Science Impact and 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)

Virtual Meeting, June 20, 2023

PESO Community Workshop

Held at

Argonne National Laboratory

June 8 – 9, 2023

All workshop content available at:

https://bit.ly/peso-workshop-june2023-sharedcontent

Agenda

Thursday, June 8, 2023			
Time (CT)	**Topic**	**Leads**	**Comments**
7:30 am	Registration / Working Breakfast		
8:30 am	Plenary Kickoff	Mike Heroux (PESO PI)	Give overall workshop charge
9:00 am	Applications Perspectives	Andrew Siegel (ECP Applications Development Director)	Challenges and opportunities for increased impact of libraries and tools on application success
9:30 am	Industry Perspectives	Jeff Larkin (NVIDIA), Berk Geveci (Kitware)	Challenges and opportunities for increased impact of libraries and tools in collaboration with industry
10:00 am	Break		
10:30 am	Advanced Computing Facilities Perspectives	Katherine Riley (ALCF), Balint Joo (OLCF), Richard Gerber (NERSC)	Challenges and opportunities for increased impact of libraries and tools in collaboration with computing facilities
11:15 am	Set up for breakouts		Describe charge questions, take Q&A, locate breakout rooms
12:00 pm	Lunch		
12:30 pm	Lunchtime Talk	Ulrike Yang (ECP xSDK Project PI)	How software product communities can enhance the productivity of teams
1:00 pm	Breakout session		
2:45 pm	Break		
3:15 pm	Breakouts resume		
4:00 pm	Report out from breakouts		
5:00 pm	Adjourn		Dinner on your own

Friday, June 9, 2023			
Time (CT)	**Topic**	**Leads**	**Comments**
7:30 am	Working Breakfast		
8:30 am	Challenges and Opportunities for Computing	Rick Stevens (ANL, Assoc Lab Director, Computing, Environment and Life Sciences)	Roles of sustainable software ecosystems in addressing next-generation computing challenges
8:45 am	Software Foundations	Todd Gamblin (PESO co-PI)	How we can leverage software foundations for DOE software sustainability
9:00 am	Workforce Development	Lois Curfman McInnes (PESO co-PI), Mary Ann Leung (Sustainable Horizons Institute)	Challenges and opportunities for broadening participation in the HPC workforce
9:15 am	Breakout session		
10:15 am	Break		
10:45 am	Breakouts resume		
12:00 pm	Working Lunch		
12:30 pm	Report out from breakouts, discussion		
2:00 pm	Adjourn		

PESO Project Description

Progress on defining PESO

Note: PESO is a work in progress!

Your contributions are essential to define and shape it!

