Accelerating Science and Engineering with **Titan**, the World’s Fastest Supercomputer

Fernanda Foertter
Jack Wells
Oak Ridge Leadership Computing Facility (OLCF)
BIG PROBLEMS REQUIRE BIG SOLUTIONS

Energy

Healthcare

Competitiveness
BIG PROBLEMS REQUIRE BIG SOLUTIONS

Climate Change

“We will respond to the threat of climate change, knowing that the failure to do so would betray our children and future generations.”
– President Obama
1/21/2013
Leadership computing capability is required for scientists to tackle the high-resolution, multi-scale/multi-physics simulations of greatest interest and impact to both science and the nation.

Leadership computing research is mission critical to inform policy decisions and advance innovation in far reaching topics such as:

- energy assurance
- ecological sustainability
- scientific discovery
- global security

Greater than other computational centers
WHAT IS THE LEADERSHIP COMPUTING FACILITY?

Collaborative, multi-lab, DOE/SC initiative ranked top domestic priority in Facilities for the Future of Science: A Twenty-Year Outlook.

**Mission:** Provide the computational and data science resources required to solve the most important scientific & engineering problems in the world.

- Highly competitive user allocation program (INCITE, ALCC).
- Projects receive 100x more hours than at other generally available centers.
- LCF centers partner with users to enable science & engineering breakthroughs.
INCREASED OUR SYSTEM CAPABILITY BY 10,000X since 2004

LCF Capacity

Peak Teraflops

<table>
<thead>
<tr>
<th>Year</th>
<th>Teraflops</th>
</tr>
</thead>
<tbody>
<tr>
<td>2004</td>
<td>1</td>
</tr>
<tr>
<td>2005</td>
<td>10</td>
</tr>
<tr>
<td>2006</td>
<td>100</td>
</tr>
<tr>
<td>2007</td>
<td>1,000</td>
</tr>
<tr>
<td>2008</td>
<td>10,000</td>
</tr>
<tr>
<td>2009</td>
<td>100,000</td>
</tr>
<tr>
<td>2010</td>
<td>1,000,000</td>
</tr>
<tr>
<td>2011</td>
<td>10,000,000</td>
</tr>
<tr>
<td>2012</td>
<td>100,000,000</td>
</tr>
<tr>
<td>Year</td>
<td>Hours Allocated</td>
</tr>
<tr>
<td>------</td>
<td>----------------</td>
</tr>
<tr>
<td>2004</td>
<td>4.9M</td>
</tr>
<tr>
<td>2005</td>
<td>6.5M</td>
</tr>
<tr>
<td>2006</td>
<td>18.2M</td>
</tr>
<tr>
<td>2007</td>
<td>95M</td>
</tr>
<tr>
<td>2008</td>
<td>268M</td>
</tr>
<tr>
<td>2009</td>
<td>889M</td>
</tr>
<tr>
<td>2010</td>
<td>1.6B</td>
</tr>
<tr>
<td>2011</td>
<td>1.7B</td>
</tr>
<tr>
<td>2012</td>
<td>1.7B</td>
</tr>
<tr>
<td>2013</td>
<td>5B</td>
</tr>
</tbody>
</table>


 Modeling of molecular basis of Parkinson's disease named #1 computational accomplishment, *Breakthroughs* (2008).


 OMEN breaks the petascale barrier using more than 220,000 cores, *Proceedings SC10*.

 Unprecedented simulation of magnitude-8 earthquake over 125-square miles, *Proceedings SC10*.
SCIENCE CHALLENGES FOR LCF IN NEXT DECADE

Climate Change Science
Understand the dynamic ecological and chemical evolution of the climate system with uncertainty quantification of impacts on regional and decadal scales.

Fusion Energy/ITER
Develop predictive understanding of plasma properties, dynamics, and interactions with surrounding materials.

Solar Energy
Improve photovoltaic efficiency and lower cost for organic and inorganic materials.

