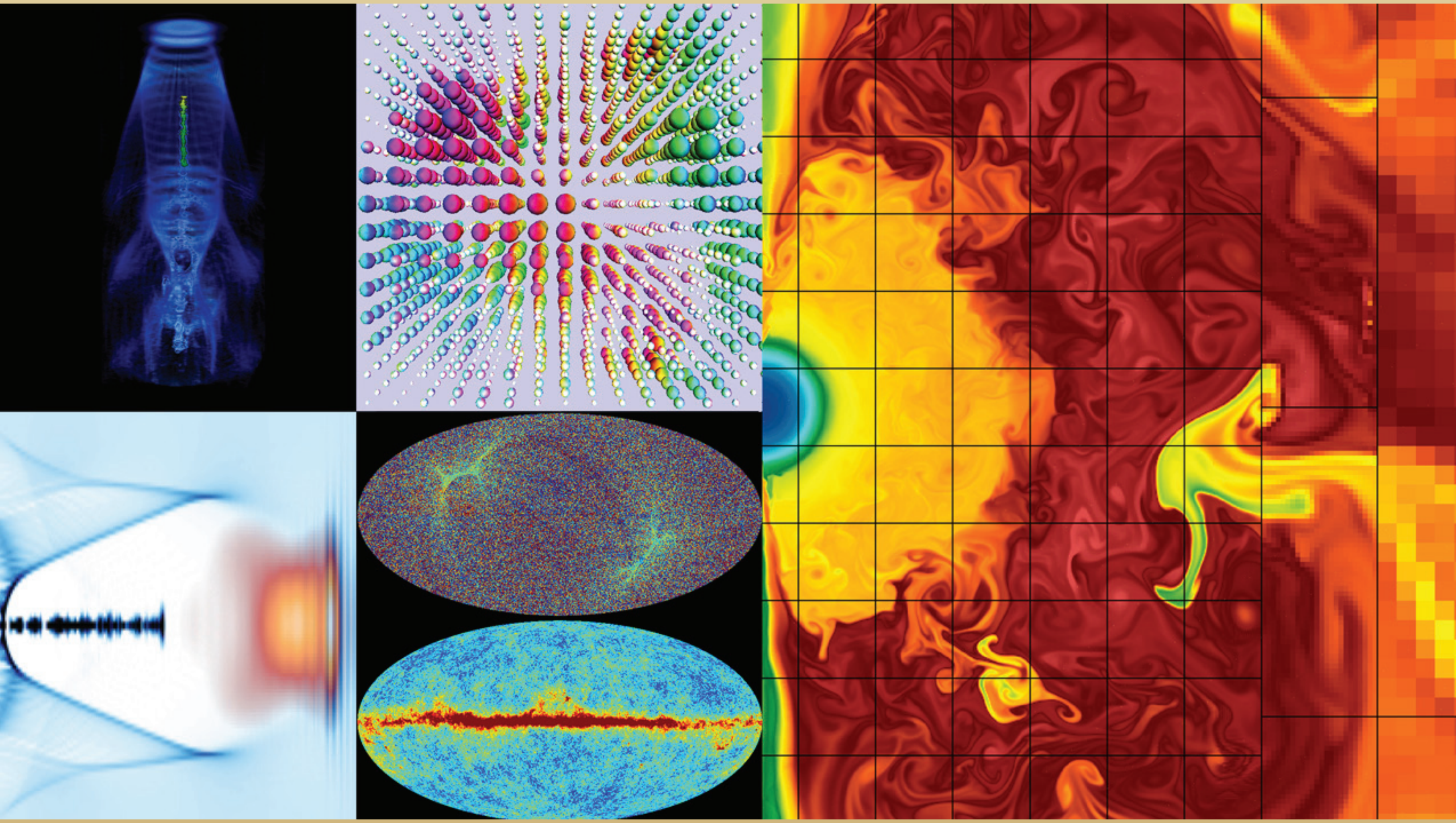
LARGE SCALE COMPUTING AND STORAGE REQUIREMENTS



High Energy Physics

Report of the NERSC / HEP / ASCR
Requirements Workshop
November 12 and 13, 2009

DISCLAIMER

This report was prepared as an account of a workshop sponsored by the U.S. Department of Energy. Neither the United States Government nor any agency thereof, nor any of their employees or officers, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of document authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof. Copyrights to portions of this report (including graphics) are reserved by original copyright holders or their assignees, and are used by the Government's license and by permission. Requests to use any images must be made to the provider identified in the image credits.

NERSC is funded by the United States Department of Energy, Office of Science, Advanced Scientific Computing Research (ASCR) program. Yukiko Sekine is the NERSC Program Manager and Alan Stone serves as the HEP allocation manager for NERSC.

NERSC is located at the Lawrence Berkeley National Laboratory, which is operated by the University of California for the US Department of Energy under contract DE-AC02-05CH11231.

This work was supported by the Directors of the Office of Science, Office of High Energy Physics, and the Office of Advanced Scientific Computing Research, Facilities Division.

This is LBNL report LBNL-4128E, published November, 2010

1. Executive Summary

The National Energy Research Scientific Computing Center (NERSC) is the leading scientific computing facility for the Department of Energy's Office of Science, providing high-performance computing (HPC) resources to more than 3,000 researchers working on about 400 projects. NERSC provides large-scale computing resources and, crucially, the support and expertise needed for scientists to make effective use of them.

In November 2009, NERSC, DOE's Office of Advanced Scientific Computing Research (ASCR), and DOE's Office of High Energy Physics (HEP) held a workshop to characterize the HPC resources needed at NERSC to support HEP research through the next three to five years. The effort is part of NERSC's legacy of anticipating users' needs and deploying resources to meet those demands.

The workshop revealed several key points, in addition to achieving its goal of collecting and characterizing computing requirements. The chief findings:

1. Science teams need access to a significant increase in computational resources to meet their research goals.
2. Research teams need to be able to read, write, transfer, store online, archive, analyze, and share huge volumes of data.
3. Science teams need guidance and support to implement their codes on future architectures**.**
4. Projects need predictable, rapid turnaround of their computational jobs to meet mission-critical time constraints.

This report expands upon these key points and includes others. It also presents a number of case studies as representative of the research conducted within HEP. Workshop participants were asked to codify their requirements in this case study format, summarizing their science goals, methods of solution, current and three-to-five year computing requirements, and software and support needs. Participants were also asked to describe their strategy for computing in the highly parallel, multi-core environment that is expected to dominate HPC architectures over the next few years.

The report includes a section that describes efforts already underway or planned at NERSC that address requirements collected at the workshop. NERSC has many initiatives in progress that address key workshop findings and are aligned with NERSC's strategic plans.

Large Scale Computing and Storage Requirements for High Energy Physics

Workshop Report

U.S. Department of Energy, Office of High Energy Physics
and
Office of Advanced Scientific Computing Research

National Energy Research Scientific Computing Center
(NERSC)

Washington, DC
May 7 and 8, 2009

2. Participants and Contributors

Amber Boehnlein	DOE Office of High Energy Physics Research and Technology Division and Fermi National Accelerator Laboratory
David Bruhwiler	Tech-X Corporation
John Bell	Center for Computational Sciences and Engineering , LBNL
Julian Borrill	Computational Cosmology Center, LBNL
Cameron Geddes	Accelerator and Fusion Research Division, LBNL
Chengkun Huang	Department of Physics and Astronomy, UCLA
Paul McKenzie	Theoretical Physics Department, Fermi Accelerator Laboratory
Lie-Quan (Rich) Lee	Advanced Computation Department, SLAC National Accelerator Laboratory
Michael Norman	Director, San Diego Supercomputer Center
Peter Nugent	Computational Research Division, LBNL
Yukiko Sekine	DOE Office of Advanced Scientific Computing Research
John Shalf	Advanced Technology Group, NERSC
Panagiotis Spentzouris	Computing Division and the Accelerator Physics Center, Fermi National Accelerator Laboratory.
Alan Stone	Research and Technology Division, DOE Office of High Energy Physics
Alex Szalay	Department of Physics and Astronomy, The Johns Hopkins University
Doug Toussaint	Physics Department, University of Arizona
Craig Tull	Computational Research Division, LBNL
Stan Woosley	Department of Astronomy and Astrophysics, University of California, Santa Cruz
Kathy Yelick	NERSC Director

Editors

Richard Gerber	NERSC User Services Group
Harvey Wasserman	NERSC User Services Group

Table of Contents

1. Executive Summary ... iii
2. Participants and Contributors ... v
Table of Contents ... vi
3. Office of High Energy Physics Mission ... 1
4. About NERSC ... 3
5. Workshop Background and Structure ... 4
6. Workshop Demographics ... 5
 6.1. Participants ... 5
 6.2. NERSC Projects Represented by Case Studies ... 6
7. Findings ... 7
 7.1. Summary of Requirements ... 7
 7.2. Other Significant Observations ... 8
 7.3. Computing Requirements ... 9
8. NERSC Response: Initiatives and Plans ... 10
 8.1. Compute Resources ... 10
 8.2. Data Storage, Sharing, and Analysis ... 10
 8.3. Support for Future Architectures ... 11
 8.4. Predictable, Rapid Job Turnaround ... 11
9. Accelerator Physics ... 13
 9.1. Accelerator Physics Overview ... 13
 9.2. Common Requirements Among Accelerator Science Projects ... 14
 9.3. Accelerator Physics Case Studies ... 14
  9.3.1. Community Petascale Project for Accelerator Science and Simulation ... 14
  9.3.2. Advanced Modeling for Particle Accelerators ... 19
  9.3.3. Electromagnetic Modeling of Accelerator Structures ... 23
  9.3.4. Simulation of Laser Plasma Wakefield Particle Accelerators ... 26
  9.3.5. Petascale Particle-in-Cell Simulations of Plasma Based Accelerators ... 30
10. Astrophysics Modeling and Simulation ... 34
 10.1. Astrophysics Modeling and Simulation Overview ... 34
 10.2. Astrophysics Modeling and Simulation Case Studies ... 34
  10.2.1. The Cosmic Frontier – Structure Formation ... 34
  10.2.2. Type Ia Supernovae ... 38
  10.2.3. Core-Collapse Supernovae ... 49
11. Astrophysics Data Analysis ... 54
 11.1. Astrophysics Data Analysis Overview ... 54
 11.2. Astrophysics Data Analysis Case Studies ... 54
  11.2.1. Palomar Transient Factory & La Silla Supernova Search ... 54
  11.2.2. Cosmic Microwave Background Data Analysis for the Planck Satellite Mission ... 57
12. Lattice QCD ... 60
 12.1. Lattice QCD Overview ... 60
 12.2. Lattice QCD Case Studies ... 60
  12.2.1. MIMD Lattice Computation (MILC) Collaboration ... 60
13. HEP Data Analysis and Detector Simulation ... 64
 13.1. Overview ... 64
 13.2. Methods of Solution ... 65
 13.3. HPC Requirements ... 65
 13.4. Access to 50X resources at NERSC ... 67
 13.5. Support Services and Software ... 67

13.6. Emerging HPC Architectures and Programming Models 68
Appendix A. Attendee Biographies 69
Appendix B. Workshop Agenda 73
Appendix C. Abbreviations and Acronyms 74
Appendix D. About the Cover 76

3. Office of High Energy Physics Mission

The Office of High Energy Physics (HEP) program mission is to understand how the universe works at its fundamental level by discovering the most elementary constituents of matter and energy, probing the interactions between them, and by exploring the basic nature of space and time. Fulfilling this mission implies theoretical and experimental programs that rigorously test the Standard Model of particle interactions, including searching for the mechanism that breaks the symmetry between the electro-magnetic and weak forces, probing for additional symmetries of nature, understanding the hierarchy of neutrino masses and investigating the nature of dark energy and dark matter.

Addressing this highly diverse set of interconnected questions requires systematic programs that move along multiple lines of investigation. To this end, HEP supports long term programs in accelerator physics, accelerators with experimental facilities, observational astrophysics, and theoretical physics. All of these programs are data- and compute-intensive and will be described in general terms below. In some cases, the computing needs are met by dedicated high-energy physics (HEP) computing centers located at national laboratories and universities; in others, HEP relies on partnerships with other offices and agencies to supply high performance computing.

HEP operates accelerator facilities such as Fermi National Accelerator Laboratory and collaborates with CERN on the Large Hadron Collider. These custom-designed and purpose-built accelerators provide beams of particles at the necessary energy and intensity to probe a particular set of physics questions. Designing, building and operating optimal accelerator facilities implies the need to simulate the full lifecycle from conceptual R&D to accelerator design through commissioning and stable operations. Furthermore, the complexity, precision, and beam intensity requirements for the next generation accelerators requires end-to-end, multi-physics simulations using highly parallel codes.

Across the full scope of the HEP experimental programs, scientists require intensive computing resources to produce scientific results. Analyzing data is a chain, starting by processing raw detector data with sophisticated algorithms to identify and reconstruct physics objects such as charged particle tracks, including the generation of detailed Monte Carlo simulations to understand the detector response and the efficiency for observing physical processes, and finishing with data mining large datasets for rare events and to make precision measurements.

Likewise, the program in ground- and space-based experiments and telescopes presents a range of computing challenges; large data sets from the instruments are stored, manipulated, and simulated to produce science, sometimes requiring parallel algorithms due to correlations in data. Different elements of the program have different challenges from the multi-petabyte volumes of data and simulation data required by LHC experiments such as ATLAS, to the need to insure the integrity of a small and precious data set as it passes from a distant experimental facility to the hands of users for the Daya Bay Reactor Neutrino experiment, or to the large scale parallel processing required to generate thousands of Monte Carlo simulation of the full mission necessary to exploit the full sensitivity of the Planck Satellite data set.

Complementing the experimental program is a broad theoretical program with computationally intensive requirements. To make meaningful tests of the Standard Model's range of validity, large-scale Lattice QCD calculations are essential to make predictions at a comparable level of precision as the experimental results. The theoretical understanding of the cosmos to shed light on the nature of dark energy and dark matter is also advanced by the use of sophisticated parallel codes to enhance the understanding of a wide range of physical observables from detailing the properties of exploding stars to understanding the clustering of baryonic matter in galaxies.

