DRAFT
Advanced Scientific Computing Advisory Committee Meeting
Argonne National Laboratory, Argonne, Illinois
November 9-10, 2010

ASCAC members present:

Marsha Berger
Roscoe Giles, Chair
Susan Graham
James Hack
Anthony Hey
Thomas Manteuffel
John Negele (by telephone)
Linda Petzold
Larry Smarr
William Tang
Victoria White

ASCAC members absent:

Jacqueline Chen
Jack Dongarra
Vivek Sarkar

Also participating:

Pavan Balaji, Mathematics and Computer Science Division, Argonne National Laboratory
Christine Chalk, ASCAC Designated Federal Officer, Office of Advanced Scientific Computing Research, Office of Science, USDOE
Jeffrey Hammond, Leadership Computing Facility, Argonne National Laboratory
Daniel Hitchcock, Acting Associate Director, Office of Advanced Scientific Computing Research, Office of Science, USDOE
Paul Messina (by telephone), Director (Ret.), Center for Advanced Computing Research, California Institute of Technology
Misun Min, Mathematics and Computer Science Division, Argonne National Laboratory
Boyana Norris, Mathematics and Computer Science Division, Argonne National Laboratory
Frederick O'Hara, ASCAC Recording Secretary
Michael Papka, Computing, Environment, and Life Sciences Division, Argonne National Laboratory
Jeannie Robinson, Oak Ridge Institute for Science and Education
Robert Rosner, Enrico Fermi Institute, University of Chicago
Robert Ross, Mathematics and Computer Science Division, Argonne National Laboratory
Rachel Smith, Oak Ridge Institute for Science and Education
Rick Stevens, Associate Laboratory Director for Computing, Environment, and Life Sciences, Argonne National Laboratory

About 40 others were in attendance during the course of the two-day meeting.

Tuesday, November 9, 2010

The meeting was called to order at 8:37 a.m. by the chair, **Roscoe Giles**. He thanked the staff of Argonne National Laboratory for their hospitality. Jeanie Robinson made safety and convenience announcements.

Daniel Hitchcock, Acting Associate Director, was asked to give an update on the activities of the Office of Advanced Scientific Computing Research (ASCR).

Because of the recent election, there has been no action on the federal budget. The current continuing resolution will go through at least mid-February 2011. There are a lot of uncertainties. No funding opportunity announcements or new starts are allowed under the continuing resolution.

China went to the top of the Top500 list with a hybrid architecture employing Intel and Nvidia chips and a Chinese-developed interconnect that delivered double InfiniBand performance. The scale and the

stability demonstrated added up to a substantial accomplishment. They have aggressive plans for Chinese-made processors as well as interconnect. This is the first big hybrid system.

In staffing, Hitchcock has been named as the Facilities Division Director. Offers have been made for three open positions. The opening of the division director's position has been announced.

Warren Washington received the National Medal of Science, James Demmell won this year's Sidney Fernbach Award, and Alexandre Chorin and James Sethian won prestigious prizes from the International Council for Industrial and Applied Mathematics.

There were several scientific accomplishments of note. Heat transport in plasmas has a new mathematical approach for modeling and simulation of plasma confinement. This is the first simulation of the fractal structure of the temperature profile in a weakly chaotic field and the first simulation on the effective radial transport from nonlocal parallel transport fully chaotic magnetic fields.

Automatic performance tuning has achieved several techniques that work well.

In the American Recovery and Reinvestment Act (ARRA) projects, the Leadership Computer Facility (LCF) upgrades were completed on time and on budget. The Advanced Network Initiative research projects on a 100-Gbs optical network are under way. The advanced computer architecture projects are on track. Magellan is demonstrating cloud computing for mid-range computational science; it is turning out to be costly.

Scientific Discovery through Advanced Computing–Extended (SciDAC-e) is advancing the high-performance computational capabilities of the Office of Basic Energy Sciences Energy Frontier Research Centers. Bottlenecks have been solved. The SciDAC-e milestone to convene an expert panel to review its activities will not be achieved because money was not made available in time. No results will be available for review this calendar year. ASCAC will be asked to review this project next spring.

Exascale research will be kicked off at a principal investigator (PI) meeting March 7–11, 2011, on advanced architecture, X.-stack, scientific data management and analysis, and codesign.

This is the second year for the Early Career Award Program. The topic descriptions were tightened this year to ensure that proposals were not submitted for topics that could not be funded. The solicitation was issued July 1, pre-applications were due August 13, full applications were due November 9, and panel reviews will be held January 11–13.

The recent committee of visitors (COV) report said that more data management is needed in the portfolio. A funding opportunity announcement (FOA) was issued in scientific data management and analysis for about $5 million per year on the topics of file systems and input/output (FSIO) and storage, triage and analysis, integration, knowledge representation (KR) and machine reasoning, and visual analysis. Thirty-seven proposals were received, asking for about $22 million per year. Ten were funded in dynamic non-hierarchical file systems for exascale storage, a runtime system for input/output (I/O) staging, bringing exascale I/O within science's reach, adding data management services to parallel file systems, scalable and power-efficient data analytics for hybrid exascale systems, a pervasive parallel processing framework for data visualization and analysis at the extreme scale, and information-theoretic framework for enabling extreme-scale science discovery, enabling scientific discovery in exascale simulations, graph-based 3-D flow field visual analysis, and topology-based visualization and analysis of multidimensional data and time-varying data at the extreme scale.