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

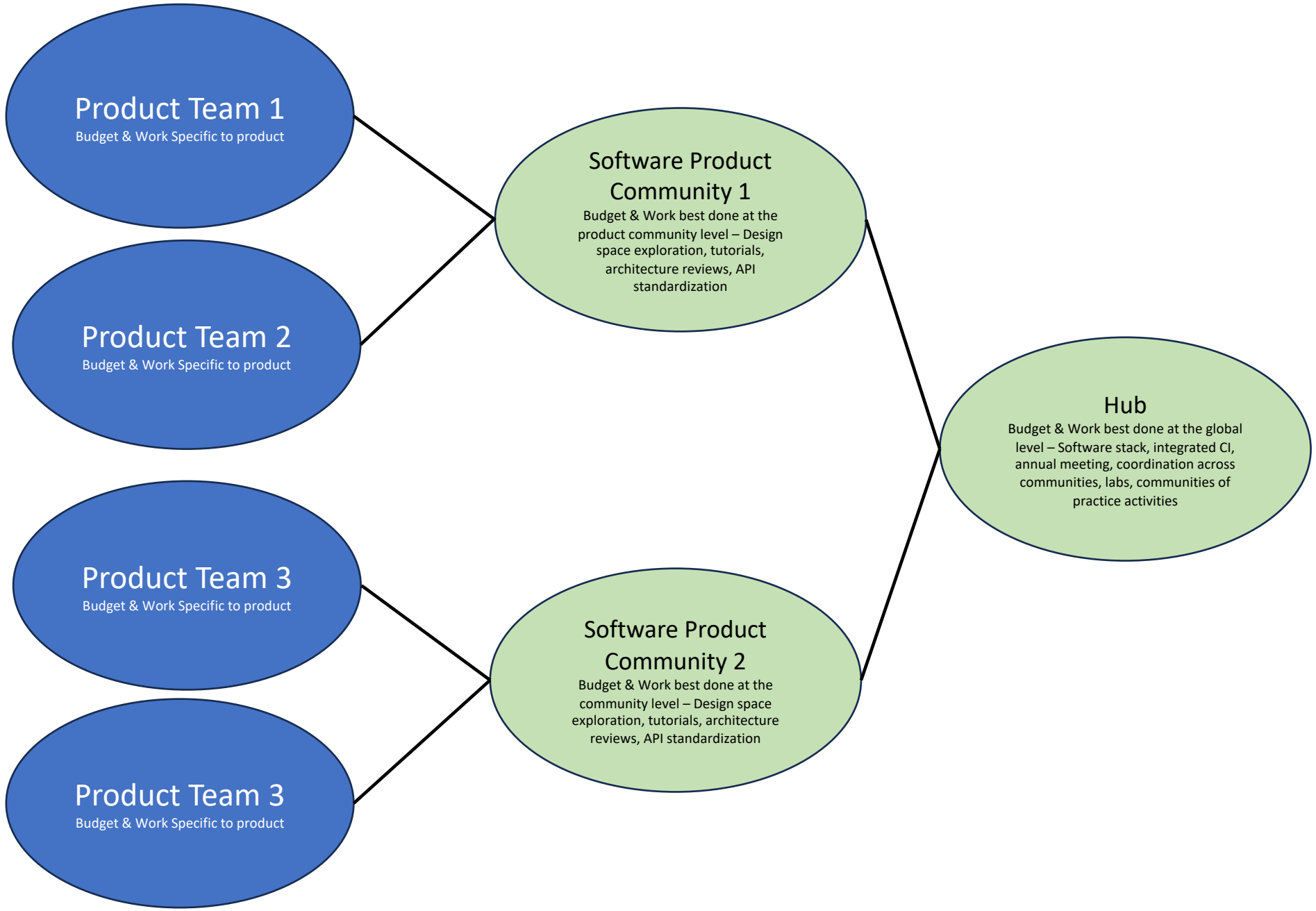
- Spack+E4S
  - Major PESO deliverable
  - Important product delivery conduits
  - Platforms for agency, industry collaboration
  - Keys for testing on new and diverse platforms & software environments
- PESO = Spack+E4S+More:
  - Impact on science via 100X efforts
  - Collaborative planning, executing tracking and reporting
  - HPC community engagement: apps, facilities, vendors, agencies
  - Cross-cutting training, community engagement, sustainability efforts
  - More

PESO 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 project teams** to coordinate development activities for long-term sustainability and benefit to stakeholders
- Work with
  - **software product communities (SPCs) and**
  - **communities of practice (COPs)**
- To provide cross-cutting services and support that are broadly needed by developers, users, and stakeholders
- Realize the full potential of DOE investments in the scientific libraries and tools ecosystem:
  - By taking a broad, strategic view
  - Through project growth, improved software quality and availability, and sustainable delivery, deployment, and support.
  - Realizing the 100X potential enabled by ECP investments

PESO Proposed Organization Strategy



- A lot of work is best done at the individual product team level
  - Everyday development work
  - Delivery of capabilities that contribute part of the whole
  - Testing and product improvement
- Some work is best done at a product community level:
  - Portfolio planning, coordination
  - Holistic tutorial delivery
  - Design space exploration for next-gen platforms
- Some work is best done at hub level:
  - Software stack management
  - Specialized CI testing
  - All-team meetings
  - Coordinated planning across portfolio
  - Community of practice activities: working with software foundations, improving software skills, community engagement

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

PESO Key Services and Activities

- **Collaboratively steer funding** to key projects to maintain a robust software ecosystem in the long term
- **Delivery and deployment** via Spack and E4S to DOE Facilities, on-premise and cloud users and developers, including CI testing, issues triage, build caches, and software quality assurance
- **Coordination** of cross-cutting engagement with DOE Facilities, DOE sponsors, and other stakeholders such as other US agencies, US industry, and international partners
- **Lightweight processes, models and tools** to support software product communities with their software lifecycle management activities, including annual planning, execution, tracking, and assessment with change management
- **Basic infrastructure** for community engagement in coordination with communities of practice for outreach, training, community development, and coordination with external collaborators
- **Outreach and workforce development** to incubate new projects and grow the contributor base and external investment in key projects

Key value proposition: By engaging with PESO each product team and community will be better off than without PESO:

- Help you improve your product quality, and availability - through PESO collaboration, better build, test, integration and distribution
- Be part of an ambitious, collaborative, and dynamic community – Contribute to something bigger than ourselves, to DOE mission

