2014 DOE Report on Software Productivity for Extreme-Scale Science: Perspectives on Recommendations, Progress, and Urgent Needs

Lois Curfman McInnes, David E. Bernholdt, Rinku Gupta, Michael Heroux, Mark C. Miller
June 2019

Abstract. The 2014 DOE Workshop on *Software Productivity for Extreme-Scale Science (SWP4XS)* brought together experts in the development of large-scale scientific applications, numerical libraries, and computer science infrastructure to determine how to address the growing crisis in software productivity caused by disruptive changes in extreme-scale computer architectures and new frontiers in extreme-scale modeling, simulation, and analysis. The resulting workshop report [1] identified gaps in software productivity and determined the following recommendations for a collection of focused efforts within the DOE Office of Advanced Scientific Computing Research (ASCR) as well as broader partnerships:

- Characterize and measure extreme-scale software productivity impediments and opportunities.
- Develop software designs that minimize the impact of ongoing computer architecture changes.
- Characterize opportunities for new frontiers at extreme scale, including advanced multiphysics, multiscale, and analysis.
- Develop composable and interoperable components and libraries.
- Identify, develop, and disseminate knowledge of productivity tools and best practices.
- Grow an extreme-scale software productivity community.
  - Develop productivity partnerships throughout DOE and other agencies.
  - Establish a Software Productivity Technical Council.

In the context of these recommendations, this whitepaper summarizes progress during the past five years in DOE software productivity and sustainability, with emphasis on work by members of the IDEAS (Interoperable Design of Extreme-scale Application Software) project (https://ideas-productivity.org) and collaborators. The scope of this whitepaper does not include comprehensive discussion of complementary efforts throughout the broader community.

History. The IDEAS project began in September 2014, jointly sponsored by ASCR and the DOE Office of Basic Energy Research (BER) to address challenges in software productivity and sustainability, with emphasis on terrestrial ecosystem modeling. The IDEAS project expanded in 2017 in the DOE's Exascale Computing Project (ECP, https://www.exascaleproject.org), which requires intensive development of applications and software technologies while anticipating and adapting to continuous advances in computing architectures. The role of IDEAS within the ECP is to ease the challenges of software development and ensure that investment in the exascale software ecosystem is as productive and sustainable as possible. On the BER front, the IDEAS project expanded in 2019 via the IDEAS-Watershed project, to accelerate watershed science through a community-driven software ecosystem.

For each recommendation of the SWP4XS report, we summarize advances and discuss remaining challenges and opportunities for future work.

Characterize and measure extreme-scale software productivity impediments and opportunities. We must gather, analyze, and categorize software productivity impediments and opportunities in order to provide insight into fundamental problems and direction for future efforts. We must also identify, measure, and track metrics for software productivity throughout software lifecycles, with the ultimate goal of improving key metrics of scientific productivity.

- **Advances.** A key activity of the IDEAS project has been interviewing ECP teams to identify current software practices and productivity challenges, to understand preferred approaches for collaboration, and to determine crosscutting, high-priority needs for training and outreach. The interviews act as the first stage in the engagement with teams that may result in Productivity and Sustainability Improvement Planning (PSIP). We created an interview protocol, interview questions, and the PSIP process—a lightweight, iterative process where teams identify their most urgent software bottlenecks and track progress on work to overcome them. These resources are available to the community in the PSIP Tools repository. A blog article discusses the PSIP process for adopting continuous integration (CI) testing for molecular dynamics simulations in the EXAALT project.

- **Observations.** These ECP team interviews and early PSIP partnerships are enabling the development of crosscutting methodologies materials based on understanding teams' most urgent bottlenecks in software productivity. The teams are directly contributing to the development of methodologies materials with concrete use cases, motivation, feedback, and refinement. Initial advances are good steps to encourage a culture of *continuous technology refreshment*, a term introduced in a recent blog article as an essential aspect of sustaining software. Much work remains, where an ongoing challenge is getting the attention of teams due to tension between investment in improving software quality and the need for continual scientific advances.

