



# Enabling a Culture of Developer Productivity and Software Sustainability

Elaine M. Raybourn  
Sandia National Laboratories  
[emraybo@sandia.gov](mailto:emraybo@sandia.gov), @elaineraybourn

SIAM CSE March 1, 2019

[exascaleproject.org](http://exascaleproject.org)   [ideas-productivity.org](http://ideas-productivity.org)



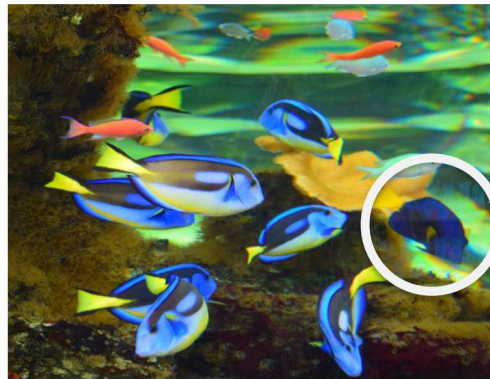
U.S. DEPARTMENT OF  
**ENERGY**

Office of  
Science



## *I help organizations tell their stories.*

- I am a social scientist who deepens understanding by being embedded in different cultures.



- Cultures: DARPA, DoD (Army, SOF, USMC, OSD, Team Orlando), BT Research, FhG FIT, INRIA, DOE Office of Science, Academia, National Labs
- Research: innovation and productivity, immersive learning environments, design of transmedia learning ecosystems, cultural awareness
- Focus on ECP productivity since 2017, transmedia learning since 2010, games, immersive virtual environments, social simulations, and intelligent community systems since 2000
- Passion: ***Seize opportunities that allow us to learn about ourselves and others***
- Favorite question: ***Why not?***

Michael Heroux (SNL), **Co-Lead PI, Director, Software Technology**

Lois Curfman McInnes (ANL), **Co-Lead PI**

David Bernholdt (ORNL), **Institutional PI, Outreach Lead**

Elsa Gonsiorowski (LLNL), **Institutional PI**

Osni Marques (LBNL), **Institutional PI, Webinars Lead**

David Moulton (LANL), **Institutional PI**

Boyana Norris (Univ of Oregon), **Institutional PI**

Elaine Raybourn (SNL) **Institutional PI, PSIP Lead**

Satish Balay (ANL)

Roscoe Bartlett (SNL)

Anshu Dubey (ANL)

Patricia Grubel (LANL)

Rinku Gupta (ANL), **BSSw Editor-in-Chief**

Stephen Hudson (ANL)

Reed Milewicz (SNL)

Mark Miller (LLNL)

Jared O'Neal (ANL)

Barry Smith (ANL)

Greg Watson (ORNL)

Jim Willenbring (SNL), **SDK Lead**

Paul Wolfenbarger (SNL)

Lisa Childers (ALCF)

Rebecca Hartman-Baker (NERSC)

Judy Hill (OLCF)

Hai Ah Nam (LANL), **BSSw Fellows**

Jean Shuler (LLNL)

