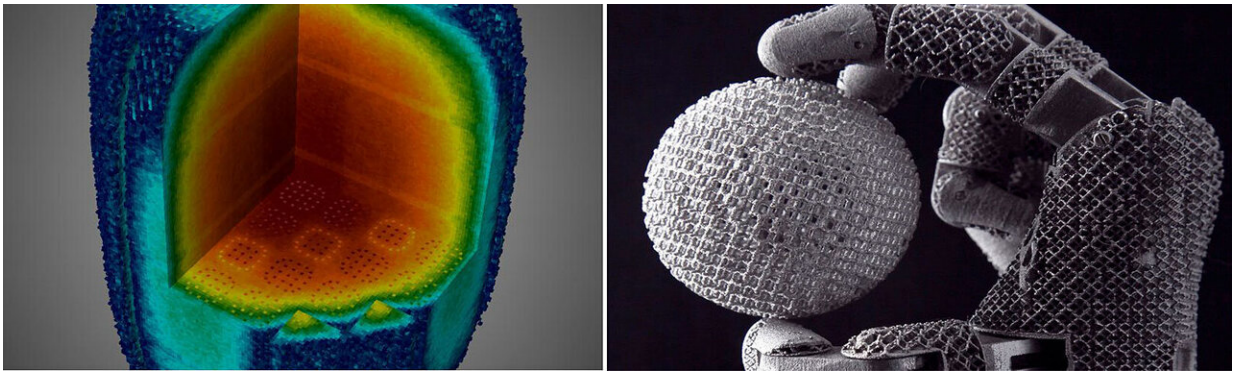
Preparing scientific applications for exascale computing

June 11 2019, by Ariana Tantillo



Exascale computers will be used to solve problems in a wide range of scientific applications, including to simulate the lifetime operations of small modular nuclear reactors (left) and to understand the complex relationship between 3-D printing processes and material properties (right). Credit: Oak Ridge National Lab

Exascale computers are soon expected to debut, including Frontier at the U.S. Department of Energy's (DOE) Oak Ridge Leadership Computing Facility (OLCF) and Aurora at the Argonne Leadership Computing Facility (ALCF), both DOE Office of Science User Facilities, in 2021. These next-generation computing systems are projected to surpass the speed of today's most powerful supercomputers by five to 10 times. This performance boost will enable scientists to tackle problems that are otherwise unsolvable in terms of their complexity and computation time.

But reaching such a high level of performance will require software adaptations. For example, OpenMP—the standard application programming interfaces for shared-memory parallel computing, or the use of multiple processors to complete a task—will have to evolve to support the layering of different memories, hardware accelerators such as graphics processing units (GPUs), various exascale computing architectures, and the latest standards for C++ and other programming languages.

Evolving OpenMP toward exascale with the SOLLVE project

In September 2016, the DOE Exascale Computing Project (ECP) funded a software development project called SOLLVE (for Scaling OpenMP via Low-Level Virtual Machine for Exascale Performance and Portability) to help with this transition. The SOLLVE project team—led by DOE's Brookhaven National Laboratory and consisting of collaborators from DOE's Argonne, Lawrence Livermore, and Oak Ridge National Labs, and Georgia Tech—has been designing, implementing, and standardizing key OpenMP functionalities that ECP application developers have identified as important.

Driven by SOLLVE and sponsored by ECP, Brookhaven Lab's Computational Science Initiative (CSI) hosted a four-day OpenMP hackathon from April 29 to May 2, jointly organized with Oak Ridge and IBM. The OpenMP hackathon is the latest in a series of hackathons offered by CSI, including those focusing on NVIDIA GPUs and Intel Xeon Phi many-core processors.

"OpenMP is undergoing substantial changes to address the requirements of upcoming exascale computing systems," said local event coordinator Martin Kong, a computational scientist in CSI's Computer Science and

Mathematics Group and the Brookhaven Lab representative on the OpenMP Architecture Review Board, which oversees the OpenMP standard specification. "Porting scientific codes to the new exascale hardware and architectures will be a grand challenge. The main motivation of this hackathon is application engagement—to interact more deeply with different users, especially those from DOE labs, and make them aware of the changes they should expect in OpenMP and how these changes can benefit their scientific applications."

