GUEST EDITORS' INTRODUCTION

Transforming Science Through Software: Improving While Delivering 100×

Richard Gerber, *Lawrence Berkeley National Laboratory, Berkeley, CA, 94720, USA*
Steven Gottlieb, *Indiana University, Bloomington, IN, 47405, USA*
Michael A. Heroux, *Sandia National Laboratories, Albuquerque, NM, 87185, USA*
Lois Curfman McInnes, *Argonne National Laboratory, Lemont, IL, 60439, USA*

The U.S. Department of Energy (DOE) Exascale Computing Project (ECP)[1,2,3] represented one of the largest scientific software projects ever. Charged with establishing a practical high-performance computing (HPC) ecosystem around systems capable of achieving a billion-billion double-precision floating point operations per second, or an exaflop, the ECP funded the development of new (and the transformation of important existing) applications, libraries, and tools that realized much more improvement in performance and capabilities than what the raw hardware performance potential would indicate. Exascale systems are nominally 50 times more capable than the supercomputing systems available at the start of the ECP, but the improvements in observed capabilities were often 100 times or more.

This exceptional gain, achieved in seven years, inspired the title of this special issue: "Transforming Science Through Software: Improving While Delivering 100×." Through the six invited articles in this special issue, our goal is to share experiences with the practice and science of scientific software development, with an emphasis on developing a coherent, portable, and sustainable software ecosystem for HPC, with an eye to the needs of next-generation computational science.

The term "100×" refers to advancing capabilities in modeling, simulation, and analysis by a factor of 100 or more using some combination of new algorithms, optimization techniques, software libraries, and programming models coupled with the next generation of HPC hardware. The work represented in this issue was performed by teams funded by the ECP who were charged with delivering novel algorithms for emerging exascale computing platforms in robust software libraries, tools, and applications as needed by stakeholders while simultaneously improving software quality to reduce technical risk and build a firm foundation for scientific advances.

1521-9615 © 2024 IEEE
Digital Object Identifier 10.1109/MCSE.2024.3400462
Date of current version 5 July 2024.

HETEROGENEOUS ARCHITECTURES AND EXASCALE COMPUTING

For decades, HPC applications steadily increased performance by relying on a relatively constant and straightforward growth in the capability of computing hardware based on CPUs. However, about 15 years ago, single-core CPU performance growth began to slow due to the end of Dennard scaling and the slowing of Moore's law. New heterogeneous computing architectures (typically incorporating hardware accelerators[4]) arose with the potential to unleash enormous gains in computational capability for science (even at the desktop level), but realizing that potential comes with substantially increased programming complexity. Developing and optimizing applications for heterogeneous systems is challenging due to massive parallel concurrency requirements, reduced bandwidth-to-compute ratios, heterogeneous memory spaces, new programming models and languages, changing storage paradigms, and a proliferation of hardware accelerators.

Beginning in 2016, the DOE Office of Science's Advanced Scientific Computing Research program and the National Nuclear Security Agency's Advanced Simulation and Computing program jointly formed the ECP to support the research, development, and deployment of science and national security applications, software, and hardware technologies specifically targeted to the DOE mission priorities. Six years after the start of the ECP, the nation saw the delivery of the world's first exascale systems. By the time of the ECP's conclusion in December 2023, ECP teams had developed a diverse suite of applications in chemistry, materials, energy, Earth and space science, data analytics, optimization, AI, and national security.[5] In turn, these applications build on a robust software ecosystem,[6] including programming models and runtimes, mathematical libraries, data and visualization packages, and development tools that comprise the ECP-developed Extreme-Scale Scientific Software Stack (E4S). Achieving the required advances in simulation and scalability in the ECP required a high level of collective efficiency, performance, and fidelity in all models, methods, software, tools, and ECP-enabled computing technologies. This, in turn, required a new level of coordination and collaboration among teams of researchers, computer scientists, and software technology specialists with hardware vendors and DOE HPC facilities that build and manage exascale computers.

TRANSFORMING SCIENCE THROUGH SOFTWARE: IMPROVING WHILE DELIVERING 100×

To inform future scientific software communities, the articles in this special issue of *Computing in Science & Engineering* focus on ECP software technologies and their deployment at DOE HPC facilities through the lens of a sustainable scientific software ecosystem, as needed to support applications at the exascale and beyond. We also discuss cultural changes that embrace continual improvement in software practices as needed for collaborative and trustworthy computational science. A complementary special issue of *Computing in Science & Engineering*, "Scientific Impact of the Exascale Computing Project," focuses on advances in ECP applications.