Combustion Science
Increase efficiency by 25%-50% and lower emissions from internal combustion engines using advanced fuels and new, low-temperature combustion concepts.

Biomass to Biofuels
Enhance the understanding and production of biofuels for transportation and other bio-products from biomass.

Globally Optimized Accelerator Designs
Optimize designs as the next generations of accelerators are planned, increasingly detailed models will be needed to provide a proof of principle and a cost-effective method to design new light sources.
**Mission:** Deploy and operate the computational resources required to tackle global challenges
- Deliver transforming discoveries in climate, materials, biology, energy technologies, etc.
- Enabling investigation of otherwise inaccessible systems, from regional climate impacts to energy grid dynamics

**Vision:** Maximize scientific productivity and progress on largest scale computational problems
- World-class computational resources and specialized services for the most computationally intensive problems
- Stable hardware/software path of increasing scale to maximize productive applications development

Jaguar: 2.3 PF
- Leadership system for science

Titan (OLCF-3):
- 10–20 PF Leadership system

OLCF-4:
- 100–250 PF

OLCF-5:
- 1 EF

2009

2012

2016

2019
THE POWER WALL

- **Moore’s Law** continues, while **CPU clock rates** stopped increasing in 2003 due to **power constraints**.

- **Power** is capped by heat dissipation and $$$

- Performance increases have been coming through increased parallelism

Power consumption of 2.3 PF Jaguar

7 megawatts

equivalent to a small city (~7,000 homes)
Power consumption of a 27 PF CPU-only system

82 megawatts

equivalent to ~80,000 homes
POWER IS THE PROBLEM

Power consumption of a 27 PF Hybrid system

8.2 megawatts

equivalent to ~8,000 homes
WHY GPUs?
High performance and power efficiency on path to exascale

- **CPU**
  - Optimized for multitasking

- **GPU**
  - 10x performance per socket
  - 10x the energy-efficiency
  - Optimized for throughput
• On Jaguar, with 299,008 cores, we were seeing the limits of a single level of MPI scaling for most applications

• To take advantage of the vastly larger parallelism in Titan, users need to use hierarchical parallelism in their codes
  • Distributed memory: MPI, SHMEM, PGAS
  • Node Local: OpenMP, Pthreads, local MPI communicators
  • Within threads: Vector constructs on GPU, libraries, OpenACC

• These are the same types of constructs needed on all multi-PFLOPS computers to scale to the full size of the systems!
Hierarchical parallelism
Improve scalability of applications

Explicit data management
Between CPU and GPU memories

Data locality: Keep data near processing
GPU has high bandwidth to local memory and large internal cache

Expose more parallelism
Code refactoring and source code directives can double performance

Heterogeneous multicore processor architecture
Using right type of processor for each task
27 Petaflops

Theoretical Peak Performance

WORLD’S MOST POWERFUL COMPUTER

#1

TOP500 SUPERCOMPUTER SITES
Jack Wells, PhD

Director of Science, NCCS at ORNL
ORNL’S “TITAN” HYBRID SYSTEM

SYSTEM SPECIFICATIONS:
• Peak performance of 27.1 PF
  • 24.5 GPU + 2.6 CPU
• 18,688 Compute Nodes each with:
  • 16-Core AMD Opteron CPU
  • NVIDIA Tesla “K20x” GPU
  • 32 + 6 GB memory
• 512 Service and I/O nodes
• 200 Cabinets
• 710 TB total system memory
• Cray Gemini 3D Torus Interconnect
• 8.9 MW peak power