4. About NERSC

The National Energy Research Scientific Computing (NERSC) Center, which is supported by the U.S. Department of Energy's Office of Advanced Scientific Computing Research (ASCR), serves more than 3,000 scientists working on about 400 projects of national importance. Operated by Lawrence Berkeley National Laboratory (LBNL), NERSC is the primary high-performance computing facility for scientists in all of the research programs supported by the Department of Energy's Office of Science. These scientists — working remotely from DOE national laboratories, universities, other federal agencies, and industry — use NERSC's resources and services to further the research mission of the Office of Science (SC). While focused on research that supports DOE's missions and scientific goals, research conducted at NERSC spans a range of scientific disciplines, including physics, materials science, energy research, climate change, and the life sciences. This large and diverse user community runs hundreds of different application codes.

Results based on work done at NERSC are citied in about 1,500 peer reviewed scientific papers per year. NERSC's activities and scientific results are also described in the center's annual reports, newsletter articles, technical reports, and extensive online documentation. In addition to providing computational support for projects funded by the Office of Science program offices (ASCR, BER, BES, FES, HEP and NP), NERSC directly supports the Scientific Discovery through Advanced Computing (SciDAC[1]) and ASCR Leadership Computing Challenge[2] Programs, as well as several international collaborations in which DOE is engaged. In short, NERSC supports the computational needs of the entire spectrum of DOE open science research.

The DOE Office of Science supports three major High Performance Computing Centers: NERSC and the Leadership Computing Facilities at Oak Ridge and Argonne National Laboratories. NERSC has the unique role of being solely responsible for providing HPC resources to all open scientific research sponsored by the Office of Science. The Leadership Computing Facilities support a more limited number of select projects, whose research areas may not span all Office of Science objectives and are not restricted to mission-relevant research.

This report illustrates NERSC's alignment with, and responsiveness to, DOE program office needs, in this case those of the Office of High Energy Physics. The large number of projects supported by NERSC, the diversity of application codes, and its role as an incubator for scalable application codes present unique challenges to the center. As demonstrated by the overall scientific productivity by NERSC users, however, the combination of effectively managed resources and excellent user support services, the NERSC Center continues its 35-year history as a world leader in advancing computational science across a wide range of disciplines.

For more information about NERSC, visit http://www.nersc.gov.

[1] http://www.scidac.gov

[2] http://www.sc.doe.gov/ascr/incite/AllocationProcess.pdf

5. Workshop Background and Structure

In support of its mission and to maintain its reputation as one of the world's most productive scientific computing facilities, NERSC regularly collects user requirements by a number of means; among them: querying all its research projects through the NERSC Energy Research Computing Allocations process, conducting workload analysis studies, holding frequent conversations with DOE program managers, meetings with the NERSC User Group, and directly engaging scientists who use the facility.

In November 2009, the DOE Office of Advanced Scientific Computing Research (ASCR, which oversees NERSC), the DOE Office of High Energy Physics (HEP), and NERSC held a workshop to gather HPC requirements for current and future research funded by HEP. This result of the workshop is this report, which includes findings that will serve as input to the NERSC/ASCR planning processes and will help ensure that NERSC continues to provide world-class resources and support to Office of Science-funded research projects. The format of the workshop and report was based on that used by DOE's Energy Sciences Network (ESnet), which has conducted a series of similar successful workshops on future networking requirements. That format was modified to accommodate the set of requirements relevant to NERSC.

This report presents a number of consensus findings. In support of these, a number of case study reports are included as specific representative samples of research conducted within HEP. The case studies were chosen by the DOE Program Office Managers and NERSC personnel to provide broad coverage in accelerator physics, astrophysics, data analysis in high-energy physics and astrophysics, and theoretical quantum chromodynamics. However, HEP funds many research endeavors in high energy physics and the case studies presented here do not necessarily represent the entirety of HEP research. Each case study describes its scientific goals today and for the next three to five years, its computational method of solution, and its current and expected future computing needs.

Since supercomputer architectures are trending toward systems with multiprocessors containing hundreds or thousands of cores per socket and perhaps millions of cores per system, participants were asked to describe their strategy for computing in such a highly parallel, multi-core environment. Science teams were also asked to list significant scientific achievements they could make if they had access to a 50-fold increase in their current computing resources.

Requirements presented in this document will serve as input to NERSC planning for systems and services, and will help ensure that NERSC continues to provide world-class resources for scientific discovery to scientists and their collaborators in support of the DOE Office of Science, Office of High Energy Physics.

6. Workshop Demographics

6.1. Participants

Name	Affiliation	NERSC Repo
Amber Boehnlein	DOE Office of High Energy Physics Research and Technology Division; Fermi National Accelerator Laboratory	
David Bruhwiler	Tech-X Corporation	m558
John Bell	Center for Computational Sciences and Engineering , LBNL	m1055, m1012, mp111, m106
Julian Borrill	Computational Cosmology Center, LBNL	planck, mp107, cmbpol
Cameron Geddes	Accelerator and Fusion Research Division, LBNL	m558
Richard Gerber	Services Department, NERSC	
Chengkun Huang	Department of Physics and Astronomy, UCLA	mp113, m778, incite14 (representing Warren Mori)
Paul McKenzie	Theoretical Physics Department, Fermi Accelerator Laboratory	National
Lie-Quan (Rich) Lee	Advanced Computation Department, SLAC National Accelerator Laboratory	m349, m778
Michael Norman	San Diego Supercomputer Center	
Peter Nugent	Computational Research Division, LBNL	m219, m106, mp90, m779
Yukiko Sekine	DOE Office of Advanced Scientific Computing Research	
John Shalf	Advanced Technology Group, NERSC	
Panagiotis Spentzouris	Computing Division and the Accelerator Physics Center, Fermi National Accelerator Laboratory.	m778
Alan Stone	Research and Technology Division, DOE Office of High Energy Physics	
Alex Szalay	Department of Physics and Astronomy, The Johns Hopkins University	
Doug Toussaint	Physics Department, University of Arizona	mp13
Craig Tull	Computational Research Division, LBNL	atlas, dayabay
Harvey Wasserman	NERSC	
Stan Woosley	Department of Astronomy and Astrophysics, University of California Santa Cruz	m106
Kathy Yelick	NERSC Director	

6.2. NERSC Projects Represented by Case Studies

The workshop attendees represented a large fraction of the HEP research performed at NERSC, with 85% of the HEP time allocated to a project for which one of the attendees was the Principle Investigator or a senior researcher on the project. The case studies for existing NERSC projects are listed below, showing the number of NERSC hours used in 2009 and the overall time allocated to that area.

Project ID (Repo)	NERSC Computational Project Title	Principal Investigator	Hours Used at NERSC in 2009
Lattice Quantum Chromodynamics			
mp13	Quantum Chromodynamics with three flavors of dynamical quarks	Doug Toussaint, U. Arizona	19,537,394
Total of projects represented by case studies (84% of total)			19,537,394
NERSC Lattice QCD Total			**23,222,929**
Accelerator Physics			
incite14	Petascale Particle-in-Cell Simulations of Plasma Based Accelerators	Warren Mori, UCLA	9,542,574
m558	Particle simulation of laser wakefield particle acceleration	Cameron Geddes, LBNL	4,741,723
m778	Community Petascale Project for Accelerator Science and Simulation	Panagiotis Spentzouris, FNAL	1,978,819
m349	Advanced Modeling for Particle Accelerators	Kwok Ko, SLAC	1,246,260
mp113	Continuing Studies of Plasma-Based Accelerators	Warren Mori, UCLA	547,774
Total of projects represented by case studies (91% of total)			18,057,150
NERSC Accelerator Physics Total			**19,858,151**
Astrophysics			
m106	Computational Astrophysics Consortium	Stan Woosley, UCSC	3,178,664
planck	Cosmic Microwave Background Data Analysis For The Planck Satellite Mission	Julian Borrill, LBNL	930,211
m779	Baryon Oscillation Spectroscopic Survey	Peter Nugent, LBNL	75,492
Total of projects represented by case studies (76% of total)			4,184,367
NERSC Astrophysics Total			**5,525,880**
Total Represented by Case Studies (85% of total)			**41,778,911**
All HEP at NERSC			**49,028,123**

7. Findings

7.1. Summary of Requirements

The following is a summary of consensus requirements derived from the workshop.

1. **Science teams need access to a significant increase in computational resources to meet their research goals.**

   1.1. Workshop participants collectively estimate needing, in three to five years, a 35-fold increase over their 2009 NERSC allocations of computing hours to meet the scientific goals of projects represented by case studies in this report.

   1.2. Investigators need allocations of computer time that are adequate for production computing, code development, and simulation validation and verification.

   1.3. Science teams need more hours now. Scientific progress is already limited by current allocations to use NERSC resources.

2. **Science teams need to be able to read, write, transfer, store online, archive, analyze, and share huge volumes of data.**

   2.1. The projects considered here collectively estimate needing a 10-fold increase in online disk storage space in three to five years.

   2.2. HEP researchers need efficient, portable libraries for performing parallel I/O. Parallel HDF5 and netCDF are commonly used and must be supported.

   2.3. Project teams need well-supported, configurable, scriptable, parallel data analysis and visualization tools.

   2.4. Researchers require robust workflow tools to manage data sets that will consist of hundreds of TB.

   2.5. Science teams need tools for sharing large data sets among geographically distributed collaborators. The NERSC Global File System currently enables high-performance data sharing and HEP scientists request that it be expanded in both size and performance.

   2.6. Scientists need to run data analysis and visualization software that often requires a large, shared global memory address space of 64-128 GB or more.

   2.7. Researchers anticipate needing support for parallel databases and access to databases from large parallel jobs.

3. **Science teams need guidance and support to implement their codes on future architectures.**

   3.1. NERSC's HEP users need help choosing programming models and implementing algorithms that can exploit the computing power of future NERSC systems. These systems are expected to include multi-core, many-core, and/or GPU architectures.

   3.2. Project teams want NERSC to host — and make available to its user community — testbed machines for developing, porting, and testing code to run on prototypes of possible future NERSC systems.

   3.3. Science teams need continued development of, and support for, third-party software like I/O and math libraries on new systems. Research teams have a large investment in existing codes that use such software, e.g., HDF5 and PETSc.

3.4. Some groups requested support for new programming models, such as partitioned global address space (PGAS) languages (e.g., UPC and Co-Array Fortran).

3.5. Some research teams require a full-featured OS with standard system calls, dynamic load libraries, and shared-object libraries, on future NERSC systems. This is to support standard codes and frameworks, high-level languages like Python, and visualization and analysis software.

4. **Projects need predictable, rapid turnaround of their computational jobs to meet mission-critical time constraints.**

4.1. Teams need stable, available, and reliable systems. Interruptions to production runs and workflows caused by system failures are expensive both in terms of lost computation time and the human effort needed to deal with disruptions. Additionally, on systems that suffer frequent interruptions, applications are forced to checkpoint often, at the cost of additional disk usage and lost simulation time.

4.2. Researchers need good throughput for many ensemble runs at modest concurrency. These runs are needed to search parameter space, test different assumptions, accumulate adequate statistics using different realizations of a single physical simulation, and/or verify and validate larger simulations.

4.3. Software developers need ready access to NERSC resources to develop code on schedule. Code development is necessary to augment and refine physical models, explore new algorithms, optimize code, and improve scaling. Rapid turnaround for small, medium and large jobs is necessary for development and testing.

4.4. Some projects need to perform real-time data processing on a fixed schedule and more projects are likely to require this in the future.

4.5. HEP scientists need workflow frameworks that allow sophisticated management of ensemble runs and failed jobs.

7.2. Other Significant Observations

1. In the next three to five years, many projects expect to be able to use 100,000 to 500,000 cores concurrently in their computations, a 10- to 50-fold increase over current runs.
2. Efforts by scientists to employ GPUs have yielded mixed results; often the solution a portion of a problem is amiable to GPU computing but others are not.
3. Manipulation and analysis of data is often performed in a collaborative environment.
4. Some groups are using mixed OpenMP/MPI hierarchical parallel constructs to run efficiently on current supercomputer architectures, but the groups recognize that this approach may not be effective in the long run.
5. Some codes run well using a small memory footprint per core (<2 GB), but others are difficult to run efficiently in an environment with limited local memory.
6. Projects are already limited by the space required to store data; the time needed to read and write it; and the limited facilities, software, and workflow tools needed to perform analysis on large data sets.
7. Some projects want to run many instances of their code for a long time on a small number of processors because they are limited by algorithm scalability or a particular input configuration.

7.3. Computing Requirements

The following table lists the computational hours required by research projects represented by case studies in this report, for projects that had an allocation of time at NERSC in 2009. "Total Scaled Requirement" represents the hours needed by all 2009 NERSC projects if increased by the same factor as that needed by the projects represented by case studies in this report.

NERSC Computational Project Title	Principal Investigator	Hours Needed in 3-5 Years	Factor Increase Over 2009 NERSC Usage
Lattice Quantum Chromodynamics			
Quantum Chromodynamics with three flavors of dynamical quarks	Doug Toussaint, U. Arizona	280 M	15
Total of projects represented by case studies		280 M	15
NERSC Lattice QCD Total Scaled Requirement		**341 M**	**15**
Accelerator Physics			
Petascale Particle-in-Cell Simulations of Plasma Based Accelerators	Warren Mori, UCLA	100 M	11
Particle simulation of laser wakefield particle acceleration	Cameron Geddes, LBNL	150 M	32
Community Petascale Project for Accelerator Science and Simulation	Panagiotis Spentzouris, FNAL	40 M	20
Advanced Modeling for Particle Accelerators	Kwok Ko, SLAC	160 M	123
Continuing Studies of Plasma-Based Accelerators	Warren Mori, UCLA	30 M	55
Total of projects represented by case studies		480 M	27
NERSC Accelerator Physics Total Scaled Requirement		**530 M**	**27**
Astrophysics			
Computational Astrophysics Consortium	Stan Woosley, UCSD	400 M	126
Cosmic Microwave Background Data Analysis For The Planck Satellite Mission	Julian Borrill, LBNL	50 M	54
Baryon Oscillation Spectroscopic Survey	Peter Nugent, LBNL	1 M	13
Total of projects represented by case studies		451 M	108
NERSC Astrophysics Total Scaled Requirement		**600 M**	**108**
Total Represented by Case Studies		**1,471 M**	**35**
All HEP at NERSC Total Scaled Requirement		**1,700 M**	**35**

8. NERSC Response: Initiatives and Plans

NERSC has initiatives underway already that address some requirements presented in this report. In addition, NERSC has long-term strategic plans that will help meet the needs of HEP research teams in five years, but this is dependent on funding increases to meet the computational and data needs, along with personnel increases in particular areas identified by the users. A summary of these initiatives and plans is presented in this section.