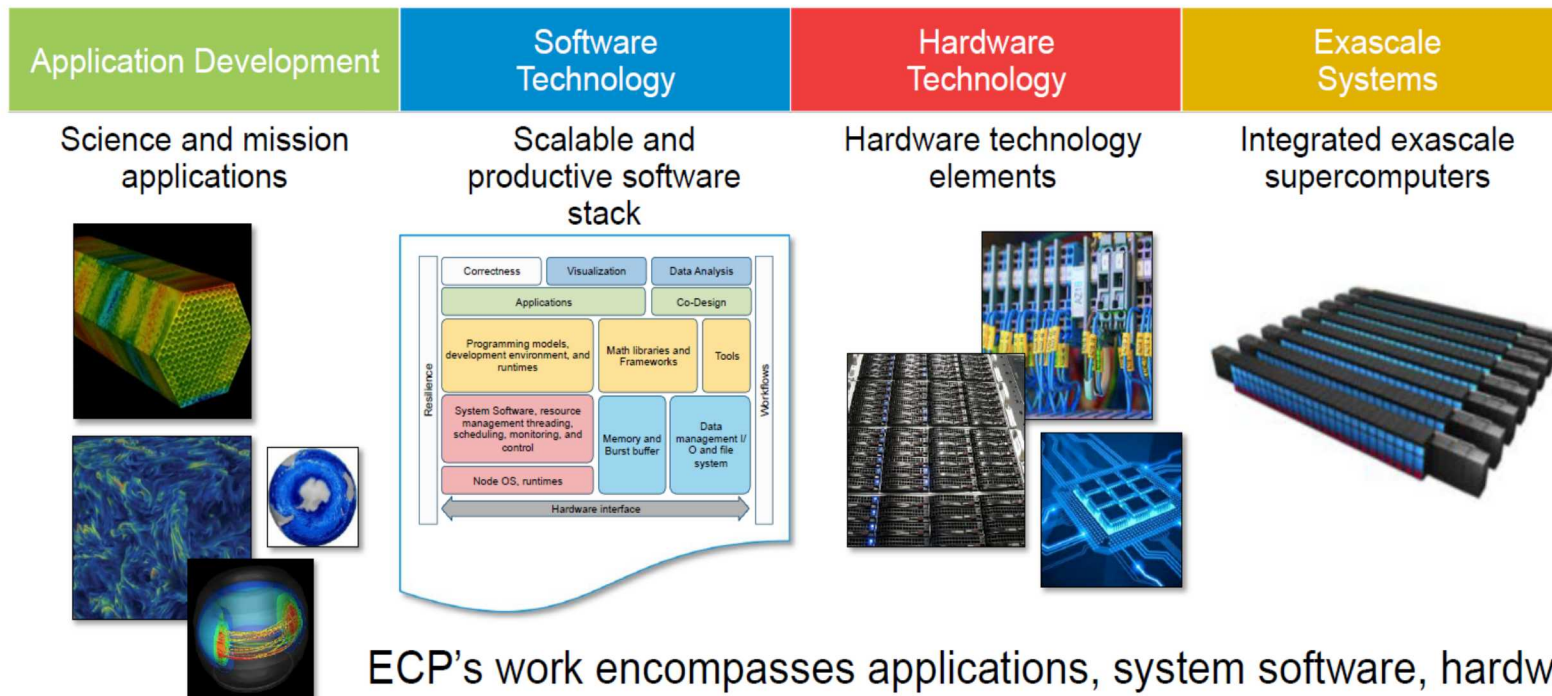
**Computing  
Facilities  
Liaisons**

## What is the Exascale Computing Project?

- As part of the National Strategic Computing initiative, ECP was established to accelerate delivery of a **capable exascale computing system** that integrates hardware and software capability to deliver approximately 50 to 100 times more performance than today's petaflop machines.
- ECP's work encompasses applications, system software, hardware technologies and architectures, and workforce development to meet the scientific and national security mission needs of DOE.

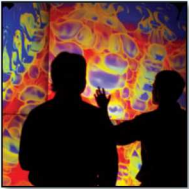
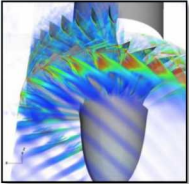

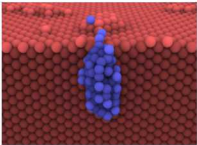
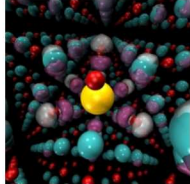
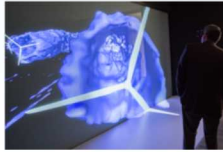
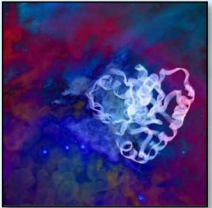
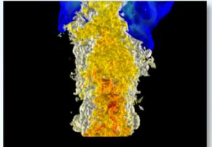


# To achieve capable exascale requires a holistic approach



ECP's work encompasses applications, system software, hardware technologies and architectures, and workforce development

# ECP applications target six strategic areas

National security	Energy security	Economic security	Scientific discovery	Earth system	Health care
<p>Stockpile stewardship</p> <p>Next-generation electromagnetics simulation of hostile environment and virtual flight testing for hypersonic re-entry vehicles</p>  	<p>Turbine wind plant efficiency</p> <p>High-efficiency, low-emission combustion engine and gas turbine design</p> <p>Materials design for extreme environments of nuclear fission and fusion reactors</p> <p>Design and commercialization of Small Modular Reactors</p> <p>Subsurface use for carbon capture, petroleum extraction, waste disposal</p> <p>Scale-up of clean fossil fuel combustion</p> <p>Biofuel catalyst design</p>	<p>Additive manufacturing of qualifiable metal parts</p> <p>Reliable and efficient planning of the power grid</p> <p>Seismic hazard risk assessment</p> <p>Urban planning</p>  	<p>Find, predict, and control materials and properties</p> <p>Cosmological probe of the standard model of particle physics</p> <p>Validate fundamental laws of nature</p> <p>Demystify origin of chemical elements</p> <p>Light source-enabled analysis of protein and molecular structure and design</p> <p>Whole-device model of magnetically confined fusion plasmas</p> 	<p>Accurate regional impact assessments in Earth system models</p> <p>Stress-resistant crop analysis and catalytic conversion of biomass-derived alcohols</p> <p>Metagenomics for analysis of biogeochemical cycles, climate change, environmental remediation</p> 	<p>Accelerate and translate cancer research</p>  

## ECP Goals

- **Application Development:** Deliver a broad array of comprehensive science-based computational applications that effectively utilize exascale HPC technology to provide breakthrough simulation and data analytic solutions for scientific discovery, energy assurance, economic competitiveness, health enhancement, and national security
- **Ease of Use:** Create software that makes exascale systems usable by a wide variety of scientists and engineers across a range of applications
- **Diverse Architectures:** Enable by 2023  $\geq$  two diverse computing platforms with up to 50 $\times$  more computational capability than today's 20 PF systems, within a similar size, cost, and power footprint
- **US HPC Leadership:** Help ensure continued American leadership in architecture, software and applications to support scientific discovery, energy assurance, stockpile stewardship, and nonproliferation programs and policies