The Summit supercomputer. Credit: Oak Ridge National Lab

Laying the foundation for application performance portability

Computational and domain scientists, code developers, and computing hardware experts from Brookhaven, Argonne, Lawrence Berkeley,

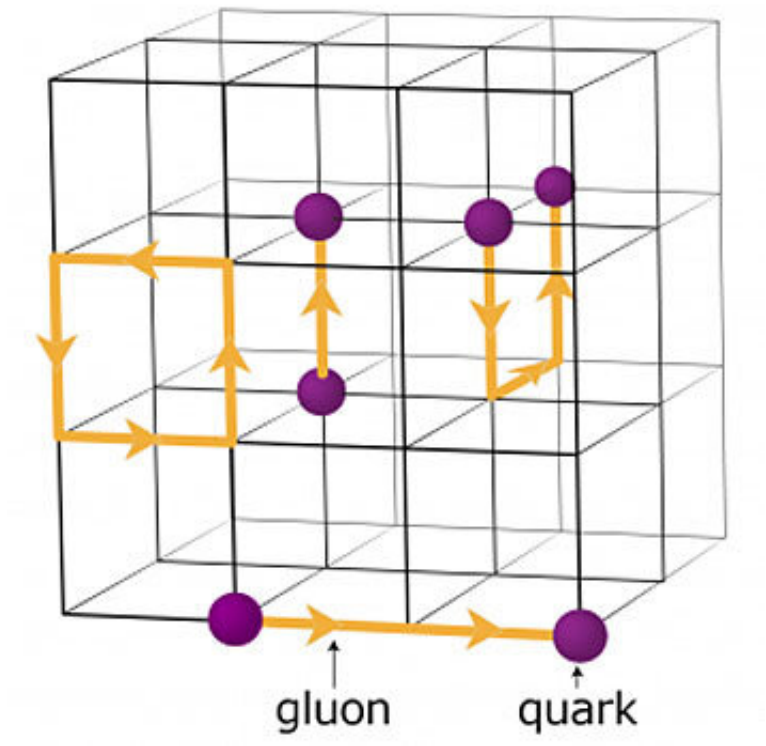
Lawrence Livermore, Oak Ridge, Georgia Tech, Indiana University, Rice University, University of Illinois at Urbana-Champaign, IBM, and the National Aeronautics and Space Administration (NASA) participated in the hackathon. The eight teams were guided by national lab, university, and industry mentors who were selected based on their extensive experience in programming GPUs, participating in the OpenMP Language Committee, and conducting research and development in tools that support the latest OpenMP specifications.

Throughout the week, the teams worked on porting their scientific applications from central processing units (CPU) to GPUs and optimizing them using the latest OpenMP version (4.5+). In between hacking sessions, the teams had tutorials on various advanced OpenMP features, including accelerator programming, profiling tools to assess performance, and application optimization strategies.

Some teams also used the latest OpenMP functionalities to program IBM Power9 CPUs accelerated with NDIVIA GPUs. The world's fastest supercomputer—the Summit supercomputer at OLCF—is based on this new architecture, with more than 9000 IBM Power9 CPUs and more than 27,000 NVIDIA GPUs.

Taking steps toward exascale

The teams' applications spanned many areas, including nuclear and high-energy physics, lasers and optics, materials science, autonomous systems, and fluid mechanics.



A schematic of the lattice for quantum chromodynamics calculations. The intersection points on the grid represent quark values, while the lines between them represent gluon values. Credit: Brookhaven National Laboratory

Participant David Wagner of the NASA Langley Research Center High Performance Computing Incubator and colleagues Gabriele Jost and Daniel Kokron of the NASA Ames Research Center came with a code for simulating elasticity. Their goal at the hackathon was to increase single-instruction, multiple-data (SIMD) parallelism—a type of computing in which multiple processors perform the same operation on many data points simultaneously—and optimize the speed at which data can be read from and stored into memory.