Efforts in the ECP on scientific libraries and tools represent one of the most significant scientific software projects ever, requiring a high degree of coordination across many teams and institutions to deliver various software capabilities in support of new exascale platforms and their users. In addition, the mission and size of the ECP allowed these teams to invest in collaboration to understand and improve how work on scientific software is accomplished. Building on ECP successes, the community is now tackling new opportunities in the computing sciences. For example, ECP outcomes set the stage for future DOE efforts in integrated research infrastructure and AI for science, which will require a rich, reliable, and portable software stack to support uniformity across systems as well as new libraries and tools seamlessly integrated with those provided by the ECP.

SYNOPSIS OF THE ARTICLES

The articles in this issue discuss complementary topics related to building this scientific software ecosystem, including cultural changes as we embrace the fundamental role of high-quality software in sustained collaboration and discovery in computational science. Topics include software portfolio leadership and management techniques; community development; increased impact on scientific progress via robust scientific libraries and tools that are portable across advanced computing architectures; and realizing scientific potential through the delivery, integration, and performance tuning of application codes built on this software. The following is a brief synopsis of articles in this special issue:

- "Scalable Delivery of Scalable Libraries and Tools: How ECP Delivered a Software Ecosystem for Exascale and Beyond"[A1] explains software portfolio leadership and management techniques in the ECP Software Technology Focus area, emphasizing applicability to other large-scale research software projects.
- "Providing a Flexible and Comprehensive Software Stack Via Spack, an Extreme-Scale Scientific Software Stack, and Software Development Kits"[A2] introduces the multilayer software ecosystem developed for the ECP and beyond, which provides a curated distribution of numerical libraries, runtime systems, and tools that lower the barrier to entry to HPC and AI/machine learning developer communities while promoting software interoperability, quality improvement, porting, testing, and deployment across a range of computing environments.
- "Creating Continuous Integration Infrastructure for Software Development on U.S. Department of Energy High-Performance Computing Systems"[A3] presents the ECP's approach for creating robust continuous integration workflows that span multiple protected HPC environments, emphasizing strategies to address challenges in automation, charging models, and security requirements.
- "Deploying Optimized Scientific and Engineering Applications on Exascale Systems"[A4] highlights how the ECP's Application Integration area served as an integration point among applications, supporting software, system environments, HPC facilities, and vendors—overcoming challenges in using first-of-a-kind computing systems while promoting portable and sustainable programming models and helping harden systems before general availability.

- "A Cast of Thousands: How the IDEAS Productivity Project Has Advanced Software Productivity and Sustainability"[A5] explains how synergistic work—in fostering software communities while incubating methodologies and disseminating knowledge to advance developer productivity and software sustainability—helps to mitigate technical risks by building a firmer foundation for reproducible, sustainable science at all scales of computing, from laptops, to clusters, to the exascale and beyond.
- "Then and Now: Improving Software Portability, Productivity, and 100× Performance"[A6] discusses how transformations in software development methodologies and interdisciplinary teams during the ECP led to improvements in software technologies and applications and why these advances are essential for next-generation computational science.

We recognize that these articles represent only a subset of the broad range of excellent work in scientific software throughout the international community. We hope the articles will benefit research teams and provide a foundation for advancing computational science and our knowledge and understanding of the world. We further hope that these articles will foster expanded community efforts related to the fundamental role of sustainable scientific software ecosystems in advancing the computing sciences.

ACKNOWLEDGMENTS

This research was supported by the ECP (17-SC-20-SC), a collaborative effort of the DOE Office of Science and the National Nuclear Security Administration. We thank all those in the ECP, DOE, and broader computational science community who are advancing software for science. We especially thank people who have contributed to the work discussed in this special issue.

We sincerely thank all reviewers of articles in this special issue for their insights and feedback, which substantially improved the manuscripts: Wolfgang Bangerth, David Bernholdt, Manoj Bhardwaj, John Cazes, Edmond Chow, Ben Cowan, Charles Ferenbaugh, Jennifer Green, Fred Johnson, Alicia Klinvex, Scott L'Ecuyer, Paul Lin, Ray Loy, Mark Miller, David Moulton, Ryan Richard, David Rogers, Joost VandeVondele, and John Quincy Wofford III. This special issue used an open review process, where the coauthors of an article were provided with the names of their article's three reviewers but not attribution of comments to a particular reviewer.