8.1. Compute Resources

NERSC plans to increase its total computational resources in the next 3 to 5 years, with the Hopper system (the NERSC-6 Project) to be installed in late 2010 and NERSC-7 approximately three years later. Hopper is a Cray XE6 Petaflop/s system with over 1 Petaflop/s of peak performance, which will nearly triple the center's total peak capacity when it goes into full production in 2011. Franklin (NERSC-5, with a peak capacity of under 0.4 Petaflop/s) will be decommissioned prior to NERSC-7 installation. Technology trends suggest that the replacement of NERSC-5 by NERSC-7 should again triple capacity with a similar footprint and hardware costs, but growing electrical costs will require budget increases even to meet this overall increase of 9x. To meet the 35x across the facility, as projected by the HEP science goals, increases in the NERSC budget will be required. The NERSC-8 system is currently planned for production in 2017, which is too late to meet the planned science needs between now and 2015.

Current price/performance trends in hardware, along with projected power and cooling needs, strongly suggest that without additional budget resources, NERSC will be underfunded to deliver the 35x increase in computing hours needed by HEP researchers by 2015.

8.2. Data Storage, Sharing, and Analysis

Over the next five years, NERSC will continue to grow and improve its data storage, bandwidth and sharing capacity in the NERSC Global Filesystem (NGF), including an increase in late 2010 that will double the shared project space to 1.5 Petabytes. NERSC plans to continue a constant investment in storage each year at the planned budget levels. However, the HEP research activities require a 10x increase in disk capacity in the next three to five years, which exceeds the projected increases based on the cost trends in storage capacity and bandwidth.

NERSC is also investing heavily in improving both capacity and bandwidth for the HPSS archival storage. Beginning in 2011, NERSC plans to add increased bandwidth to achieve 10 percent of the fastest file systems' aggregate bandwidth. This is in line with conventional bandwidth guidelines at other centers. NERSC is also working to significantly improve data movement between HPSS and NGF.

NERSC has been successfully fielding web-based science gateways that allow researchers to collaborate remotely for sharing and analyzing data, e.g. the Gauge Connection (qcd.nersc.gov). HEP users can leverage those interfaces to develop their own HEP Science Gateways, but additional staff are needed to ensure robust continued development and support of these interfaces.

8.3. Support for Future Architectures

NERSC will continue in its leading role in identifying performance, programmability, and other adoption issues raised by new architectures and in working with vendors and users to resolve them. However, the advent of radically different future architectures — such as many-core and GPU systems — will require more staff to help users adapt to the hardware and to support its associated software models.

NERSC is fielding early architecture testbeds, including a 42-node GPU system largely in response to this requirements workshop, and the Magellan testbed to explore various aspects of Cloud computing for science. NERSC has the expertise to play a sorely needed leadership role in the looming architecture and software transformation, but the group is small and staff and testbeds are largely supported by funds outside the NERSC Center budget. To translate these activities to training, software support, and hands-on consulting, all of which are personnel-intensive activities, the center will require higher staffing levels. While NERSC can field a testbed such as Magellan on a one-off basis, the center is not currently staffed to support multiple testbeds on an ongoing basis, nor to deal with associated software issues.

NERSC is aware of the varied operating system requirements among its HPC users and has worked closely with Cray to implement a "shared root" environment on its systems to support dynamic load libraries and thus support an expanded list of system calls and applications (e.g. python), compared those available on the original Cray XT compute-node operating system. At NERSC's behest, the new Hopper Cray XE6 system will support dynamic libraries and all vendor-supplied libraries will be available in both static and shared-object format. Users can choose between two operating systems: a limited kernel OS with static linking for minimal runtime intrusion and high scalability, or a more fully featured Linux OS that includes support for shared libraries.

8.4. Predictable, Rapid Job Turnaround

NERSC monitors job queues regularly and tries to optimize them to minimize waiting times, although in general, longer wait times are an effect of the DOE goal of running systems at high utilization. NERSC offers premium queues, for example, with double the charge factor, to shorten wait times, and we are exploring various ways of scheduling the resources to provide better predictability for some of the specific usage scenarios raised by this report. Through the Magellan project, NERSC is studying the applicability of a cloud-computing model for scientific applications. As part of this project NERSC is working with Adaptive Computing (the company behind the Moab Scheduler) to investigate advanced scheduling capabilities, including dynamic provisioning, virtual private clusters, and workflow scheduling, which might give NERSC users the kind of predictable throughput and specialized software services they are seeking. For the kind of short-turnaround jobs that arise in code development, NERSC continues to work closely with users to balance interactive use of computing resources with production runs. We have considered a separate system dedicated to development, but feel that a scheduling adjustment would provide more flexibility: a dedicated development machine would not support testing and debugging at scale, and would statically divide resources into development and production, would not allow provisioning the resources as workload changes. To address testing at scale, and in response to an earlier requirements workshop, NERSC is experimenting with allowing users to schedule dedicated time on the Cray XT4 Franklin system. Some users have already taken advantage of this reservation system to run and debug at scale interactively. NERSC is investigating the ramifications of offering such a service on a regularly scheduled basis to projects that require data processing on a predictable schedule.

NERSC has a long history of working closely with vendors to reduce system software and hardware failures. Vendors are subject to strict contractual uptime and availability metrics on the major NERSC systems. NERSC plans to have major systems overlap in lifetimes, typically supporting two major systems at a time. This strategy mitigates the impact of downtimes during a new system's "breaking in" period by having the stability provided by a mature system. As systems continue to scale component failures will continue to be a problem, and in anticipation of this trend, NERSC has been proactive in working with vendors on innovations to help users cope with transient failures. This includes an in-depth tracking and analysis of job failures, and support through the vendors for the Berkeley Lab Checkpoint Restart (BLCR) software. With the installation of Hopper, NERSC is addressing requirements related to both system failure mitigation and the varied OS needs of users. A key Hopper feature is its external services: login nodes and file systems that are external from the core XT/E system allowing users to login, compile, view files, and submit jobs even if the compute node array is down.

9. Accelerator Physics

Contributors: Panagiotis Spentzouris, Fermilab; David Bruhwiler, Tech-X Corp., Cameron Geddes, LBNL; Chengkun Huang, LANL; Lie-Quan Lee, SLAC

9.1. Accelerator Physics Overview

Particle accelerators are invaluable tools for making fundamental scientific discoveries and DOE has clearly identified them as critical facilities for advancing research. Of the 28 facilities listed in the 2003 DOE report Facilities for the Future of Science: A Twenty-Year Outlook, 14 involve accelerators. The development and optimization of accelerators are essential for advancing our understanding of the fundamental properties of matter, energy, space, and time directly in two of the three frontiers supported by the Office of Science program The Energy and The Intensity Frontiers, and indirectly in The Cosmic Frontier. High-energy collider experiments are used to discover new particles and directly probe the properties of nature, high-intensity beam experiments are used to uncover the elusive properties of neutrinos and observe rare processes that probe physics beyond the Standard Model, and precision measurements from accelerator experiments are required to interpret cosmological observations. Modeling of accelerator components and simulation of beam dynamics are necessary for understanding and optimizing the performance of existing accelerators, as well as for optimizing the design and cost-effectiveness of future accelerators.

In the next decade the HEP community will explore the energy frontier by operating the Large Hadron Collider (LHC), by designing LHC upgrades, and by developing the novel concepts and technologies needed for the design of the next lepton collider. The community will also be exploring the Intensity Frontier by designing and possibly operating high intensity proton sources for neutrino physics and rare process searches, and designing high intensity muon sources for neutrino factories. At the same time, it is imperative to maximize the physics reach of the ongoing DOE/HEP program by optimizing the performance of the Fermilab Tevatron accelerator complex. Additionally, DOE/HEP is supporting a world-class R&D program to develop new accelerator technologies, including laser and plasma wakefield accelerators, as well as other advanced accelerator concepts.

The design, cost optimization, and successful operation of such accelerators require the optimization of many parameters, and the understanding and control of many physics processes. This can only be accomplished using high fidelity computational accelerator models. In addition, massive computations requiring tightly coupled parallel computing and advanced algorithms are necessary to model many important physical processes that occur in accelerators. It is crucial to understand collective effects in beam dynamics, the properties of complex electromagnetic structures, and new accelerator technologies such as plasma and laser wakefield acceleration. High fidelity three-dimensional modeling of space charge effects, beam-beam interactions, generation of wakefields in electromagnetic structures, electron cloud effects, RF fields in structures with realistic geometry, and laser- and plasma-based systems can only be accomplished through large-scale high-performance computing (HPC) accelerator modeling. The increased complexity, precision, higher beam intensity and energy requirements — and ultimately the cost of the next-generation particle accelerators — further increase the demands on computational accelerator science. Not only is it required to accurately model single-stage, single-physics, single-scale systems, but it is also necessary to have end-to-end (multi-stage or complete system), multi-physics, multi-scale simulations. Such simulations can only be performed with extreme scale computing resources and with algorithms that can effectively exploit these resources. The results of the these simulations will allow designers to achieve high model accuracy, to shorten the turnaround time in designing and optimizing accelerators, and to significantly control costs by reducing the number of trial-and-error cycles for producing accelerator component prototypes.

9.2. Common Requirements Among Accelerator Science Projects

The accelerator physics case studies have the following requirements in common:

- The ability to run very large simulations using 200,000 cores in three to five years, 500,000 cores beyond 5 years;
- Scalable, efficient, and perhaps asynchronous I/O so run times at high concurrency are not dominated by I/O operations;
- Support for many mid-size ensemble runs for design optimization studies, using 10,000 cores in three to five years and 30,000 cores beyond five years;
- Robust workflows to handle job error detection and recovery;
- Tools to extract and organize massive amounts of data;
- Platforms to efficiently run large-scale parallel data analysis and visualization tools (e.g. VisIt and Paraview). Systems with large high-performance shared memory pools (e.g., 100 nodes with 128GB per node) are needed to do this;
- Support for flexible scripting languages like IDL and Python for driving analysis workflows;
- Support for shared libraries;
- Facilities for testing new hardware and programming paradigms and tools (e.g., GPU computing and UPC);
- R&D support for understanding how a change in programming paradigms will affect commonly used computational algorithms;

9.3. Accelerator Physics Case Studies

9.3.1. Community Petascale Project for Accelerator Science and Simulation

Principal Investigator: Panagiotis Spentzouris, Fermi National Accelerator Laboratory
Contributors: James Amundson, Fermilab; David Bruhwiler, Tech-X Corp.; Cameron Geddes, LBNL; Chengkun Huang, LANL; Lie-Quan Lee, SLAC
NERSC Repo: m778

9.3.1.1. Summary and Scientific Objectives

In the past seven years, under the Scientific Discovery through Advanced Computing (SciDAC) program, the Accelerator Science and Technology project and its successor, the Community Petascale Project for Accelerator Science and Simulation (ComPASS), have developed a powerful suite of HPC simulation tools that allow the development of large multi-scale, multi-physics accelerator simulations. The ComPASS (compass.fnal.gov) collaboration is currently developing a virtual accelerator modeling environment for realistic, inclusive simulation of beam dynamics effects (single and multi-particle dynamics, realistic geometry and parameters), a virtual prototyping environment for realistic simulation of all relevant accelerator component effects (thermal, mechanical, and electromagnetic), and a toolkit for supporting and guiding R&D for new high-gradient acceleration techniques, advanced accelerator modeling environment.

Currently, ComPASS codes are being applied in three major areas of interest to HEP:

- Large-scale electromagnetic modeling of superconducting RF cavities for the Project-X proton driver at Fermilab and the design of the proposed International Linear Collider;
- Assessment of the impact of wakefields on particle beam dynamics;
- Multi-physics modeling of the Fermilab accelerator chain for performance optimization under the current (Booster, Tevatron) and Project-X (Main Injector, Debuncher) operating conditions.

We also perform design optimization studies of accelerator components with complicated geometries such as in the LHC crab cavity, which includes couplers with very fine features. In addition, the project's applications assist the development of advanced accelerator concepts and support the experimental program of the BELLA and FACET facilities.

Most ComPASS applications are being run at NERSC, where the scientific computing infrastructure benefits the further development of the applications. The current NERSC HEP allocation for projects that use codes developed (or partially developed) under ComPASS is 2.2M core-hours distributed by HEP through the NERSC ERCAP allocation process and 4.2M core-hours awarded by the DOE INCITE program. ComPASS applications are used for both "discovery" and "design optimization," resulting in demands for both very large-scale and medium-scale runs. The medium-scale runs typically require many simulations using a few thousand cores per job. These runs are used for parameter scans and optimization studies and are referred to as large "volume" computations (many core-hours of jobs). The different application areas have different computational methods of solution:

- Beam Dynamics: electrostatic particle-in-cell (PIC) models.
- Electromagnetics: time and frequency domain solutions, finite difference techniques, and finite element models.
- Advanced Accelerators and Laser -and Particle-Driven plasma acceleration: full electromagnetic PIC and reduced PIC models.

This case study will cover the computational requirements and objectives of the beam dynamics area. The other areas are covered by the case studies in the following sections.