## ECP by the Numbers

7  
YEARS  
\$1.7B

A seven-year, \$1.7 B R&D effort that launched in 2016

6  
CORE DOE  
LABS

Six core DOE National Laboratories: Argonne, Lawrence Berkeley, Lawrence Livermore, Los Alamos, Oak Ridge, Sandia

- Staff from most of the 17 DOE national laboratories take part in the project

3  
TECHNICAL  
FOCUS  
AREAS

Three technical focus areas (Application Development, Software Technology, Hardware and Integration)

100  
R&D TEAMS  
1000  
RESEARCHERS

More than 100 top-notch R&D teams

- Hundreds of consequential milestones delivered on schedule and within budget since project inception

IDEAS  
productivity



EXASCALE  
COMPUTING  
PROJECT



# Many ECP ST products are available (many github)

For example...

## Programming Models and Runtimes Products

Legion  
ROSE  
Kokkos  
DARMA  
Global Arrays  
RAJA  
CHAI  
Umpire  
MPICH  
PaRSEC  
Open MPI  
Intel GEOPM  
LLVM OpenMP compiler  
OpenMP V&V Suite  
BOLT  
UPC++  
GASNet-EX  
Qthreads

<http://legion.stanford.edu>  
<https://github.com/rose-compiler>  
<https://github.com/kokkos>  
<https://github.com/darma-tasking>  
<http://hpc.pnl.gov/globalarrays/>  
<http://github.com/LLNL/RAJA>  
<https://github.com/LLNL/CHAI>

## Mathematical Libraries Products (16)

xSDK  
hypre  
FleCSI  
MFEM  
Kokkoskernels  
Trilinos  
SUNDIALS  
PETSc/TAO  
libEnsemble  
STRUMPACK  
SuperLU  
ForTrilinos  
SLATE  
MAGMA-sparse  
DTK  
Tasmanian

<https://xsdk.info>  
<http://www.llnl.gov/casc/hypre>  
<http://www.flecsi.org>  
<http://mfem.org/>  
<https://github.com/kokkos/kokkos-kernels/>  
<https://github.com/trilinos/Trilinos>  
<https://computation.llnl.gov/projects/sundials>  
<http://www.mcs.anl.gov/petsc>  
<https://github.com/Libensemble/libensemble>  
<http://portal.nersc.gov/project/sparse/strumpack/>  
<http://crd-legacy.lbl.gov/~xiaoye/SuperLU/>  
<https://trilinos.github.io/ForTrilinos/>  
<http://icl.utk.edu/slate/>  
<https://bitbucket.org/icl/magma>  
<https://github.com/ORN-CEES/DataTransferKit>  
<http://tasmanian.ornl.gov/>

etc...

## Development Tools (19)

SICM  
QUO  
Kitsune  
SCR  
Caliper  
mpiFileUtils  
Gotcha  
TriBITS  
Exascale Code Generation Toolkit  
PAPI  
CHILL Autotuning Compiler  
Search using Random Forest

<https://confluence.exascaleproject.org/display/STSS07>  
<https://github.com/lan/libquo>  
<https://github.com/lan/kitsune>  
<https://github.com/llnl/scr>  
<https://github.com/llnl/caliper>  
<http://github.com/hpc/mpifileutils>  
<http://github.com/llnl/gotcha>  
<https://tribits.org>  
<http://icl.utk.edu/exa-papi/>

<http://hpc toolkit.org>  
[www.paradyn.org](http://www.paradyn.org)  
[www.cs.uoregon.edu/research/tau](http://www.cs.uoregon.edu/research/tau)  
[ft.ornl.gov/research/papyrus](http://ft.ornl.gov/research/papyrus)  
[ft.ornl.gov/research/openarc](http://ft.ornl.gov/research/openarc)  
[m.org/](http://m.org/)  
[www.cs.uoregon.edu/research/pdt/home.php](http://www.cs.uoregon.edu/research/pdt/home.php)



IDEAS  
productivity



EXASCALE  
COMPUTING  
PROJECT

## Challenges of CSE for ECP – *so many stories!*

### *Technical*

- All parts of the ecosystem can be under research
- Requirements change throughout the lifecycle as knowledge grows
- Importance of reproducibility, sustainability
- Verification is complicated
- Real world is messy, so is the software

### *Social*

- Competing priorities and incentives
- Limited resources
- Perception of “invisible work” with deferred or no benefit
- Need for interdisciplinary interactions
- Boutique operations must scale!

***Science through computing is only as good as the software that produces it.***

## **IDEAS-ECP enables a culture of developer productivity**

- Deliver value to ECP application teams and software technologies. This overarching goal drives all of our work.
- Engagement with the broader software community -- adapting and adopting approaches, and raising awareness of the particular needs of extreme-scale computational science.
- Development and dissemination of better scientific software practices
- Incorporation of informed strategies for human systems: focusing on changing the way scientific teams and individuals work.
- Engagement with leadership computing facilities: direct liaison roles for project leaders at all major DOE computing facilities.
- Web-based content development and delivery: expanding the development and usability of the Better Scientific Software web portal (<https://bssw.io>) as a go-to site for content on developer productivity and software sustainability.