APPENDIX: RELATED ARTICLES

A1. M. Heroux, "Scalable delivery of scalable libraries and tools: How ECP delivered a software ecosystem for exascale and beyond," *Comput. Sci. Eng.*, vol. 26, no. 1, pp. 9–18, Jan./Mar. 2024, doi: 10.1109/MCSE.2024.3384937.

A2. J. M. Willenbring, S. S. Shende, and T. Gamblin, "Providing a flexible and comprehensive software stack via spack, an extreme-scale scientific software stack, and software development kits," *Comput. Sci. Eng.*, vol. 26, no. 1, pp. 20–30, Jan./Mar. 2024, doi: 10.1109/MCSE.2024.3395016.

A3. R. Adamson et al., "Creating continuous integration infrastructure for software development on U.S. department of energy high-performance computing systems," *Comput. Sci. Eng.*, vol. 26, no. 1, pp. 31–39, Jan./Mar. 2024, doi: 10.1109/MCSE.2024.3362586.

A4. R. Gerber, B. Joo, and S. Parker, "Deploying optimized scientific and engineering applications on exascale systems," *Comput. Sci. Eng.*, vol. 26, no. 1, pp. 41–47, Jan./Mar. 2024, doi: 10.1109/MCSE.2024.3397209.

A5. L.C. McInnes et al., "A cast of thousands: How the IDEAS productivity project has advanced software productivity and sustainability," *Comput. Sci. Eng.*, vol. 26, no. 1, pp. 48–60, Jan./Mar. 2024, doi: 10.1109/MCSE.2024.3383799.

A6. H. Anzt, A. Huebl, and X. S. Li, "Then and now: Improving software portability, productivity, and 100× performance," *Comput. Sci. Eng.*, vol. 26, no. 1, pp. 61–70, Jan./Mar. 2024, doi: 10.1109/MCSE.2024.3387302.

REFERENCES

1. "U.S. DOE Exascale Computing Project." Exascale Computing Project. Accessed: Jun. 14, 2024. [Online]. Available: https://exascaleproject.org
2. P. Messina, "The Exascale Computing Project," *Comput. Sci. Eng.*, vol. 19, no. 3, pp. 63–67, May/Jun. 2017, doi: 10.1109/MCSE.2017.57.

3. D. Kothe, S. Lee, and I. Qualters, "Exascale computing in the United States," *Comput. Sci. Eng.*, vol. 21, no. 1, pp. 17–29, Jan./Feb. 2019, doi: 10.1109/MCSE.2018.2875366.
4. M. Rosso and A. Myers. "A gentle introduction to GPU programming." Better Scientific Software (BSSw). Accessed: Jun. 14, 2024. [Online]. Available: https://bssw.io/blog_posts/a-gentle-introduction-to-gpu-programming
5. F. Alexander et al., "Exascale applications: Skin in the game," *Philos. Trans. Roy. Soc. A*, vol. 378, no. 2166, pp. 1–31, 2020. [Online]. Available: https://royalsocietypublishing.org/doi/10.1098/rsta.2019.0056
6. L. C. McInnes, M. A. Heroux, E. W. Draeger, A. Siegel, S. Coghlan, and K. Antypas, "How community software ecosystems can unlock the potential of exascale computing," *Nature Comput. Sci.*, vol. 1, pp. 92–94, Feb. 2021.

RICHARD GERBER is the senior science advisor at the National Energy Research Scientific Computing Center at Lawrence Berkeley National Laboratory, Berkeley, CA, 94720, USA, and also serves as director of hardware and integration for the U.S. Department of Energy's Exascale Computing Project. Contact him at ragerber@lbl.gov.

STEVEN GOTTLIEB is a distinguished professor emeritus and provost professor emeritus of physics at Indiana University, Bloomington, IN, 47405, USA. Contact him at sg@iu.edu.

MICHAEL A. HEROUX is a senior scientist at Sandia National Laboratories, Albuquerque, NM, 87185, USA; the director of software technology for the U.S. Exascale Computing Project; and a scientist in residence at St. John's University, Collegeville, MN, 56321, USA. Contact him at maherou@sandia.gov.

LOIS CURFMAN MCINNES is a senior computational scientist and Argonne distinguished fellow in the Mathematics and Computer Science Division of Argonne National Laboratory, Lemont, IL, 60439, USA. She served as deputy director of software technology for ECP. Contact her at mcinnes@anl.gov.