4,352 ft²  
(404 m²)
## Titan Nodes

<table>
<thead>
<tr>
<th>Component</th>
<th>Description</th>
<th>Frequency</th>
<th>Memory</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Node</strong></td>
<td>AMD Opteron 6200 Interlagos (16 cores)</td>
<td>2.2 GHz</td>
<td>32 GB (DDR3)</td>
</tr>
<tr>
<td><strong>Accelerator</strong></td>
<td>Tesla K20x (2688 CUDA cores)</td>
<td>732 MHz</td>
<td>6 GB (DDR5)</td>
</tr>
<tr>
<td><strong>Network</strong></td>
<td>Gemini High Speed Interconnect</td>
<td></td>
<td>3D Torus</td>
</tr>
<tr>
<td><strong>Storage</strong></td>
<td>Luster Filesystem</td>
<td></td>
<td>5 PB</td>
</tr>
<tr>
<td><strong>Archive</strong></td>
<td>High-Performance Storage System (HPSS)</td>
<td></td>
<td>29 PB</td>
</tr>
</tbody>
</table>
Material Science (WL-LSMS)
Illuminating the role of material disorder, statistics, and fluctuations in nanoscale materials and systems.

Molecular (LAMMPS)
A molecular description of soft materials, with applications in biotechnology, medicine and energy.

Combustion (S3D)
Understanding turbulent combustion through direct numerical simulation with complex chemistry.

Climate Change (CAM-SE)
Answering questions about specific climate change adaptation and mitigation scenarios; realistically represent features like precipitation patterns / statistics and tropical storms.

Astrophysics (NRDF)
Radiation transport – important in astrophysics, laser fusion, combustion, atmospheric dynamics, and medical imaging – computed on AMR grids.

Nuclear Energy (Denovo)
Discrete ordinates radiation transport calculations that can be used in a variety of nuclear energy and technology applications.
**LAMMPS EARLY RESULTS**

- **Liquid crystal** mesogens are represented with biaxial ellipsoid particles, Gay-Berne potential, isotropic phase, isothermal-isobaric ensemble, $4\sigma$ cutoff with a $0.8\sigma$ neighbor skin (High arithmetic intensity)

![Graph showing time vs. nodes with different configurations and time percentages.]
EFFICIENT ORGANIC PHOTOVOLTAIC MATERIALS

- Organic photovoltaic (OPV) solar cells are promising renewable energy sources:
  - Low costs, high-flexibility, and light weight
  - Bulk-heterojunction (BHJ) active layer is critical for device performance
  - High ratios of donor/acceptor interfaces per volume
  - Detailed structure of BHJ is unknown
  - Use Titan to converge early pioneering MD simulations of BHJ interfaces

P3HT (electron donor)

PCBM (electron acceptor)
COARSE-GRAIN MD SIMULATION OF P3HT:PCBM HETEROJUNCTION

- Acceleration for neighbor-list, short-range forces, and long-range electrostatics
- Portability: Builds with CUDA or OpenCL
- Speedups on Titan (GPU+CPU vs. CPU: 2X to 15x (mixed precision) depending upon model and simulation
- Titan simulations are 27x larger and 10x longer
- Converged P3HT:PCBM separation in 400ns CGMD time
- Increasing polymer chain length will decrease the size of the electron donor domains
- PCBM (fullerene) loading parameter results in an increasing, then decreasing impact on P3HT domain size

217 Cray XK7 nodes per simulation during March 2013

Speedup of 2.5-3x for OPV simulation used here
LAMMPS ACCELERATOR SPEEDUP

Speedup with Acceleration on XK6/XK7 Nodes
1 Node = 32K Particles
900 Nodes = 29M Particles

<table>
<thead>
<tr>
<th></th>
<th>XK6 (1 Node)</th>
<th>XK7 (1 Node)</th>
<th>XK6 (900 Nodes)</th>
<th>XK7 (900 Nodes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Atomic Fluid (cutoff = 2.5(\sigma))</td>
<td>1.92</td>
<td>2.90</td>
<td>1.68</td>
<td>2.75</td>
</tr>
<tr>
<td>Atomic Fluid (cutoff = 5.0(\sigma))</td>
<td>4.33</td>
<td>8.38</td>
<td>3.96</td>
<td>7.48</td>
</tr>
<tr>
<td>Bulk Copper</td>
<td>2.12</td>
<td>3.66</td>
<td>2.15</td>
<td>2.86</td>
</tr>
<tr>
<td>Protein</td>
<td>2.6</td>
<td>3.36</td>
<td>1.56</td>
<td>1.95</td>
</tr>
<tr>
<td>Liquid Crystal</td>
<td>5.82</td>
<td>15.70</td>
<td>5.60</td>
<td>10.14</td>
</tr>
</tbody>
</table>
DENONO EARLY RESULTS

