Includes existing AI products, and can expand easily
- Quantum: quantum processors will be accelerators, like GPUs
- Community policies drive quality
- Frank system provides testing resources
- Built on Spack, the backbone for much of HPC software going forward
- Flexible business models for agency, industry, international collaboration
- Central to post-ECP efforts

Leveraging the Future Potential of ECP Investments

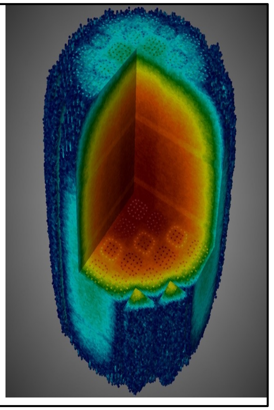
100X

100X Demonstrated: ECP-sponsored application FOMs

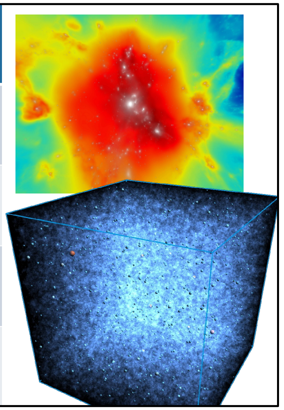
Project/PI	EXAALT: Molecular Dynamics Danny Perez
Challenge Problem	Damaged surface of Tungsten in conditions relevant to plasma facing materials in fusion reactors • 100,000 atoms • T=1200K
FOM Speedup	398.5
Nodes Used	7000
ST/CD Tools	Used in KPP Demo: **Kokkos, CoPa**



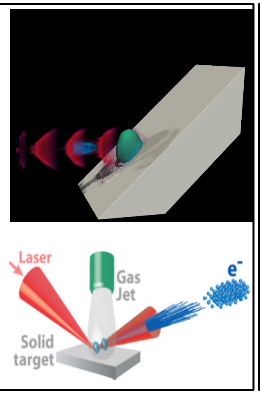
Project/PI	ExaSMR: Small Modular Reactors Steve Hamilton
Challenge Problem	NuScale-style Small Module Reactor (SMR) with depleted fuel and natural circulation • 213,860 Monte Carlo tally cells/6 reactions • 5.12×10^{12} particle histories/cycle, 40 cycles • 1098×10^{6} CFD spatial elements • 376×10^{9}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



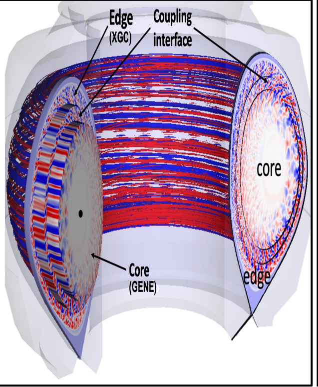
Project/PI	ExaSky: Cosmology Salman Habib
Challenge Problem	Two large cosmology simulations • gravity-only • hydrodynamics
FOM Speedup	271.65
Nodes Used	8192
ST/CD Tools	Used in KPP demo: **none** Additional: CoPa, VTK-m, CINEMA, HDF5.0



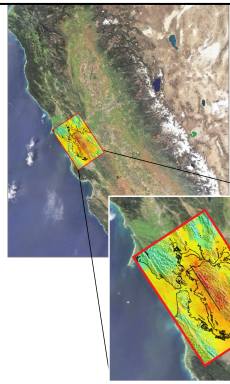
Project/PI	WarpX: Plasma Wakefield Accelerators Jean-Luc Vay
Challenge Problem	Wakefield plasma accelerator with a 1PW laser drive • 6.9×10^{12} grid cells • 14×10^{12} macroparticles • 1000 timesteps/1 stage
FOM Speedup	500
Nodes Used	8576
ST/CD Tools	Used in KPP Demo: **AMReX**, **libEnsemble** Additional: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