"Scientists at NASA are trying to understand how and why aircraft and spacecraft materials fail," said Wagner. "We need to make sure that these materials are durable enough to withstand all of the forces that are

present in normal use during service. At the hackathon, we're working on a mini app that is representative of the most computationally intensive parts of the larger program to model what happens physically when the materials are loaded, bent, and stretched. Our code has lots of little formulas that need to run billions of times over. The challenge is performing all of the calculations really fast."

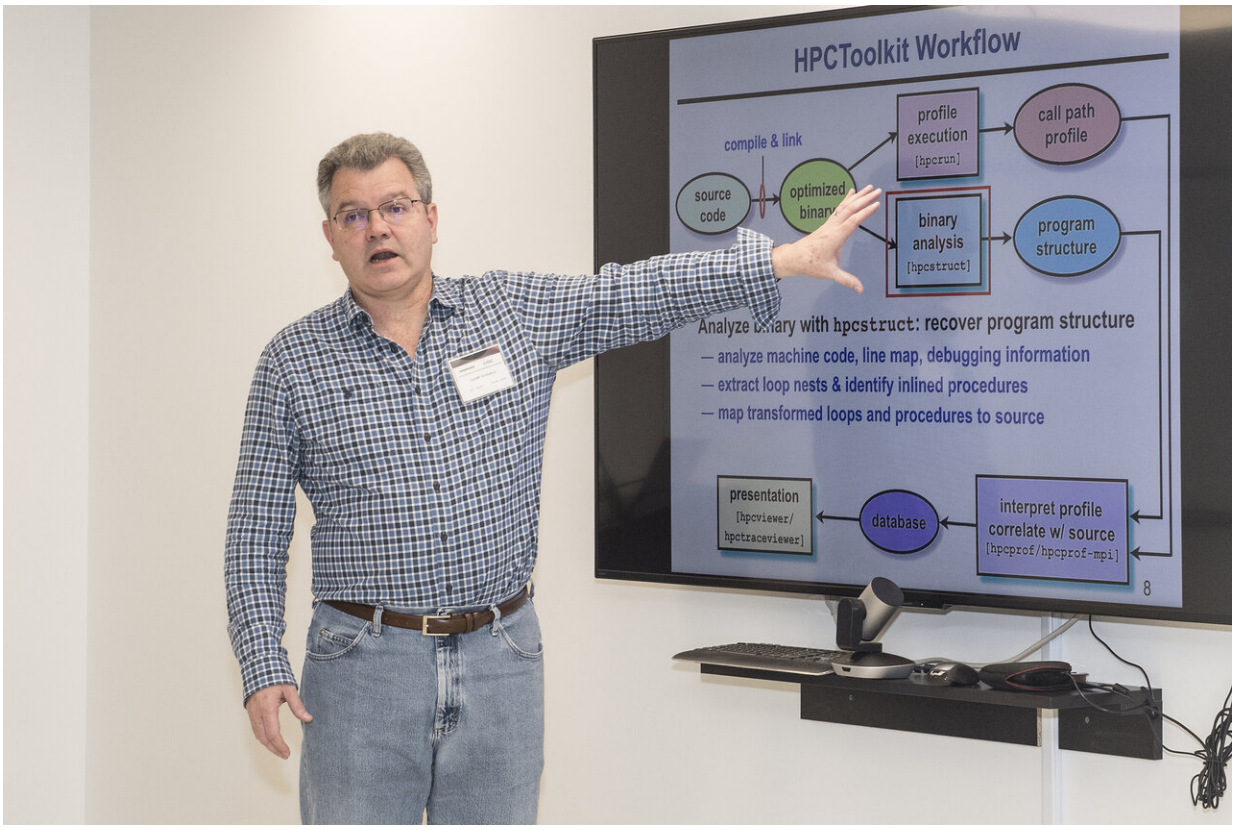
According to Wagner, one of the reasons NASA is pushing for this computational capability now is to understand the processes used to generate additively manufactured (3-D printed) parts and the different material properties of these parts, which are increasingly being used in aircraft. Knowing this information is important to ensuring the safety, reliability, and durability of the materials over their operational lifetimes.