The project emphasis in the area of beam dynamics is on simulating different accelerators of the FNAL's Tevatron and the CERN's LHC accelerator complexes (Energy Frontier applications), and different designs of accelerators for FNAL's future proton driver (Project-X) complex (Intensity Frontier applications). These simulations aim to quantify and understand the effects of space-charge and beam-beam interactions, impedance, and electron-cloud effects on the performance of these accelerators. The goal is to use simulation to achieve higher beam throughput and minimize beam losses. Our simulations have played an important role in understanding space-charge effects at the Fermilab Booster, optimizing operating parameters at the Fermilab Tevatron in the presence of beam-beam and impedance effects, study beam-beam effects and their mitigation for LHC, and study space-charge and impedance effects in the Fermilab accelerator chain for the proposed Mu2e experiment and Project-X. We are currently moving from single physics to multi-physics simulations. The most computationally challenging applications are parameter scans for performance optimization of operating and design optimization of future accelerators. Such applications involve a large number of modest size runs, where the timely availability of the results is of great importance (especially for parameter optimization of operating accelerators). In most cases, the physics under study requires a long time to develop (but critically depends on the initial conditions and the evolution of the system), thus we need to run these simulations for a long time (many simulation steps) to accurately model beam dynamics effects. Given the queue wall-clock limits (and possibly stability issues) this can only be achieved by checkpointing and using many runs per simulation (see also wall clock requirements in table below).

9.3.1.2. Methods of Solution

The main codes used for ComPASS beam dynamics applications are Synergia, the Impact family of codes (ML/Impact and Impact), BeamBeam3D and NIMZOVICH. Synergia and ML/Impact are multi-physics parallel frameworks, while the other codes are single-purpose codes for different LINAC and ring applications. All the codes utilize electrostatic particle-in-cell algorithms with structured grids, with a variety of different strategies and solver implementations:

a) Particles: Depending on the physics of the problem, the codes might use domain decomposition, particle decomposition, or hybrid decomposition. There may be communication of particle data, grid data, or both. Particle movement between Poisson solves may be slight or large; hence, for some applications a particle manager is employed while in others no particle manager is used.

b) Solvers: Accelerator modeling codes utilize spectral based, finite difference based, and hybrid discretization algorithms, with both FFT and multi-grid based solvers.

Depending on the type of algorithm, we have different grid size limitations (due to memory requirements): a typical large grid size is 1024^3 for the solver implementation where both particles and grids are distributed (hybrid decomposition) and a typical size is 256^3 for the case where just the particles are distributed (particle decomposition). These typical problem sizes result in requirements of 10M to 100M macroparticles, depending on the type of simulation. In certain applications of the LINAC codes (Impact-T, for example) the physics under study (microbunching instability for light source design, for example) demands a very large number of macroparticles, on the order 1-5B.

Parallelism is expressed using the MPI message-passing library, the HDF5 library is used for I/O, some of our spectral solver implementations depend on the FFTW Fast Fourier Transform library, and analysis and visualization applications employ VisIt. The I/O of a typical job involves both checkpointing and the particle and grid dumps that are used for post-processing, including visualization, trace-back through particle history for “interesting” samples of particles (lost particles, for example), statistical analysis, etc.

Checkpointing requires that about 75 percent of the data stored in run-time memory be written per snapshot (mostly particle state information), with each new file typically overwriting the previous one. For post-processing analysis a typical particle dump is a fraction of one percent of the total number of particles times the number of dumps (typically on the order of 10,000), so a data volume equivalent to 50-100 checkpoint files is needed.

9.3.1.3. HPC Requirements

Production simulations for beam dynamics typically run on 2,000 cores for 48 wall-clock hours on systems such as Franklin at NERSC, Intrepid at ALCF, and Linux clusters. Most of the production runs involve parameter scans for performance or design optimization, and we've been experimenting with ensemble runs using MPI groups. In such runs the typical core count is 10,000. We are working to further incorporate workflows for true parallel optimization.

Beam dynamics parameter optimization runs require fast turnaround for medium size jobs (queue priority) and high availability and stability. For example, the recent campaign to improve Tevatron luminosity by optimizing chromaticity settings (a task requiring detailed representation of beam orbits and accurate modeling of beam-beam and impedance effects) started on NERSC's Franklin system, but because of stability issues (N.B., Franklin stability issues were largely resolved by 3Q 2009) the work was moved to Intrepid at the ALCF. The completion of this task required 5M hours on Intrepid, of which about 10 percent was used for development and about 10 percent was lost to failed jobs for a variety of reasons. The data generated and moved from such runs is approximately 1TB per run, split into many smaller files (~100,000 files). These are particle dump files (once per turn for a ring simulation) and various files containing on-the-fly calculated quantities per simulation step. The number of the particle dump files could be significantly reduced with more intelligent utilization of HDF5 capabilities.

Multi-physics, multi-scale simulations (currently used for Intensity Frontier studies, such as the optimization of extraction for the proposed Mu2e experiment at Fermilab and Project-X accelerators) employ beam dynamics frameworks that require shared libraries on the worker nodes. For example, the Mu2e extraction design and Main Injector simulations for Project-X (space-charge, electron-cloud, and impedance) are currently performed on Intrepid and on Linux clusters. This is not an ideal situation, because the resources we have available on Linux clusters are not adequate (both in core count and total number of hours available), while running long jobs of medium core counts on Intrepid is probably not the best match for a leadership-scale machine (although availability of such resources is very much appreciated!).

Access to a 50-fold increase in computing resources will allow the community to make significant advances in two key areas:

a) Understanding and controlling beam loss and activation in Intensity Frontier accelerators using simulations over the full range of relevant scales, from 10^{-3} m beams, to 10 m wakefields, to many 10^{3} m propagation. Such simulations require the deployment of multi-scale, multi-physics beam dynamics codes and will be important for the design optimization and operation parameter improvements of the short and mid-term FNAL plans (Project-X and current accelerator chain improvements). Accurate modeling of beam losses requires detailed modeling of the tails of the beam using a large number of macroparticles – 10 to 100 times more than used in current simulations. These simulation problems will be relevant in the next five years as the Intensity Frontier program of Fermilab is increasing in priority and getting more defined within HEP.

b) Maximizing luminosity in Energy Frontier accelerators by deploying and using multi-scale, multi-physics beam dynamics simulations. Such simulations will be important for helping maximize the output of the last years of the Tevatron, helping diagnose potential LHC problems, and contributing to the design of the next generation lepton collider.

In both cases, utilization of ensemble runs (without any algorithmic improvements) and parallel parameter optimization runs (with the development and implementation of appropriate workflow algorithms) will be essential. The availability of additional resources will allow this type of running to become the common mode of operation for fast turnaround of parameter optimization runs.

9.3.1.4. Computational and Storage Requirements Summary

	Current (2009 at NERSC)	In 3-5 Years
Computational Hours	2 M	40 M
Parallel Concurrency	2,048 - 32,000	10,000 – 100,000
Wall Hours per Run	48	72
Aggregate Memory Needed	2 TB	10 TB
Memory per Core Needed	0.5 GB	1 GB
I/O per Run Needed	100 GB-1 TB	200 GB-2 TB
On-Line Storage Needed	2 TB	15 TB
Data Transfer Needed	500 GB/week	2 TB/week

Note –requirements include both ensemble runs and smaller “single parameter set” jobs (thus the range in values). I/O size per run varies depending on visualization detail.

9.3.1.5. Support Services and Software

In order to develop such applications on "50X HPC resources" more processor hours will be required with better throughput for medium-size jobs (queue policies) and better access to debug queues (i.e., larger core counts and longer run times) to allow for scaling studies of new algorithms, especially for parallel optimization algorithms. In addition, error checking and recovery service implementation will be essential for the larger long-running jobs.

Robust parallel file I/O becomes an important issue, as is the development and availability of parallel visualization and analysis tools that will provide similar functionality to commonly used serial analysis tools such as MATLAB and Octave. Continuing support and development of VisIt, for example, on the future “50X HPC capability” infrastructure is important, since our beam dynamics frameworks utilize it for analysis and visualization.

The Synergia framework solvers depend on FFTW and PETSc, so support and further development of these libraries on any new hardware is a requirement for Synergia’s applications.

Finally, support of shared libraries is essential for any multi-physics framework application, since we use Python-driven frameworks with dynamic loading of physics modules during run-time.

9.3.1.6. Emerging HPC Architectures and Programming Models

The ComPASS team is starting a research program to understand how to effectively employ GPUs. The beam dynamics applications (for machine design and optimization) have two main components: particle tracking and field solves. Efforts to date have demonstrated efficient tracking with high-order-optics on GPUs. Field solves on GPUs and hybrid schemes involving a mixture of conventional processors and GPUs are being investigated. However, in order to design efficient multi-level parallelism schemes., the development team will need more information on the architecture of the future machines incorporating GPUs. It appears that extensive algorithmic changes will be needed, but It will be imperative to support the algorithmic development in a fashion that will not affect our current applications. Availability of test systems utilizing the new HPC architectures at NERSC will be very helpful, since it would allow consolidation and coordination of such efforts.

9.3.2. Advanced Modeling for Particle Accelerators

Principal Investigator: Kwok Ko, SLAC National Accelerator Laboratory
Contributor: Lie-Quan Lee, SLAC National Accelerator Laboratory
NERSC Repo: m349

9.3.2.1. Summary and Scientific Objectives

Beyond basic scientific research, particle accelerators benefit the nation across a broad range of applications in medicine, national security and industry. The goal of this project is to use advanced modeling to design and optimize the function of particle accelerators. A set of computationally intensive problems that will have significant impacts on accelerator R&D in the next three to five years has been identified. Tackling these problems is critical to the design, optimization and analysis of accelerator projects and their R&D. Results of computations are applicable to the LHC upgrade, Project X, Compact Linear Collider (CLIC), high gradient structure, Muon Collider, and laser acceleration.

9.3.2.2. Methods of Solution

High-fidelity modeling using high-order, curvilinear finite-element methods enables accurate solutions of complex accelerator structures with disparate length scales. In this method, a set of hierarchical H(curl) basis functions is employed to discretize the Maxwell's equations, which govern the behavior of electromagnetic fields inside accelerator structures. With the support of DOE's SciDAC-1 and SciDAC-2 programs, SLAC has developed a comprehensive suite of Advanced Computational Electromagnetics (ACE3P) codes, based on the parallel finite-element method, for accelerator applications in the areas of cavity design, wakefield calculation, dark current and multipacting simulation, RF gun modeling, and multiphysics (electromagnetic, thermal, mechanical) analysis. ACE3P has seven simulation modules for different applications: Omega3P, S3P, T3P, Pic3P, Track3P, Gun3P and TEM3P. Here is a brief description of the methods used in Omega3P for frequency-domain analysis and T3P for time-domain analysis; each has drastically different computational requirements.

For frequency domain analysis, Omega3P, a parallel finite-element eigensolver, has been developed for finding resonant modes in accelerator structures. The Maxwell's equations in

Omega3P are modeled as linear or nonlinear eigenvalue problems and the mathematical algorithms include explicit and implicit restarted Lanczos for solving real generalized eigenvalue problems, second-order Arnoldi for complex quadratic eigenvalue problems, inverse iterations and Jacobi-Davidson for complex nonlinear eigenvalue problems. The code also includes sparse direct solvers and Krylov subspace methods with a spectral multilevel preconditioner for shifted linear systems because the eigenvalues of interest are interior in most cases. Omega3P uses MPI for coarse-grained parallelism. Inside the basic linear algebra operations, vendor libraries are often used, which can provide more fine-grained parallelism through threading to take advantage of on-node parallelism. Examples of such vendor libraries include libsci_mp from Cray and libessl_mp from IBM.

For time-domain analysis, T3P has been developed to study wakefields created by beam-environment interactions following the transit of a particle bunch through an accelerator structure. It uses the same finite-element discretization as that in Omega3P for space and the Newmark-beta scheme for implicit time stepping. At each time step, a linear system with different right hand sides is solved. The matrix in T3P is symmetric positive definite and is solved using the conjugate gradient method with a block Jacobian preconditioner where each core owns one block and performs an incomplete factorization. The method has been proved to be very efficient and scalable.

9.3.2.3. HPC Requirements

In a typical simulation, the number of elements (N_e) in the mesh and the basis function order (p) for each element together determine the matrix size N. For example, using 2nd order basis functions (p=2), the matrix size is roughly $N = 6.2 * N_e$, and with p=3, the matrix size is $N = 18 * N_e$. In frequency-domain analysis, the run time of Omega3P is dominated by the time for solving the algebraic eigenvalue problem. In the time-domain analysis, the run time is proportional to the number of time steps while a sparse linear system is solved at each time step. The time step is determined by the element size and the highest frequency to be resolved in the structure. The time step is chosen so that at each step the beam or particles will not move more than one element size.

In the next three to five years, the team expects to improve the spatial resolution of simulations by two to three times, which corresponds to the increase of computational problem sizes by a factor of 20 to 100. Here are two extreme-scale computational requirements. For frequency domain analysis using Omega3P, researchers want to have 64 GB to 128 GB memory per node (not per core) and about 1,000 nodes in order to model large, complex accelerator structures proposed in the scientific objective section. However, realizing that such systems may not be available, we are developing algorithms to reduce the need for large-memory nodes, including discontinuous Galerkin methods for frequency-domain analysis and novel solvers. The state-of-the-art sparse direct solvers that are currently in Omega3P have non-scalable per-node memory usage. Simply increasing the node count will not necessarily solve problems that do not fit into memory with smaller node counts. More details are discussed in the next paragraph.

For time domain analysis using T3P, higher core count jobs will be required to improve spatial and temporal resolution in simulations of wakefields and perform self-consistent particle-in-cell calculations. The computational resource requirements for the two cases are drastically different and are summarized separately.

For eigenvalue problems using Omega3P, of the linear systems are highly indefinite because the eigenvalues of interest are interior and a spectral transformation is needed to solve the algebraic eigenvalue problem. One way to solve these linear systems is to use sparse direct solvers.

Unfortunately, sparse direct solvers suffer from imbalanced and non-scalable per-node memory usage. Scalability of per-node memory usage is defined similar to scalability of speed, but its focus is on how the memory is consumed by the application code, not on how fast the code is executed. The amount of physical memory available on each node certainly is the most significant constraint on how large a problem size can be simulated. The team has been developing new algorithms to address the memory usage scalability issue. For example, a spectra multi-level preconditioner has been developed, enabling solution of problem sizes one order of magnitude larger than before. In addition, working with scientists at SciDAC Centers for Enabling Technology, the team is developing a general-purpose hybrid linear solver, which will further improve the scalability of memory usage.