Develop software designs that minimize the impact of ongoing computer architecture changes. We must transition software design and data structures to take full advantage of new features of emerging extreme-scale architectures in order to achieve portable performance as well as resilience, reproducibility, and use of high-level abstractions. We must determine how to achieve good performance while at the same time provide stability for end users. This work is particularly important for libraries that insulate domain scientists from architecture-specific details.

- **Advances and observations.** This is an area of active concern and activity in the computational science and engineering (CSE) community. We have moved past the stage of thinking that there must be some silver bullet solution to the performance portability challenge and are now in a phase of gathering and sharing practical experience on the diverse architectures that now confront us. We will have to wait for these technologies to mature further before we understand their full implications in the *productivity* discussion. For the most part, this issue will need to play out in the technical performance portability space, but it behooves those interested in productivity to keep an eye on these discussions and consider the implications for productivity and sustainability.

Characterize opportunities for new frontiers at extreme scale, including advanced multiphysics, multiscale, and analysis. Exascale systems open up the opportunity to couple multiple physics and scales, as well as to introduce parameter sensitivity studies, optimization, uncertainty quantification, machine learning, and other so-called "outer loop" computations. Enabling technologies can greatly improve our ability to couple existing components and to transition simulation and analysis toward predictive science.

- **Advances and observations.** While much progress has been made throughout the scientific community, in this early phase we are merely scratching the surface in exploring directions of next-generation CSE. Given the broad scope of this topic, we should consider issues in productivity and sustainability as workflows continue to advance toward predictive science. Such workflows are likely to comprise a fusion of multiple different technologies that are still maturing.

Develop composable and interoperable components and libraries. Research is needed to ensure that our widely used and independently developed software libraries can be readily used *in combination* in extreme-scale multiphysics and multiscale applications that require the complementary contributions of diverse teams. We must transform libraries, tools, and applications into robust, interoperable, and easy-to-use software ecosystems, with support for composability and library interoperability, multiphysics and multiscale coupling methodologies, configuration and build systems, as well as testing, verification, validation, and debugging at scale.

- **Advances.** Work on the Extreme-scale Scientific Software Development Kit (xSDK, https://xsdk.info) began in 2014 as part of the IDEAS project and in 2017 transitioned to support in ECP as a primary delivery mechanism for math libraries. Work focuses on community development and a commitment to combined success via quality improvement policies, better build infrastructure, and the ability to use diverse, independently developed libraries in combination to solve large-scale multiphysics problems. xSDK community policies (https://xsdk.info.policies) govern activities and set expectations for future xSDK members. Any package that satisfies the community policies is welcome; xSDK-0.4.0 (December 2018) included 17 numerical packages and two application components. The xSDK community has defined three levels of xSDK library interoperability, and developers have designed and implemented interoperability among complementary solvers in selected xSDK packages, including tests for inter-library functionality. Broader work is under way on the Extreme-scale Scientific Software Stack (E4S, https://e4s-project.github.io), a community effort to provide open source software packages for developing, deploying and running scientific applications on HPC platforms.
- **Observations.** Work on these coordinated software releases has pushed community advances on tools for building, packaging, and testing diverse software at scale. For example, xSDK and E4S have adopted Spack (https://spack.io) as a meta-build tool, and ECP working groups are advancing capabilities for CI testing and container technologies on extreme-scale architectures.

Identify, develop, and disseminate knowledge of productivity tools and best practices. Effective tools and practices provide the most tangible means to improving productivity. We must adapt mainstream software productivity and software engineering methodologies in order to address the unique requirements of extreme-scale scientific computing. Specific focus on identifying the needs for productivity tools, developing and evaluating solutions to those needs, and deploying the best tools and practices will accelerate productivity across all activities, independent of other initiatives.