It is expected that SciDAC will be re-competed. Institutes will be addressed first with an announcement in January or February, but that timing will depend on the continuing resolution. After the institutes are dealt with, the programs will be addressed; there will be an announcement during the summer. A strategic realignment will be sought.

A phase change is going on in computing. Everything will go to concurrency. Today the typical server node chip has about eight cores, and a 1000-note cluster has about 8000 cores. A laptop has about two cores at the low-power end of the spectrum. By 2020, it is expected that the typical server node chip will have about 400 cores, and a 1000-node cluster will have about 400,000 cores. A typical laptop will have about 100 cores at the low-power end of the spectrum. To achieve this change will require codesign, the X-stack, advanced architecture, and data management and analysis at the extreme scale. In the new

regime, moving data is more difficult than carrying out operations. Replacing the Message-Passing Interface (MPI) would be very painful.

Hey asked if there were any reports on the Magellan program. Hitchcock replied that several open-source programs were being evaluated. A report will be made at the Supercomputing 2010 (SC-10) meeting. September 30 is the program report date.

Smarr noted that, at billion-core scales, there will likely also be a phase shift from classic peer architecture to more biological approaches. Not enough out-of-the-box thinking is being invested in. There will not be enough time to understand the change. He asked what DOE was doing in this area. Hitchcock answered that the Office was talking with vendors about disruptive technologies they might produce (e.g., nonvolatile memory on the nodes; self-healing networks, which is already evolving; and transactional memory). A lot will not be manufacturable until the mid-2020s. The current roadmap for memory is not sustainable from a power perspective.

Manteuffel asked what the "strategic realignment" plans were for the SciDAC recompetition. Hitchcock responded that institutes would be the place to go to get software and help. The project topics would be more strategically selected to have the most impact.

Tang asked if there were any thoughts of collaborations with the Chinese. Hitchcock answered that there have not been any talks about collaborations, but there may be. There are significant intellectual-property issues. Tang asked whether all the codesign people who got funding would be invited to the PI meeting. Hitchcock answered that all of the researchers will be invited to the meeting to see what they have to offer, and then it will be seen what can be funded.

Berger asked, given the continuing resolution, what the Office's priorities will be. Hitchcock replied that there will be a lot of influence on those decisions from higher management. The Office will try to keep the core going; but to be funded, work will have to have pertinence to the future. Both FY11 and FY12 will be influenced by the new Congress. Giles added that this Committee will try to have an impact and input, also. He asked if there were any interactions with the National Science Foundation (NSF). Hitchcock responded, yes, there was interaction with individual program managers. NSF has a lot of new personnel and vacancies, and it does not have the same mission as DOE. An attempt is made to see where the overlaps are and to work with NSF on those overlapping topics.

Robert Rosner was asked to report on the ASCAC Exascale Subcommittee. The Subcommittee took disciplinary workshop reports and tried to decide if the next step in computing growth could be taken. The draft report has resulted from feedback from ASCAC and many online comments. It is now entirely defensible. The question about the exascale matters in terms of competitiveness. China has an industrial policy; the United States does not. China seeks to be a world industrial power, and it is being successful. It is now the leader in computing. For the United States to ignore this fact would be devastating. The question is, can the United States do something that is transformative and that really matters.

The Subcommittee focused on applications with high visibility, such as national security, and that are key to U.S. competitiveness, such as aerospace. DOE is the lead agency in going to the exascale.

The conclusions of the report are that the world is about to enter a new era in industrial competition. There were applications that were hungry for transformative changes in capability. Going to the exascale is not just a matter of improving, but of reinventing the technology and allowing the total rethinking of how disciplines (like hydrodynamics) are done. Those working on the algorithms each know what those working on hardware are doing (codesign). It can work. Computers are built because one wants to compute something that matters. The software has to be of interest to society. Codesign centers are needed.

The case has been made for the exascale. There are numerous applications where the exascale will make a difference. There are challenges.

Giles said that ASCAC has been asking for a charge like this since its inception.

Smarr noted that some areas will need to be rethought from the ground up. Rosner asked rhetorically: Where have people gotten to in first-principle calculations? That is a huge challenge, as is plasma physics. There will be transformations in these areas because of the exascale.

Graham stated that the report has come a long way since the past meeting. However, it offers no strategy for getting to the exascale. She asked if the milestones were left for another subcommittee. Rosner responded that the Subcommittee was not set up to develop a roadmap. All the players have to be involved in developing such a roadmap.

Hey agreed that this draft was a great improvement. He was intrigued by the terms "algorithmic challenges" and, elsewhere, the statement that "incremental changes" were needed. He asked whether these statements were contradictory. Rosner answered that one can predict incremental changes but not necessarily transformative changes, where unpredictable challenges may arise.

Tang said that this case was quite compelling. The Subcommittee should lead with the science. Also, the United States cannot expect the Chinese to sit back on their haunches; they will continue to move forward. An example of a method that is very promising is the particle-and-cell method. Algorithmic treatments will need to move forward. Rosner pointed to a calculation done at Lawrence Livermore National Laboratory (LLNL) that took an alternative path to the Navier-Stokes method. To get the rollup right, one has to get a lot of first-principal predictions right.

Manteuffel stated that what one can do with this machine is important to justify going forward. The software has to be kept in pace with the hardware to do science. Rosner added that there are a lot of ways to look at how to go forward. One way is the National Nuclear Security Administration (NNSA) program. There is the invention of a new probe that would run on the hardware. There is a hardware issue. Software has to produce a correct result, requiring experimental data to test it against. Within the past 15 years, it has been recognized that there is a need for an experimental program to back up the validation program. ASCR does not have the funds for an experimental program; that component has to be provided by the disciplines.

