Application Modernization for the Exascale Era

J. Robert Neely, Bronis R. de Supinski, and Charles H. Still | Lawrence Livermore National Laboratory

Application modernization is a term that succinctly embodies a challenge that any application developer in high-performance computing (HPC) ultimately must face if he or she is to continue riding the wave of increasing compute power unleashing continued scientific advancement through simulation. While application modernization can be pursued for any number of reasons, including updating to new language standards, incorporation of emerging technologies taking root in the broader community, or simply the desire to remove years of technical debt that can weigh down an application’s ability to evolve, we focus here on a specific need for application modernization around the sea change in HPC hardware taking place before our eyes. Application modernization involves an asset—in this case, HPC scientific applications—and a process, the active undertaking of modifications necessarily to avail emerging HPC hardware technology. The impending paradigm shift in HPC system architectures driven by technology concerns related to power, parallelism, and resilience cannot be completely insulated from the applications and codes that must effectively use these systems. If applications are to continue the pursuit of better science enabled by almost seven decades of performance gains enabled by Moore’s law, parallel scaling, stability in programming models, and countless hardware innovations, HPC developers must recognize that resting on their laurels and ignoring the need for potentially
significant code and algorithmic refactoring is a high-risk venture. They must in fact “modernize,” a term that indicates a level of effort somewhere on a long continuum that embodies more than simply porting, yet shy of completely rewriting from scratch.

The process of co-design was born out of the realization that exascale performance will require broader coordination within the HPC ecosystem. Technologies such as massively multicore CPUs and heterogeneous architectures answer the need for hardware speed but require application developers to handle the commensurate complexity of massive fine-grained parallelism. Developers must also consider that the memory wall—how CPU speed increases have far outpaced the performance of memory—has finally manifested itself as data motion replaces floating-point performance as the most common application bottleneck. Therefore, application developers must now deeply understand diverse architectures, while architects of hardware, the supporting software stack, and programming language standards must deliver innovations that ease this transition. The relatively stable architectures and programming models of the past 25 years in HPC allowed application developers and users to focus primarily on scientific discovery and less on the disruptive churn caused by continuous change in the underlying technology. The end of this trend could require major changes in these often large, complex, and validated applications that divert these users from their primary purpose.

This issue highlights three approaches being taken within the US Department of Energy (DOE) at Lawrence Livermore National Laboratory (LLNL), Los Alamos National Laboratory (LANL), and Lawrence Berkeley National Laboratory (LBNL). Each of these facilities houses world-leading supercomputing facilities as well as a cadre of application developers who use those facilities for scientific advancement in the pursuit of both national security and open science goals through the use of HPC simulation. LLNL and LANL are part of the DOE’s National Nuclear Security Administration (NNSA) and perform their work under the aegis of the Advanced Simulation and Computing (ASC) program focused on stockpile stewardship, as well as a portfolio of basic science applications. LBNL is home to the National Energy Research Scientific Computing Center (NERSC), a large DOE user facility with thousands of users from across the US and world who pursue accelerated scientific discovery in support of energy, physics, biological, and data science applications. Each of these sites is undertaking application modernization efforts as a result of recent and soon-to-come procurements of HPC systems that are challenging their application teams with new approaches, including heterogeneous systems involving general-purpose graphical processing units (GPGPUs) as accelerators and manycore systems based on the Intel Xeon Phi. Recognizing the challenge, these DOE sites now include their vendor partners in tight collaborations that far exceed the standard partnership of deploying and supporting hardware to include co-design activities that team lab scientists with vendor application experts long in advance of platform delivery to ensure applications hit the ground running upon deployment.

In “Application Modernization at LLNL and the Sierra Center of Excellence,” the authors outline the process they’re undertaking in a new Center of Excellence (CoE) for the Sierra system, which will be delivered by IBM in 2017–2018 and consist of Power9 CPUs and Nvidia Volta GPUs. The Sierra CoE at LLNL supports a wide range of classified and unclassified applications. This article focuses on the CoE’s engagement process as a template for other HPC programs to integrate application teams and architecture.

In “The Trinity Center of Excellence Co-Design Best Practices,” the authors focus on the process used to stand up their CoE to address the Trinity system delivered in 2015–2016 by Cray and consisting of a mix of Xeon and Xeon Phi Knights Landing CPUs. Like the first article, the authors define an emerging set of best practices for co-design through the focused efforts of a CoE.

Finally, in “Toward Exascale Earthquake Ground Motion Simulations for Near-Fault Engineering Analysis,” the authors talk about a specific application modernization effort to optimize seismic simulations for the NERSC Cori system that was delivered in 2015–2016 along with LANL’s Trinity system under the Alliance for Application Performance at Extreme Scale (APEX). The article focuses the use of the SW4 seismic code on the Phase 2 CORI deployment that consists of Intel Xeon Phi nodes. SW4 is undergoing modernization with support from NESAP, the NERSC Center of Excellence, and the NERSC Exascale Science Application Program.
While these three leading DOE sites customized their approaches around their specific needs and vendor partnerships, several commonalities in their processes naturally emerge. From the development of proxy applications to rapidly explore algorithmic and programming model changes and co-design of hardware and software features between the vendors and application teams well in advance of delivery to early access to hardware and software stacks through dedicated hands-on activities with the vendors, these articles provide others with a starting point for their own roadmap to their application modernization efforts.

Acknowledgments
This work was prepared by LLNL under contract DE-AC52-07NA27344. This document was prepared as an account of work sponsored by an agency of the United States government. Neither the US government nor Lawrence Livermore National Security nor any of their employees makes any warranty, expressed or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the US government or Lawrence Livermore National Security. The views and opinions of authors expressed herein do not necessarily state or reflect those of the US government or Lawrence Livermore National Security and shall not be used for advertising or product endorsement purposes.

J. Robert Neely is a Weapons Simulation & Computing Program Coordinator for Computing Environments at Lawrence Livermore National Laboratory. Contact him at neely4@llnl.gov.

Bronis R. de Supinski is CTO for Livermore Computing at Lawrence Livermore National Laboratory. Contact him at bronis@llnl.gov.

Charles H. Still is the Leader for Application Strategy within Livermore Computing at the Lawrence Livermore National Laboratory. Contact him at still1@llnl.gov.