The project needs the following to improve its computational capabilities:

1) Use more accurate workload and communication models in the partitioning scheme for better load balancing and domain decomposition. When the number of cores used in the execution is larger than 50,000, the imbalance can be significant as the current partitioning scheme uses the number of elements as the balancing measure while the actual workload is proportional to the number of degrees of freedom. By using a more accurate workload model, the load is expected to be more balanced.

2) Investigate using a more scalable linear solver in the computational kernel of the finite element simulation suite.

3) Explore the discontinuous Galerkin method for accelerator modeling. This is a high-risk, high-yield research activity. If successful, it could greatly improve the scalability of simulation, possibly up to one million cores.

9.3.2.4. Computational and Storage Requirements Summary

	Current (2009 at NERSC)	In 3-5 Years
Total Computational Hours	1.2 M	160 M

Frequency Domain Analysis with Omega3P

	Current (2009 at NERSC)	In 3-5 Years
Computational Hours		10 M
Parallel Concurrency	1,024 to 8,192	10,000
Wall Hours per Run	5	10
Aggregate Memory	16 TB	128 TB
Memory per Node	8 GB to 16 GB	128 GB
I/O per Run Needed	1 GB	50 GB
On-Line Storage Needed	1 GB	50 GB
Data Transfer Needed	1 GB	50 GB

Time domain analysis with T3P

	Current (2009 at NERSC)	In 3-5 Years
Computational Hours		150 M
Parallel Concurrency	8,192	75,000
Wall Hours per Run	12	20
Aggregate Memory	12 TB	100 TB
Memory per Core	1.5 GB	1.5 GB
I/O per Run Needed	1 TB	10 TB
On-Line Storage Needed	500 GB	5 TB
Data Transfer Needed	100 GB	1 TB

Access to a 50-fold increase in HPC resources will allow the following key scientific goals to be met:

1) Accurately determine wakefield effects of beam-environment interactions with realistic bunch size for large complex accelerator structures to understand the performance of an accelerator and to provide information for further design optimization. This cannot be done now because it requires a threefold increase in spatial resolution in order to resolve the bunch size. This translates into 50 to 100 X computational resources compared to what's currently available.
2) Model self-consistent field-particle interactions in space-charge dominated devices such as electron sources over long time scales with high accuracy to provide capabilities for designing high-quality and high-brightness beams for basic and applied scientific research. This cannot be done now because longer time-scales with high-accuracy requires finer spatial and temporal resolution that is beyond the capability of currently available resources.
3) Understand dark-currents and RF breakdown issues that limit accelerating structures from operating at high gradients so as to provide insights for designing more efficient accelerating structures. Today's resources limit this kind of understanding because issues like RF breakdown require multi-scale, multi-physics simulation involving not only self-consistent electromagnetic analysis but also surface physics and other analysis.

9.3.2.5. Support Services and Software

Visualization of large data sets will be increasingly difficult as the problem size increases. Researchers need infrastructure support for remote and interactive visualization and analysis. Specifically, they need NERSC to support the latest versions of Paraview and parallel netCDF. Paraview is the software used for visualization and analysis because of its superb unstructured mesh support. VisIt is a possible alternative, but it is better suited for applications using structured grids. Team members have compared the images generated by VisIt and Paraview and have concluded that images from Paraview are far superior because of its native support for unstructured grids and curvilinear elements.

The parallel netCDF library is the I/O library used for checkpointing and outputting data.

The team requests that NERSC make emerging architectures available to its users to explore future architectures and programming models. Certainly this is better than individual teams acting alone.

9.3.2.6. Emerging HPC Architectures and Programming Models

The SciDAC accelerator team has employed a hybrid MPI/thread programming model in their sparse linear algebra kernel. They attempt to implement a full-scale hybrid programming model in other parts of their codes such as in mesh handling.

They are also actively exploring GPU computing and other alternatives. In summer 2009, the team hired an intern to study the effectiveness of iterative linear solvers on a single Nvidia GPU on a Dell workstation.

The design of their simulation codes is very modular, making it s easy to swap some components in the codes with their alternatives, or to plug in new components for the same functionality. For example, they can easily incorporate a new linear solver or a new preconditioner, which usually is the key to better scalability and performance. If new components that can more efficiently use emerging many-core or heterogeneous architectures, they expect to be able to quickly adapt them to their simulation codes.

9.3.3. Electromagnetic Modeling of Accelerator Structures

Prepared by: David Bruhwiler, TechX Corp.
Contributors: J.R. Cary, K. Amyx, T. Austin, B. Cowan, P. Messmer, P. Mullowney, K. Paul, P. Stoltz, D. Smithe and S. Veitzer, Tech-X Corp.; G. Werner, U. of Colorado
Supports the following HEP funded NERSC repositories:

“Community Petascale Project for Accelerator Science and Simulation (m778),” Principal Investigator: P. Spentzouris

“Particle simulation of laser wakefield particle acceleration,” Principal Investigator: C.G.R. Geddes

“Simulation of photonic crystal structures for laser driven particle acceleration,” Principal Investigator: B. Cowan

9.3.3.1. Scientific Objectives

The overarching scientific objective for the next three to five years is to enable rapid high-fidelity simulation and design of a wide range of accelerating structures of relevance to the Office of High Energy Physics.

For example, modeling of superconducting radio frequency (SRF) accelerating cavities requires rapid and accurate calculation of frequencies for all resonant modes (fundamental and high-order), the associated Q (quality factor) for each mode, and also the spatial structure of the modes. Surface heating and multipacting are key physical processes that limit the gradient of SRF cavities and must be modeled, with emphasis on moving from analysis to designs that reduce risk and cost while improving performance for new accelerator facilities. Relevant DOE/HEP applications include the Large Hadron Collider (LHC), Project X and the International Linear Collider (ILC).

Normal conducting (warm) RF cavities and waveguides are also critical technologies for present and future facilities. There is a worldwide effort to accurately simulate RF breakdown in warm structures. This will require major advances in modeling surface physics under extreme conditions, as well as the ability to couple very small scale surface simulations to large-scale electromagnetic simulations. A particular example explores the concept of “magnetic insulation” of novel RF cavities for muon acceleration. Relevant DOE/HEP applications include muon collider and neutrino factory concepts, RF power transport, and CLIC-like alternatives to the baseline SRF design of the ILC.

High-gradient RF cavities based on dielectric structures are key to the “advanced concepts” portfolio of research and development within the DOE Office of High Energy Physics. Two examples include: a) laser-driven photonic band gap (PBG) accelerating cavities; and b) novel, larger-scale RF structures with ultra-high Q and ultra-low wakefields. Field emission and secondary emission of electrons from dielectric surfaces, and resulting surface damage, are important issues that can limit the achievable gradients and must be addressed in future simulations.

9.3.3.2. Methods of Solution

The main project code, VORPAL, is a parallel framework for finite-difference time domain (FDTD) simulations of fields and particles of various types, employing a variety of algorithms. VORPAL uses an explicit stencil to update E and B fields with 2nd-order accuracy on structured 1D, 2D and 3D Cartesian meshes. The operators are all local, which enables efficient communication via MPI and excellent scaling up to 16,000 cores.

The use of cut-cell techniques makes it possible to accurately treat complicated metal geometries, and a recently developed 2nd-order FDTD algorithm for dielectrics with arbitrary geometry makes it possible to do new large-scale simulations of complicated RF cavities made from novel dielectric structures. Using the broadly filtered diagonalization technique, frequency related information can be obtained efficiently from a time-domain code. Multi-physics capabilities, like secondary emission of electrons, are made available in VORPAL through the freely available TxPhysics library. Algorithms for coupling implicit heat advection in metal surfaces to the explicit update of Maxwell's equations in vacuum are present in VORPAL and are under active development.

9.3.3.3. HPC Requirements

Recent simulations have been modest in size, typically using 10^6 to 10^7 cells and running for fewer than $1x10^5$ time steps on approximately1,000 cores. A 50-fold increase in resources would allow modeling of 50X larger structures by increasing the size of the mesh, while holding resolution constant. This would enable VORPAL to simulate full cryomodule assemblies containing multiple SRF cavities. Also, a 50X increase in resources would enable VORPAL to directly address multi-physics problems, coupling EM and heat transport solvers over order-of-magnitude longer time scales, and also coupling surface physics models to much larger scale EM simulations in order to understand and eventually mitigate RF breakdown physics.

Developers anticipate that VORPAL simulations will scale well from ~1,000 cores at present to ~100,000 cores in the near future, as the problem size is expanded to meet future challenges over the next three to five years. In addition, developers envision a new mode of routine operation, in which ~100 VORPAL simulations each using ~1,000 cores are run in a task-parallel fashion on ~10,000 cores, under the control of a parallelized nonlinear optimization algorithm. This approach to parallel computing at NERSC will enable a much-needed shift from the present workflow — doing a few large simulations in order to obtain physical insight — towards a new and more powerful workflow of optimizing existing accelerator designs and also creating completely novel designs.

9.3.3.4. Computational and Storage Requirements Summary

	Current (2009 at NERSC)	In 3-5 Years
Computational Hours	< 1 M	100 M
Parallel Concurrency	~1 K	~100 K
Wall Hours per Run	10	10
Aggregate Memory	10 GB	1 TB
Memory per Core	0.01 GB	0.01 GB
I/O per Run	10 GB	1 TB
On-Line Storage Needed	< 1 TB	~5 TB
Data Transfer	1 TB	100 TB

9.3.3.5. Support Services and Software

Parallel file I/O using HDF5 must be scaled to hundreds of thousands of processors.

Error checking and job-relaunch services that detect if a job has terminated partway through and automatically restart it will become more important as jobs take up increasing numbers of nodes, with corresponding increases in the possibility of failure.

To allow simulations to predictively guide experiments, scans of parameter space are needed (as are conducted in experiments), which will require the above job monitoring services together with automation to generate and run sequentially large numbers of jobs, and to extract the data from them.

Visualization services for visualizing and analyzing large datasets, and in extracting physics data from them, are important. VisIt and IDL are important for visualization. Parallel visualization and analytics tools must be further developed, to provide similar functionality to well-known serial tools such as IDL/MATLAB while providing access to petascale datasets. This is a serious challenge, as even simple operations such as smoothing require communication or guard cells. The tools need to be robust and script-callable so as to be integrated into a batch workflow providing automatic analysis after batch compute jobs complete.

9.3.3.6. Emerging HPC Architectures and Programming Models

The FDTD electromagnetic algorithms of VORPAL have already been ported to the NVIDIA GPU hardware, showing more than an order of magnitude speedup over modern processor performance, updating >$4x10^8$ cells per second. The new implementation works on a simple heterogeneous system – a small Opteron cluster with one GPU per node. Effective use of multiple GPUs per Opteron core is being considered and no major technical difficulties are anticipated. Hence, the FDTD algorithm has been shown to be well situated to take advantage of these new architectures, and we expect to obtain comparable benefits for simulations that include particles.

9.3.4. Simulation of Laser Plasma Wakefield Particle Accelerators

Principal Investigator: Cameron G.R. Geddes, LBNL
Contributors: E. Cormier-Michel, E. Esarey, C.B. Schroeder, J.-L. Vay, W.P. Leemans, LBNL; D.L. Bruhwiler, J.R. Cary, B.M. Cowan, C. Nieter, K. Paul, V. Ranjabar , Tech-X Corp.
NERSC Repository: m558

9.3.4.1. Summary and Scientific Objectives

Laser-plasma acceleration of charged particles shows great promise for reducing the cost and size of next-generation electron and positron accelerators. Plasmas are not subject to the electrical breakdown that limits conventional accelerators; accelerating gradients thousands of times those obtained in conventional accelerators have been obtained using the electric field of a plasma wave (wakefield) driven by an intense laser. Such accelerators will be important to scale beyond TeV energies for high energy physics and to provide brighter and smaller (laboratory- and hospital-scale) radiation sources including free-electron lasers, Thomson sources and ultrafast THz oscillators.

This project's emphasis is on simulating experiments run at LBNL's LOASIS center, and planning future experiments to be run there. The goal is to quantitatively understand internal dynamics, evaluate controlled injection, understand beam propagation and improve interpretation of diagnostics. The plasma interaction in this regime is fully nonlinear, and particle distribution effects are important, making simulation essential but challenging. Time-explicit particle-in-cell (PIC) simulations have played a key role in supporting the rapid progress of laser plasma accelerators, including the physics behind the first narrow energy spread beams (Nature 2004), and GeV e-beams in 3cm (Nature Physics 2006). Applications of laser-plasma to colliders and advanced radiation sources (such as X-ray free-electron lasers or Thomson scattering generation of gamma-rays) require high-quality electron beams, and we therefore work both to improve numerical modeling of beam injection and propagation (PRE 2008, SciDAC Review 2009) and use the codes to understand how controlled injection of particles can improve beam quality (PRL 2008).

The team is now designing next-generation experiments at 10 GeV and staging of multiple LWFAs for the BELLA plasma wakefield laser facility in progress at LBNL, as well as controlled injection experiments to increase beam quality and stability. Over the next three years the BELLA facility will become operational, and is designed to test collider-relevant accelerator stages. In the three-to-five year time frame, the simulations will therefore be developed to accurately design efficient 10 GeV stages for this facility, to understand staging of multiple modules and transport of low emittance bunches required for collider and other applications, as well as to design injectors to create the required low emittance bunches. Other laser facilities are being planned in the same time frame. To address these goals, the project uses codes which include three-dimensional, time-explicit particle-in-cell, laser envelope particle-in-cell, fluid plasma models, and Lorentz boosted simulation frames as described in detail below. There are other laser plasma accelerator simulation projects at NERSC using codes with similar algorithms, including OSIRIS.

Note that electron beam-driven plasma accelerators are also being developed, including the new FACET facility at SLAC, as detailed by Tsung et al. A key difference between laser and electron beam driven plasma accelerators is that in the laser case the laser wavelength (micron) must be resolved, while in the electron beam-driven case the longitudinal resolution requirement may be significantly less stringent. Additionally, the laser wavelength shifts as the laser depletes its energy into the wake, which makes laser cases more challenging for some reduced models, requiring increased resolution.