Negele said that he was happy to see the case made in the Executive Summary. He said that the success of quantum chromodynamics (QCD) should have been included and that the explanation of subatomic particles in the nuclei from first principles should have been described. Rosner replied that the Subcommittee got a lot of requests like that but that the report was already too long. The Subcommittee, therefore, referred to the workshop reports where those advancements were described. The report focused on the question, "Should we do this?" Giles stated that the theme of the impact of science came through. Negele applauded the Subcommittee for its work and report.

Messina commented that the application groups should be involved from the beginning. The applied mathematics and algorithm groups should also be involved from the beginning. There are opportunities for their making an impact all the way down to the instruction-set scale.

Berger noted that, in the report, explicit time-stepping was highlighted but the scaling was not there. The applied-mathematics component will be a great challenge. One should not just throw money at hardware but take a more-measured approach. Rosner said that that is absolutely correct, but that the charge to go ahead is needed first.

Giles asked the Committee members if they were happy with the Executive Summary and its five sentences "findings."

Petzold said that the report is much improved. She was disappointed with the algorithms section. Algorithm development is intimately involved with the science. Algorithms need to be codesigned with the science, and that case should be made better.

Smarr referred to in the second bullet in the Executive Summary and said that it is not just science or technology but also plasma physics. He asked whether the Chinese had put forward what the grand challenges are and pointed out that the United States is making a huge investment in genome sequencing. Rosner answered that, in Shanghai 2 years ago, it was made clear that the general category of national security is a priority for them (i.e., design codes for nuclear weapons). It is ironic that the report does not say anything about the need for codesign on hardware and algorithms. A small change could be made in the report about that issue. Graham said that the role of hardware could be discussed in that rewrite, also.

Hack commented that this report has come a long way. It is not comprehensive in all the science applications, but a 700-page report is not wanted. It strikes a good balance. Some words about algorithms would be appreciated.

White said that it is unfortunate that all the successes of the past have not been described. One elevator-talk point that is missing is that the scientists stand ready to work on this transformational approach because they will otherwise be unable to do the science. That case has to be made in this report. Rosner agreed. However, this report is not where that point should be made. ASCR has to make that case; they have the evidence. Messina added that a good example of scientists taking the time to develop the hardware is Norman Christ. White still thought that some recognition of the point should be included in the postcard version. It speaks to the viability of the roadmap.

A break was declared at 10:19 a.m. to allow the rewriting of the report. The meeting was called back into session at 10:41 a.m. During the break, changes focused on the Executive Summary were suggested, and those changes were presented to the Committee. Rosner was comfortable with the changes. Giles moved the acceptance of the report, as modified. Berger seconded. Negele asked for a reading of the changes, which was done. There being no further discussion, a show of hands and voices showed that the motion passed unanimously. Giles thanked Rosner and the Subcommittee for their hard work and asked that the report be distributed at SC-10.

Michael Papka was asked to provide an update on computing at Argonne National Laboratory (ANL) and the Argonne Leadership Computing Facility (ALCF).

The Laboratory has a rich mix of computing resources: 504 computer servers, storage service with 200 nodes, and memory resources. The Computational Institute has the Petascale Active Data Server (PADS), and the NSF Beagle supercomputer for biomedical research is coming by the end of the year.

ACLF-2 is the coming machine. The early science program is already started; the Cray machine is to be delivered in early 2011; the Blue Gene/Q will operate at 10 petaflops. Early-science proposals were recently awarded with 75 participants. They will use ALCF-1 resources to begin with and then will shake out the new machine with 16 projects (climate, materials, astrophysics, etc.). These projects underwent a computational-readiness review as well as a scientific review.

Giles asked how much time had been allocated. Papka replied that the codes need to be ready to go when the Q arrives, so 20% of the Blue Gene/P's time has been allocated. Allocations on Blue Gene/Q will parallel those on Blue Gene/P.

Christine Chalk was asked to present the Office's response to the Applied Mathematics COV report. The report, presented to ASCAC in May, was very helpful. These COV charges will be routinely issued each fall. They will (1) look at the processes involved in solicitation/review/recommendation proposal actions, (2) monitor active awards, and (3) examine the portfolio and its balance.

In the proposal-review process, the Applied Mathematics COV's first recommendation was to improve the level of outreach for new funding opportunities. The Office is developing lists of individuals to contact. The second recommendation was to limit proposals to 15 pages. Applied Mathematics' targeted solicitations will limit project descriptions to 15 pages. The third recommendation was that large multi-investigator proposals should include an evaluation that ensures that the elements of the supposed research are appropriately integrated, coordinated, and synergistic. ASCR will evaluate the integration, coordination, and/or synergy of the proposed research within the DOE merit-review criteria. The fourth recommendation was to accelerate the processing of approved grants. The time to award has been improved significantly.

On monitoring active awards, the COV recommended that explicit guidelines should be instituted for progress reports. Explicit guidelines for progress reports will be instituted for all Applied Mathematics projects.

In breadth and depth of the portfolio, the COV found the portfolio to be exceptionally strong with regard to both depth and breadth. The program was found to be of extremely high quality and standing, both nationally and internationally. A great strength of the program is its willingness to invest in projects with a longer-term perspective than is possible with most U.S. agencies, enabling the support of breakthrough research and ensuring success and eventual adoption.