"The hackathon was a success for us," said Wagner. "We got our code set up for massively parallel execution and running correctly on GPU hardware. We'll continue with debugging and parallel performance tuning, as we expect to have suitable NASA hardware and software available soon."

Another team took a similar approach in trying to get OpenMP to work for a small portion of their code, a lattice quantum chromodynamics (QCD) code that is at the center of an ECP project called Lattice QCD: Lattice Quantum Chromodynamics for Exascale. Lattice QCD is a numerical framework for simulating the strong interactions between elementary particles called quarks and gluons. Such simulations are important to many high-energy and nuclear physics problems. Typical simulations require months of running on supercomputers.

"We would like our code to run on different exascale architectures," said team member and computational scientist Meifeng Lin, deputy group lead of CSI's new Quantum Computing Group and local coordinator of

previous hackathons. "Right now, the code runs on NVIDIA GPUs but upcoming exascale computers are expected to have at least two different architectures. We hope that by using OpenMP, which is supported by major hardware vendors, we will be able to more easily port our code to these emerging platforms. We spent the first two days of the hackathon trying to get OpenMP to offload code from CPU to GPU across the entire library, without much success."



John Mellor-Crummey gives a presentation about the HPCToolkit, an integrated suite of tools for measuring and analyzing program performance on systems ranging from desktops to supercomputers. Credit: Brookhaven National Laboratory

Mentor Lingda Li, a CSI research associate and a member of the SOLLVE project, helped Lin and fellow team member Chulwoo Jung, a physicist in Brookhaven's High-Energy Theory Group, with the OpenMP offloading.

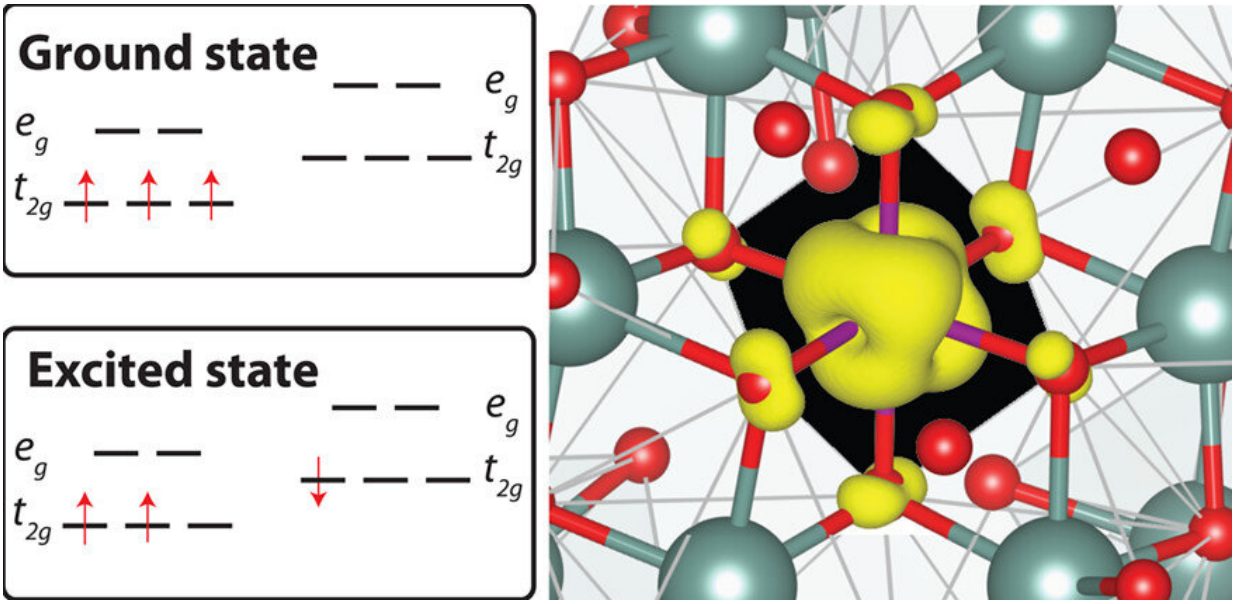
Though the team was able to get OpenMP to work with a few hundred lines of code, its initial performance was poor. They used various performance profiling tools to determine what was causing the slowdown. With this information, they were able to make foundational progress in their overall optimization strategy, including solving problems related to initial GPU offloading and simplifying data mapping.