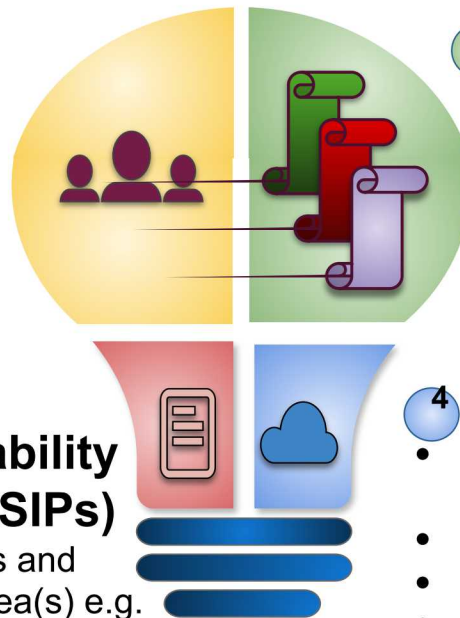
**Goal: Improve Exascale Computing Project (ECP) developer productivity and software sustainability while ensuring continued scientific success.**

**1 Interviews with Exascale Computing teams**

- Applications & Software Technology
- Understand crosscutting productivity challenges, priorities, and opportunities

**2 Productivity and Sustainability Improvement Planning (PSIPs)**

- Work with team to define focus and track progress on particular area(s) e.g. research software engineering



**3 Customize, create, and curate methodologies**

- Targeting application productivity and sustainability
- Create user stories to convey requirements from interview & PSIPs to determine priorities, plans for work

**4 Outreach and training**

- In partnership with US Department of Energy facilities
- Documents: WhatIs, HowTo, PSIP policies
- Webinar series and tutorials
- Better Scientific Software site (<https://bssw.io>)



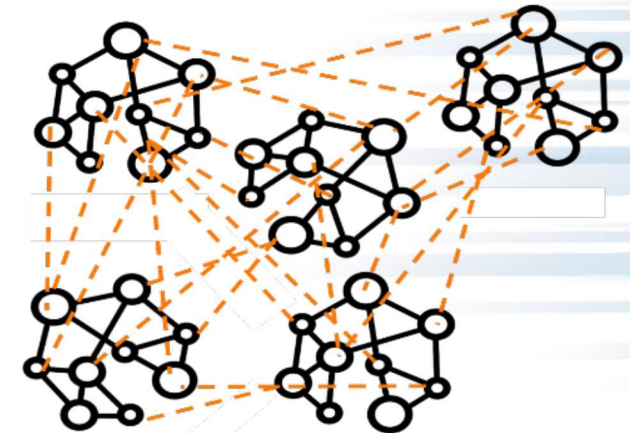
## Interactions with ECP teams have illuminated needs

- Testing/verification of scientific software
- Team onboarding and team member transitions
- Intermediate/advanced Git (especially for aggregate teams)
- Agile team management
- Agile workflows for scientific software
- Use of (interoperable) scientific libraries

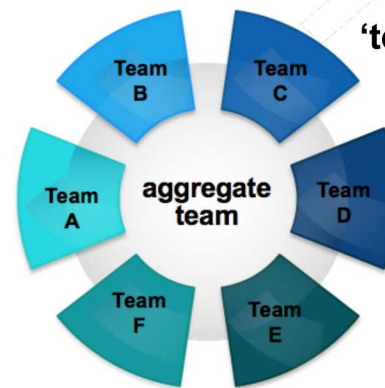
### Process for interviews, synthesis, outreach

#### ECP Application, Co-Design, and Software Teams:

CANDLE, ExaGraph, Exascale MPI, ExaStar, E3SM-MMF, EXAALT, MARBL, NWChemEx, UnifyCR, QMCPack, and WDMApp



‘teams of teams’



IDEAS  
productivity

ECP

EXASCALE  
COMPUTING  
PROJECT

# HI Training and Productivity: Integration with ST and AD

## Members of the ECP Productivity Project (i.e., IDEAS-ECP) work with ST teams to:

- **Understand** current software practices
- **Identify** crosscutting, high-priority needs for training and outreach
- **Collaborate** on Productivity and Sustainability Improvement Planning (PSIP)
- **Improve** software practices while maintaining scientific productivity; share best practices experiences