PESO Financial Model

- **Clear values and transparency:** Product and community funding is based on a transparent set of criteria applied through an open review and assessment process
- **Low overhead:** Funds for all efforts at a particular lab are sent directly from the sponsor as a lump sum to the lab with itemized amounts for individual projects at that lab. Funds meant for university and industry subcontracts are sent from sponsor to the most appropriate partner lab. All funds at each lab are managed by the normal lab funding infrastructure

PESO Sustainability Strategy

- Sustained resources – funding, effort, and infrastructure
- Robust user and developer base
- Focus: HPC libraries and tools ecosystem sustainability
  - Includes DOE, ECP, Advanced Scientific Computing scope as relevant
  - Critical functionality is sustained even as specific products come and go
  - Transitions into, out of, and within ecosystem are explicitly managed
  - Vendor products are integral to planning and collaborations
  - Evolution to meet community needs
- Critical mass:
  - Using hub and spoke approach to combine efforts
  - Leverage our aggregate scale for external influence

PESO PIER Plan

- Multifaceted approach to advance diversity, equity and inclusion throughout all work in the project, with emphasis on two complementary layers of scope:
  - Activities within the project and
  - Partnerships with others to plan and lead work toward culture change in our community overall
- In both contexts, we will address:
  - **Recruitment and inclusion**, with emphasis on engaging diverse individuals from underrepresented groups as members of our teams and community
  - Cultivating **work environments** that promote mutual respect and professionalism, with emphasis on sharing best practices and effecting culture change
  - Planning for scholarly and **professional growth of community members**, with particular emphasis on research software engineers (RSEs) and early-career staff

Seed project interactions

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
- OSSF: Open Scientific Software Foundation
  - Lead Greg Watson, explore the role of software foundations to address sustainability

All-seed projects May 2023 collaborations

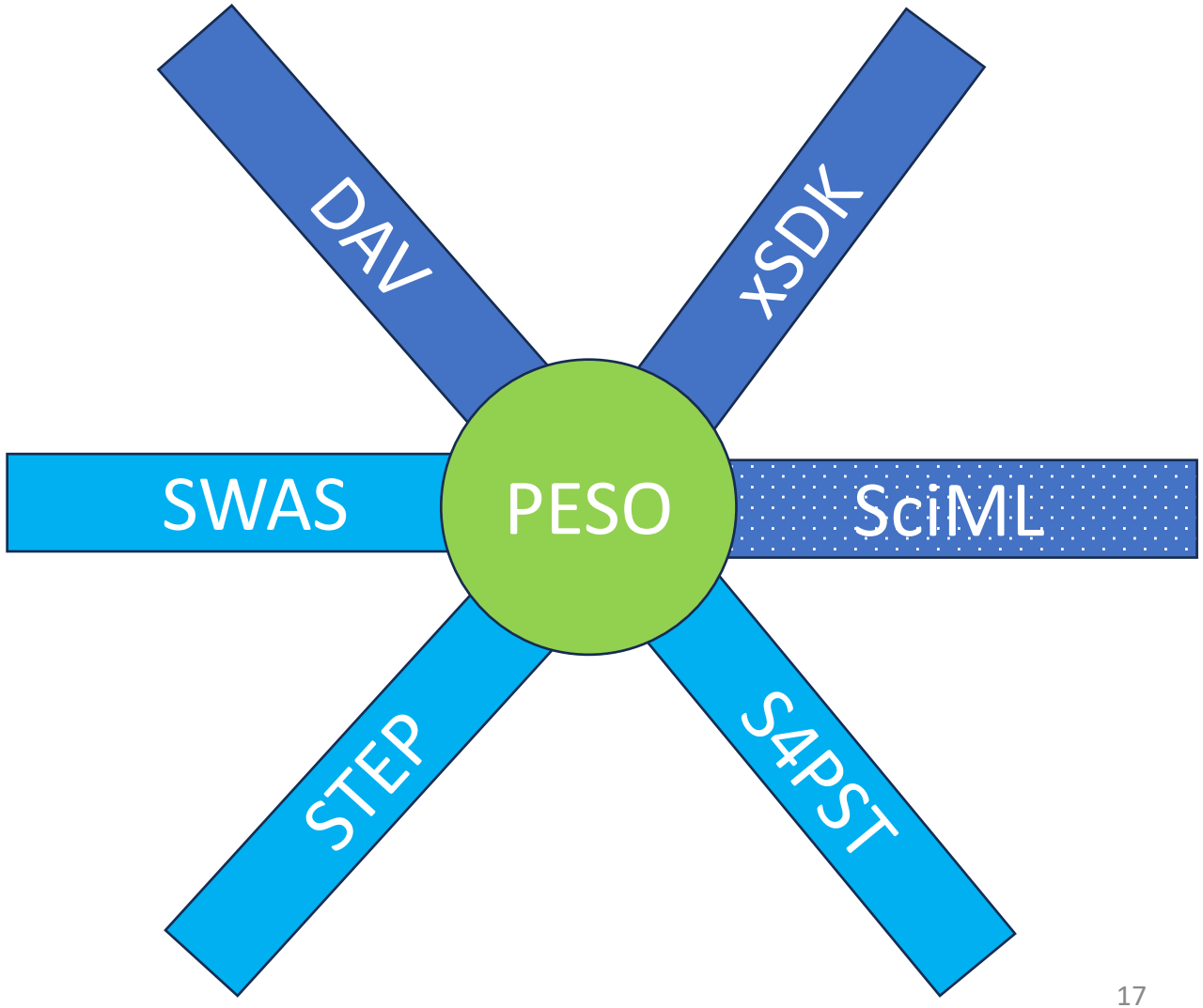
- May 2023 updates
  - LSSw.io (https://lssw.io)
    - Pointer to seed project proposals and web pages https://lssw.io/SeedProjects
    - Meeting 12: (May 18) Promoting inclusive and equitable research https://lssw.io/Meetings/Meeting12
    - Meeting 13: (June 15) Seed projects update https://lssw.io/Meetings/Meeting13
    - Meeting 14: (July 20) Software foundations https://lssw.io/Meetings/Meeting14
  - Bi-weekly all-PI meetings with all seed projects
  - Focused meetings and workshops
    - Pair-wise sync up and outbriefing: May 18 – 19
    - Pair-wise sync up and outbriefing: July 21
  - PESO has initiated collaboration plans with STEP, S4PST, and SWAS seed projects
  - Collaboration options with COLABS and OSSF are under discussion