9.3.4.2. Methods of Solution

The main project code, VORPAL, is a parallel framework for finite-difference time domain (FDTD) simulations of fields and particles of various types, employing a variety of algorithms. Its algorithms and computational requirements are representative of PIC codes used in other projects such as OSIRIS. Fields and fluids are represented on a structured Cartesian mesh, while particles move through space. For this project, VORPAL is being used to model laser-plasma interactions and the associated particle acceleration. In these simulations we solve the relativistic Maxwell's equations. The electron plasma is usually represented by particles via PIC (particle-in-cell) methods, but can also be represented as a cold, charged fluid. In addition to the standard algorithm, which fully resolves the laser wavelength, a computationally faster method can be used that represents the laser fields with an "envelope" that interacts with a full PIC treatment of the wakefields in the plasma. Computations can be conducted in a Lorentz-boosted computational frame used to minimize disparity in scales between the laser and plasma and hence reduce the number of required time steps. Field-induced tunneling ionization and electron-impact ionization can be included. Particle tracks can be exported for radiation calculations. Collisions can also be modeled.

In simulating electromagnetics, VORPAL uses an explicit stencil to update E and B fields with 2nd-order accuracy on the standard dual Yee mesh. Relativistic particle motion is modeled by the 2nd-order leap-frog algorithm, and VORPAL uses high-order spline-based shapes for the current deposition. Linear finite difference operators are appropriately centered in space and time, to obtain global 2nd-order accuracy. E and B fields are leap-frogged in time. Particle (or fluid) and field updates are also leap-frogged.

The operators are all local, which enables local communication via MPI and excellent scaling up to 8,000 processors on Franklin. We have run production simulations up to 11K-processors, with excellent scaling, and are limited in going further primarily by allocated processor hours and by machine availability (that is, it is easier to schedule 11K-processors for 24 hours than 33K for 8 hours). A primary issue to be resolved (as noted elsewhere) is scaling of parallel I/O (such as HDF5) for such machines.

For simulations using the laser envelope model, the Trilinos library suite (Aztec) is used to iteratively solve a Crank-Nicholson treatment of the field update.

The project also makes use of other PIC and fluid codes that use computational methods similar to those in VORPAL. A primary example is Warp, which is a both a code and a general purpose framework for parallel three-dimensional Particle-In-Cell simulations of beams in accelerators, plasmas, laser-plasma systems, non-neutral plasma traps, sources, and other applications. It contains multiple field solvers (electrostatic FFT, multigrid, electromagnetic), internal conductors (cut-cell method with electrostatic solver), surface physics (space-charge limited emission, secondary emission of electrons or gas from impact of electrons or ions), and volumetric ionization. It employs advanced methods such as cut-cell boundaries, and has been used to develop Lorentz boosted computational frame techniques and Adaptive Mesh Refinement.

9.3.4.3. HPC Requirements

Based on past simulations, and using recently developed algorithms to reduce cost at the same time increased resources are available, it is anticipated that of order 50X scaling in computational resources will be needed to accurately design collider-scale stages.

Past explicit-PIC simulations of 100-MeV experiments required million-hour simulations to resolve the laser wavelength (nm) over the acceleration length (mm). As noted above, these simulations run on 11K cores at NERSC with excellent scaling. The plasma length and diameter must both increase to increase beam energy, and meter-scale 10 GeV BELLA and collider relevant stages would then require at least a billion hours with no new computational models and without accounting for the higher resolution needed to provide increased accuracy in emittance transport. Because of the increased plasma size, we anticipate that as many as 500,000-processors can be used for these simulations with weak scaling.

Newly developed computational models will be used in conjunction with new computers to simulate the required stages. Laser envelope models reduce costs by not resolving the laser period, while performing calculations in a Lorentz-boosted frame reduces cost by reducing the disparity in scale between the laser and plasma characteristic lengths. These models can save of order 10,000X in computational resources for meter-scale 10 GeV stages.

Using these models, it is anticipated that 10 GeV stages can be simulated using ~50X resources with order 10X the resolution of current simulations, as will be needed to provide accurate modeling of emittance transport. At the same time, multiple 3D runs will be possible both to explore parameter space to improve beam quality and to simulate staging of the beam through multiple modules for high energies. Many runs at modest resolution will be made possible, and will be very important to allow simulations to predictively explore parameter space to guide experiments rather than being restricted to a few runs conducted after experiments to explain results. High-resolution simulations of the particle injector will also be conducted to determine what combination of techniques is required to produce the required emittances for colliders and other applications.

Multiple models may need to be integrated to handle the physics of each stage of the accelerator. These may include explicit PIC in the laboratory frame to handle the injection (creation) of the particle beam, envelope, quasistatic, or boosted frame PIC for long accelerator stages, and fluid models for the plasma structure together with PIC or other representations of the accelerated beam to reduce noise. Radiation and scattering contributions to beam quality will need to be modeled as well. Free-space propagation modules will be used to model the drift of the beams between stages in multi-stage collider designs to be developed, and for focusing.

9.3.4.4. Computational and Storage Requirements Summary

	Current (2009 at NERSC)	In 3-5 Years
Computational Hours	4.7 M	150 M
Parallel Concurrency	11,000	500k (large) / 50k (sml)
Wall Hours per Run	24	12 / 12
Aggregate Memory	100 GB	100 TB / 10 TB
Memory per Core	<0.1 GB	0.5 GB / 0.5 GB
I/O per Run Needed	2 TB	50 TB / 5 TB
On-Line Storage Needed	2 TB	50 TB
Data Transfer Needed	5 TB	10 TB

Note – requirements reflect need for a few large runs and many smaller runs, noting typical sizes for each. Due to good scaling, longer clock times with fewer processors could be used depending on queue policies.

9.3.4.5. Support Services and Software

Parallel file I/O using interfaces such as provided by HDF5 must be scaled to 10's to 100's of thousands of processors, and must be made robust to varying mesh sizes on different processors.

Error checking and job-relaunch services that detect if a job has terminated partway through and automatically restart it will become more important as jobs take up increasing numbers of nodes, with corresponding increase in failure possibilities.

To allow simulations to predictively guide experiments, scans of parameter space are needed (as are conducted in experiments), which will require the above job monitoring services together with automation to generate and run sequentially large numbers of jobs, and to extract the data from them.

Visualization services for visualizing and analyzing large datasets, and in extracting physics data from them are important. VisIt and IDL are important for visualization. Parallel visualization and analytics tools must be further developed to provide similar functionality to well-known serial tools such as IDL/Matlab while providing access to petascale datasets. This is a serious challenge, as even simple operators such as smoothing require communication of guard cells. The tools need to be robust and script-callable so as to be integrated into a batch workflow providing automatic analysis after batch compute jobs complete.

Envelope simulations will require continued development of Aztec/Trilinos and FFTW libraries. The laser envelope model in VORPAL makes use of the Aztec00 component of the Trilinos library from Sandia National Lab (SNL). This is also true for the Poisson solver and the implicit electromagnetic updates in VORPAL. In scaling such simulations up to and beyond 100k cores, we will depend on NERSC to continue supporting Trilinos and we are assuming Trilinos development at SNL will continue, so that the library remains relevant on future supercomputing hardware. The PETSc and FFTW libraries are viable alternatives for future use with VORPAL, both for new algorithmic development and also for replacing Trilinos, so it would be very valuable if NERSC would continue to support these libraries as well.

9.3.4.6. Emerging HPC Architectures and Programming Models

Tech-X is working actively to develop VORPAL for GPUs and for other advanced architectures, and preliminary results are very promising. For further details see input of D. Bruhwiler. Related PIC codes such as UPIC (with which we collaborate under SCIDAC) have also shown good results on GPU architectures, as has VPIC on cell (Roadrunner). The internal structure of the codes must be reorganized in some cases to take advantage of these architectures. Advanced accelerator simulations generally are not memory bound, and use large numbers of processors with relatively little memory/processor. Communication of the edge information from each processor to processors handling neighboring domains is required at each step, so that communication is important (and may need to be multi-layered on many-core or GPU systems). The PIC algorithm is hence well situated to take advantage of these new architectures.

9.3.5. Petascale Particle-in-Cell Simulations of Plasma Based Accelerators

Principal Investigator: Warren Mori, UCLA
Contributors: F. S. Tsung, UCLA, C. K. Huang, LANL
NERSC Repos: incite14, mp113

9.3.5.1. Summary and Scientific Objectives

For the past 80 years, the tool of choice in experimental high-energy physics has been the particle accelerator. The Large Hadron Collider (LHC) at CERN came online in 2008. The construction cost alone for the LHC machine was nearly 10 billion dollars and it is clear that if the same technology is used, the world's next “atom smasher” will cost at least several times that in today’s dollars. The long-term future of experimental high-energy physics research using accelerators depends on the successful development of novel ultra high-gradient acceleration methods. New acceleration techniques using particle beams/lasers and plasmas have already been shown to exhibit gradients and focusing forces more than 1,000 times greater than conventional technology, raising the possibility of ultra-compact accelerators for applications in science, industry, and medicine.

In plasma-based acceleration the Coulomb force of a particle beam or the radiation pressure of a laser beam pushes (or pulls) to create a plasma wake that moves near the speed of light. The accelerating gradients in plasma wakefields are more than 1,000 times higher than in conventional accelerators. Properly placed particles surf these wakes to ultra-high energies. Plasma-based accelerator science and engineering has been a fast-growing field due to a combination of breakthrough experiments, parallel code developments, and a deeper understanding of the underlying physics of the nonlinear wake excitation in the so-called blowout regime. In a recent PWFA experiment at SLAC, electrons in the tail of a 42 GeV electron beam were accelerated out to ~80 GeV in only 80 cm. This corresponds to a greater than 40 GeV/m energy gain for nearly one meter! In recent LWFA experiments at LBNL, quasi-monoenergetic electron beams at 1GeV have been reported (in a recent experiment by scientists from UCLA and Lawrence Livermore National Laboratory, maximum electron energy of 1.7 GeV have also been observed). In each case the wakefield was excited in the nonlinear regime in which plasma electrons are radially expelled. Additionally, in the past few years, parallel simulation tools for plasma-based acceleration have been verified against each other [paul:2008], against experiment [blumenfeld:07], and against theory [lu:07].

Based on this progress in experiment, theory and simulation, linear collider concepts using wakefields have been developed and two facilities in the U.S. have been approved [tuttle:09]. The FACET facility at SLAC will provide 25 GeV electron and positron beams as drivers for plasma wakefields. The other facility is BELLA at LBNL, which uses a 30 Joule/30 fs laser as its driver. The goal for each facility is to experimentally test key aspects of a single plasma accelerator stage within the multi-stage collider concepts. Furthermore, there are other lasers within the U.S. and in Europe and Asia that are currently or will be able to experimentally study LWFA in nonlinear regimes.

While some simulations will be conducted to help design and interpret near-term experiments, an important step forward is to use advanced simulation tools to study the feasibility of a plasma-based linear collider concept and to optimize the parameters of the conceptual designs. This work will dramatically advance the rate of discovery and progress in plasma-based accelerator research. Because much of the physics involved in plasma-based acceleration is nonlinear such that fluid approaches are not appropriate, particle-in-cell modeling is generally necessary. In the case of the UCLA simulation group, these tools include a fully explicit particle-in-cell code, OSIRIS [fonseca:08] and a quasistatic particle-in-cell code, QuickPIC [huang:06]. These codes are described in the following section.

9.3.5.2. Methods of Solution

Two codes are used to study the problem of plasma-based accelerators (both PWFA and LWFA): one code is OSIRIS, which is a relativistic, fully-explicit particle-in-cell code which uses an FDTD (finite-difference, time domain) solver for the electromagnetic fields; and a second order Boris solver for the particles. OSIRIS is used mainly to study the LWFA problem in the lab frame, and also in the boosted frame [vay:07,martins:09]. It is also used to study the electron trapping in PWFA and used extensively to benchmark QuickPIC, which employs the quasistatic model for the LWFA and the PWFA problems.

QuickPIC is a highly efficient, fully parallelized, fully relativistic, three-dimensional PIC code for simulating particle- or laser beam-driven wakefield acceleration. The algorithm is based on the quasi-static approximation, which separates the time scales of the driver and plasma evolution and reduces a fully three-dimensional electromagnetic field solve and particle push for the plasma to a sequence of two-dimensional transverse field solves and particle pushes. The particle or laser beam can be updated using a time step much larger than the plasma oscillation period. Overall this algorithm speeds up the computational time by two to three orders of magnitude without losing accuracy for problems of interest.

QuickPIC solves the reduced Maxwell's equations (Poisson equations for the vector and scalar potentials) in Fourier space. It is built on a PIC framework (UPIC) developed at UCLA. The framework supports periodic, conducting boundary conditions using FFT (or Fast Sine/Cosine Transform for better performance). The particle beam pusher uses a 2D domain decomposition that allows small transverse cell size, a major computation challenge for the plasma-based linear collider design with tightly focused particle beams. The two-dimensional field solve and particle push is implemented with 1D domain decomposition and dynamic load balancing. The scaling limitation of the spectral solver due to the global all-to-all communication pattern can be relaxed using a pipelining algorithm [feng:09] that divides a large computation task (domain) into several smaller consecutive ones that are then pipelined. QuickPIC achieves good strong scaling up to 16,384 cores on Franklin at NERSC with the pipeline algorithm.

Because the LWFA problem is described elsewhere, we will focus our attention on PWFA simulations using QuickPIC in the rest of this section.

9.3.5.3. HPC Requirements

The design of the FACET PWFA experiments at SLAC is shown in Figure 7.2.5.1, where electron or positron beams with widths on the order of 10 microns will be accelerated to 50 GeV from 25 GeV in a meter-long plasma section. This experiment will come online in the next two years (2010-2011) and the goal is to achieve high quality (low energy spread, high energy efficiency and emittance preservation) accelerated beams through fine-tuning the experimental parameters. Each individual simulation for this experiment will have about the same computational requirement, e.g., 128 processor cores for 12 hours or 512 processor cores for three hours, as in the previous energy doubling experiment and can be readily simulated on the existing computing resource. However, many simulations of this type are needed to explore and optimize the parameter dependence and to guide the experiment. This is different from the previous simulation mission, which was to explain the experiment outcome. A 50X increase of the resources will enable simultaneous multi-parameter optimization and acceptable turn-around time for experimental feedback. The ability to secure the required computing resource for a dedicated period of time would be beneficial.