## High-priority needs for ECP teams:

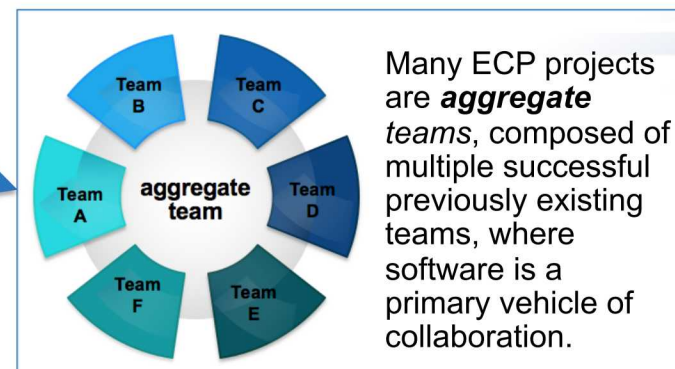
Tools that foster productive and sustainable collaboration (through software) for **aggregate** ECP science teams

- testing/verification of scientific software
- team onboarding and team member transitions
- intermediate/advanced Git (especially for aggregate teams)
- code reviews for identifying defects
- agile team management,
- agile workflows for scientific software
- use of (interoperable) scientific libraries

## Process for interviews, synthesis, outreach

### ECP Application, Co-Design, and Software Teams:

CANDLE, ExaGraph, Exascale MPI, ExaStar, E3SM-MMF, EXAALT, MARBL, NWChemEx, UnifyCR, QMCPack, and WDMApp



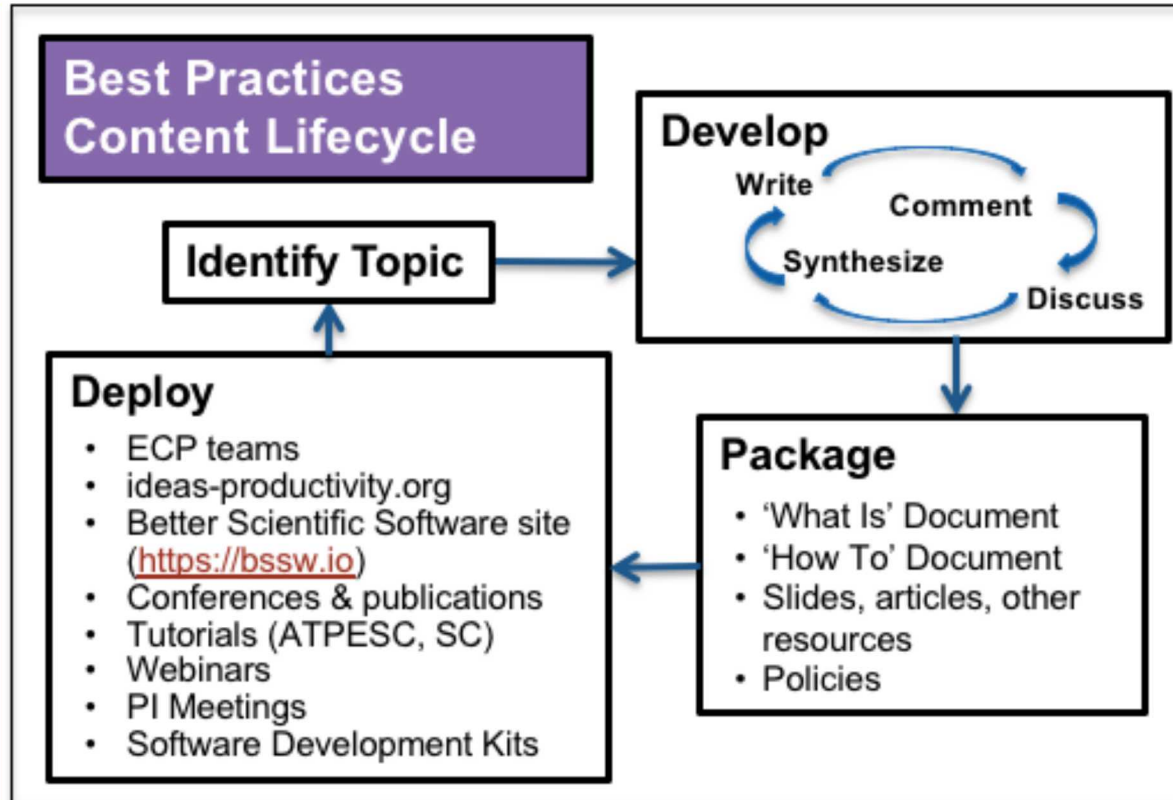
# Workflow for Best Practices Content Development

## Modern learning theory:

Build from knowledge base:  
Elaboration and models

Vast body of SE content from broad community

Learn, adapt, adopt, assimilate



# Productivity and Sustainability Improvement Planning (PSIP) Examples: EXAALT & MPICH



PSIP workflow helps a team create user stories, identify areas for improvement, select a specific area and topic for a single improvement cycle, and then develop those improvements with specific metrics for success.