The Office greatly appreciates the COV's careful evaluation of the Applied Mathematics research program. Even the Office of Management and Budget (OMB) reviewer reads and acts upon these findings.

A new charge from the Director of the Office of Science was presented to the Committee for a COV for next-generation networking and science. The report of the COV will be due in August 2011. Giles pointed out that this charge has been e-mailed to the entire Committee. Suggestions for a chair from ASCAC would be welcome.

Hey was pleased with the progress on the grant funding.

Giles stated that this seemed like a great response and thanked the Office.

Pavan Balaji was asked to discuss runtime infrastructure for exascale computing. This work is being done cooperatively by workers at ANL, Oak Ridge National Laboratory, Sandia National Laboratories, Lawrence Berkeley National Laboratory, Pacific Northwest Laboratory, and Lawrence Livermore National Laboratory.

Current programming models fall into three categories: high-level compilers, high-level libraries, and low-level runtime systems. Each model provides its own capabilities and challenges. Applications have relied mostly on MPI. Some use alternative models, such as global arrays or Unified Parallel C (UPC). Each has strengths and weaknesses. One does not want to throw these away but to develop ways for them to work together (multi-model programming), primarily based on MPI but using UPC or Coarray Fortran (CAF).

The exascale era will require more complexity in data, software, and hardware. Current computational and communication systems will not be able to migrate to the new systems. N-body coulomb interactions will move from 20–100 atoms to thousands of atoms, requiring more computation, combining large and small interactions that produce areas of both dense and sparse data and computation. Another application example is Green's Function Monte Carlo (GFMC) ab initio calculations in nuclear physics, a nontrivial master/slave algorithm scaled to 2000 processors on Blue Gene/L. However, the memory needed will grow as an exponential of the atomic weight of the nucleus being studied.

Each application is packaged with its own high-level programming library. This practice is not sustainable at the exascale. The vision is a unified stack with communication libraries, threading routines, and hardware management that adds new components as they are needed. The fundamental concept is an integrated runtime infrastructure, high-level libraries, and high-level languages.

The six laboratories have already had meetings about what the unified stacks and programming infrastructure should look like. A white paper will be developed.

Technical challenges include memory consistency, computation management and load balancing, unified communication runtime and progress model, coordinated management of shared resources, hybridization and interoperability, heterogeneous/accelerator computing, memory hierarchy and threading, fault tolerance, and interaction with performance and debugger tools.

The challenges for the unified programming infrastructure include unified semantics, interoperability of data objects, and integrated resource management.

Currently, ad hoc interactions are being studied, like some form of MPI plus threads, MPI plus UPC, or Asynchronous Dynamic Load Balancing (ADLB) plus MPI. However, a more truly unified (drag-and-drop) model is needed as a migration path for applications to start using other models. A hybridized MPI and UPC is being tried on several applications. A combination of MPI and the Aggregate Remote Memory Copy Interface (ARMCI) is also being looked at.

In conclusion, several programming models are out there, but applications might need more than what each of the models provides. It might be time to allow applications to use multiples of these models together. The unified programming infrastructure is a community-wide effort to bring together the capabilities of virtually all of the existing programming models available today.

Graham noted that she did not hear the terms "error" or "fault." The probability of faults is high at the exascale. The unified infrastructure should provide some support for that. Balaji responded that that need was recognized. Some things can be looked at but not all things. It is hard for the system software to benchmark and check and restart a calculation.

Hey said that this is a worthy cause, but it is very difficult to check accuracy. Once one mixes these programs, one introduces the possibility of errors. Balaji replied that the uniform stack will produce some bugs and interaction bugs. That problem has been identified to be studied.

Giles asked what the benefits of this effort were. Balaji answered that the investment already made should be utilized, and preparations need to be made for the exascale.

Tang suggested, as one puts a new paradigm into use, cross-benchmarking for an application to show that the new codes can deliver science similar to that of older, validated ones. Balaji said that they hoped to do so for about five applications. Some people are codesign leads and are involved in determining the requirements that would need to be met.

Giles asked what systems would host the early implementations. Balaji replied that four or five systems would be used, such as the Blue Gene, Cray, and medium-class systems.

The floor was opened to public comment. There being none, the meeting was adjourned for the day at 11:45 a.m. During the afternoon, the Committee toured the ALCF, Theory and Computing Science Center, Advanced Photon Source, and Center for Nanoscale Materials.

Wednesday, November 10, 2010

The meeting was called back into session by the chair, **Roscoe Giles**, at 8:14 a.m.

Christine Chalk was asked to discuss the selection of Joule-metric codes for FY 11. The Joule metric calls for ASCR to improve the computational science capabilities of a subset of application codes by simulating the same problem in less time or by simulating a larger problem in the same time. The metric is expressed as the average percentage increase in the computational effectiveness. The increase can be achieved by scaling up the code to run on a larger/faster machine or by enhancing or optimizing the code so it runs faster on the same machine, allowing larger problems to be run in the same time on the same machine. The FY11 goal is to average a 100% improvement in the computational efficiency of four selected applications: OMEN, LAMMPS, COMPASS, and STOMP. The original state of the code, machine, and problem is referred to as Q2; the improved state is referred to as Q4.