KPP ID	Description of Scope	Threshold KPP	Objective KPP	Verification Action/Evidence
KPP-1	11 selected applications demonstrate performance improvement for mission-critical problems	✓ 6 of 11 applications demonstrate Figure of Merit improvement ≥50 on their base challenge problem	All 11 selected applications demonstrate their stretch challenge problem	Independent assessment of measured FOM results and base challenge problem demonstration evidence
KPP-2	14 selected applications broaden the reach of exascale science and mission capability	5 of 10 DOE Science and Applied Energy applications *and* 2 of 4 NNSA applications demonstrate their base challenge problem	All 14 selected applications demonstrate their stretch challenge problem	Independent assessment of base challenge problem demonstration evidence
KPP-3	76 software products selected to meet an aggregate capability integration score	Software products achieve an aggregate capability integration score of at least 34 out of a possible score of 68 points	Software products achieve the maximum aggregate capability integration score of 68 points	Independent assessment of each software product's capability integration score
KPP-4	Delivery of 267 vendor baselined milestones in the PathForward element	✓ Vendors meet 214 out of the total possible 267 PathForward milestones	✓ Vendors meet all 267 possible PathForward milestones	Independent review of the PathForward milestones to assure they meet the contract requirements; evidence is the final milestone deliverable

EXASCALE COMPUTING PROJECT

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**

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**
- **Transitioning software stacks to GPUs is essential**
  - **CPU-based HPC system realize only modest energy efficiency improvements**
  - **Migrating to GPUs is key to improving HPC environmental impact**

Next Steps and Opportunities

7 Software Stewardship Organizations (SSOs)*

- PESO: Stewarding, evolving and integrating a cohesive ecosystem for DOE software.
- OASIS: Stewardship, advancement, and 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.

*Members of the Scientific Software Stewardship Consortium (S3C)

PESO: Partnering for Scientific Software Ecosystem Stewardship Opportunities

Team Member	Affiliation
Michael Heroux	Sandia National Laboratories; PI
Lois Curfman McInnes	Argonne National Laboratory; co-PI
Satish Balay	Argonne National Laboratory
Roscoe Bartlett	Sandia National Laboratories
Keith Beattie	Lawrence Berkeley National Laboratory
Greg Becker	Lawrence Livermore National Laboratory
David Bernholdt	Oak Ridge National Laboratory
Tamara Dahlgren	Lawrence Livermore National Laboratory
Todd Gamblin	Lawrence Livermore National Laboratory
Berk Geveci	Kitware
William Godoy	Oak Ridge National Laboratory
Elsa Gonsiorowski	Lawrence Livermore National Laboratory
Patricia Grubel	Los Alamos National Laboratory
Mahantesh Halappanavar	Pacific Northwest National Laboratory
Bill Hoffman	Kitware
Damien Lebrun-Grandie	Oak Ridge National Laboratory
Mary Ann Leung	Sustainable Horizons Institute
Xiaoye Sherry Li	Lawrence Berkeley National Laboratory
Dan Martin	Lawrence Berkeley National Laboratory
Mark Miller	Lawrence Livermore National Laboratory
Patrick O'Leary	Kitware
Erik Palmer	Lawrence Berkeley National Laboratory
Suzanne Parete-Koon	Oak Ridge National Laboratory
Lavanya Ramakrishnan	Lawrence Berkeley National Laboratory
Sameer Shende	University of Oregon
Rajeev Thakur	Argonne National Laboratory
Matteo Turilli	Brookhaven National Laboratory
Terece Turton	Los Alamos National Laboratory
James Willenbring	Sandia National Laboratories
Hui Zhou	Argonne National Laboratory

https://pesoproject.org

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)

Stakeholders:
Applications Community
Commercial HPC Companies
Industrial Users
US Agencies

DOE Computing Facilities:
ALCF NERSC
OLCF

CRLC: Computational Research Leadership Council: ANL, BNL, LBNL, LLNL, LANL, ORNL, PNNL, SNL