- **Advances.** IDEAS initiatives focus on developing, customizing, curating, and deploying best practices as the fundamental way to improve software sustainability and programmer productivity for extreme-scale science. The IDEAS project and broader HPC community are composed of team members with many years of valuable experience in designing and producing high-quality, reusable scientific software. This collective experience provides a foundation for focused discussion, distillation, and development of useful practices for CSE software development. We have developed a *best practices content lifecycle model* and successfully used this lifecycle model to produce a large and growing collection of content to promote best practices in CSE software engineering, as available through the newly established Better Scientific Software site (https://bssw.io). For example, some content is produced as part of our "What Is" and "How To" documents, which provide concise characterizations and best practices for important topics in CSE software projects (such as software configuration, documentation, testing, version control, agile methodologies, and others), thus enabling teams to consider improvements at a small but impactful scale.
- **Observations.** While this work is a solid start, much more remains to be done to address the full range of needs of next-generation scientific software and to update guidance as technologies continue to advance.

Grow an extreme-scale software productivity community. Provide productivity training for software developers, and create forums for information exchange about software productivity issues.

- **Advances.** This is an area in which the IDEAS project has been very active, together with others. IDEAS seeks to grow the community via tutorials, a webinar series, and the creation of venues for technical discussions about software development productivity within larger conferences. We have found members of the leadership teams in both important institutions (e.g., computing facilities) and conferences to be strongly supportive of these kinds of activities, and willing to facilitate them. These events are tracked on the IDEAS project web site (https://ideas-productivity.org/events**).** Among these, the "Best Practices for HPC Software Developers" webinar series is noteworthy. Since being established in 2016, we have presented 30 webinars to more than 1700 attendees. The IDEAS project is certainly not alone in this space. The Blue Waters Webinar Series recently launched a new topical area on "Scientific Software Ecosystems". At the recent PASC19 and ISC2019 conferences, the number of "software-related" events organized by various international groups was notable, and we have noticed similar trends in other conferences that are open to this kind of input (software productivity topics are still unlikely to compete in venues that involve peer-reviewed papers exclusively).
- **Observations.** While the progress in this area has been gratifying, we must continue the "pressure" to bring discussions of software productivity and sustainability out of the shadows. As stated in a recent article on community organizations [2] in a special issue of CiSE, a useful next step would be to increase communication among the various organizations who have been working at the grassroots level to make this happen, to move from occasional discussions to a more active level of coordination and collaboration.

The SWP4XS report also recommended developing productivity partnerships throughout DOE and other agencies. Informal interactions have promoted loose collaboration among complementary groups who are working on issues in software productivity and sustainability [2]. Several venues have promoted discussions, including the 2015 multi-agency *Computational Science and Engineering Software Sustainability and Productivity (CSESSP) Challenges Workshop* [3] and the very event to which this whitepaper is submitted, the *2019 Collegeville Workshop on Sustainable Scientific Software* (https://collegeville.github.io/CW3S19). Also, the NSF-funded *Conceptualization of a US Research Software Sustainability Institute* (URSSI, http://urssi.us) is making the case for and planning a possible institute to improve science and engineering research by supporting the development and sustainability of research software in the United States. While conversations on software issues are advancing throughout the DOE community, an additional recommendation from the SWP4XS report on establishing a Software Productivity Technical Council for DOE has not been realized.

References:

[1] *Software Productivity for Extreme-Scale Science*, DOE Workshop Report, H. Johansen, L.C. McInnes, et al., January 2014, https://science.osti.gov/-/media/ascr/pdf/research/cs/Exascale-Workshop/SoftwareProductivityWorkshopReport2014.pdf

[2] Community Organizations: Changing the Culture in Which Research Software is Developed and Sustained, D. Katz, L.C. McInnes, D.E. Bernholdt, A.C. Mayes, N.P. Chue Hong, J. Duckles, S. Gesing, M.A. Heroux, S. Hettrick, R. Jimenez, M. Pierce, B. Weaver, N. Wilkins-Diehr, special issue of *Computing in Science and Engineering (CiSE)* on "Accelerating Scientific Discovery with Reusable Software", vol 21, issue 2, March-April 2019, pp. 8-24, https://dx.doi.org/10.1109/MCSE.2018.2883051.

[3] *Computational Science and Engineering Software Sustainability and Productivity (CSESSP) Challenges Workshop Report*, M.A. Heroux, G. Allen, et al., October 2015, https://www.nitrd.gov/PUBS/CSESSPWorkshopReport.pdf.