## EXAALT PSIP: Continuous integration (CI) testing

BSSw blog article:

[Adopting Continuous Integration for Long Timescale Materials Simulation](#), Rick Zamora (Sept 2018)

PSIP Process: Continuous Integration (CI)	PSIP Process: Testing
<b>Target:</b> Implement and document a basic CI pipeline to act as the foundation for automated build and functionality testing.	<b>Target:</b> Implement and document practical testing examples for ongoing EXAALT development.
<b>0. Initial Status:</b> No comprehensive CI framework in place	<b>0. Initial Status:</b> No comprehensive testing framework in place
1. Develop a minimal docker image, with EXAALT dependencies	1. Add 1-3 example tests using the existing CMake infrastructure (CTest)
2. Implement a minimal 'ymf' script for the CI pipeline	2. Add 1-3 example tests using the 'Boost Test' library
3. Update EXAALT docker image to leverage CMake, and create a ParSpice-specific image for build testing	3. Integrate the CTest infrastructure with the new Boost tests
4. Generate step-by-step 'how-to' Docker-image documentation	4. Integrate the Boost-enabled CTest framework into the CI pipeline
5. Extend CI to automate build and functionality testing with both CMake and Boost.	5. <b>Bonus:</b> Work with EXAALT team to add more advanced tests to improve code coverage
Score (0-5): 4	Score (0-5): 3

## MPICH PSIP: Onboarding new team members

Practice: Create Centralized Training Resources		
Score (0 - 4)	Description	Tracking
0	Initial Status : No training process in place.	
1	Understand MPICH requirement for developers and typical challenges for new hires	✓
2	Review and gather specific training materials	✓
3	Design "MPICH Training Base" website	✓
4	Solicit feedback, improve, add and prune content to ensure effectiveness	2019



# Enabling culture change by involving others

**IDEAS productivity** [www.ideas-productivity.org](http://www.ideas-productivity.org)

## Software Development Kits (SDKs)

An ECP Software Technology Initiative

### SDKs: An ECP ST Initiative

**SDK Initiative.** The ECP Software ramps up a new initiative to create collections of ECP software product external collaboration across ECP ST product delivery strategy. P coordination points to better n

**SDK (Defn):** A collection of packages) where coordinated improve usability and prod among teams that develop capabilities. SDKs have!

- Domain scope:
- Interaction model:
- Community policy:
- Meta-infrastructure
- Coordinated r
- Community

**Policy creation ship criteria, developed!**

### A first SDK: The xSDK

**Motivation**

Now more than ever, the development of software is critical to the practice of science. However, the scientific software community is facing a crisis created by the confluence of disruptive changes in computing architectures and new opportunities for greatly improved simulation capabilities. This crisis brings with it a unique opportunity to fundamentally change how scientific software is designed, developed, and supported.

### PSIP

**Background.** The Interoperable Design of Extreme-scale Application Software (IDEAS) project is an interdisciplinary coalition of domain experts, productivity through innovative practices, processes, and tools targeting all phases of the software development lifecycle.

**What is a PSIP?** A Productivity and Sustainability Improvement Plan (PSIP) is a living document that is a planning and communication tool for capturing and conveying the practices, processes, policies and tools of a given software project. It serves a two-fold purpose:

- Improving developer productivity, increasing software quality while reducing the effort, time, and cost of development and deployment;
- Improving software sustainability, to maintain and extend a software product over its intended lifespan.

### The PSIP Workflow

Workflow for Productivity and Sustainability Improvement Plans

### Tracking Progress

**Incremental Improvement.** Both near term and long term targets for improvement are naturally identified as the development of the Current Project Practices Improvement Document. These targets are expressed and recorded using Progress Tracking Cards, usually selected from our growing catalogue.

**Progress Tracking Card: Test Coverage**

**Building a Healthy Soft**

**Other People's Code.** Use the cost (time and effort) compared to but, it also increases risk and complexity. Test the TPLs for correct behavior. Maintain builds of TPLs over a range of books. Manage the loss of functionality via the IDEAS making significant progress helping to be a foundation of an extensible scientific software ecosystem.

### Building a Healthy Soft

- We have helped develop a Current Project Practices Improvement Document (CPIP) for many of the IDEAS partner projects, and we use Progress Tracking Cards. Want to learn more?
- <https://www.lawrence Livermore National Laboratory>
- <https://github.com/better-scientific-software/psip>

## User Stories: Communicating about Software Requirements

**Goal**

• Popularize the "user story" approach to support scientific development.

• A simple, simplified description of a requirement, that can assist in planning and development.

• A simple, simplified description of a requirement, that can assist in planning and development.

### IDEAS-ECP User Story Sam

**Category**

**Example**

**As a** [role]

**I want to** [action]

**so that** [benefit]

**Example:**

As a scientist,

I want to integrate software components,

so that I can develop a new application.

## IDEAS-ECP: Advancing Software Productivity for Exascale Applications

**Goal**

• Improve scientific software application developer productivity and sustainability, as key aspects of increasing overall scientific productivity.

• With ECP community technologies for ECP development Kit of tools.

### User Stories

User stories provide flexible means of specifying and conveying requirements gathered from interviews, PSIPs, and informal interactions.

As a casual user of IDEAS, I want more details about the IDEAS project and its goals so that I can better understand the project and its impact on my work.

To be made public available (not for internal use only).

Create a document that details the IDEAS project and its goals.

Include a section on the IDEAS project and its goals.

Include a section on the IDEAS project and its goals.

Include a section on the IDEAS project and its goals.