Another big need for computing resources is for the design study of the plasma-based linear collider concept. A linear collider based on PWFA will use 20 stages, with each stage resembling the FACET experiment. However, a linear collider requires 100~1,000 times smaller beam emittance than currently achievable in the FACET experiment and therefore requires a (transverse) beam size of tens of nm, increasing the problem size by more than 10,000 times. At this extreme beam size, the plasma ions will also affect the emittance transport and the simulation needs to properly resolve their motion. Currently, we can model only a small fraction (<20 percent) of the beam transport distance under such conditions with the computing resources allocated to us each year at NERSC.

The problem described in the previous paragraph requires 100 wall-clock hours on 8,192 processors on the Franklin supercomputer. The simulation uses 8192 x 8192 x 1024 cells, and requires roughly 1 GB of memory per core (including temporary memory) and 8 TB total. Each simulation produces roughly 100 GB of data (mostly grid quantities such as electron densities and fields). To finish such a full scale PWFA simulation requires five times our current computing time allocation. As the emittance of the witness beam becomes asymmetric as needed by a collider, the transverse resolution of the simulations will need to increase by a factor of three; therefore, we expect the computation requirements to increase 15 fold in the next three to five years.

9.3.5.4. Computational and Storage Requirements Summary

	Current (2009 at NERSC)	In 3-5 Years
Computational Hours	10 M	30 M
Parallel Concurrency	8,192 cores	25,000
Wall Hours per Run	100	100
Aggregate Memory	8 TB total	25 TB total
Memory per Core	1 GB	1 GB
I/O per Run Needed	100 GB	1,000 GB
On-Line Storage Needed	4,500 GB	30 TB
Data Transfer Needed	100 GB/month	1 TB/month

The computation and storage requirements for the QuickPIC PWFA simulation described in this section. Note that the online storage and data transfer requirements reflect combined need of simulations for the FACET experiments and PWFA linear collider conceptual design.

9.3.5.5. Emerging HPC Architectures and Programming Models

High Performance Computing (HPC) has been dominated for the last 15 years by distributed memory parallel computers and the Message-Passing Interface (MPI) programming paradigm. The computational nodes have been relatively simple, with only a few processing cores per node. This computational model appears to be reaching a limit, with several hundred thousand simple cores in the IBM Blue Gene. The future computational paradigm will likely consist of much more complex nodes, perhaps with graphical processing units (GPUs) or Cell processors, which can have hundreds of processing cores requiring different and still evolving programming paradigms, such as CUDA. One anticipates that new HPC computers, unlike Blue Gene, will consist of a relatively small number (<1,000) of nodes, each of which will contain hundreds of cores. To obtain high performance on the node will, in most cases, require new algorithms, which are being developed by Viktor Decyk of our group, and some of the programming techniques learned on the GPU can also be applied on current multi-core processors. Nonetheless, between nodes it is likely that MPI will continue to be effective.

9.3.5.6. References

[fonseca:08] R. A. Fonseca et al, Plas. Phys. Cont. Fus., 50, 124034 (2008).
[huang:06] C. K Huang et al, Jour. Comp. Phys., 217, 658 (2006).
[martins:09] S. F. Martins et al, submitted to Nature Physics (2009).
[vay:07] J. L. Vay, Phys. Rev. Lett., 98, 130405 (2007).
[feng:2009] B. Feng et al, Jour. Comp. Phys., 228, pp. 5340-5348 (2009).
[paul:08] K. Paul et al, AIP Conference Proceedings, 1086, 315 (2008) (Proc. Adv. Accel. Concepts Workshop, Santa Cruz, CA 2008).
[lu:07] W. Lu et al, Phys. Rev. STAB, 10, 061301 (2007).
[blumenfeld:07] I. Blumenfeld et al, Nature, 445, p. 741 (2007).
[tuttle:09] Symmetry magazine, page 22, volume 06, Issue 05, October 09

10. Astrophysics Modeling and Simulation

10.1. Astrophysics Modeling and Simulation Overview

Computational astrophysics encompasses many research areas of interest and relevance to high-energy physics. Modeling of explosive astrophysical events, including Type Ia supernovae, gamma-ray bursts, X-ray bursts and core collapse supernovae is needed, not only for the quantitative understanding of the mechanics of supernovae, but also because all of these types of events produce unique nucleosynthesis products responsible for nearly all of the elements in the solar system and in living creatures. Additionally, detailed simulations of these objects, in conjunction with astrophysical data, can shed new light on the physics of particle interactions and the properties of fundamental particles. Scientific efforts at the Cosmic Frontier provide unique opportunities to discover physics beyond the Standard Model and directly address fundamental physics questions involving the study of energy, matter, space and time. Numerical simulations are currently the dominant tool in theoretical astrophysics and cosmology.

There are two known mechanisms that create supernova, one involving the collapse of a massive star's core and the second resulting from a thermonuclear explosion of material accreted onto the surface of a white dwarf in a binary star system. A core collapse, or Type II, supernova occurs at the end of the life of a massive star when the iron core collapses to nuclear matter density and then bounces. Type Ia supernovae result from the thermonuclear burning of a carbon-oxygen white dwarf that accretes mass from a companion until it approaches a critical mass (the Chandrasekhar mass, beyond which the white dwarf itself would begin to collapse) at which nuclear carbon burning is ignited. Type Ia supernovae are significant scientifically because they are used as "standard candles" in determining the size and expansion rate of the universe.

In the same way that supernova experiments provide a standard candle for determining absolute brightnesses (and therefore distances from us), the clustering of baryonic material (e.g. galaxies, dark matter, intergalactic gas) imprinted by sound waves in the early universe, or Baryon Acoustic Oscillations (BAO), provides a "standard ruler" for length scale in cosmology.

In this section we provide case studies for three key areas of astronomical simulation: Type Ia Supernova, core-collapse Type II supernova, and cosmic structure signatures imprinted by BAO.

10.2. Astrophysics Modeling and Simulation Case Studies

10.2.1. The Cosmic Frontier – Structure Formation

Principal Investigator: Michael Norman, University of California, San Diego

10.2.1.1. Summary and Scientific Objectives

The study of Baryon Acoustic Oscillations (BAO) has emerged as the most promising probe of cosmic dark energy. BAO leaves an imprint on all kinds of cosmic large-scale structure at a characteristic wavelength of ~150 Megaparsec (Mpc). It has been detected in galaxy large-scale structure by the Sloan Digital Sky Survey (SDSS), and predicted in the “Lyman alpha forest” of absorption lines seen in the spectra of distant background quasars as their light passes through intervening gas. Observing this feature at a variety of redshifts will give astronomers a “standard ruler“ to measure the expansion history of the universe and thereby the dark energy equation of state. BAO is complementary to standard candle techniques, such as high-redshift supernovae, and they together form the basis for dark energy surveys. Unlike high-redshift supernovae, though, BAO is a true standard (i.e., it does not require calibration). The BAO wavelength is unique, and set by the simplest physics: it is simply the distance a baryon acoustic wave can travel in the photon-baryon fluid until decoupling (recombination), which is now precisely measured using cosmic microwave background observations. It therefore offers, in principle, the most precise measurements of the dark energy equations of state parameters, provided observational systematic errors can be corrected.

We will use very large cosmological hydrodynamic and N-body simulations of structure formation to simulate BAO observations of both galaxies and Lyman alpha absorbers in gigaparsec volumes in order to better quantify observational systematics in current and upcoming surveys (DES, BOSS, LSST, JDEM). The dynamic range requirements are very large, necessitating grids/particles of 4,0003 or more, and adaptive multiscale methods.

Two specific goals for the next five years are:

> (1) Carry out a small suite of hydrodynamic Adaptive Mesh Refinement (AMR) simulations of the Lyman alpha forest in a 1 Gpc volume with a base grid of 40963 particles/cell and a maximum resolution of 1 kpc including UV and X-ray background photoheating, optically thin radiative cooling, star formation and feedback.
>
> (2) Carry out a small suite of N-body dark matter only simulations of galaxy large-scale structure in a 2 Gpc volume with 8,1923 particles and a force softening length of 1 kpc.

10.2.1.2. Methods of Solution

The primary code used in this project is Enzo, but GADGET-3 will also be used when it becomes available. Enzo solves the equations of collisionless N-body dynamics and multispecies hydrodynamics on a structured adaptive mesh (AMR). The N-body solver employs the particle-mesh algorithm, while gas dynamics employs the piecewise parabolic method (PPM). Poisson’s equation is solved using FFTs on the base grid and local multigrid solves on the AMR subgrids. We are transitioning this to a global FAC multigrid solver using the Hypre library. Enzo is a hybrid MPI/OpenMP code.

The current largest uniform grid calculations are $4{,}096^3$ particles/cells. The current largest AMR calculation is $1{,}024^3$ base grid/particles with seven levels of refinement that generates over 1 million subgrids.

The GADGET-2 code is a gravitational N-body code that uses the tree-particle-mesh (TPM) algorithm. It supplements a long-range gravity solve on a uniform root grid with a local tree solve. GADGET-2 is an MPI parallel code and a hybrid parallel MPI/OpenMP version (GADGET-3) is under development. GADGET-2 was used to carry out the 10-billion particle Millennium Simulation in 2005 at the Max Planck Institute.

10.2.1.3. HPC Requirements

In the near term, we will perform a small suite of $4{,}096^3$ uniform grid Enzo simulations of BAO in the Lyman alpha forest with different assumed dark energy equations of state to assess observability and systematic effects. Longer term, we will transition to $4{,}096^3$ AMR simulations using Enzo in order to better resolve high column density absorbers and reduce the galaxy cross sections. Such simulations are expected to have tens of millions of subgrids and require higher levels of concurrency to execute in a reasonable amount of wall time.

The team will also use GADGET-3 to carry out simulated galaxy large-scale structure surveys in gigaparsec volumes when the code is available. Initially these will be done with 4,0963 particles, moving to 81923 particles when feasible.

These simulations will produce large amounts of data—at least 100 TB per run. Total archival storage in the first few years will be 1-2 PB, growing to 5 PB by year five.

10.2.1.4. Computational and Storage Requirements Summary

	Current (not at NERSC in 2009)	**In 3-5 Years**
Main science driver	Enzo unigrid @ 4096^3	Enzo AMR @ 4096^3 GADGET @ 4096^3, 8192^3
Computational hours	12 M	100 M
Parallel concurrency	16,384	100,000
Wall hours per run	600	1000
Aggregate memory	8 TB	100 TB
Memory/core	2 GB	4 GB
I/O per run	200 TB	500 TB
Online storage	100 TB	250 TB
Data transfer	100 TB/month	100 TB/month
Archival Storage	0.5 PB	5 PB

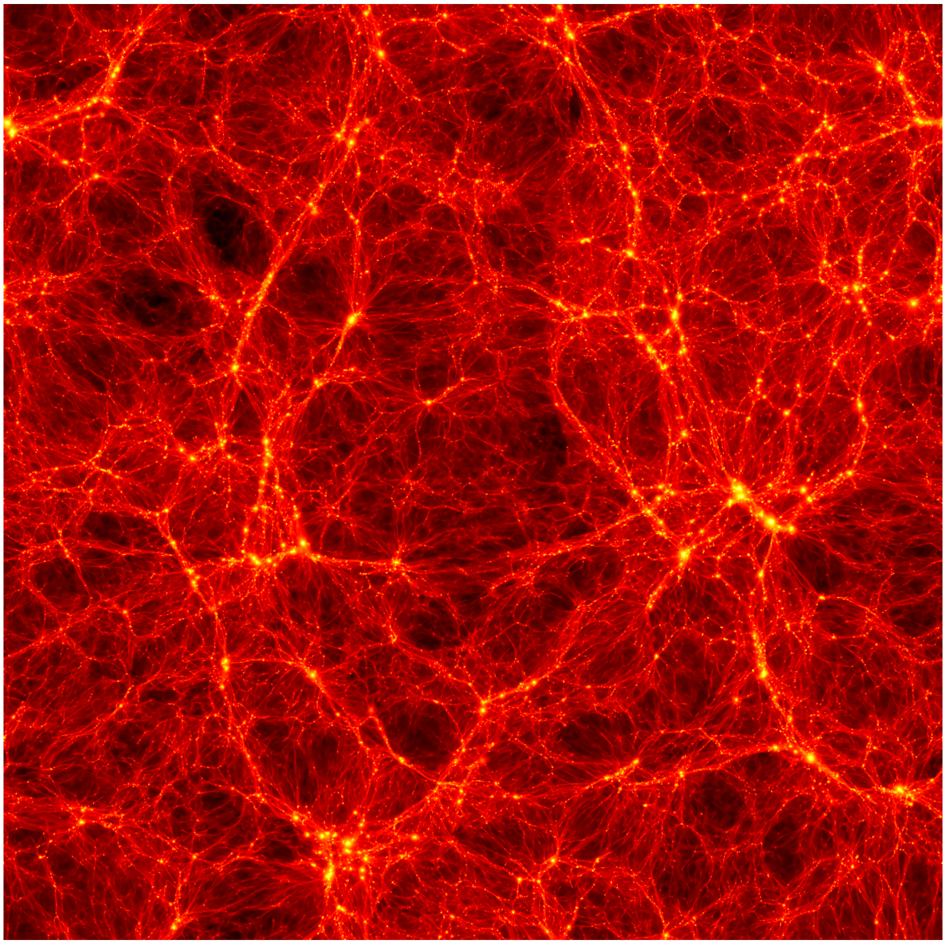
10.2.1.5. Support Services and Software

Our codes depend on optimized FFT and multigrid linear solver libraries. We currently use our own scalable FFT solver and the hypre multigrid solver library for elliptic solves. We use on HDF5 for I/O.

Scalability problems with large uniform grid runs are manageable. I/O is currently our largest bottleneck. The $4{,}096^3$ simulation generates 8-TB snapshot files.