PESO Advisory Board
Reps from ANL, LBNL, LLNL, LANL, ORNL, SNL

S3C Consortium
PESO, COLABS, CORSA, OASIS, STEP, SWAS, S4PST

DOE Program Managers
ASCR: Hal Finkel, Ben Brown, Saswata Hier-Majumder, Robinson Pino, Bill Spotz, David Rabson
NNSA: Si Hammond

PESO: Partnering for Scientific Software Ecosystem Stewardship Opportunities

Mike Heroux, SNL - PI **Lois Curfman McInnes, ANL - Co-PI**

PESO Partnerships

Stakeholder Engagement (Mike Heroux, SNL)

Partnerships Coordinator (Terece Turton, LANL)

Strategic engagement with consortium partners, applications, facilities, industry and agencies (in collaboration with and co-funded by SSOs)

- William Godoy, ORNL, On-node programming systems (w. S4PST)
- Rajeev Thakur, ANL, Inter-node programming systems (w. S4PST)
- Sameer Shende, Univ of Oregon, Tools (w. STEP)
- Sherry Li, LBNL, Math libraries (w. OASIS)
- Berk Geveci, Kitware, Data and viz (w. OASIS)
- Lavanya Ramakrishnan, LBNL, Workflows (w. SWAS)
- Mahantesh Halappanavar, PNNL, AI/ML (w. OASIS)

Unfunded partners: Strategic engagement with NNSA, communities of practice, applications, facilities, industry, agencies

- David Bernholdt, ORNL, RSE engagement (funded by COLABS)
- Addi Malviya-Thakur, ORNL, Foundation engagement (funded by CORSA)
- Elaine Raybourn, SNL, Consortium-wide community development (funded by CORSA)
- Ulrike Yang, LLNL, NNSA software (funded by NNSA)
- Partners at ALCF, NERSC, OLCF (funded by facilities, SW integration)

Community Development (Lois Curfman McInnes, ANL)

Broadening Participation Initiative

- Mary Ann Leung, Sustainable Horizons Institute, PIER planning, lead of Sustainable Research Pathways (SRP)
- Daniel Martin, LBNL, lab lead of Sustainable Research Pathways
- Suzanne Parete-Koon, ORNL, lead of HPC Workforce Development and Retention Action Group

Better Scientific Software (BSSw) Fellowship Program

- Elsa Gonsiorowski, LLNL, Coordinator of BSSw Fellowship Program
- Erik Palmer, LBNL, Deputy Coordinator of BSSw Fellowship Program

PESO Services

Integration Coordinator (Jim Willenbring, SNL)

Software portfolio management and integration (in collaboration with and co-funded by SSOs)

- Damien Lebrun-Grandie, ORNL, On-node prog systems (w. S4PST)
- Hui Zhou, ANL, Inter-node programming systems (w. S4PST)
- Bill Hoffman, Kitware, Tools (w. STEP)
- Satish Balay, ANL, Math libs (w. OASIS)
- Patrick O'Leary, Kitware, Data & viz (w. OASIS)
- Matteo Turilli, BNL, Workflows (w. SWAS)
- Sam Browne, SNL, NNSA software (funded by NNSA)

SQA & Security (David Bernholdt, ORNL)

- Ross Bartlett (SNL)
- Berk Geveci (Kitware)
- Jim Willenbring (SNL)

PESO Products

E4S (Sameer Shende, U Oregon)

- Luke Peyralans, Erik Keever, Wyatt Spear, Jordi Rodriguez

Spack (Todd Gamblin, LLNL)

- Greg Becker, LLNL
- Tammy Dahlgren, LLNL

Port & Test Platforms (Gamblin & Shende)

- In partnership with Univ of Oregon, Cloud, etc.

