A cast of thousands: How the IDEAS Productivity project has advanced software productivity and sustainability

David E. Bernholdt (he/him)
Oak Ridge National Laboratory

On behalf of the entire IDEAS-ECP team

a Best Practices for HPC Software Developers webinar

learn more about IDEAS at https://ideas-productivity.org
and https://doi.org/10.48550/arXiv.2311.02010

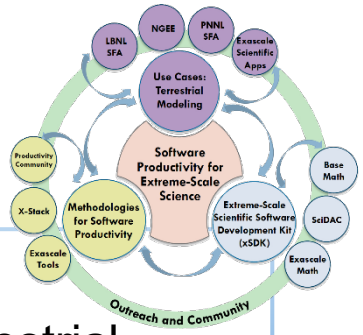
This work was supported by the U.S. Department of Energy Office of Science, Office of Advanced Scientific Computing Research (ASCR), and by the Exascale Computing Project (17-SC-20-SC), a collaborative effort of the U.S. Department of Energy Office of Science and the National Nuclear Security Administration

A Brief History of IDEAS

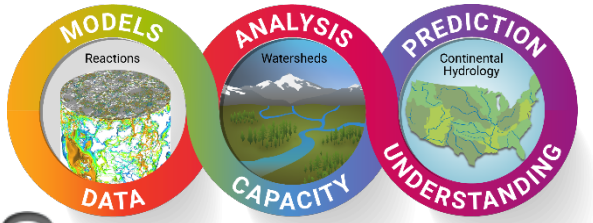
- IDEAS = Interoperable Design of Extreme-Scale Application Software
- First of its kind (in U.S.) with a focus on incubating, curating, and disseminating knowledge and methodologies about the sustainment of scientific software
  - Inspired by UK Software Sustainability Institute
- IDEAS is now a family of related projects
  - Different sponsors, different time frames, different people (but significant overlap), different approaches
  - **Common focus on improving developer productivity and software sustainability and trustworthiness**

Acronyms

DOE	= U.S. Department of Energy
ASCR	= Office of Advanced Scientific Computing Research
BER	= Office of Biological and Environmental Research
ECP	= Exascale Computing Project



- IDEAS-Classic (2014—2017)
  - Focus: multiscale multiphysics terrestrial ecosystem modeling
  - Sponsors: DOE/ASCR and BER
- IDEAS-ECP (2017—2023)
  - Focus: supporting the ecosystem of applications, libraries, and tools developed by ECP
  - Sponsor: DOE/ECP
- IDEAS-Watersheds (2019—present)
  - Focus: accelerating watershed science through a community driven software ecosystem
  - Sponsor: DOE/BER



IDEAS productivity

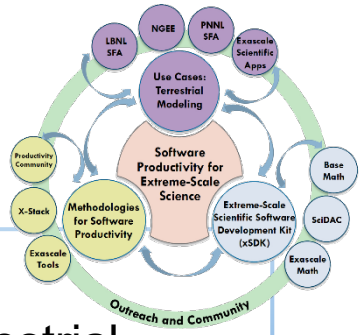
ECP EXASCALE COMPUTING PROJECT

A Brief History of IDEAS

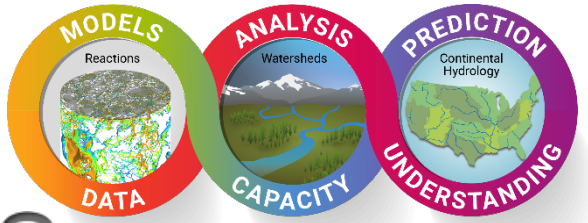
- IDEAS = Interoperable Design of Extreme-Scale Application Software
- First of its kind (in U.S.) with a focus on incubating, curating, and disseminating knowledge and methodologies about the sustainment of scientific software
  - Inspired by UK Software Sustainability Institute
- IDEAS is now a family of related projects
  - Different sponsors, different time frames, different people (but significant overlap), different approaches
  - **Common focus on improving developer productivity and software sustainability and trustworthiness**

Acronyms

DOE	= U.S. Department of Energy
ASCR	= Office of Advanced Scientific Computing Research
BER	= Office of Biological and Environmental Research
ECP	= Exascale Computing Project



- IDEAS-Classic (2014—2017)
  - Focus: multiscale multiphysics terrestrial ecosystem modeling
  - Sponsors: DOE/ASCR and BER
- IDEAS-ECP (2017—2023)
  - Focus: supporting the ecosystem of applications, libraries, and tools developed by ECP
  - Sponsor: DOE/ECP
- IDEAS-Watersheds (2019—present)
  - Focus: accelerating watershed science through a community driven software ecosystem
  - Sponsor: DOE/BER



IDEAS productivity

ECP EXASCALE COMPUTING PROJECT

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

80+ R&D teams, 1000 researchers!

Performant mission and science applications at scale

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

Application Development (AD)

Develop and enhance the predictive capability of applications critical to DOE

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

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


Andrew Siegel, AD Director
Erik Draeger, AD Deputy Director

Software Technology (ST)

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

70 unique software products developed by 35 teams spanning programming models and runtimes, math libraries, data and visualization, development tools


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

Hardware and Integration (HI)

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

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


Richard Gerber, HI Director
Susan Coghlan, HI Deputy Director

Science and beyond: Applications and discovery in ECP

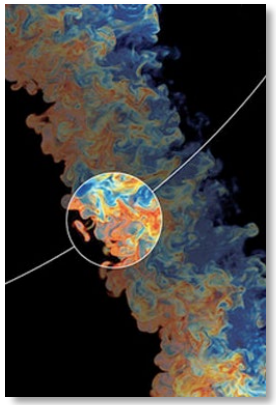
24 applications
+ 6 co-design projects
62 separate codes!

National security

Next-generation, **stockpile stewardship** codes

Reentry-vehicle-environment simulation

Multi-physics science simulations **of high-energy density physics** conditions



Energy security

Turbine **wind plant** efficiency

Design and commercialization of **SMR**s

Nuclear fission and fusion reactor **materials design**

Subsurface use for **carbon capture**, petroleum extraction, waste disposal

High-efficiency, low-emission **combustion engine** and gas turbine design

Scale up of **clean fossil fuel** combustion

Biofuel catalyst design

Economic security

Additive manufacturing of qualifiable metal parts

Reliable and efficient planning of the **power grid**

Seismic hazard risk assessment

Scientific discovery

Cosmological probe of the standard model of particle physics

Validate fundamental laws of nature

Plasma wakefield accelerator design

Light source-enabled **analysis of protein and molecular structure** and design

Find, predict, and control materials and properties

Predict and control **magnetically confined fusion plasmas**

Demystify **origin of chemical elements**

Earth systems

Accurate regional impact assessments in **Earth system models**

Stress-resistant crop analysis and catalytic conversion of **biomass-derived alcohols**

Metagenomics for analysis of biogeochemical cycles, climate change, environmental remediation

Health care

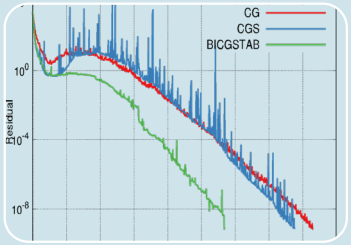
Accelerate and translate **cancer research** (partnership with NIH)

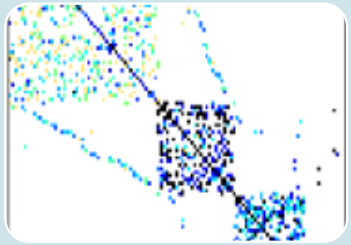
Thank you to Andrew Siegel, Erik Draeger and ECP applications teams