OMEN is a 1-D, 2-D, and 3-D quantum transport software based on the nearest-neighbor, tight-binding method and dedicated to the simulation of the next generation of transistors. LAMMPS is a classical electoral-dynamics code that models an ensemble of particles (from a few to billions) in a liquid, solid, or gaseous state. LAMMPS is a freely available open-source code that is distributed under the terms of the GNU [GNU's Not Unix] public license; in the past five years, it has been downloaded roughly 50,000 times and used to produce 250 publications. It is desired to port LAMMPS to hybrid architectures. COMPASS [Community Petascale for Accelerator Science and Simulation] is a computational infrastructure for accelerator ongoing and optimization. STOMP was added this year. It is a subsurface multi-fluid flow and reactive transport simulation capability for environmental restoration of legacy nuclear waste sites, geologic sequestration of carbon dioxide, and the development of alternative-energy resources.

These are the four codes that the Office is suggesting. All the metrics have four milestones. Identifying the codes is the first milestone.

Manteuffel asked how this works. Chalk replied that one team documents the code. They identify a target platform. A team is put together for code improvement. They then put out Q2 and Q4 reports. Everything along the way is documented.

Smarr said that this is a great way to ensure that the codes are developed. If a program were run on half of the Jaguar and then on all of the Jaguar, its performance would double. However, what happens if one uses all the resources of the biggest machine? Chalk replied that one would then try to run it in half the time or double its problem size. If one has a code and scales it to twice the number of processors or double the problem size, that counts. There have been teams that took a number of approaches that the scientists could relate to. Giles pointed out that these examples benefit the scientists in the long run *and* meet the external requirements.

Chalk pointed out that the Office does not financially support anyone to do this except the PI who owns the application. This is not a grant.

White asked what the penalty was for not meeting the goal. Chalk responded that ASCR has never missed the metric. It is possible for a team not to meet their metric but for the average performance of all teams to meet the metric.

Hack noted that, if one doubled OMEN's performance, it would be HPC performance, which would be spectacular. He asked if there were a well-thought-out process for selecting the codes. Chalk replied that there was, the relative risk is considered in choosing the codes.

Tang stated that this is an excellent program. It improves performance and efficiency, and it is on the record, giving the code credibility.

Manteuffel asked if teams lobby to get into this arena. Chalk responded that the process is very resource dependent. The selection team looks for codes, and some that can be improved are identified. The team rotates through disciplines, also. Hitchcock added that all lessons learned are used to build expertise and are applied to other codes. Smarr agreed; the biggest payoff is that one has exemplars that can be replicated. He asked whether there was any tracking of how the improvements were used. Hitchcock said that the Innovative and Novel Computational Impact on Theory and Experiment (INCITE) program has a computer-readiness review, so a code that has not been optimized gets rejected.

Giles asked if the Joule metrics were reviewed periodically. Chalk said that the definitions of the metrics evolve each year. It is not possible to completely replace the metric.

Giles stated that he would write a "sense of ASCAC" letter, saying that this process is fine.

Jeffrey Hammond was asked to speak on computational chemistry beyond the petascale.

Computational chemistry consists of classical molecular dynamics (MD) with empirical potentials, ab initio MD based upon density-function theory (DFT), and quantum chemistry with wave functions [e.g., coupled cluster (CC)]. Classical MD solves the Newton's equations of motion with empirical terms and classical electrostatics. It handles 100,000 to 10,000,000 atoms, and its scaling is proportional to the number of atoms. In ab initio molecular dynamics, the forces are obtained from solving an approximate single-particle Schrödinger equation. It handles 100 to 1000 atoms, and its scaling is proportional to 1 to 3 powers of the number of atoms. The coupled-cluster theory uses an infinite-order solution to a many-body Schrödinger equation truncated via clusters. It handles 10 to 100 atoms, and its scaling is proportional to 4 to 7 powers of the number of atoms. Today, one can only do five or six time steps; one cannot do dynamics with coupled-cluster theory.

Both classical and ab initio MDs have essentially reached algorithmic maturity. Most research is fighting Amdahl's law and related concepts (the fast Fourier transform does not scale). DEShaw has turned classical MD into an engineering problem. But quantum many-body methods are far from algorithmic maturity because they have been constrained to tiny systems. So the *N*-body problem is hidden behind dense linear algebra. Dense linear algebra is great for Gordon Bell prizes but terrible for science because one is still bound by algorithms.

With the GPU, everyone had a teraflop on the desktop. The computational chemistry community does not care unless they can afford hundreds of jobs per chemist. One cannot discriminate between the thousands of important questions that can be answered by computation. Chemists need a lot of capabilities across a lot of needs. The impact of heroic simulations is often psychological.

Two things happen with the exascale. The algorithms start to look familiar, and one can have a wealth of types of algorithms. If a lot of energy is pumped into biological systems to achieve high resolution, they can be destroyed. The probe di-8-ANEPPS has specific absorption peaks that can indicate how it is affected by the environment. But computations of the system produced wildly different results. Many-body methods were tuned to converge on experimental results (within 0.2 of an electron volt). This answer is not heroic, but it is still impossible with single-node codes because of the memory wall.

In graphitic materials, the polarization of polyacenes cannot be solved by electron density function theory but can be solved by many-body methods. Coupled-cluster single and double excitation (CCSD) has gone from impossible in 2000 to mundane in 2010. Automatic code generation was critical in implementing CCSD-LR [linear response] in parallel. Finally, in 2005, there was a machine that could hold everything in memory. In 2010, there are enough resources that an intern did not need to be afraid to burn millions of hours a year.