BSSw.io Content (w. COLABS)

- Ross Bartlett, SNL
- Keith Beattie, LBNL
- Patricia Grubel, LANL
- Mark Miller, LLNL

PESO

ECP did leadership science on leadership systems
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 modest changes 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 (to assure their SW works with ours)**
  - Provide a common test and evaluation setting when working with non-DOE users
- 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

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

Heads up: We would like to contact you about ECP Impact statements

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 dives ongoing

DoD

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

https://e4s.io

E4S lead: Sameer Shende (U Oregon)

https://spack.io

Spack lead: Todd Gamblin (LLNL)

We intend to form the

HIGH PERFORMANCE SOFTWARE FOUNDATION

- Cloud HPC and AI are growing
  - Accelerated computing is becoming ubiquitous
- The world has invested billions in open source HPC software
- After ECP, We can leverage our momentum with a foundation

Initial Projects

HPCToolkit, WarpX, Charliecloud, Trilinos, VTK-m, AMReX, Spack, Apptainer, kokkos, E4S

Initial Working Groups

CI & Testing	Facility Engagement
Software Stacks	Architecture Support
End Users	Performance testing

HPSF will build, promote, and advance a portable core software stack for HPC by increasing adoption, lowering barriers to contribution, and supporting development efforts.

THE LINUX FOUNDATION

Get in touch to become a member!
Email Todd Gamblin: tgamblin@llnl.gov

Post-ECP and Final Remarks

- DOE/ECP 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 community to realize the legacy of ECP, and beyond**
  - We have many new means to interact
  - Many new opportunities to pursue

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.

EXASCALE COMPUTING PROJECT

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.**

Leave behind ECP libraries and tools product sample

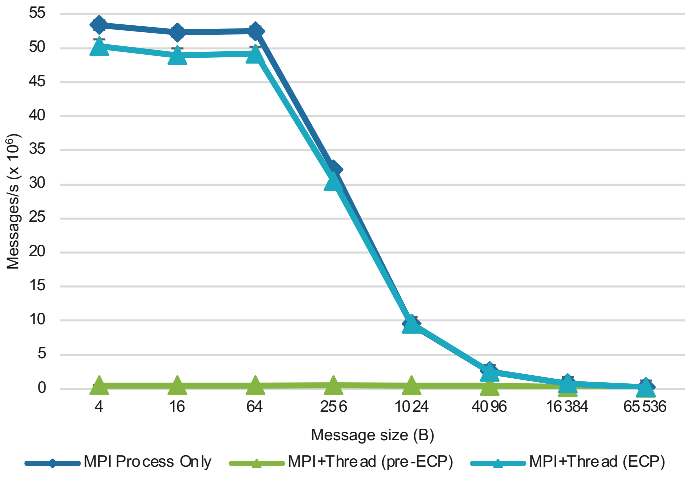
MPICH: Exascale MPI

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

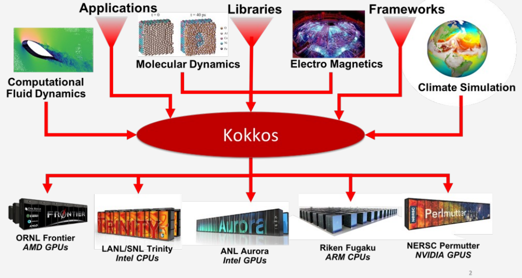
ECP Team and Funding

- 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.

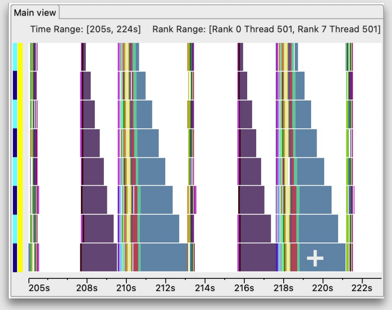
HPCToolkit : Performance Analysis Tools for Exascale

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

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.