ECP ST has six technical areas

35 projects
70 software products

ECP ST Director: Mike Heroux
ECP ST Deputy Director: L.C. McInnes





E4S

NNSA
National Nuclear Security Administration

Programming Models & Runtimes

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

Area Leads: Rajeev Thakur

Development Tools

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

Jeff Vetter

Math Libraries

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

Sherry Li

Data and Visualization

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

Jim Ahrens

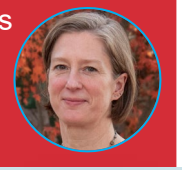
Software Ecosystem

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

Todd Munson

NNSA ST

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

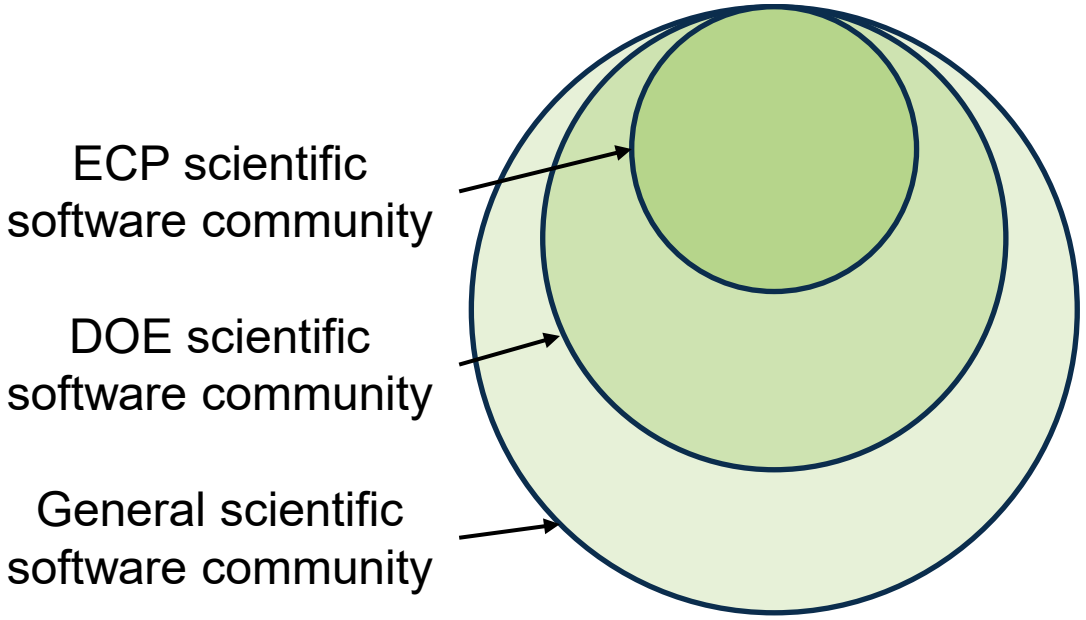


Kathryn Mohror

IDEAS-ECP Had to Be Different

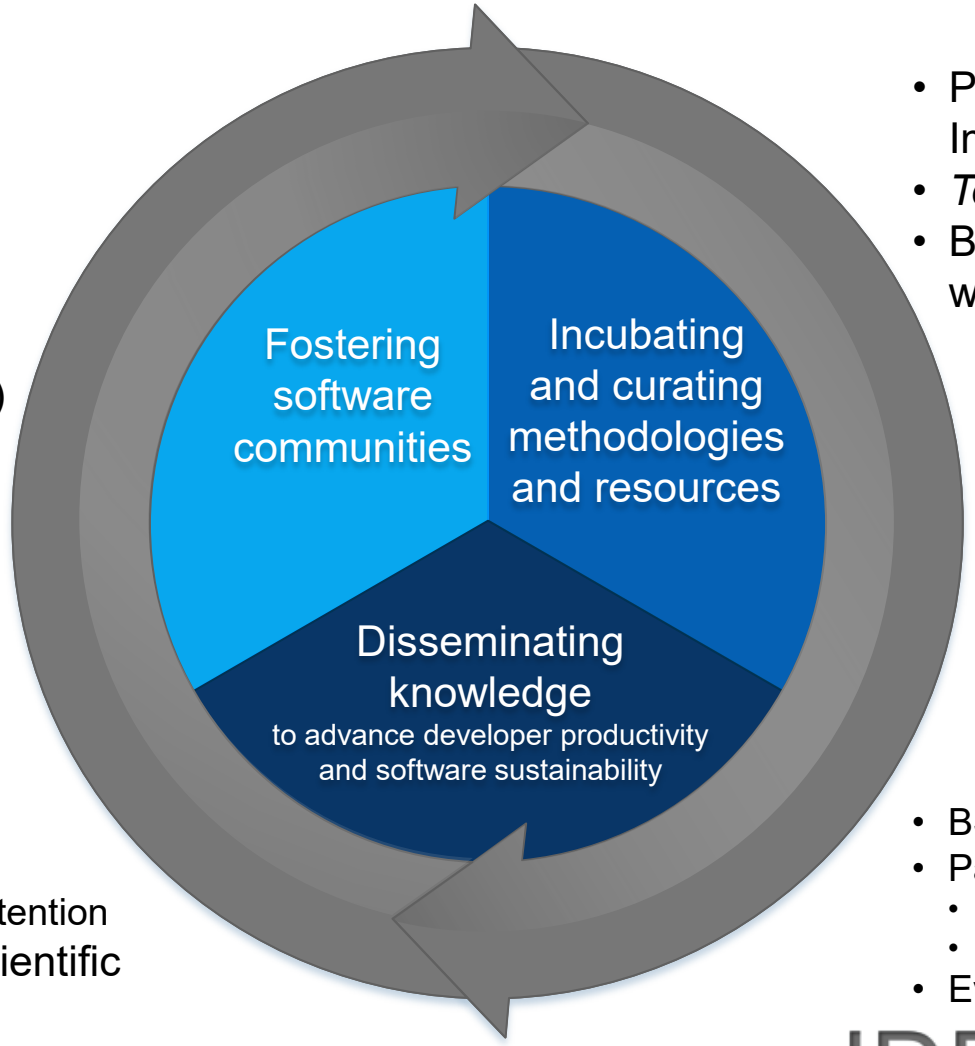
- The ECP environment and needs were very different from IDEAS-Classic
  - Stringent deliverable set for performance and capability
  - Requirement to utilize new exascale hardware platforms
    - Actual exascale hardware available only late in the project
    - Strong likelihood of encountering bugs or inadequacies in developing software environment
    - Extensive exploration and experimentation with code
  - Impossible to do 1:1 interactions with every ECP software team
    - 80+ software teams, 100+ software products, ~1000 people
- So, we had to develop a different approach for IDEAS-ECP, focusing on…
  - Fostering software communities,
  - Broadly applicable methodologies and resources, and
  - Disseminating knowledge broadly

Scientific Software Ecosystems of Interest



Diffusion of Innovations via Teams, Teams of Teams, and Communities

- Accelerating design space exploration
- Advancing quality, usability and interoperability, while respecting team autonomy
- Software community policies
- Software Development Kits (SDKs) and E4S
  - xSDK, CAT-SDK, DAV-SDK, SWAS, etc



- Productivity and Sustainability Improvement Planning (PSIP)
- *Team of teams* concepts
- Better Scientific Software (BSSw.io) website

- Webinar Series:
  - HPC Best Practices
  - HPC Workforce Development and Retention
- Tutorials on Practices for Better Scientific Software

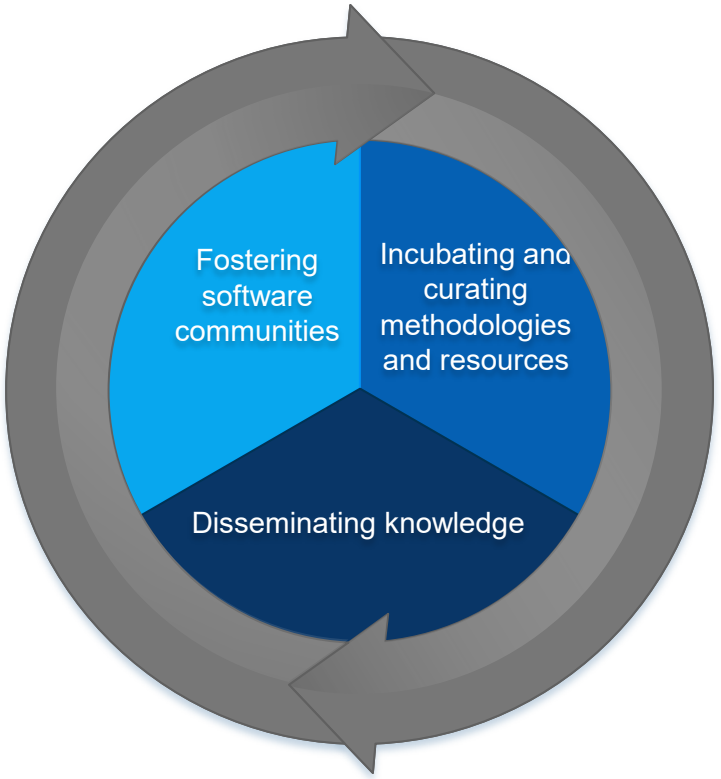
- BSSw Fellowship Program
- Panel Series:
  - Strategies for Working Remotely
  - Performance Portability
- Events: BOFs, workshops, and more

IDEAS productivity

ECP EXASCALE COMPUTING PROJECT

Fostering Software Communities

- Accelerating design space exploration
- Advancing quality, usability, interoperability, and sustainability, while respecting team autonomy
- Software community policies
- Software Development Kits (SDKs) and E4S
  - xSDK, CAT-SDK, DAV-SDK, etc.