PESO Hub and Spoke Approach

- Software Product Communities (SPCs), (aka, SDKs, Spokes)
  - PESO intends to be a hub that aggregates with communities compose of like products
  - We anticipate SPCs will self-organize and have community-specific governance
  - We anticipate SPCs to include DOE-sponsored and commercial/community software
  - SPC value proposition includes – Shared design space exploration, coordination, more
- Communities of Practice (COPs)
  - PESO anticipates engaging with community leaders in important cross-cutting efforts
  - Examples include:
    - Scientific software developers: (e.g., IDEAS, HPC Best Practices webinars)
    - Community outreach (e.g., Center for Scientific Collaboration and Community Engagement (CSCCE)
    - Software foundations (e.g., NumFOCUS, Linux Foundation)
    - Workforce development (e.g., US RSE, BSSw Fellows, and Sustainable Research Pathways)
- Funding: Transparent criteria and process, with block funding

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, STEP, S4PST** – Funded seed projects for workflows, tools, programming systems
- **DAV, xSDK** – 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



The Seedling Landscape - Spokes

3 product community-focused seedlings: SWAS, STEP, S4PST

- If funded, these orgs would aggregate well with PESO
- We also expect to collaborate with other SPCs:
  - Math Libraries – xSDK
  - Other libraries and tools that are not part of SWAS, STEP, S4PST but high priority
  - Emerging areas: AI for Science
- Focus on how these seeds and PESO will align
  - What they would need from PESO *beyond* their communities
  - How PESO could engage with key communities through them
  - Expected that these seeds will include non-DOE funded products
- Spoke attributes:
  - Tight domain compatibility: Math Libs, Workflows and app services, programming systems
  - Products sponsored by DOE **and** products not: Vendor teams, other community products (NSF)
  - Deep community collaboration in exploring next-gen features, computing environments, etc.
  - Holistic community engagement in tutorials, requirements gathering, etc.

PESO – Communities of Practice

Software quality improvement often cuts across many product efforts:

- Software practice improvement helps individual and teams
  - Across the ecosystem – common needs
  - Special funding only needed for coordination, coaching
- Examples:
  - How to incorporate generative AI tools and workflows into development
  - How to build an intentional community of engagement
  - If and how to join a foundation
  - More

The Seedling Landscape – Communities of Practice

3 broader meta-organizations (Hubs): PESO, COLABS, OSSF

- If COLABS or OSSF funded, PESO would
  - coordinate with and
  - coexist *alongside* these other organizations
- Looking for synergies and potential mismatches
- Trying to understand how funding models fit together