Among the profiling tools available to teams at the hackathon was one developed by Rice University and University of Wisconsin.

"Our tool measures the performance of GPU-accelerated codes both on the host and the GPU," said John Mellor-Crummey, professor of computer science and electrical and computer engineering at Rice University and the principal investigator on the corresponding ECP project Extending HPCToolkit to Measure and Analyze Code Performance on Exascale Platforms. "We've been using it on several simulation codes this week to look at the relative performance of computation and data movement in and out of GPUs. We can tell not only how long a code is running but also how many instructions were executed and whether the execution was at full speed or stalled, and if stalled, why. We also identified mapping problems with the compiler information that associates machine code and source code."

Other mentors from IBM were on hand to show the teams how to use IBM XL compilers—which are designed to exploit the full power of IBM Power processors—and help them through any issues they encountered.

"Compilers are tools that scientists use to translate their scientific software into code that can be read by hardware, by the largest supercomputers in the world—Summit and Sierra [at Lawrence Livermore]," said Doru Bercea, a research staff member in the Advanced Compiler Technologies Group at the IBM TJ Watson Research Center. "The hackathon provides us with an opportunity to discuss compiler design decisions to get OpenMP to work better for scientists."



QMCPack can be used to calculate the ground and excited state energies of localized defects in insulators and semiconductors—for example, in manganese $(Mn)^{4+}$-doped phosphors, which are promising materials for improving the color quality and luminosity of white-light-emitting diodes. Credit: Brookhaven National Laboratory

According to mentor Johannes Doerfert, a postdoctoral scholar at ALCF, the applications the teams brought to the hackathon were at various

stages in terms of their readiness for upcoming computing systems.

"Some teams are facing porting problems, some are struggling with the compilers, and some have application performance issues," explained Doerfert. "As mentors, we receive questions coming from anywhere in this large spectrum."

Some of the other scientific applications that teams brought include a code (pf3d) for simulating the interactions between high-intensity lasers and plasma (ionized gas) in experiments at Lawrence Livermore's National Ignition Facility, and a code for calculating the electronic structure of atoms, molecules, and solids (QMCPack, also an ECP project). Another ECP team brought a portable programming environment (RAJA) for the C++ programming language.

"We're developing a high-level abstraction called RAJA so people can use whatever hardware or software frameworks are available on the backend of their computer systems," said mentor Tom Scogland, a postdoctoral scholar in the Center for Applied Scientific Computing at Lawrence Livermore. "RAJA mainly targets OpenMP on the host and CUDA [another parallel computing programming model] on the backend. But we want RAJA to work with other programming models on the backend, including OpenMP."

"The theme of the hackathon was OpenMP 4.5+, an evolving and not fully mature version," explained Kong. "The teams left with a better understanding of the new OpenMP features, knowledge about the new tools that are becoming available on Summit, and a roadmap to follow in the long term."

"I learned a number of things about OpenMP 4.5," said pf3d team member Steve Langer, a computational physicist at Lawrence Livermore. "The biggest benefit was the discussions with mentors and

IBM employees. I now know how to package my OpenMP offload directives to use NVIDIA GPUs without running into memory limitations."

Provided by Brookhaven National Laboratory