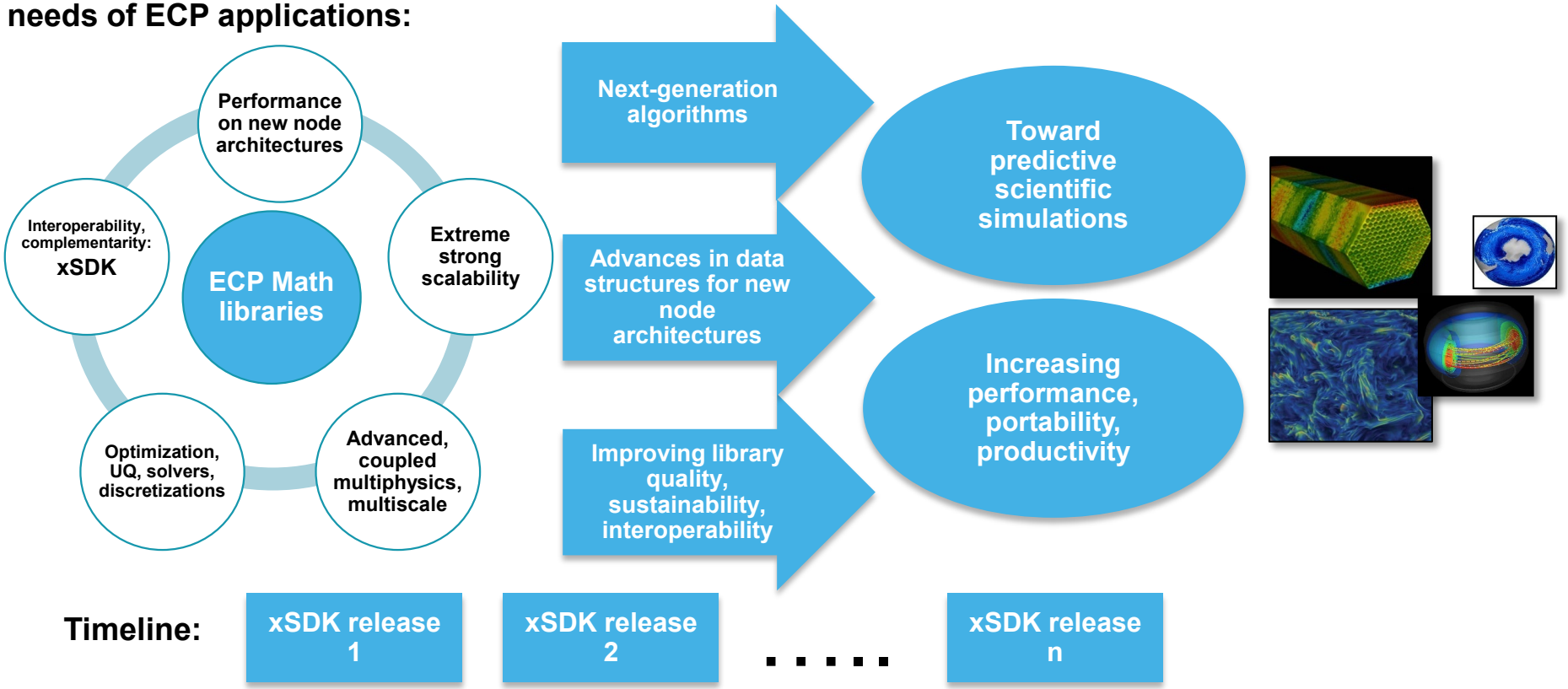
xSDK: Primary delivery mechanism for ECP math libraries' continual advancements


xSDK lead: Ulrike Meier Yang (LLNL)
xSDK release lead: Satish Balay (ANL)

xSDK release 0.8.0 (Nov 2022)

- hypre, PETSc/TAO, SuperLU, Trilinos — the initial xSDK 0.1.0 libraries
- Alquimia, PFLOTRAN — domain libraries included in xSDK 0.1.0 or 0.2.0
- AMReX, ArborX, ButterflyPACK, DTK, Ginkgo, heFFTe, libEnsemble, MAGMA, MFEM, Omega_h, PLASMA, PUMI, SLATE, Tasmanian, SUNDIALS, Strumpack
- deal.II, preCICE, PHIST, SLEPc — from the broader community

As motivated and validated by the needs of ECP applications:



Refs: xSDK: Building an Ecosystem of Highly Efficient Math Libraries for Exascale, **SIAM News**, Jan 2021; Building Community through xSDK Software Policies, HPC-BP webinar, Dec 2019

xSDK Community Policies (v1.0.0, Feb 2023)
https://doi.org/10.6084/m9.figshare.13087196.v1

Mandatory Policies

1. Support portable installation through Spack
2. Provide a comprehensive test suite
3. Packages using MPI must not directly use MPI_COMM_WORLD
4. Package must support common platforms
5. Must provide a reliable way to contact developers
6. Respect decisions by other previously called packages about system resources (e.g., exception and signal handlers)
7. Permissive open-source licence
8. Runtime API call to return library version
9. Must not pollute namespaces
10. Must have a public repository
11. Must be able to turn off/off and redirect I/O
12. If a package embeds externally developed, it must be buildable against an outside copy of that software
13. Installations must conform to certain conventions for names and locations of header and library files
14. Must be buildable with 64-bit pointers
15. xSDK compatibility changes go into the regular development/release versions, not special branches
16. Build options must include a “debug” mode
17. Packages must provide sufficient documentation

Recommended Policies

1. Should have at least one “smoke” test
2. Should be possible to run test suite under Valgrind
3. Adopt and document a consistent system for handling error conditions
4. Free all resources acquired
5. Ability to export an ordered list of library dependencies
6. Document the versions of packages it works with or depends upon
7. Include README, SUPPORT, LICENSE, and CHANELOG files
8. Provide preprocessor macros to allow for version comparisons

IDEAS productivity

ECP 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: E4S 23.08 – 115 packages

Spack

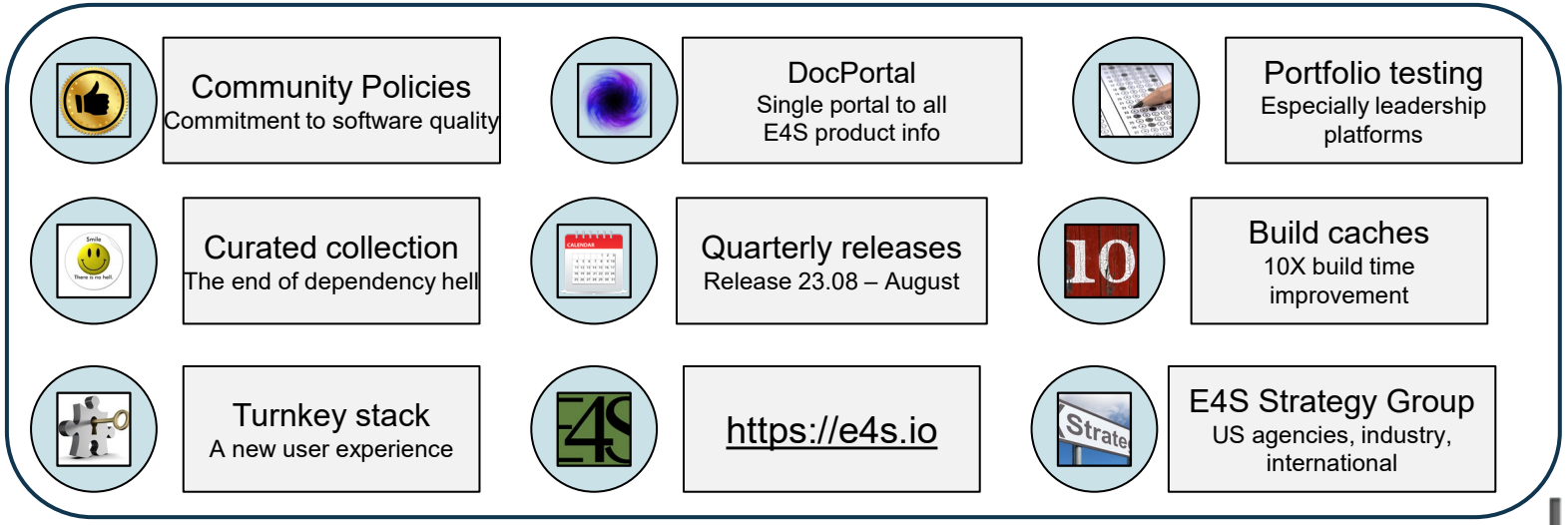
https://spack.io

Spack lead: Todd Gamblin (LLNL)