### Improving Exascale MPI Onboarding Process through ECP IDEAS PSIP

Pavan Balaji (PI - Exascale MPI), Ken Raffanetti, Neelima Baryapu, Hui Zhou, Lois Curfman McInnes (PI - ECP IDEAS), Rinku Gupta

#### The Exascale MPI Project

- Funded by DOE for 26 years
- Has been a key influencer in the adoption of MPI MPI standard
- Firstmost comprehensive implementation of every MPI feature
- Allows supercomputing centers to not compromise on what features they demand from vendors
- DOE RND100 award in 2005
- MPI and its derivatives are the world's most widely used MPI implementations
- Supports all versions of MPI including the recent MPI-3.1

#### IDEAS Project and PSIP Overview

**ECP IDEAS Goal:** Improve scientific software application developer productivity and sustainability, as key aspects of increasing overall scientific productivity.

**Interviews with ECP**

**Productivity and Sustainability Improvement Plan (PSIP) process** is used for capturing and conveying the practices, processes, policies and tools of a given software project.

**A Productivity and Sustainability Improvement Plan (PSIP) process** is used for capturing and conveying the practices, processes, policies and tools of a given software project.

#### Executing the Exascale MPI PSIP

IDEAS PSIP process is being applied to Exascale MPI (MPICH) onboarding training process as follows:

1. Lack of Training Infrastructure
2. Expedite the Process of Training
3. Training resources, time and bandwidth are identified as a set of
4. Current values are recorded and not satisfactory
5. Planned for a Centralized Training Resources
6. Identify the Requirements for the Creation of Training Resources and Collect Resources
7. Solicit Reviews and Feedback, and Update PTC Values

Both near term and long term targets for this improvement PSIP are expressed and recorded using Progress Tracking Cards (PTC)

**And much. Review and identify next work links for improvement**

#### Future Plans

stand portal to make it a strong resource for the ECP community

house material on set topics (and not set MPI)

contributions to the website

### IDEAS-EXAALT Collaboration: Adopting Continuous Integration for Long-Timescale Materials Simulation

Richard J. Zamora<sup>1</sup>, Christoph Junghans<sup>2</sup>, and David Mouton<sup>2</sup>  
<sup>1</sup>Argonne National Laboratory, <sup>2</sup>Los Alamos National Laboratory

#### Overview

Sustainability Improvement Planning (PSIP) methodology being developed by the Interoperable Design of Extreme-scale Application Software (IDEAS) project. In this Application Software (IDEAS) project, we highlight a recent PSIP-based effort to implement an end-to-end continuous-work integration pipeline within the EXAALT integration pipeline within the EXAALT application-project software repository.

This work is highlighted in a September 2018 Better Scientific Software (BSSW) blog post: <https://bssw.io/>

#### The EXAALT Simulation Framework

Figure 1. One application of EXAALT is modeling the surface of a fusion reactor (shown above is the interior of a tokamak at CTR, photographed by Chris Bohn, Lawrence Livermore National Laboratory). Simulation image credit: Luis Delgado.

Figure 2. Illustration of the EXAALT framework. The three main software components (LAMMPS, LATTE, and ParSplice) are represented as concentric circles, while other components are represented as grey circles (grey eggs). The connections between the various software components are represented as lines (green eggs).

#### Continuous Integration (CI) in EXAALT

Boost: Used to implement and organize functionality tests (integration, regression, and unit) inside CTest.

GitLab CI: Used to automatically build and test the software framework (using CMake) to validate new repository commits.

Docker: Used to generate standard system images (with library dependencies) for use in CI.

#### Applying the PSIP Workflow

Continuous Integration (CI) is a software development process that relies on the automated compilation and testing of all new features to detect bugs early in the development cycle. In EXAALT, we leverage the following tools:

- CMake: Used to manage the compilation of EXAALT and then to execute functionality tests (using CTest) for each build.

Figure 3. Illustration of a continuous integration pipeline for EXAALT: the preliminary pipeline was implemented using GitLab CI (along with other complementary tools).

PSIP Process: Testing

Continuous Integration (CI) is a software development process that relies on the automated compilation and testing of all new features to detect bugs early in the development cycle. In EXAALT, we leverage the following tools:

- CMake: Used to manage the compilation of EXAALT and then to execute functionality tests (using CTest) for each build.

Figure 3. Illustration of a continuous integration pipeline for EXAALT: the preliminary pipeline was implemented using GitLab CI (along with other complementary tools).

PSIP Process: Testing

Continuous Integration (CI) is a software development process that relies on the automated compilation and testing of all new features to detect bugs early in the development cycle. In EXAALT, we leverage the following tools:

- CMake: Used to manage the compilation of EXAALT and then to execute functionality tests (using CTest) for each build.

Figure 3. Illustration of a continuous integration pipeline for EXAALT: the preliminary pipeline was implemented using GitLab CI (along with other complementary tools).

PSIP Process: Testing



Information For ▼

Contribute To BSSw

Receive Our Email Digest



Find Resources ▼

Blog

Events

About



➤ BSSw Site Launch At SC17 ... Contribute To Better Scientific Software!