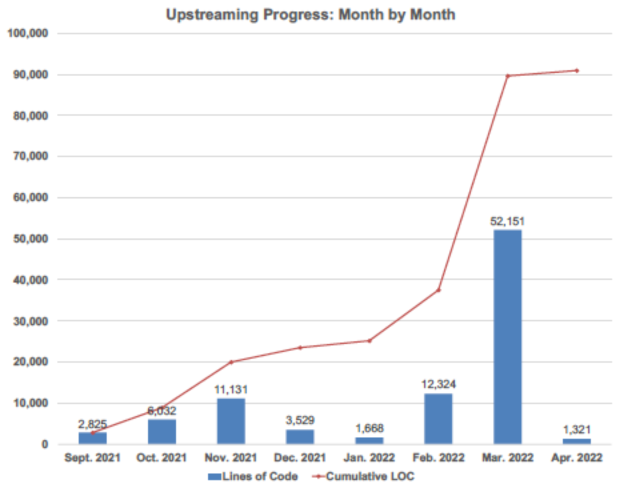
Flang: Fortran front-end for LLVM

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/

EXASCALE COMPUTING PROJECT

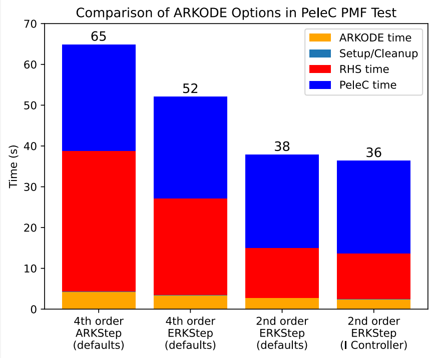
SUNDIALS: Time Integrators for High Performance Systems

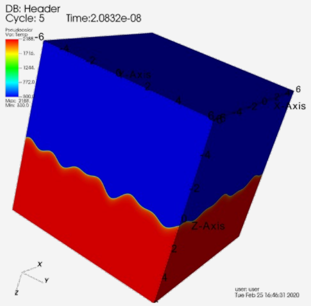
ECP Team and Funding

- 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.

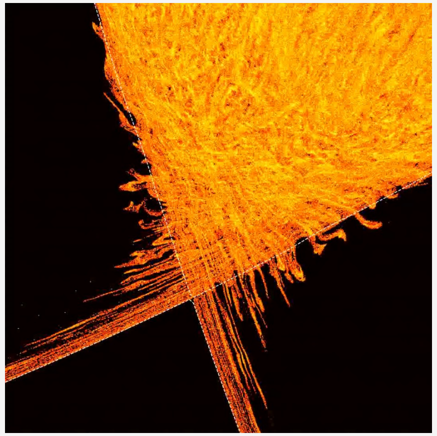
VTK-m: Visualization on Accelerators

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

VTK-m

https://m.vtk.org

AMReX: Block-structured Adaptive Mesh Refinement

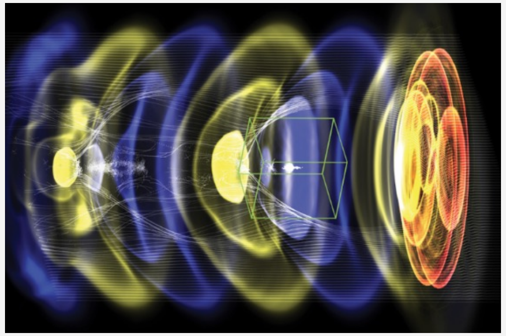
ECP Team and Funding

- 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

EXASCALE COMPUTING PROJECT

AMReX

https://amrex-codes.github.io

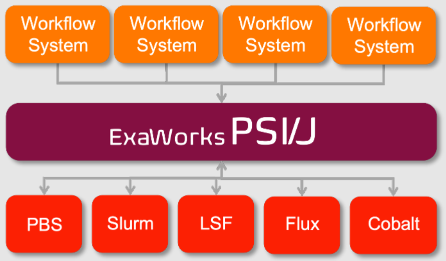
ExaWorks SDK: an SDK for Exascale Workflows

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

EXASCALE COMPUTING PROJECT

ExaWorks

https://exaworks.org

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)

Umpire kokkos flux CINEMA RAJA Charliecloud SMALL SOFTWARE DELIVERING BIG FLEXIBILITY TO HPC Spack Caliper VTK-m

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