E4S

https://e4s.io

E4S lead: Sameer Shende (U Oregon)

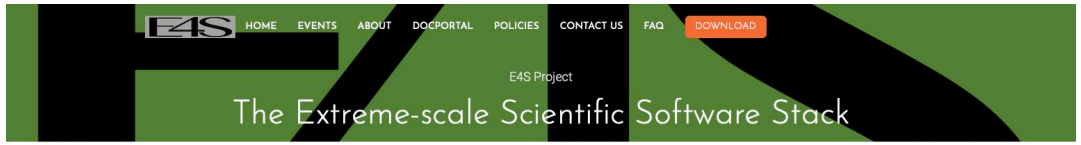


Also includes other products, e.g.,
AI: PyTorch, TensorFlow, Horovod
Co-Design: AMReX, Cabana, MFEM

IDEAS productivity

ECP EXASCALE COMPUTING PROJECT

E4S Community Policies: *A commitment to quality improvement*



- Purpose: Enhance sustainability and interoperability
- Will serve as membership criteria for E4S
  - Membership is not required for *inclusion* in E4S
  - Also includes forward-looking draft policies
- Modeled after xSDK community policies
- Multi-year effort led by SDK team
  - Included representation from across ST
  - Multiple rounds of feedback incorporated from ST leadership and membership

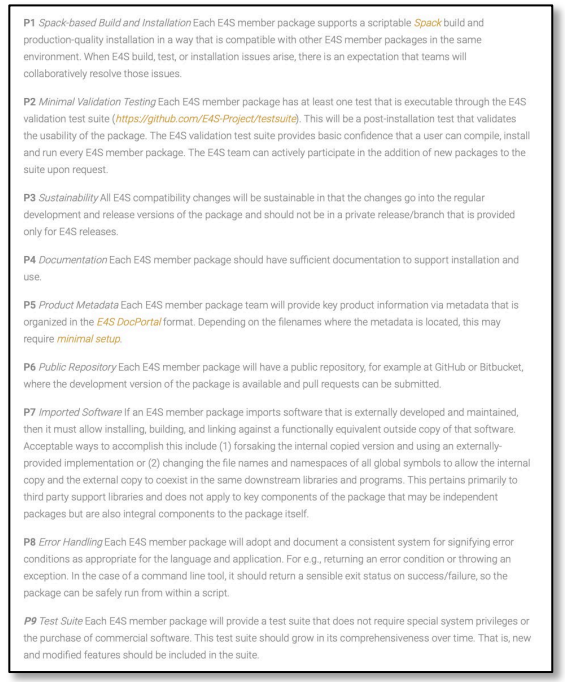
SDK lead: Jim Willenbring (SNL)

Policies: Version 1

Feedback welcome. What policies make sense for your software?

https://e4s-project.github.io/policies.html

- **P1: *Spack-based Build and Installation***
- **P2: *Minimal Validation Testing***
- **P3: *Sustainability***
- **P4: *Documentation***
- **P5: *Product Metadata***
- **P6: *Public Repository***
- **P7: *Imported Software***
- **P8: *Error Handling***
- **P9: *Test Suite***



P1 *Spack-based Build and Installation* Each E4S member package supports a scriptable *Spack* build and production-quality installation in a way that is compatible with other E4S member packages in the same environment. When E4S build, test, or installation issues arise, there is an expectation that teams will collaboratively resolve those issues.

P2 *Minimal Validation Testing* Each E4S member package has at least one test that is executable through the E4S validation test suite (*https://github.com/E4S-Project/testsuite*). This will be a post-installation test that validates the usability of the package. The E4S validation test suite provides basic confidence that a user can compile, install and run every E4S member package. The E4S team can actively participate in the addition of new packages to the suite upon request.

P3 *Sustainability* All E4S compatibility changes will be sustainable in that the changes go into the regular development and release versions of the package and should not be in a private release/branch that is provided only for E4S releases.

P4 *Documentation* Each E4S member package should have sufficient documentation to support installation and use.

P5 *Product Metadata* Each E4S member package team will provide key product information via metadata that is organized in the *E4S DocPortal* format. Depending on the filenames where the metadata is located, this may require *minimal setup*.

P6 *Public Repository* Each E4S member package will have a public repository, for example at GitHub or Bitbucket, where the development version of the package is available and pull requests can be submitted.

P7 *Imported Software* If an E4S member package imports software that is externally developed and maintained, then it must allow installing, building, and linking against a functionally equivalent outside copy of that software. Acceptable ways to accomplish this include (1) forsaking the internal copied version and using an externally-provided implementation or (2) changing the file names and namespaces of all global symbols to allow the internal copy and the external copy to coexist in the same downstream libraries and programs. This pertains primarily to third party support libraries and does not apply to key components of the package that may be independent packages but are also integral components to the package itself.

P8 *Error Handling* Each E4S member package will adopt and document a consistent system for signifying error conditions as appropriate for the language and application. For e.g., returning an error condition or throwing an exception. In the case of a command line tool, it should return a sensible exit status on success/failure, so the package can be safely run from within a script.

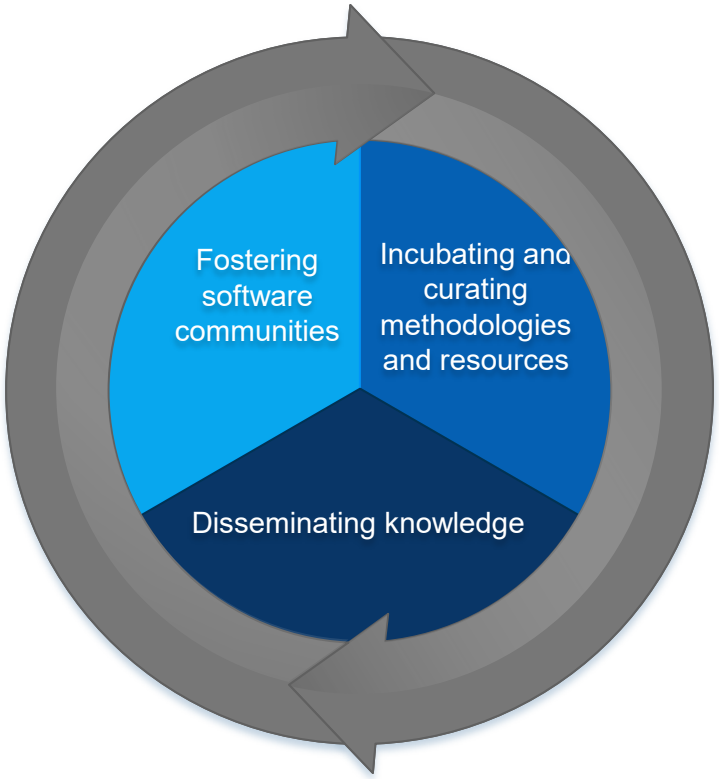
P9 *Test Suite* Each E4S member package will provide a test suite that does not require special system privileges or the purchase of commercial software. This test suite should grow in its comprehensiveness over time. That is, new and modified features should be included in the suite.