• Solving synthetic reactor eigenvalue problem with Denovo

• Weak scaling, per node problem size (NX, NY, NZ) = (16, 32, 64), NE=16, NM=16, NA=256, LD elements, GMRES tolerance 1e-4, Arnoldi tolerance 1e-2

• Runs on Titan, preliminary results

• Total runtime 2.5X-3X faster using GPU-enabled sweeper

• Shows need to run non-sweep components on GPU as well, e.g., GMRES solver
## HOW EFFECTIVE ARE GPUs ON SCALABLE APPLICATIONS?

Very early performance measurements

### OLCF-3 early science codes compared to performance on Jaguar

<table>
<thead>
<tr>
<th>Application</th>
<th>Description</th>
<th>Jaguar workload</th>
<th>Speedup</th>
</tr>
</thead>
<tbody>
<tr>
<td>S3D</td>
<td>Turbulent combustion</td>
<td>6%</td>
<td>1.8</td>
</tr>
<tr>
<td>Denovo sweep</td>
<td>Sweep kernel of 3D neutron transport for nuclear reactors</td>
<td>2%</td>
<td>3.8</td>
</tr>
<tr>
<td>LAMMPS</td>
<td>High-performance molecular dynamics</td>
<td>1%</td>
<td>7.4*</td>
</tr>
<tr>
<td>WL-LSMS</td>
<td>Statistical mechanics of magnetic materials</td>
<td>2%</td>
<td>3.8**</td>
</tr>
<tr>
<td>CAM-SE</td>
<td>Community atmosphere model</td>
<td>1%</td>
<td>~1.8</td>
</tr>
</tbody>
</table>

*mixed precision  **gordon bell winner
## Criteria for Selecting Early Readiness Applications

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
</tr>
</thead>
</table>
| **Science**                               | Science results, impact, timeliness  
Alignment with DOE and U.S. science mission  
Broad coverage of science domains |
| **Implementation (models, algorithms, software)** | Broad coverage of relevant programming models, environment, languages, implementations  
Broad coverage of relevant algorithms and data structures (motifs)  
Broad coverage of scientific library requirements |
| **User community (current and anticipated)** | Broad institutional and developer/user involvement  
Good representation of current and anticipated INCITE workload |
| **Preparation for steady state ("INCITE ready") operations** | Mix of low ("straightforward") and high ("hard") risk porting and readiness requirements  
Availability of OLCF liaison with adequate skills/experience match to application  
Availability of key code development personnel to engage in and guide readiness activities |
ACTION PLAN FOR CODE PORTING
We developed a plan for porting these applications, which involved the following steps:

1. Multidisciplinary code team for each code – OLCF application lead, Cray engineer, NVIDIA developer, also cross-cutting support from tool and library developers
2. Early testbed hardware – white box GPU cluster “yona” for code development
3. Code inventory for each code to understand characteristics – application code structure, code suitability for GPU port, algorithm structure, data structures and data movement patterns. Also code execution profile – are there performance “hot spots” or is the profile “flat”
4. Develop parallelization approach for each application – ascertain which algorithm and code components to port to GPU, how to map work to GPU threads, how to manage data motion CPU-GPU and between GPU main memory and GPU caches/shared memory
5. Decide GPU programming model for port to GPU, e.g., CUDA for more close-to-the-metal programming, OpenACC for a higher abstraction level and a more incremental porting approach, OpenCL for portability advantages, or libraries when appropriate
6. Address code development issues – rewrite vs. refactor, managing portability to other platforms, incorporating GPU code into build system, relationship to the code repository main trunk
7. Representative test cases, e.g., early science problems, formulated as basis for evaluating code performance and setting priorities for code optimization. Also formulate comparison metric to measure success, e.g., time to solution on dual Interlagos Cray XE6 vs. Titan Cray XK7 Interlagos+Kepler
### APPLICATION CHARACTERISTICS INVENTORY