In calculating force fields from first principles, it was debatable whether dialanine represents the real torsional potential. Many force field potentials use Møller-Plesset second-order perturbation theory (MP2) results for dialanine, which is useful for calibrating methods without polluting cooperative effects. Calculating force fields from first principles was computationally tractable for CCSD(T) [triples calculated with perturbation theory]. It was the best of all the methods tested. These tests use MP2 for fitting torsional parameters.

Oil will not disappear, but the price will go way up. One cannot live without plastic and cannot pay more for commodities. Levulinic acid is a precursor for polymers, plastics, and pharmaceuticals. If one could convert cheap, abundant, nonpetroleum chemicals into levulinic acid, there would be the potential for a chemical industry after oil. Glucose can be converted to levulinic acid. The CCSD(T) calculations on the G4 cluster processor were prohibitively slow (lasting for weeks) using Gaussian but ran in less than 1 hr on 1024 nodes of Blue Gene/P using NWChem. G3/G4 methods are the standard model for thermochemistry. If they were to run on supercomputers rather than workstations, the possible applications would grow exponentially.

In summary, four things were described that went from "can't do" to "done easily." The exascale means the democratization of such capability as well as a paradigm shift in quantum many-body algorithms. The Chemistry Exascale Codesign Center will deliver the transformative software capability required to realize the potential of accurate simulations in many critical areas.

Smarr said that the talk had been brilliant. Putting everything in random access memory (RAM) had been done for a long time. He asked how memory was going to scale up in going to the exascale. Hammond replied that there are still things people will put on local disk. FUSION's CPU is so fast that reading from disk slows things down. One can recompute things faster than reading them from dynamic random access memory (DRAM). One no longer wants to touch DRAM if one can get it from cache. As cache gets bigger, one can do more. DRAM is going to become today's disk.

Tang asked where the low-hanging fruit was on the validation-test front. Hammond responded that wave function theory is called first principles; density functional theory (DFT) has a lot of these. By going to many-body methods, one eliminates a lot of these first principles that need to be justified. As one goes in this direction, the results get better, and that forms a lot of the validity. A lot of heroic calculations are needed to provide benchmarks at high resolution. If one cannot include a lot of physics, comparison with experiment is silly. One has to simulate what nature does. The petascale allows this validation with the full physics system.

Hey asked whether moving to the petascale allowed a lot of smaller simulations. Hammond answered, yes. One needs to be at the bleeding edge because it will allow thousands of people to do petascale calculations every day. Hey asked what focus had been put on quantum computers. Hammond responded that, 30 years from now, people will be working on quantum computers. Today, a primitive algorithm can be run on a primitive quantum computer.

Berger asked whether cache will scale up to the exascale. Hammond answered that he would like to see 1 MB of fast memory close to the CPUs; however, he was not concerned about this. People will program to the architecture.

Misun Min was asked to discuss high-order algorithms for electromagnetic and fluid modeling.

The team at ANL is working on three projects in nano science, accelerator modeling, and fluid simulation. High-order methods are used because of the need to represent geometry accurately, because long time integration requires minimal numerical dispersion, and because high-order methods deliver engineering accuracy with fewer points per wavelength.

The Nek5000 is an open-source code with Maxwell and Boltzmann solvers in Fortran and C. It has excellent scaling on the Blue Gene/P. A weak formulation of the Maxwell equation is used that defines numerical flux and integrates by parts using the matrix diagonal. Communication latency is reduced by a factor of 6. The team is also looking into high-order time integration.

In the first application problem, nanophotonic devices, the standard finite-difference time-domain (FDTD) calculation does not capture the correct profile on the surface (with strong oscillations). A very dense grid resolution is required to represent high-gradient fields.

In the second application problem, accelerator modeling for the International Linear Collider (ILC), the team is looking at beam-propagation behaviors, wake fields, and wave potentials.

In the third application problem, fluid simulation, a simplified Boltzmann equation is used for the Couette flows, vortex rollup, flow past a cylinder, and drag coefficient.

In the future, applications and collaborations will be expanded to nano solar cell applications, geometric flexibility will be enhanced, efficient time-stepping methods will be developed, and alternative programming models will be developed for the extreme scale.

Tang asked why a high-order fluid approach was used rather than the cell-and-particle methods and what the differences were. Min replied that no comparisons had been done. It was planned to compare wakefield calculations.

Giles asked if these methods were being inserted into libraries. Min replied, no; this is application code.

A break was declared at 9:46 a.m. The meeting was called back into session at 10:05 a.m. **Robert Ross** was asked to discuss planning for the Exascale Software Center (ESC).

Design-build partnerships for the extreme scale are being pulled together. However, today, software development is uncoordinated with hardware features; only basic acceptance-test software is delivered with the platform; vendors often "snapshot" key open-source components and then deliver a stale code branch; community codes are unprepared for a sea change in architectures; "coordination via contract" is poor and only involves two parties, and there is no global evaluation of key missing components. A planning team has come together to explore these problems. The goal of the ESC is to ensure successful deployment of a coordinated exascale software stack on exascale-initiative platforms. It will identify required software capabilities; identify gaps; design and develop open-source software components; ensure functionality, stability, and performance; collaborate with platform vendors to integrate software; coordinate outreach to the broader open source; and track development progress and milestones.

It is assumed that there will be several vendor platform partnerships; that in about 2015, it will be early scalability demonstration systems; that in about 2018, there will be an exascale system; and that codesign centers will provide initial applications. Therefore, the ESC is a partnership that is responsible for the common software environment for exascale-initiative systems. All development will be open source. Some components will be integrated and supported by vendor; others will be provided atop the basic platform supported by the ESC. Vendor-specific components will be part of the platform strategy.