IDEAS productivity

ECP EXASCALE COMPUTING PROJECT

Incubating and Curating Methodologies and Resources

- Productivity and Sustainability Improvement Planning (PSIP)
- Team of teams concepts
- Better Scientific Software (BSSw.io) resource portal



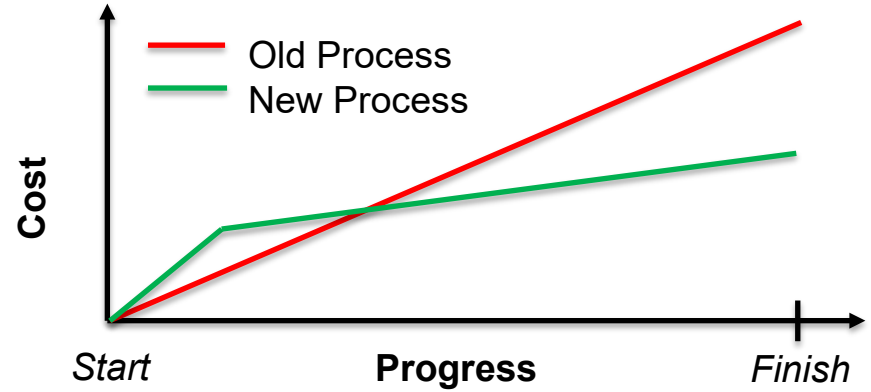
PSIP: Productivity and Sustainability Improvement Planning
Continual, Incremental Software Process Improvement

https://bssw.io/psip

1. Identify your team's "pain points" in your software development processes
   - Help: RateYourProject assessment tool: https://rateyourproject.org/
2. Set a goal for something to improve
   - Target processes and behaviors, not just tasks
   - Pick something that you can address in a few months that will give you a noticeable benefit
3. Agree on a plan to address it, identify markers of progress and what is "done"
   - Write them down
   - Help: Progress tracking card examples: https://bssw-psip.github.io/ptc-catalog/catalog
4. Work your plan, track your progress
5. When you are done, celebrate…

…then pick a new pain point to address

Lead: Elaine Raybourn (SNL)



The new process costs something to implement, but it pays off over time

Target: your project should include "just enough" software engineering so that you can meet your short-term and longer-term scientific goals effectively

IDEAS productivity

ECP EXASCALE COMPUTING PROJECT

PSIP multi-pronged, socio-technical strategy

EDUCATION
Share knowledge via tutorials, paper publications, and seminars to engender a culture of productivity by engaging PSIP liaisons, early adopters, and champions

TEAM ENGAGEMENT
By directly working with ECP teams, PSIP facilitators help software teams to **IDENTIFY** opportunities to iteratively and incrementally **IMPROVE** software team practices and processes.

RESEARCH
Conduct Team of Teams data-driven research (data mining + interviews) to characterize ECP organization, draw general conclusions about observable behaviors that contribute to team productivity, collaboration, and success

METHOD & TOOL DEVELOPMENT
Develop tools that automate PSIP to support ECP teams, scientific software quality, and developer productivity. Development of PSIP tools based on needs and salient issues identified during Team of Teams research and ECP team engagement

PSIP: Collaborative and **team-oriented.**
Measurable and **specific.** **Realistic increments.**

IDEAS productivity

ECP EXASCALE COMPUTING PROJECT

Examples of Who's using PSIP

The HDF Group

Improvements to documentation, setting code style standards, transition to GitHub (blog article)

> "The PSIP project had an immediate impact on our community. With the GitHub move we see increasing amounts of small but very valuable contributions to make HDF5 code and documentation better." **– Elena Pourmal, Director of Engineering, The HDF Group**

FLASH5

Improve testing and verification, transition development workflow to GitHub

Revamp build system, implement a CTest-based testing framework, implement a basic CI pipeline

EXAALT

ALPINE/ZFP

Created a VTK-m filter for APLINE in situ algorithm users

OAK RIDGE National Laboratory

Using a more detailed version for internal project assessment

Completed PSIP tutorial, investigating how it can be used in academic context

TECHNISCHE UNIVERSITÄT DARMSTADT

Sandia National Laboratories

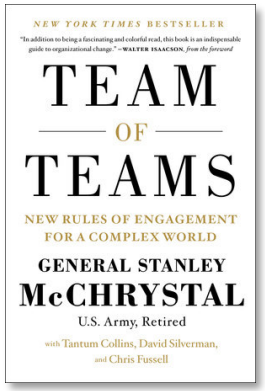
Using internally for reproducibility LDRD research, and for large projects updating version control systems, and updating documentation to support better onboarding

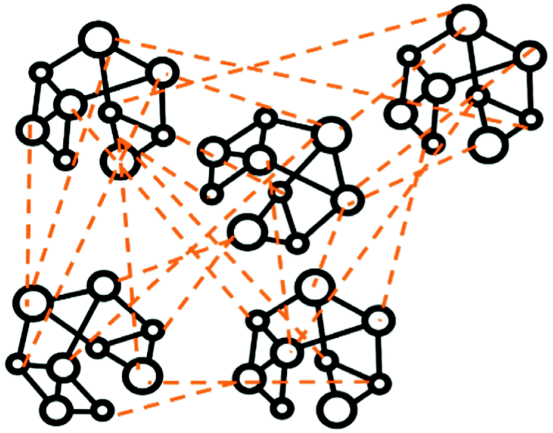
IDEAS productivity

ECP EXASCALE COMPUTING PROJECT

Collaboration via Teams of Teams

- The “team of teams” concept (ToT) was popularized by Stanley McChrystal’s 2015 book
  - IDEAS efforts are an offshoot of PSIP, led by Elaine Raybourn (SNL)
  - Using tools from the CAT-SDK software community for repository analysis
- ToT provides a powerful lens through which to better understand the ECP, as well as many other software ecosystems, and to improve their effectiveness
  - Strengthen community partnerships
  - Scaling productivity typically experienced in small teams (where it's easy), to larger groups via the team of teams paradigm
- ToT principles facilitated contributions of the HDF5 team to the E4S and Data & Viz SDK
  - Supported applications in modeling earthquakes, electronic structures, subsurface flow, reacting flow, stellar explosions, wind plants, and cosmology
- Distributed, Interconnected Teams through the Lens of Team of Teams Principles
  - Panel discussion with members of PETSc, Trilinos, xSDK, and E4S ECP projects
- Scaling productivity and innovation on the path to exascale with a “team of teams” approach
  - Case study of the ASC Ristra ECP project