<table>
<thead>
<tr>
<th>App</th>
<th>Science Area</th>
<th>Algorithm(s)</th>
<th>Grid type</th>
<th>Programming Language(s)</th>
<th>Compiler(s) supported</th>
<th>LOC</th>
<th>Comm Libraries</th>
<th>Math Libraries</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAM-SE</td>
<td>climate</td>
<td>spectral finite elements, dense &amp; sparse linear algebra, particles</td>
<td>structured</td>
<td>F90</td>
<td>PGI, Lahey, IBM</td>
<td>500K</td>
<td>MPI</td>
<td>Trilinos</td>
</tr>
<tr>
<td>LAMMPS</td>
<td>Biology, materials</td>
<td>molecular dynamics, FFT, particles</td>
<td>N/A</td>
<td>C++</td>
<td>GNU, PGI, IBM, Intel</td>
<td>140K</td>
<td>MPI</td>
<td>FFTW</td>
</tr>
<tr>
<td>S3D</td>
<td>combustion</td>
<td>Navier-Stokes, finite diff, dense &amp; sparse linear algebra, particles</td>
<td>structured</td>
<td>F77, F90</td>
<td>PGI</td>
<td>10K</td>
<td>MPI</td>
<td>None</td>
</tr>
<tr>
<td>Denovo</td>
<td>nuclear energy</td>
<td>wavefront sweep, GMRES</td>
<td>structured</td>
<td>C++, Fortran, Python</td>
<td>GNU, PGI, Cray, Intel</td>
<td>46K</td>
<td>MPI</td>
<td>Trilinos, LAPACK, SuperLU, Metis</td>
</tr>
<tr>
<td>WL-LSMS</td>
<td>nanoscience</td>
<td>density functional theory, Monte Carlo</td>
<td>N/A</td>
<td>F77, F90, C, C++</td>
<td>PGI, GNU</td>
<td>70K</td>
<td>MPI</td>
<td>LAPACK (ZGEMM, ZGTRF, ZGTRS)</td>
</tr>
<tr>
<td>NRDF</td>
<td>radiation transport</td>
<td>Non-equilibrium radiation diffusion equation</td>
<td>structured</td>
<td>AMR</td>
<td>PGI, GNU, Intel</td>
<td>500K</td>
<td>MPI, SAMRAI</td>
<td>BLAS, PETSc, Hypre, SAMRSolvers</td>
</tr>
</tbody>
</table>
CAAR: SELECTED LESSONS LEARNED

• Repeated themes in the code porting work
  • finding more threadable work for the GPU
  • Improving memory access patterns
  • making GPU work (kernel calls) more coarse-grained if possible
  • making data on the GPU more persistent
  • overlapping data transfers with other work (leverage HyperQ)
  • use as much asynchronicity as possible (CPU, GPU, MPI, PCIe-2)

• The difficulty level of the GPU port was in part determined by:
  • Structure of the algorithms—e.g., available parallelism, high computational intensity
  • Code execution profile—flat or hot spots
  • The code size (LOC)
• We estimate possibly 70-80% of developer time is spent in code restructuring, regardless of whether using CUDA / OpenCL / OpenACC / …

• More available flops on the node should lead us to think of new science opportunities enabled—e.g., more DOF per grid cell

• We may need to look in unconventional places to get another ~30X thread parallelism that may be needed for exascale—e.g., parallelism in time
Primary Objective:

“Provide substantial allocations to the open science community through an peered process for a small number of high-impact scientific research projects”
## OLCF Allocation Programs