<https://bssw.io>

Collaborative content development on general topics related to developer productivity and software sustainability for CSE

We want and *need* contributions from the community ... Join us!

# Better Scientific Software (BSSw)

Scientific software has emerged as an essential discipline in its own right. Because computational models, computer architectures, and scientific software projects have become extremely complex, the Computational Science & Engineering (CSE) community now has a unique opportunity—and an implicit mandate—to address pressing challenges in scientific software productivity, quality, and sustainability.

GET ORIENTED

Communities Overview

Site Overview

Intro To CSE

Intro To HPC



## What is BSSw?

Community-based resource for sharing information on practices, techniques, and tools to improve developer productivity and software sustainability for computational science and engineering.

**We want and *need* contributions from the community ... Join us!**

- **Types of content**

- Informative articles
- Curated links
  - Highlight other web-based content
- Events
- WhatIs, HowTo docs
- Blog articles

Information For [Contribute To BSSw](#) [Receive Our Email Digest](#)

Find Resources [Blog](#) [Events](#) [About](#) [https://bssw.io](#)

New blog article ... Better Scientific Software: 2018 Highlights

### Better Scientific Software: 2018 Highlights

Vol 2018

- [Better Science through Software Testing](#), Tom Evans
- [SuperLU: How Advances in Software Practices Are Increasing Sustainability and Collaboration](#), Xiaoye Li
- [Building Connections and Community within an Institution](#), Greg Watson and Elsa Gonsiorowski
- [Can You Teach an Old Code New Tricks?](#), Charles Ferenbaugh
- [Adopting Continuous Integration for Long-Timescale Materials](#), Rick Zamora
- [Porting Code to New Architectures](#), Bronson Messer

And many more!

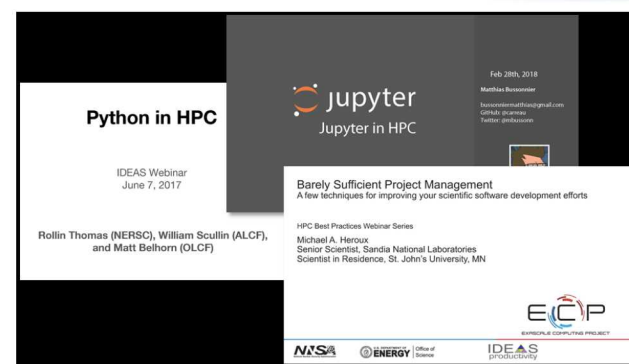
**Receive our  
email digest**

**Many ECP  
contributors**



# HPC Best Practices Webinar Series

- 2017-06-07 Python in HPC, Rollin Thomas (NERSC), William Scullin (ALCF) and Matt Belhorn (OLCF)
- 2017-09-13 Barely Sufficient Project Management: A few techniques for improving your scientific software development, Mike Heroux (SNL)
- 2017-11-01 Managing Defects in HPC Software Development, Tom Evans (ORNL)
- 2018-01-17 Bringing Best Practices to a Long-Lived Production Code Charles Ferenbaugh (LANL)
- 2018-04-18 Software Citation Today and Tomorrow, Daniel Katz (NCSA and UIUC)
- 2018-06-13 Popper: Creating Reproducible Computational and Data Science Experimentation Pipelines, Ivo Jimenez (UCSC)
- 2018-07-18 How Open Source Software Supports the Largest Computers on the Planet, Ian Lee (LLNL)
- 2018-08-21 Software Sustainability: Lessons learned from different disciplines, Neil Chue Hong (Software Sustainability Institute, UK)
- 2018-09-19 Modern CMake, Bill Hoffman (Kitware)







# BSSw Fellowship Program

Recognition & funding to leaders and advocates of high-quality scientific software

Class of 2019



Class of 2018



We will begin accepting applications for the 2020 BSSw Fellowship Program in September, 2019

We are looking for people who are:

- Passionate about scientific software.
- Interested in contributing powerful ideas, tools, methodologies, and more that improve the quality of scientific software.
- Able to use the fellowship to broadly benefit the scientific software community.
- Willing to participate as an alum in subsequent years to guide selection of future fellows and promote better scientific software in their community.

<https://bssw.io>

*So your code will see the future*

**Subscribe to BSSw mailing list to be notified about BSSw Fellowship for 2020**



EXASCALE  
COMPUTING  
PROJECT

# License, citation and acknowledgements



## License and Citation

- This work is licensed under a [Creative Commons Attribution 4.0 International License](https://creativecommons.org/licenses/by/4.0/) (CC BY 4.0).
- Requested citation: Raybourn, E.M. Enabling a Culture of Developer Productivity and Software Sustainability. 2019 SIAM Conference on Computational Science and Engineering, Spokane, WA, 2019. March 1, 2019. SAND2019-XXXX C. DOI: <https://doi.org/XXXXX>.

## Acknowledgements

- Special thanks to the members of IDEAS-ECP.
- 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.
- Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA-0003525. Images used by permission.