Schematic illustration of a team of teams, from doi:10.1007/978-3-030-22338-0_33

better scientific software

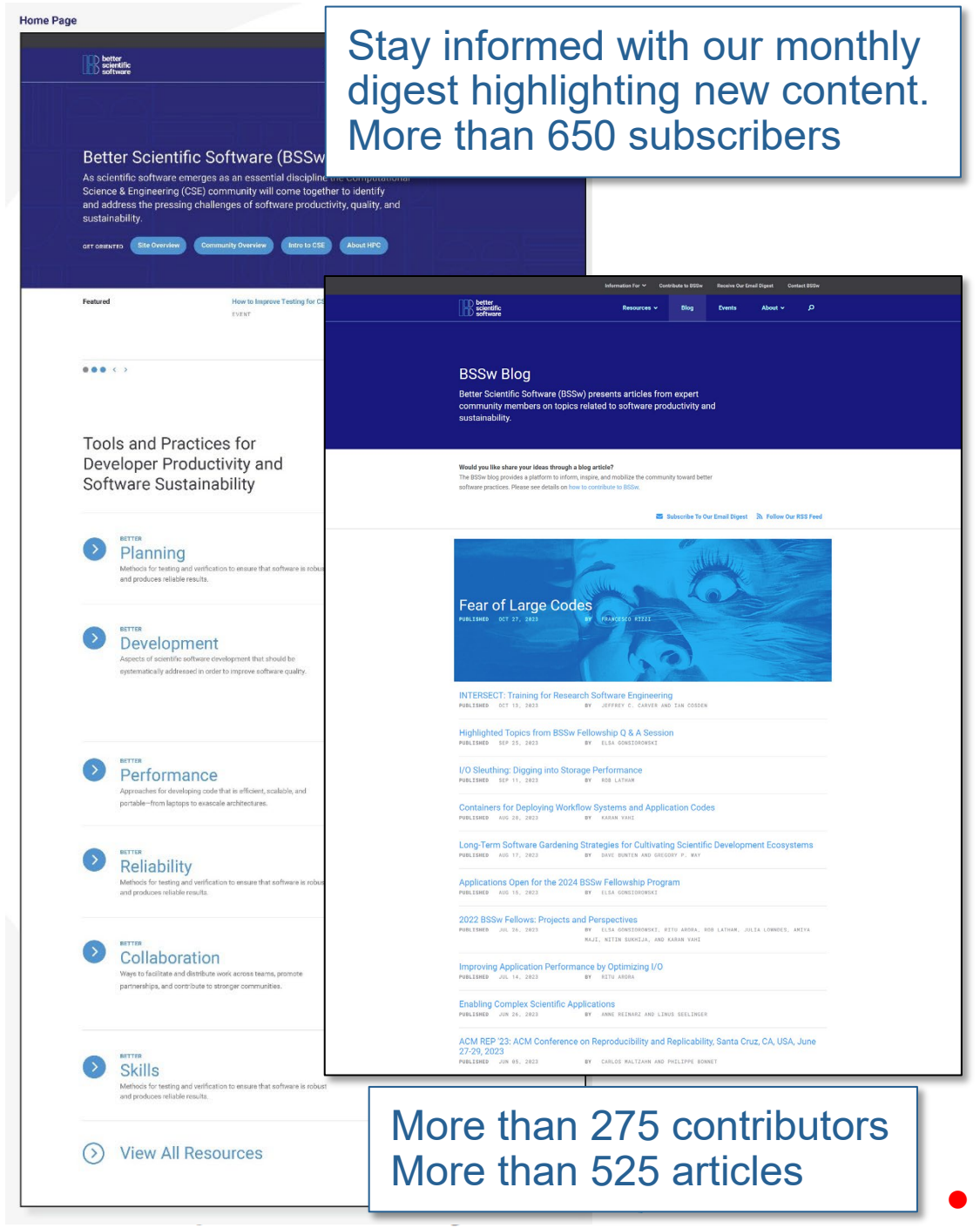
https://bssw.io

A central hub for sharing information on practices, techniques, experiences, and tools to improve developer productivity and software sustainability for computational science & engineering (CSE)

- **Find information** on scientific software topics
- **Contribute new resources** based on your experiences
- Editor-in-chief: Rinku Gupta (ANL)

Types of content on BSSw

- **Blog articles**: success stories, perspectives, opportunities, technical deep-dives, and more
- **Curated content:** short pointers to useful material already hosted elsewhere
- **Events:** increase awareness of events related to better scientific software

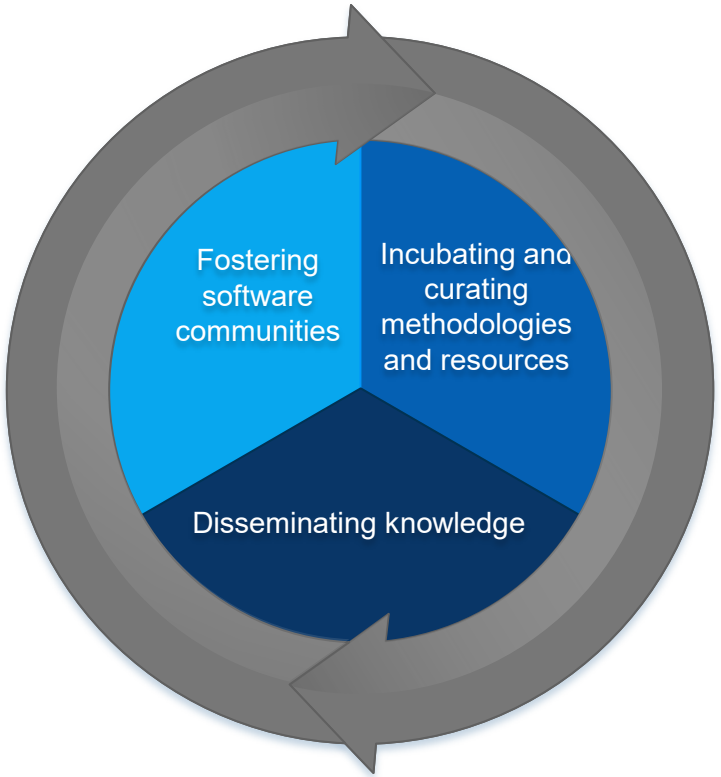


Stay informed with our monthly digest highlighting new content.
More than 650 subscribers

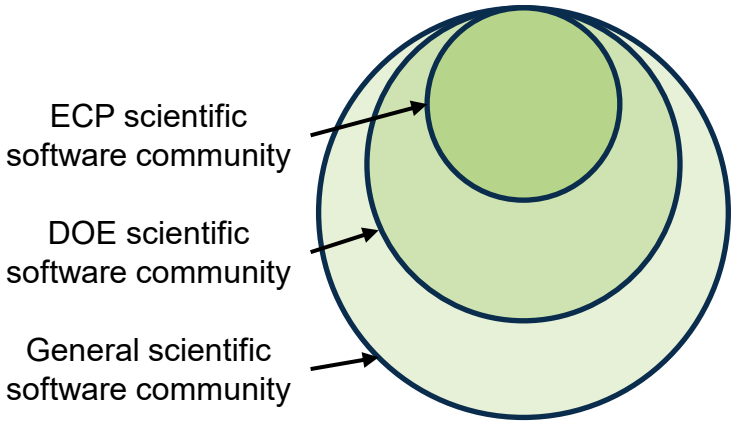
More than 275 contributors
More than 525 articles

Disseminating Knowledge

- BSSw Fellowship Program
- Better Scientific Software tutorials
- Best Practices for HPC Software Developers webinar series
- Panel Series:
  - Strategies for Working Remotely
  - Performance Portability
- Events: BOFs, workshops, and more



Scientific Software Ecosystems of Interest



IDEAS dissemination and outreach activities have always targeted the general scientific software community

IDEAS productivity

ECP EXASCALE COMPUTING PROJECT

Better Scientific Software (BSSw) Fellowship Program

better scientific software

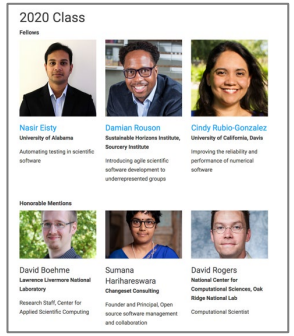
Meet Our Fellows

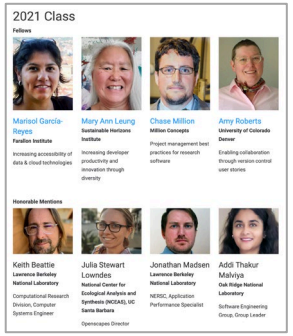
The BSSw Fellowship program gives recognition and funding to leaders and advocates of high-quality scientific software. Meet the Fellows and Honorable Mentions and learn more about how they impact Better Scientific Software.

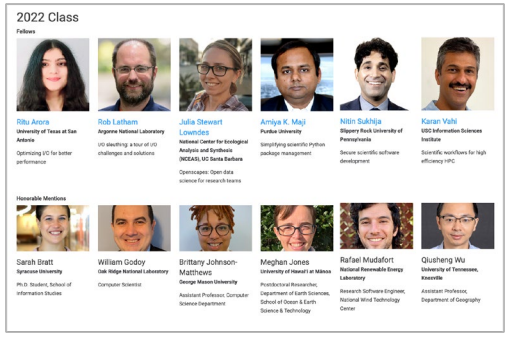
Fellowships Overview | Apply | Meet Our Fellows | BSSw Fellowship FAQ

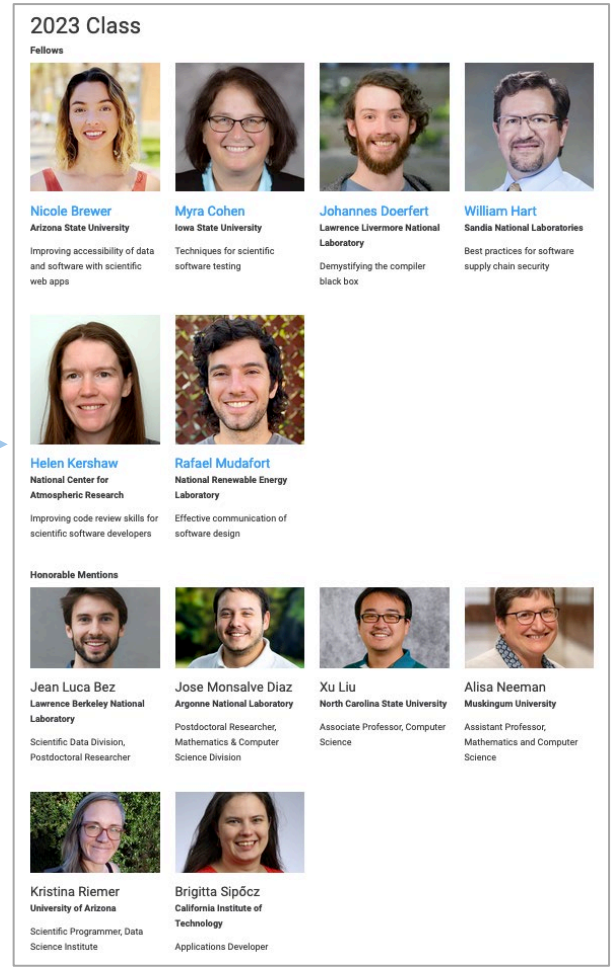
Recognizing leaders

2018 - 2023









Goal: Foster and promote practices, processes, and tools to improve developer productivity and software sustainability of scientific codes. #somycodewillseethefuture