<table>
<thead>
<tr>
<th></th>
<th>INCITE</th>
<th>ALCC</th>
<th>Director’s Discretionary</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Mission</strong></td>
<td>High-risk, high-payoff science that requires LCF-scale resources</td>
<td>High-risk, high-payoff science aligned with DOE mission</td>
<td>Strategic LCF goals</td>
</tr>
<tr>
<td><strong>Call</strong></td>
<td>1x/year (Closes June)</td>
<td>1x/year (Closes February)</td>
<td>Rolling</td>
</tr>
<tr>
<td><strong>Duration</strong></td>
<td>1-3 years, yearly renewal</td>
<td>1 year</td>
<td>3m, 6m, 1 year</td>
</tr>
<tr>
<td><strong>Typical Size</strong></td>
<td>30 - 40 projects 20M - 100M core-hours/yr.</td>
<td>5 - 10 projects 1M - 75M core-hours/yr.</td>
<td>100s of projects</td>
</tr>
<tr>
<td><strong>Review Process</strong></td>
<td>Scientific, Peer-Review Computational Readiness</td>
<td>Scientific, Peer-Review Computational Readiness</td>
<td>Strategic impact and feasibility</td>
</tr>
<tr>
<td><strong>Managed by</strong></td>
<td>INCITE management committee (ALCF &amp; OLCF)</td>
<td>DOE Office of Science</td>
<td>OLCF management</td>
</tr>
<tr>
<td><strong>Availability</strong></td>
<td>Open to all scientific researchers and organizations including industry</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
EXPANDING ACCESS TO SUPERCOMPUTING

- Innovative and Novel Computational Impact on Theory and Experiment (INCITE)
- Awards time for “grand challenges” in science and engineering
- Open to researchers from academia, government labs, industry
- Overwhelming demand, record number of proposals
- Demand exceeds awards by 3x

Distribution of INCITE (2012)
2014 INCITE CALL FOR PROPOSALS

- Planning Request for Information (RFI)
- Call opens April, 2013. Closes June, 2013
- Expect to allocate more than 5 billion core-hours
- Expect 3X oversubscription
- Awards to be announced in November for CY 2014
- Average award to exceed 50 million core-hours
- INCITE Proposal Writing Webinars!

Contact information
Julia C. White, INCITE Manager
whitejc@DOEleadershipcomputing.org

Reaching out to Researchers: Nearly 50% of the non-renewal proposals are by new PIs.
TITAN UPDATE

- Jaguar to Titan upgrade was in place
- Titan is still going through acceptance

<table>
<thead>
<tr>
<th>Date</th>
<th>Nodes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Feb 2nd</td>
<td>9,716 (CPU Only)</td>
</tr>
<tr>
<td>March 11</td>
<td>8,972 (GPUs available)</td>
</tr>
<tr>
<td>Early April</td>
<td>0 (Acceptance)</td>
</tr>
<tr>
<td>May</td>
<td>18,688 (ALL)</td>
</tr>
</tbody>
</table>
CONCLUSIONS

• Leadership computing is for the critically important problems that need the most powerful compute and data infrastructure
• Our user resources are in high demand and are effectively used.
• Computer system performance increases through parallelism
• Clock speed trend flat to slower over coming years
• Applications must utilize all inherent parallelism
• Accelerated, hybrid-multicore computing solutions are performing very well on real, complex scientific applications.
• OLCF resources are available through open, peer-reviewed allocation mechanisms.
ACKNOWLEDGEMENTS

• OLCF-3 CAAR Team: Bronson Messer, Wayne Joubert, Mike Brown, Matt Norman, Markus Eisenbach, Ramanan Sankaran
• OLCF Users: Jackie Chen, Tom Evans, Markus Eisenbach,
• OLCF-3 Hardware Vendor Partners: Cray, AMD, and NVIDIA

This research used resources of the Oak Ridge Leadership Computing Facility at the Oak Ridge National Laboratory, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC05-00OR22725.
ADVANCING TOWARD EXASCALE

Comprehensive Earth System Model at 1 km scale, enabling modeling of cloud convection and ocean eddies.