The effort will deliver high-quality system software for exascale platforms; identify software gaps, R&D solutions, test, and support deployment; and increase the productivity and capability and reduce the risk of exascale deployments. There would be 10 to 20 distributed teams of 3 to 7 people each. The facility would be a large, primarily centralized quality assurance (QA), integration, and verification center. The effort is now in a planning process; a technical review is expected in April 2011; software will need to be ready for integration in 2014. There is a large international community that has to be included in considering software, QA, and codesign.

Three challenges are relevant:

- How does the Center participate in codesign activities, and what are vendors and application teams looking for from the Center?
- Given resource constraints, how does the Center select components to be supported? What does the Center require of these components and the teams that develop them? How does the Center interact with these teams?
- How does the Center engage with the larger DOE, U.S., and international communities in the development of this software?

One needs to look at the requirements for a 2015 machine and then determine what software is needed. The gaps need to be filled, and current software needs to be upgraded. Codesign has been going on for some time in DOE [e.g., (1) the Blue Gene/P and /Q experience in math libraries and systems management with IBM and (2) the Message Passing Interface Chameleon (MPICH) codesign with IBM and Cray].

Vendors want something like the ESC to coordinate and take real responsibility for features and milestones. They do not want a "tossed-over-the-wall" strategy. "Hardening" cannot be done by different teams. There is a need to manage the risk of the final machine's functionality, performance, stability, and acceptance. The software might be ESC-developed and vendor-integrated and -supported; or it might be ESC-developed, provided, and supported. Formalized roles between the ESC and the vendors must be established for the software's development, risk, support, and acceptance. There is a split in that ESC may or may not take responsibility in debugging and supporting the software it develops.

The application teams want something like ESC to coordinate and take real responsibility for features and milestones. They want to know the specifics about hardware and available software. They also need help managing the risk of the final machine's functionality, performance, stability, and acceptance.

The answers that come from teams are not always helpful. There are four categories of software and function: I/O storage, math libraries, performance tools, and others. An initial set of software for which it makes sense for the center to support is programming models, operating systems and runtime programs, application programmer tools, numerical libraries and frameworks, data management and analysis, and system management and cybersecurity.

It is an open question whether this is the minimum set for the ESC to support. The Center has to do technical evaluation of and assess the riskiness of software. The team has to be evaluated, too (including the institution's culture). The core staff will be taken over by a vendor, and the Center will support the rest.

Successful applied R&D teams are built around a clear goal of delivering working, supported packages. Good software hygiene cannot be someone else's job. ESC must work with successful teams and existing processes or, in some cases, boot new teams within institutions with an excellent history of deployed software. Formal plans and milestones and reviews are necessary for each component. Codesign feedback and risk-based assessments work well with spiral development discipline for software.

Clear deliverables will need to be formulated with specific targets for functionality, performance, and stability. A team management plan and risk tracking will need to be defined. Software development plans will need to be documented. And resource accounting, a technical review schedule, a release schedule, and an integration plan will need to be developed.

A number of groups in the United States are looking at this problem, and there is a European exascale initiative. The ESC will also need to interact with third-party software developers, codesign centers, and ASCR/NNSA institutes.

We are currently developing software planning documents; building application codesign liaisons and developing a plan for jointly evaluating key software; building links to the International Exascale Software Project (IESP) organizational plan; beginning the technical evaluation and ranking of key software components; and establishing links to NSF, the National Aeronautics and Space Administration (NASA), the Defense Advanced Research Projects Agency (DARPA), and other groups.

Graham noted that one of the benefits of the exascale would be to allow thousands of scientists to work at the petascale level. The Center's staff should think about not only scaling up to the exascale but also scaling down to the petascale. Ross agreed; one must be aware of how the value of the exascale software trickles down.

Smarr noted that there was a country that was not on the chart. He asked how the Center will help the Chinese get to the exascale and to its national-security applications faster. Ross replied that the open-science uses of exascale computing will be useful to all communities. The United States can only gain from getting all international teams to become invested in the problem, but that process has challenges.

Berger asked what Ross was getting at in separately discussing application codesign centers and hardware codesign centers when these two are commonly considered as one. Ross replied that hardware developers are all vendors. On the application side, there are five groups interested in codesign. Everyone has unique sets of demands and constraints that will need to be traded off.

White asked why he thought that the current codesign experience that already works will apply. Ross replied that the team is still working out how that selection process will work, what software will need to be developed by the Center, and what are the best bets for the next machines. It would be good to have all

new software evolve from old software, but risks will also have to be taken with the development of new software. A new programming model will be needed, and the United States needs to get in the game early.

White said that this situation reminded her of the NSF group on networking that uses proven resources and that has metrics and benchmarks. The Center's approach sounds ad hoc. Ross agreed that a formal process was needed. The April plan will outlive such a process.

Giles said that a lot of the ability to be successful appears to depend on the voluntariness of vendors, development teams, etc. One needs muscle to keep them in the fold in the future. Ross replied that an investment needs to be made in software teams and that vendors need to be involved in those teams. That is different from their being volunteers; they become invested. The Center can help them stand by their products.