BSSw Fellowship Coordinator: Elsa Gonsiorowski (LLNL)

Deputy Coordinator, Community Building: Erik Palmer (LBNL)

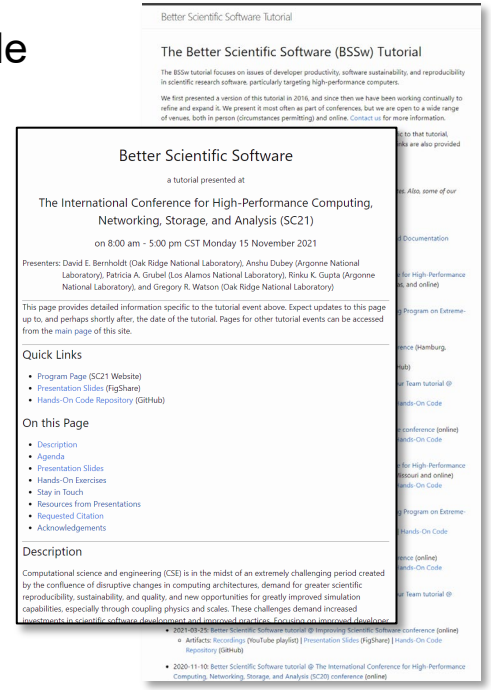
Also supported by the National Science Foundation since 2021

IDEAS productivity

ECP EXASCALE COMPUTING PROJECT

Better Scientific Software Tutorials

- Covering issues of developer productivity, software sustainability and reliability, with a special focus on the challenges of complex, large-scale HPC
  - software design, agile methodologies, Git workflows, reproducibility, software testing, continuous integration testing, refactoring, and more
- https://bssw-tutorial.github.io
- Lead: David Bernholdt (ORNL)
- 32 tutorials since 2016
  - Presentations (all) and recordings (some) available
- Topics and content under continuous refinement
- Frequent venues
  - Supercomputing (2016-2023)
  - ATPESC (2016-2023)
  - ISC (2018-2019, 2021-2023)



Webinar Series: Best Practices for HPC Software Developers (HPC-BP)

- Covering topics in software development and HPC
- https://ideas-productivity.org/resources/series/hpc-best-practices-webinars/
- Lead: Osni Marques (LBNL)
- Presented by the community to the community
- Monthly series, since May 2016 (offered live and archived)
  - To date: 80 webinars, >12,000 registrations, >5,300 attendees
  - 84 attendees per webinar, on average

Panel Series: Performance Portability & ECP

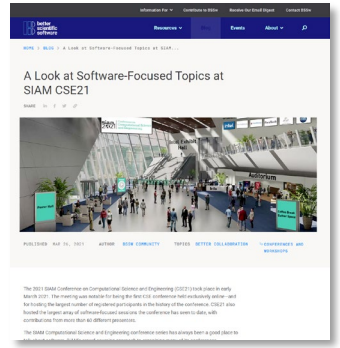
- Lead: Anshu Dubey (ANL). Refs:
  - Performance Portability in the Exascale Computing Project: Exploration Through a Panel Series, A. Dubey et al, IEEE CiSE, Sept 2021
  - SIAM CSE21 minisymposium: https://doi.org/10.6084/m9.figshare.c.5321441
  - ECCOMAS 2022 minisymposium

Panel Series: Strategies for Working Remotely

- Exploring strategies for working remotely, with emphasis on how HPC teams can be effective and efficient in long-term hybrid settings
- https://www.exascaleproject.org/strategies-for-working-remotely
- Lead: Elaine Raybourn (SNL)
- Quarterly series, since April 2020 (offered live and archived)
- Ref: Why We Need Strategies for Working Remotely: The ECP Panel Series, E. Raybourn, SC20 State of the Practice, Nov 2020

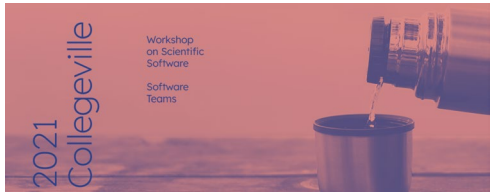
Technical Meetings and Birds of a Feather Sessions

- Creating opportunities to talk about software development, productivity, and sustainability
- https://ideas-productivity.org/resources/series/technical-sessions-and-meetings/
- Minisymposia
  - SIAM CSE, SIAM PP (2015-2023), PASC (2018, 2019)
  - Ref: A Look at Software-Focused Topics at SIAM CSE21, March 2021
- Thematic poster sessions
  - SIAM CSE (2017, 2019, 2021)
- BOF sessions
  - Software Engineering and Reuse in Modeling, Simulation and Data Analytics for Science and Engineering
    - http://bit.ly/swe-cse-bof
    - Supercomputing (2015-2023), ISC (2019, 2022-2023)



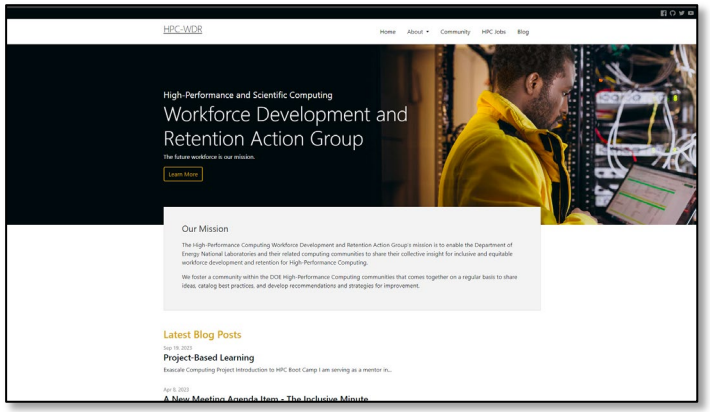
- Collegeville Workshop Series on Scientific Software,
  - Ref: Software Team Experiences and Challenges, K. Beattie et al, Oct 2021

IDEAS productivity



Promoting Culture Change

- In all of our work, IDEAS promotes a change in the culture around scientific software
  - Recognizing and valuing the software professionals who are increasingly contributing to the development and maintenance of research software
  - Recognizing and valuing the software itself as a first-class product of compute-intensive research
- Engagement with sponsors
- Support for the developing research software engineer (RSE) community
  - Many IDEAS team members are also members of, and leaders in, US-RSE
- HPC Workforce Development and Retention Action Group
  - Part of the ECP Broadening Participation Initiative to expand the pipeline and workforce for DOE high-performance computing
  - HPC-WDR website and webinar series
    - Topics have included ally skills, diversifying computing, mentoring, and normalizing inclusion by embracing difference



Lead: Suzanne Parete-Koon (ORNL)

IDEAS-ECP Impacts

- Feedback underscores IDEAS's role in **enhancing software quality, promoting best practices, and expanding awareness** of the importance of software development
- Curating best practices for software development and team productivity has **empowered teams** to build new practices into their workflows and increase cross-project collaboration
  - Many community members express a desire for additional resources
- Software communities have proven to be a **source of inspiration** for building shared foundations for software ecosystems while respecting team autonomy
- IDEAS outreach mechanisms have enabled innovators in scientific software practices to **share knowledge** with the community
- *Model for other multi-institutional software ecosystems*

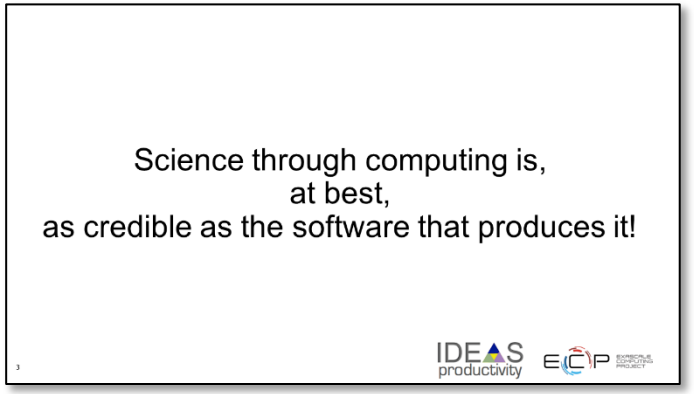
Moving Forward

doi:10.1109/MCSE.2023.3260475

doi:10.2172/1846009

We believe that IDEAS has been an important catalyst for the scientific software community in ECP and beyond, but two further elements are essential for continued qualitative growth...