Highlights from PESO Request for Input Responses

Full summary of 15 questions and responses is available here

How to improve software sustainability

- **Testing and Quality Assurance:** There's a strong emphasis on enhancing the testing infrastructure, including unit testing, continuous integration (CI), and deployment testing. This also involves establishing testing environments similar to user environments and using diverse application sets for testing. Also mentioned is the creation of a comprehensive test suite and regular code refactoring. Formalization and automation of development practices and procedures are seen as crucial.
- **Portability and Maintenance:** Several responses highlighted the importance of making the software compatible with newer languages and architectures, as well as improving portability on future systems. This includes maintenance activities for tools such as repositories, issue tracking, and CI, as well as humans for code reviews, bug fixing, training, and documentation. Maintenance of the project on major systems for testing and support was also mentioned.
- **Documentation and Training:** This includes improving and developing user and developer documentation, online short courses for users, tutorials, and other training materials. Some suggested that a software sustainability organization could provide a venue for sharing and highlighting documentation and tutorial development.
- **Community Engagement and User Support:** Responses suggested expanding the developer community, using modern community communication tools, and organizing training and outreach events. Efforts should also be made to increase the software's visibility, encourage more registrations, and identify users' potential needs. A suggestion was made to foster collaborations through a software sustainability organization to smoothen the user experience.
- **Integration and Interoperability:** This involves improving modularity of various features, integrating the software into larger communities, and providing interfaces to other packages. Better architectural documentation and support for application developers were also considered important.
- **Funding and Support:** Dedicated funding for software maintenance, quality assurance, documentation development, porting, and testing was recommended. If sustainability organizations provide personnel, the preference is for the personnel to work with the project team.
- **Forward Planning and Innovation:** Respondents mentioned feasibility studies for emerging technology and software engineering practices, as well as the need to innovate to address potential roadblocks such as language sustainability. Early access to upcoming technologies was also viewed as important.
- **Standards and Automation:** Some respondents advocated for driving standards and automating more aspects of the developer workflow, like code formatting and updates.
- **Development of Advanced Features:** The need for more efficient implementations for advanced hardware support, more fine-grained power monitoring, and support for modern workflows consisting of AI, Big Data, and Data Science workloads were also pointed out.

PESO, Software Product Communities, and Communities of Practice can positively impact most of these

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 methods.

Some factors for selecting and assessing a product as part of the portfolio - **developers**

- **User Base & Community:** Assess the current and potential user base, both within specific domains (like the Department of Energy - DOE) and across wider scientific and industrial applications. In addition, consider the size of the developer community, the participation of the product developers in the software community, and the vibrancy of the community that maintains and develops the product.
- **Relevance & Importance:** Evaluate the product's relevance and importance to its users and to the organization's mission, such as its criticality to technology, level of usage in high-performance computing (HPC) communities, and its inclusion in significant initiatives like large DOE hardware procurement or Software Development Kits (SDK).
- **Software Quality & Maintainability:** Look at the quality of the product, its maturity, and best software development practices. This includes unit testing, continuous integration/continuous delivery (CI/CD) practices with high test coverage, code quality, clear documentation, a good build system, and responsive, active development. The design of the product should encourage sustainable development.
- **Usability & Scalability:** Ensure the product is easy to use and can scale well to large problems and machines. This includes considering installation procedures, user support mechanisms, and deployability.
- **Performance & Portability:** The product should exhibit superior performance and be portable on future systems, taking into account the overall performance, scalability, and programmability of the software.
- **Support & Sustainability Plan:** Consider the level of support for the product, such as superior customer support, a supporting ecosystem around the product, and a dependable developer community. Furthermore, there should be a sustainable plan in place for long-term sustainment.
- **Innovation & Potential for Growth:** Evaluate the potential for the product to fill an unmet need in the community and its potential for growth, especially in emerging areas. A quick adoption rate could indicate high future impact.
- **Alignment with Existing Portfolio:** The product should be assessed in context with the existing portfolio and its alignment with the mission of the organization's portfolio. It's crucial to avoid unnecessary overlap with other products.
- **Openness & Extendibility:** Open-source products with numerous contributors, adherence to standards, and a well-maintained development community are highly preferred. Moreover, the software should be extensible and maintainable.
- **Uniqueness & Added Value:** The product's unique capabilities, either in performance or functionality, should be assessed, as well as the added value it brings to the community.

Across all factors: Balance between supporting successful, mature products and nurturing innovative, emerging ones.