Petzold said that it would seem obvious to direct the Center's attention to the JOULE codes. Ross pointed out that those are current codes; the Center is looking at the codes most likely to be running on 2015–2017 machines. Petzold said that some of the codes on the list really do not need an exascale machine. Ross said that those lists are not the software needed on exascale machines but are typical current-day software applications. These are the ones in the codesign process today. The people doing the big applications are yet to be identified. Petzold asked what motivates the vendors to work with the Center. Ross answered that the Center provides them with an open-source software product so they do not have to develop it themselves. Competitive bidding will motivate them to work with the Center from their home locations.

Boyana Norris was asked to discuss automatic differentiation (AD) in computational science. If one has a numerical model, the program may have a lot of source codes. One could hand-code the differentiations; however, coding time grows with program size and complexity, there is no natural way to compute derivative matrix-vector products without forming a full matrix, and maintenance is a problem. One could also use finite-difference approximations, but they introduce truncation errors, and their cost grows with the number of independent variables. Tools are available for at least semiautomatic differentiation. They are used in applications for computing gradients, Jacobian projections, Hessian projections, and higher-order derivatives.

Derivatives are used for measuring the sensitivity of a simulation to unknown or poorly known parameters, assessing the role of algorithm parameters in a numerical solution, computing a descent direction in numerical optimization, and solving discretized nonlinear partial differential equations (PDEs). Applications include sea-ice modeling, sensitivity analysis of climate models (improving performance from 23 days by the finite-difference method to 22 minutes), general circulation models (running one simulation in 51.75 hours rather than 204.2 hours), and the solution of nonlinear PDEs.

AD is a technique for computing the analytic derivatives used in many numerical algorithms. AD is equivalent to analytic differentiation of elementary functions plus propagation by the chain rule. Associativity of the chain rule leads to two main modes: the forward mode and the reverse mode. AD can be implemented via source transformation or operator overloading. In the forward mode, one propagates derivative vectors that contained derivatives with respect to independent variables. Time and storage are proportional to vector length. In the reverse mode, one propagates adjoints that contained derivatives of dependent variables, and the time is proportional to the adjoint vector length.

To accumulate one's derivatives, one computes the sum of weights over all paths and finds the order of computing the path weights that minimizes cost. Pre-accumulation reduces flops and the memory required; the optimal strategy can reduce the number of flops by another factor of 2.
AD delivers higher productivity, improved quality, higher performance, and improved software maintenance for consistency. AD tools require robust compiler infrastructure, traditional and domain-specific compiler analyses, and combinatorial algorithms to identify effective strategies for combining partial derivatives.

The tools that have been developed at ANL include Open AD/F (Automatic Differentiation/Fortran, which operates in a language-independent manner and supports Fortran 95); ADIC (Automatic Differentiation of C, which supports C and some C++); and ADIFOR (Automatic Differentiation of

Fortran, which supports Fortran 77). ADIFOR and ADIC have propagated through the user community very well.

There are research challenges/opportunities, such as producing more-efficient derivative computations by exploiting structure, exploiting cheap derivative quantities, using elimination strategies, and doing compiler analysis. Other challenges are embodied in mathematics, language features, multi-language applications, different parallel programming models, exploiting parallelism in derivative computations, efficient checkpointing strategies, and derivative propagation.

Exploiting scarcity reduces both the number of flops to pre-accumulate local partials and the number of flops to propagate global derivatives. Matrix coloring can reduce the cost of Jacobian computations for nonlinear PDEs discretized on regular grids. And fully automated derivatives can be used when standard interfaces are available. Matrix coloring is used when Jacobian matrices are sparse. This is equivalent to the distance-2 color of the bipartite graph of the Jacobian.

Math challenges include derivatives of intrinsic functions at points of non-differentiability, implicitly defined functions, and functions computed with numerical methods (e.g., a linear solver). The limitations of black-box AD can be addressed by detecting points of non-differentiability, exploiting mathematics to avoid differentiating through an adaptive algorithm, and modifying the termination criterion for implicitly defined functions.

For automatic differentiation to succeed in parallel computing, the data-flow analysis framework must become MPI-aware. The reverse mode dramatically reduces derivative cost for scalar functions but requires control and data-flow reversal relative to function evaluation. New prefix-like algorithms must be developed for derivatives of parallel reduction operations.

When one thinks about the exascale, AD is a semantic transformation, and the resulting code may exhibit different concurrency characteristics than the original computation. The reverse mode will have to be done more efficiently.

In summary, AD provides a (semi-)automated way for generating accurate derivatives, spans multiple areas (like applied mathematics, combinatorial algorithms, and compilers), and must keep pace with increasingly complex applications, evolving hardware, increasing levels of parallelism, and changing programming models and languages.

Petzold said that support for reverse-mode code disappears. Norris replied that Open ADF fully supports the reverse mode.

Graham said that debugging of the AD code is a problem. Norris answered that the usual mode of use would not require debugging. If one had a bug that caused the AD code to crash, one would need to understand how it was transformed.

Giles asked about locality as an influence on cost. Norris said that most algorithms focus on flops. One can look at limited memory and other factors. Giles asked about auto tuning the output of AD. Norris said that scalar-vector products are subject to tenability, but the best gains are available at the booster level.

Giles noted that two members had left the Committee, Robert Voigt and Ronald Bailey. The Committee appreciates their efforts, and letters of thanks will be sent to them.

The floor was opened to public comment. There being none, the meeting was adjourned at 11:47 a.m.

Respectfully submitted,
Frederick M. O'Hara, Jr.
Recording Secretary
Nov. 16, 2010