- Increasing focus on **research software science**
  - Applying our experience with hypothesis-driven science to improve our understanding of how scientific software is developed and used
  - Social-, cognitive-, hard-science, and engineering viewpoints needed
- Changing the prevailing attitude that scientific software productivity, sustainability, and trustworthiness are not just “nice to have” but **“must-have”**
  - Innovators and early adopters are there
  - But many still prioritize more scientific results at the expense of beneficial investments in the software behind them



A key message of the BSSw tutorials

What's Next?

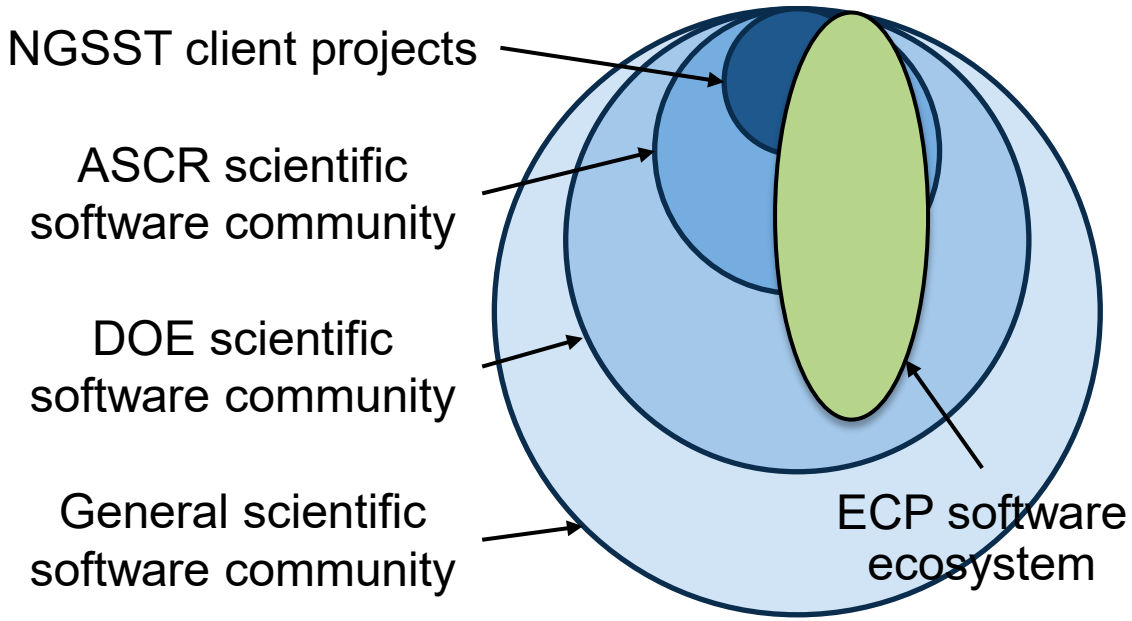
- As part of the ECP project, IDEAS-ECP ends at the end of 2023
  - IDEAS-Watersheds continues, with focus on watershed modeling
- **BSSw Fellowship** class of 2024 to be announced soon
  - Thanks to support from DOE/ASCR, DOE/NNSA, and NSF
- **Next Generation Scientific Software Technologies (NGSST)** initiative is emerging to help address scientific software stewardship needs across the ASCR software ecosystem
  - Focus is different than IDEAS-ECP
  - But you can see IDEAS influence in some of the planned work
- Looking for increased engagement from the community to continue some IDEAS legacy activities

Acronyms
NNSA = National Nuclear Security Administration
NSF = National Science Foundation

Next Generation Scientific Software Technologies (NGSST)

- Seven teams submitted interlocking proposals for a **Scientific Software Stewardship Consortium (S3C)** with distinct but complementary foci
- Focus on the ASCR scientific software ecosystem, which includes software products from
  - Exascale Computing Project (ECP) Software Technologies
  - ECP Co-Design Centers
  - Other ASCR-supported products
- At anticipated funding levels, the NGSST will be able to support only a subset of the eligible projects
- Phase 2 proposals under review, most awards expected to start in January 2024

Scientific Software Ecosystems of Interest



Continuing the IDEAS Legacy in NGSST

- Fostering software communities
  - Extreme-Scale Scientific Software Stack (E4S)
  - Some of the other community-focused NGSST organizations may pursue SDKs
- Incubating and Curating Methodologies and Resources
  - Software quality assurance (emphasis on ecosystem integration)
  - And possibly in other areas
  - BSSw.io resource portal
- Disseminating knowledge
  - Best Practices for HPC Software Developers webinar series
  - Better Scientific Software tutorials
    - Complemented by trainings from other NGSST organizations
- Promoting culture change
  - HPC Workforce Development and Retention
  - Building a community of practice for research software engineers in the DOE national labs

Items in green are specific IDEAS activities that we plan to continue in NGSST – with additional help from the larger community!

IDEAS productivity

ECP EXASCALE COMPUTING PROJECT

You Can Help Change the Culture around Scientific Software!

- IDEAS (or NGSST) can't do it alone!
- If you're a developer or user of scientific software (or a manager), **you have a role to play in making scientific software better**
  - Be **thoughtful about the stewardship** of your own software
  - Work with your team to **learn about and implement better software development practices**
    - Focus on incremental, but continual software process improvement, tailored to your needs
- **Share** your software development knowledge, experience, and resources with others
- **Engage with communities** relevant to your work and interests (see also doi:10.1109/MCSE.2018.2883051
- **Talk with your sponsors** about the importance of software stewardship, what you're doing, the benefits, and how they can help
- Remember: **Science through computing is, at best, as credible as the software that produces it!**

Acknowledgements: The IDEAS-ECP Team and the ECP


Ann Almgren (LBNL)


Ross Bartlett (SNL)


David E. Bernholdt (ORNL)
Institutional PI


Kita Cranfill (ORNL)


Anshu Dubey (ANL)
Institutional PI


Don Frederick (LLNL)


William Godoy (ORNL)


Elsa Gonsiorowski (LLNL)
Institutional PI


Patricia Grubel (LANL)


Rinku Gupta (ANL)


Rebecca Hartman-Baker (LBNL)
Computing Facility Liaison


Michael Heroux (SNL)
Lead Co-PI


Axel Huebl (LBNL)


Megan Phinney (LANL)
Computing Facility Liaison


Rose Lynch (ANL)


Addi Malviya Thakur (ORNL)


Osni Marques (LBNL)
Institutional PI


Lois Curfman McInnes (ANL)
Lead Co-PI


Reed Milewicz (SNL)


Mark Miller (LLNL)


J. David Moulton (LANL)
Institutional PI


Miranda Mundt (SNL)


Hai Ah Nam (LBNL)


Boyana Norris (U. Oregon)
Institutional PI


Erik Palmer (LBNL)


Suzanne Parete-Koon (ORNL)
ECP Workforce Development and Retention Action Group Lead


Elaine Raybourn (SNL)
Institutional PI


Katherine Riley (ANL)
Computing Facility Liaison


David Rogers (ORNL)
Computing Facility Liaison


Ben Sims (LANL)


Deborah Stevens (ANL)


Greg Watson (ORNL)


Jim Willenbring (SNL)

Alumni: Satish Balay (ANL), Lisa Childers (ANL), Todd Gamblin (LLNL), Judy Hill (ORNL), Steve Hudson (ANL), Christoph Junghans (LANL), Alicia Klinvex (SNL), Shannon Lindgren (ANL), Lena Lopatina (LANL), Jared O'Neal (ANL), Michele Rosso (LBNL), Jean Shuler (LLNL), Barry Smith (ANL), Louis Vernon (LANL), Paul Wolfenbarger (SNL)