Some factors for selecting and assessing a product as part of the portfolio - **facilities**

- **Interaction with Vendors:** The relationship and communication channels with the software's creators or suppliers.
- **Performance Portability:** The software should have the ability to efficiently operate across different hardware configurations or operating environments.
- **User and Project Demand:** The software should have a significant user demand or meet the requirements of major projects.
- **Broad and Strategic Impact:** The software should have a widespread influence, particularly on strategic objectives such as enabling portability or enhancing key investments.
- **Support and Sustainability Models:** The mechanisms for maintaining and updating the software, and its long-term viability.
- **Utilization by the HPC Applications Community:** The software's usage rate and acceptance within the high-performance computing (HPC) community.
- **Community Interest and Participation:** The software should generate interest from the user community, who should be actively involved in its maintenance and enhancement.
- **User Base and Potential for Growth:** The software should have a large user base or the potential for one, and it should be vital enough to generate organic growth.
- **Impact and Value Added:** The software's influence on its user community and the added value it brings.
- **Infrastructure:** The software should provide valuable infrastructure to support its user base and future growth.
- **Vision for the Future:** The software should be forward-looking in all aspects, anticipating and accommodating future needs and trends.

Critical factors that determine whether you will use an open-source library or tool

- **Licensing:** The library or tool must have acceptable license constraints and restrictions. It should be compatible with the project's or organization's licensing terms.
- **Functionality:** The tool or library should be able to fulfill specific needs and improve productivity.
- **Developer and Community Support:** The tool or library should have ongoing support from its developers, including regular updates and bug fixes. A robust user and developer community is also crucial for problem-solving and help.
- **Maturity and Documentation:** The tool or library should have reached a level of maturity, proven stability and should have comprehensive documentation to assist its users.
- **Compatibility and Portability:** The library or tool should support the programming languages used in the project and be portable across different platforms.
- **Quality:** The library or tool should have a high standard of development quality, including extensive testing and active maintenance on bug reports and pull requests.
- **Ease of Use and Learning Curve:** The tool or library should be easy to use, have a short learning curve, and be capable of integration with other tools and libraries.
- **Active and Welcoming Development Team:** The team behind the library or tool should value user feedback and provide a supportive environment for users.
- **Performance:** The library or tool should deliver high performance and low latency.
- **Sustainability:** There should be recent development activity and a long-term support model, indicating a high likelihood that development will continue.
- **Vendor Support:** For some users, particularly in High Performance Computing (HPC) environments, vendor support may be a significant factor.
- **Part of a Larger Ecosystem:** The tool or library should preferably be part of a larger, friendly user base, like StackOverflow or other forums, which can offer additional support and resources.

Barriers to adopting an open-source library or tool

- **Long-Term Viability Concerns:** There are reservations about the durability of open source solutions for industry issues. Open-source projects often depend on community support and development, and their sustainability can be uncertain. A dedicated software sustainability effort can alleviate these concerns by ensuring ongoing support, development, and maintenance of these projects.
- **Adapting to Evolving Hardware:** With rapidly changing hardware platforms, software obsolescence is a significant concern. Sustainable software initiatives can help address this issue by ensuring that software remains compatible and efficient with the latest hardware advancements.
- **Software Licensing Issues:** Some pointed out the need for "Apache or better" software licenses that permit free commercial use. This barrier suggests a need for clear and open-source friendly licensing, allowing companies to adopt software without concerns over legal ramifications or costs. Software sustainability efforts can help ensure that the licensing of software promotes its widespread use and longevity.

User Base Growth – How a sustainability org can help

- **Collaboration and Community Engagement:** The user base for these packages primarily grows through collaborations, leveraging existing communities, and word of mouth. Various stakeholders advocate for more active engagement with communities through training sessions, workshops, and conferences.
- **Trust and Support:** Trust in the longevity and support for a software package is a crucial factor in its adoption. Sustainability organizations can play a key role in endorsing a package, thereby fostering trust among prospective users.
- **Focus on Quality and Functionality:** The four pillars of successful software projects are performance, robustness, usability/portability, and documentation & user support. Improving these aspects can significantly increase the user base.
- **Sustainable Funding and Support:** Continuous funding and support are necessary for maintaining and improving these software packages. Sustainability organizations could help provide this support, allowing for the development of new features and ensuring the software remains up-to-date and relevant.
- **Promotion and Awareness:** The role of sustainability organizations in promoting software packages is emphasized. This can involve various activities, including creating consultant positions, promoting the software in widely distributed newsletters, and providing opportunities for package developers to participate in relevant events and publications.

Outreach Support – How a sustainability org can help

- **Organization and Support:** Assistance in organizing and supporting these activities is frequently requested. This includes support for staff and infrastructure, coordinating courses and workshops, and helping with technical aspects like containerization, integration through Jupyter notebooks, advertising, etc.
- **Endorsement and Advocacy:** Some organizations could benefit from official endorsements, or advocacy within specific networks and programs.
- **Expanding User Base:** A sustainability organization could help identify new groups that could benefit from these tools or assist in diversifying the user base by reaching out to communities like AI, Big Data, and Data Science.
- **Increasing Visibility:** There's a desire for sustainability organizations to help increase the visibility of these tools beyond current circles.
- **Development Support:** This could involve helping with the addition of functionalities, documentation, and improvement of build systems.