Coupled simulation of entire cells at molecular, genetic, chemical and biological levels.

First-principles simulation of combustion for new high-efficiency, low-emission engines.

Predictive calculations for thermonuclear and core-collapse supernovae, allowing confirmation of theoretical models.
CAAR: LESSONS LEARNED

• Repeated themes in the code porting work:
  – finding more threadable work for the GPU
  – Improving memory access patterns
  – making GPU work (kernel calls) more coarse-grained if possible
  – making data on the GPU more persistent
  – overlapping data transfers with other work

• Helpful to use as much asynchronicity as possible, to extract performance (CPU, GPU, MPI, PCIe-2)

• Codes with unoptimized MPI communications may need prior work in order to improve performance before GPU speed improvements can be realized

• Some codes need to use multiple MPI tasks per node to access the GPU (e.g., via proxy)—others use 1 MPI task with OpenMP threads on the node

• Code changes that have global impact on the code are difficult to manage, e.g., data structure changes. An abstraction layer may help, e.g., C++ objects/templates

• Two common code modifications are:
  – Permuting loops to improve locality of memory reference
  – Fusing loops for coarser granularity of GPU kernel calls

• Tools (compilers, debuggers, profilers) were lacking early on in the project but are becoming more available and are improving in quality

• Debugging and profiling tools were useful in some cases (Allinea DT, CrayPat, Vampir, CUDA profiler)
CAAR: SELECTED LESSONS LEARNED

- The difficulty level of the GPU port was in part determined by:
  - Structure of the algorithms—e.g., available parallelism, high computational intensity
  - Code execution profile—flat or hot spots
  - The code size (LOC)

- Since not all future code changes can be anticipated, it is difficult to avoid significant code revision for such an effort

- Up to 1-3 person-years required to port each code
  - Takes work, but an unavoidable step required for exascale
  - Also pays off for other systems—the ported codes often run significantly faster CPU-only (Denovo 2X, CAM-SE >1.7X)

- We estimate possibly 70-80% of developer time is spent in code restructuring, regardless of whether using CUDA / OpenCL / OpenACC / …

- Each code team must make its own choice of using CUDA vs. OpenCL vs. OpenACC, based on the specific case—may be different conclusion for each code

- Science codes are under active development—porting to GPU can be pursuing a “moving target,” challenging to manage

- More available flops on the node should lead us to think of new science opportunities enabled—e.g., more DOF per grid cell

- We may need to look in unconventional places to get another ~30X thread parallelism that may be needed for exascale—e.g., parallelism in time
LAMMPS EARLY RESULTS

**Speedup with Acceleration on XK6/XK7 Nodes**
- 1 Node = 32K Particles
- 900 Nodes = 29M Particles

<table>
<thead>
<tr>
<th></th>
<th>XKK (1 Node)</th>
<th>XKK (1 Node)</th>
<th>XKK (900 Nodes)</th>
<th>XKK (900 Nodes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Atomic Fluid (cutoff = 2.5σ)</td>
<td>1.92</td>
<td>2.90</td>
<td>1.68</td>
<td>2.75</td>
</tr>
<tr>
<td>Atomic Fluid (cutoff = 5.0σ)</td>
<td>4.33</td>
<td>8.38</td>
<td>3.96</td>
<td>7.48</td>
</tr>
<tr>
<td>Bulk Copper</td>
<td>2.12</td>
<td>3.66</td>
<td>2.15</td>
<td>2.86</td>
</tr>
<tr>
<td>Protein</td>
<td>2.6</td>
<td>3.36</td>
<td>1.56</td>
<td>1.95</td>
</tr>
<tr>
<td>Liquid Crystal</td>
<td>5.82</td>
<td>15.70</td>
<td>5.60</td>
<td>10.14</td>
</tr>
</tbody>
</table>

**Graph:**
- X-axis: Time (s)
- Y-axis: Speedup

Tuesday, March 19, 13