Takeaways from Input Responses

- Strong community awareness of importance of sustainability
- Strong common requirements and strategies
- Conjecture: An organization like PESO can be defined to provide:
  - Better – What we produce will be better than with PESO
  - Faster– We will get our work done more quickly
  - Cheaper – We can focus our efforts on other important things

Day 1 Questions

1. Using the following definitions (Draft PESO Definition of Sustainability) of sustainability as a reference,
    1. **What are we missing in the definition?**
    2. **Can we remove anything?**
2. The success of DOE-sponsored libraries and tools hinges on addressing emerging and anticipated applications, facilities, and other stakeholder requirements in service of DOE's mission.
    1. **What are some ways to assure we are identifying and meeting these needs?**
    2. **How can we optimize our impact in collaboration with stakeholders?**
3. How do we transition activities out of our funding portfolio to create room for new activities?
    1. **How can we transition software products to community ecosystems, vendors, and software foundations?**
    2. **What are other ways to create space for new efforts?**
4. PESO is proposing a decentralized Draft PESO Financial Model.
    1. **Do you see any problems with the model?**
    2. **The model is a high-level sketch. What important details must be considered?**
5. PESO proposes to be a hub for software product communities for these Draft PESO Key Services and Activities that are beneficial across all communities and product teams.
    1. **Do you see any issues with this approach?**
    2. **What are some important details that must be considered?**
    3. **How can we assure the ability to fund new projects even if budgets remain level or grow only modestly?**
6. PESO proposes to support software product communities and teams by fostering crosscutting activities that lead to better practices, processes, tools, and community growth.
    1. **What are some of the most important crosscutting activities PESO should promote and support?**
    2. **How should these activities be organized and provided to the software communities and teams?**
7. PESO proposes to sponsor annual events across the entire community.
    1. **How important is an annual in-person meeting that brings together teams, stakeholders, and key members of the external community?**
    2. **The ECP sponsored a virtual Community BOF Days. Is it useful to continue this event in the future?**
8. Prior to this workshop, the PESO team requested input from the community. The key questions and a summary of the 40 responses are found here: 00-Workshop Input Questions and Responses
    1. **Which topics are most important to consider carefully?**
    2. **From the questions and summaries, what would you change?**

Day 2 Questions

9. PESO is committed to workforce development, especially by reaching out to under-represented groups and creating a culture that is inviting, and by promoting the continued training of workforce members and stability of career paths.
    1. **What are the top three workforce challenges or impediments that you see in your organization(s)?**
    2. **What strategies and activities are currently helping to address workforce challenges? Should these be continued?**
    3. **What are the most promising new strategies and activities to address workforce challenges for the future and why?**

10. Last question: What have we not discussed that is important for software-ecosystem sustainment?

Ways to participate in PESO efforts

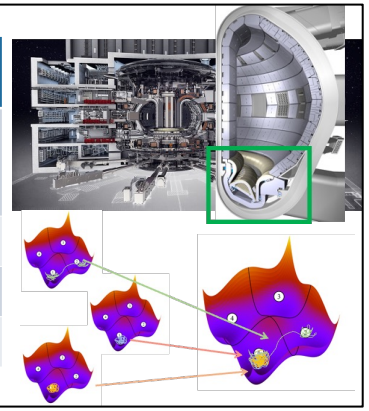
- Invitation to engage in community planning:
- Provide input via the PESO Planning Input Google Form
- Engage in PESO Community Discussions (https://lssw.io/PESO)

PESO Vision

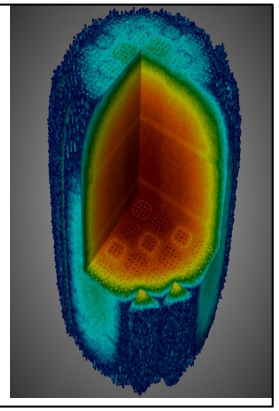
100X

100X Demonstrated: ECP-sponsored application FOMs

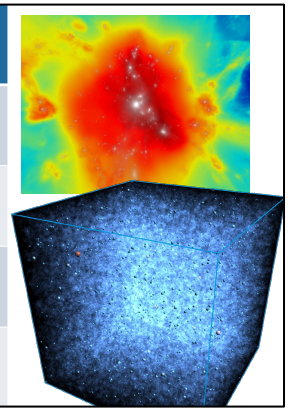
Project/PI	EXAALT: Molecular Dynamics Danny Perez
Challenge Problem	Damaged surface of Tungsten in conditions relevant to plasma facing materials in fusion reactors • 100,000 atoms • T=1200K
FOM Speedup	**398.5**
Nodes Used	**7000**
ST/CD Tools	Used in KPP Demo: **Kokkos, CoPa**



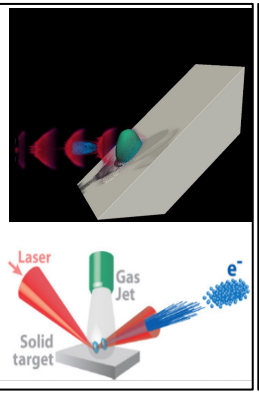
Project/PI	ExaSMR: Small Modular Reactors Steve Hamilton
Challenge Problem	NuScale-style Small Module Reactor (SMR) with depleted fuel and natural circulation • 213,860 Monte Carlo tally cells/6 reactions • 5.12×10^{12} particle histories/cycle, 40 cycles • 1098×10^{6} CFD spatial elements • 376×10^{9}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



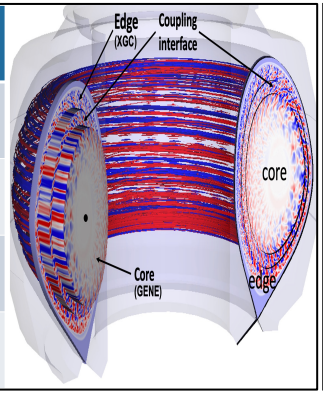
Project/PI	ExaSky: Cosmology Salman Habib
Challenge Problem	Two large cosmology simulations • gravity-only • hydrodynamics
FOM Speedup	**271.65**
Nodes Used	**8192**
ST/CD Tools	Used in KPP demo: **none** Additional: CoPa, VTK-m, CINEMA, HDF5.0



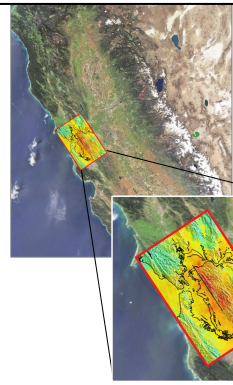
Project/PI	WarpX: Plasma Wakefield Accelerators Jean-Luc Vay
Challenge Problem	Wakefield plasma accelerator with a 1PW laser drive • 6.9×10^{12} grid cells • 14×10^{12} macroparticles • 1000 timesteps/1 stage
FOM Speedup	**500**
Nodes Used	**8576**
ST/CD Tools	Used in KPP Demo: **AMReX, libEnsemble**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

100X* your impact: Leveraging DOE/ECP investments

- Many communities still largely using home-grown solutions with room to improve. Opportunities:
  - Migrate from CPU to GPU – For scale out to larger problems, or scale in, to smaller GPU-enable systems (e.g., laptop)
  - Introduce modern software tools, workflows – leverage the outreach, training and culture focused on improvement
  - Integrate into larger software communities – E4S, xSDK, other software product communities
- How can we engage with these communities to realize the 100X improvement in science impact?
- DOE/ECP provides libraries, tools, expertise, and community connections that can be leveraged to realize 100X
- What are the best opportunities?
- What are the impediments?
- Can we produce strategic and tactical plans?
- Selling libraries and tools directly is hard to do …
- Selling 100X impact across DOE, other agencies and industry could be much easier

*100X (verb): *to increase (your scientific impact) by two orders of magnitude*

Opportunities to realize 100X

- 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

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 by running same problems 100X cheaper
- 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

Why are we here?

- To make 100X real
- To bring the people, communities, libraries, tools, methodologies from ECP into the future
- Many communities are here:
  - Application teams, Facilities, lab management, industry partners, US agencies, stakeholders
  - Library and tool developers, software community leaders
  - All: People who care about the impact of software on science
- To expand the community of engagement to include people who were not part of ECP – we need you
- To organize the DOE scientific software community for optimal impact after ECP and beyond
- Outcomes:
  - Better understanding of the challenges, opportunities, and paths to success – by everyone
  - Understand of next steps to pursue after leaving the workshop – by everyone
  - To create a workshop report that furthers our strategic and tactical plans for success
  - Ultimately:
    - What do we gain financially by establishing a sustainability effort: Goal is success that brings funding
    - What do we lose in terms of management oversight: Goal is only the things that better and more easily done in aggregate