10.2.1.6. Emerging HPC Architectures and Programming Models

There are substantial scalability problems with very large AMR simulations that researchers believe can be solved with the UPC programming language and runtime environment. The team would like to partner with NERSC to implement a hybrid MPI/UPC version of Enzo. Scalability is impacted in the current implementation of Enzo due to replication of the AMR grid hierarchy in every process. Moving to hybrid MPI/OpenMP parallelism has mitigated this, but it is not believed this will take the project to petascale. The second thing that limits scalability is the current N-body solver, which assigns particles to the subgrid object that occupies their region of space. While this encapsulation is convenient, it leads to excess MPI traffic as particles move from one grid to another. The team would like to use UPC to store the AMR grid hierarchy and particle lists as global objects that get pointed to by specific processors.



Hydrodynamic cosmological simulation of BAO in the Lyman alpha forest. The simulation used 2,0483 particles and cells in a cubic domain 330 Mpc on a side and was performed on NERSC's POWER 3 IBM Seaborg supercomputer in 2006. Shown here is the projected gas density through the simulation cube. The filamentary patterns are not the BAO signal, which is too subtle to see, but rather filaments and voids cause by the clumping of dark matter, which is also modeled in the simulation. From Norman et al. (2009).

10.2.2. Type Ia Supernovae

Principal Investigators: Stan E Woosley, UC Santa Cruz: John Bell, LBNL
Contributors: Ann Almgren, Andy Aspden, LBNL; Daniel Kasen, Haitao Ma, UC Santa Cruz; Michael Zingale, M. J. Malone. SUNY Stony Brook
NERSC Repository: m106

10.2.2.1. Project Summary & Scientific Objectives for the Next Five Years

A Type Ia supernova (SN Ia) occurs when a carbon-oxygen white dwarf accreting mass in a binary system reaches a critical mass and explodes. No gravitationally bound remnant is left behind and most of the star is burned to iron-group and intermediate mass elements. Simple parameterized 1D models give qualitative agreement with the observed light curves and spectra, but just how all this works in detail has been a robust problem in astrophysics for over 50 years [1]. For almost as long, the problem has been studied with numerical simulation [2]. The solution is interesting because SN Ia are the origin of most of the iron-group elements in nature and because their highly regular light curves are useful as cosmological "standard candles." Those deviations of the luminosity that do occur from a standard value at peak are correlated with other measurables, especially the width of the light curve, and this empirical calibration has proven sufficiently precise for SN Ia to reveal the first convincing evidence for an accelerating expansion of the cosmos. This acceleration has been attributed to "dark energy."

In the future, SN Ia observations will be used at a higher level of precision. Without a full understanding of the explosion physics, there may be uncertainties arising from evolution of the progenitor population over the billions of years and large variety of galaxy types that are studied. It is also hoped that complete models of SN Ia will provide additional diagnostics of peak luminosity besides the well-known "width-luminosity correction" and that the spectral evolution of the models can provide a useful data base for interpreting the observations. Large surveys (JDEM, LSST, PTF, Pan-STARRS, etc.) will also discover a wide variety of supernova-like transients and advances in simulation are needed in order to understand the variety of transients that will be – and in fact already have been – seen. DOE is interested in this modeling effort because of the nucleosynthesis (NP), the connection to dark energy experiments and missions (HEP), and the need for advanced simulation and radiation transport in the models (ASCR, NNSA). The NSF and NASA are also interested for similar reasons.

Understanding SN Ia from first principles is difficult because simulation results are quite sensitive to how the thermonuclear runaway is ignited (e.g., at the center or slightly off center on one side); to the details of turbulent (nuclear) combustion once ignition is achieved; and to the multi-dimensional treatment of radiation in an exploding medium that may not be in thermal equilibrium. It is a hard problem, but one where numerical simulation is already making major headway.

The overall "SN Ia problem" can be segregated into four separable sub-problems, each occurring one after the other, and each treatable with codes specifically appropriate for the purpose. Each problem requires a three dimensional calculation with as fine a resolution as is feasible. This is because the Reynolds number in the star is very high (10^{12}) and some of the most interesting problems involve turbulence. Resolving at least the integral scale of that turbulence is essential.

1) Ignition. For several centuries prior to the explosion the accreting white dwarf burns carbon stably in its center. Accretion is negligible during this time and the energy from nuclear fusion is distributed throughout the star by convection. The white dwarf thaws as both its central and average temperature increase by about a factor of two. Because of the extreme sensitivity of carbon fusion (T23), the evolution accelerates towards the end, with the most interesting time being the last hour as the temperature approaches 7 x 10^8 K. At this point, convection becomes unable to carry away the catastrophic rise in central energy and carbon starts to burn in localized hot spots. Once the temperature reaches about 9 x 10^8, the burning becomes essentially instantaneous. Carbon and oxygen then burn completely to iron and the temperature rises to 10^{10} K. The flame is born.

2) Burning. Once a region has ignited and burned to 10^{10} K, its density goes down about 15 percent and the matter becomes buoyant. Because of its huge density, the gravitational acceleration inside a white dwarf star is very large, even near the center. In a tenth of a second, burning plumes float at over 1000 km/s. At the same time the mass of the ash grows because of continued burning at its edge, the “flame.” Floatation leads to Kelvin-Helmholtz instability and the ash itself, once it becomes large enough, is Rayleigh-Taylor unstable. Jointly, these instabilities lead to turbulence that reacts back on the burning front, increasing its area, folding the flame, and making it move faster. Next to the uncertain ignition conditions, the effective flame speed for this burning is the second most important parameter of interest. The burning at this stage is still subsonic so the white dwarf expands ahead of the flame, eventually making it go out. A faster flame thus burns more of the white dwarf before dying and makes a brighter explosion.

The scenario just described is a classic problem in turbulent combustion. How does a flame move in response to the turbulence that it itself creates? It is a very different problem from terrestrial combustion, though, for a variety of reasons – the extreme temperature sensitivity of the burning rate, the importance of gravity, the large Reynolds number, the large length and time scales, and the very large turbulent energy to name a few. Initially, the flame propagation can be followed using MAESTRO, but when the Mach number becomes greater than about 0.1 we must switch to a compressible code, CASTRO. This is the same code being used to study core collapse supernovae, but here, the radiation transport is not such a major issue. Simple thermal conduction suffices. Instead, though, one needs special coding to follow the flame. We are adapting tools from the combustion community. A level set will be used to track the flame interface and a turbulent subgrid model will describe the burning below the grid resolution. The flame needs much greater resolution than the rest of the star and the star expands so adaptive mesh must be employed, along with multiple time steps.

3) Transition to detonation. So far, all pure deflagration models, i.e., models where the flame speed always remains subsonic, fail to give agreement with the most common SN Ia events. The models are too faint and move too slowly. This reflects inadequate burning and a sub-energetic explosion. Spectroscopically, common SN Ia supernovae also show no evidence for the large masses of unburned carbon and oxygen ejected in pure deflagrations. These problems can be remedied if the subsonic flame, after burning through most of the star, makes a transition to a detonation (a prompt detonation is not allowed, as it would not produce adequate quantities of intermediate mass elements like silicon and calcium that are abundant in the spectrum).

The physics of this transition is very uncertain. It might happen due to compression effects as “tongues” of flame break through the surface of the star or, in the Chicago model, because of collisions between waves of fuel pushed by the breakout to the far side of the star. Or it might happen spontaneously as the flame enters a regime of highly turbulent, slow burning. In the so-called "distributed regime," hot ash and cold fuel can mingle on a fine scale producing a potentially explosive mixture. Transition to detonation in an unconfined medium without hard obstacles is an interesting issue in modern combustion science that has no terrestrial analog, but it may be possible in the unusual supernova environment due to the high degree of turbulence and large length scales.

4) Spectra and light curves. In order to critically evaluate hydrodynamic explosion models and to optimize future dark energy missions, the radiative transfer problem in expanding supernova atmospheres must be addressed. The radiative transfer code SEDONA, which uses a Monte Carlo approach, takes the output of the hydrodynamic models (i.e., the density, velocity and compositional structure of the material ejected in the explosion) and synthesizes emergent model spectra, light curves and polarization, which can then be compared directly against observations. This provides the ultimate validation of the model predictions. Very few multidimensional light curve calculations of supernovae have been attempted in the past. The team's recent survey of 2D SN Ia supernova light curve models —run primarily on Jaguar at ORNL — were the first to systematically explore the effects of asymmetry on the light curves of supernovae. Two-dimensional simulations, however, do not properly characterize the geometry of the turbulent, explosive burning. At NERSC, the team plans to calculate light curves and spectra for a number of 3D explosion models.

References

[1] F. Hoyle and W. A. Fowler. Nucleosynthesis in Supernovae, Astrophysical Journal, 132, 565, 1960.

[2] W. D. Arnett. A Possible Model for Supernovae: Detonation of 12C. Astrophysics and Space Science, 5, 180, 1969.

10.2.2.2. Current HPC Usage and Methods

MAESTRO

MAESTRO is a low Mach number hydrodynamics algorithm for simulating stratified flows in astrophysical conditions. In the low Mach number formulation [1,2], pressure is decomposed into a dynamic and thermodynamic component, the ratio of which is of order Mach number (M) squared. The star is in hydrostatic equilibrium, so a base state density is defined (the average density at a given radius) which is in hydrostatic equilibrium with the base state (thermodynamic) pressure. The total pressure is replaced by the thermodynamic pressure everywhere, except in the momentum equation. The result is that sound waves are filtered out of the system. To enforce the thermodynamics, an elliptic constraint on the velocity field is enforced, derived from constraining the pressure to be constant along particle paths. The resulting system allows for much larger time steps to be taken than with corresponding compressible codes (>> 1/M larger). The elliptic constraint is similar to that of incompressible hydrodynamics, modified to account for background stratification with the addition of sources corresponding to the compressibility effects of thermal diffusion and reactions. The algorithm [3, 4, 5] utilizes a second-order accurate approximate projection method, developed first for incompressible flows. First, the species density, velocities and enthalpy are advanced to the new time level using an unsplit Godunov method. As part of this process, reactions are incorporated through an operator splitting approach that retains second-order accuracy of the overall algorithm. The updated thermodynamic variables are then used to evaluate the generalized divergence constraint that represents compressibility effects due to heat release and thermal diffusion associated with low Mach number reacting flow models. The provisional velocities are then projected onto the space that satisfies this divergence constraint. The projections involve elliptic solves, computed using the multigrid algorithm. MAESTRO is written primarily in Fortran 95 and is parallelized using MPI.

CASTRO

CASTRO is based on an Eulerian radiation hydrodynamics code that includes self gravity and reaction networks. The radiation module, which supports multigroup diffusion radiation, is not used for SN Ia modeling (see the core collapse case study, below). CASTRO incorporates hierarchical block-structured adaptive mesh refinement and supports 3D Cartesian, 2D Cartesian and cylindrical, and 1D Cartesian and spherical coordinates. The hydrodynamics in CASTRO is based on the unsplit methodology introduced by [6]. The code has options for the piecewise linear method in [6] and the unsplit piecewise parabolic method (PPM) in [7]. The unsplit PPM has the option to use the less restrictive limiters introduced in [8]. All of the hydrodynamics options are designed to work with a general convex equation of state. CASTRO supports two different methods for including Newtonian self-gravitational forces. One approach uses a monopole approximation to compute a radial gravity consistent with the mass distribution. The second approach is based on solving the Poisson equation for the gravitational field. The Poisson equation is discretized using standard finite difference approximations and the resulting linear system is solved using geometric multigrid techniques. A third approach in which gravity is externally specified is also available.

Our approach to adaptive refinement in CASTRO is based on the BoxLib package developed at Berkeley Lab by the Center for Computational Sciences and Engineering (CCSE). It uses a nested hierarchy of logically rectangular grids with simultaneous refinement of the grids in both space and time. The integration algorithm on the grid hierarchy is a recursive procedure in which coarse grids are advanced in time, fine grids are advanced multiple steps to reach the same time as the coarse grids and the data at different levels are then synchronized. During the regridding step, increasingly finer grids are recursively embedded in coarse grids until the solution is sufficiently resolved. An error estimation procedure based on user-specified criteria evaluates where additional refinement is needed and grid generation procedures dynamically create or remove rectangular fine grid patches as resolution requirements change.

CASTRO uses the same interface to equations of state and thermonuclear reaction networks as MAESTRO. This general interface allows to the easy incorporation of different approximations for nucleosynthesis. It also enables users to switch from a low Mach number simulation with MAESTRO to a fully compressible simulation using CASTRO without changing the underlying physics.

The parallelization strategy for CASTRO is to distribute grids to processors using MPI. This provides a natural coarse-grained approach to distributing the computational work. When AMR is used a dynamic load balancing technique is needed to adjust the load. Both a heuristic knapsack algorithm and a space-filling curve algorithm are used for load balancing. Criteria based on the ratio of the number of grids at a level to the number of processors dynamically switches between these strategies.

SEDONA

SEDONA is a multi-dimensional time-dependent multi-wavelength radiation transport code that calculates the light curves and spectra of supernovae using an implicit Monte Carlo approach

[13]. SEDONA is coded in C++ and uses MPI and OpenMP for parallelization. In order to access the full memory resources of massively parallel machines, developers have implemented scalable algorithms for domain decomposition. The specific method alternates between propagation and communication cycles: Monte Carlo particles reaching the boundary of the local processor domain are buffered until the end of the propagation phase, then communicated in batches to the neighboring processor using MPI. Once the number of nodes needed to store the 3D stellar model has been determined, the model can be replicated over a large number of processors in a manner that is trivially parallel. These advances have allowed the team to compute the first high-resolution 3D spectra of SN Ia.

A major part of the radiative transfer simulation is computing the wavelength-dependent opacity of the supernova debris. This requires reading large atomic data files and the calculation of numerous atomic level populations. Most previous SEDONA calculations have adopted the simplifying assumption of local thermodynamic equilibrium, which requires only a solution of the non-linear Saha/Boltzmann equations. However, the team has also written a module to solve the nonlocal thermodynamic equilibrium (NLTE) rate equations (in the Sobolev approximation), which consist of a set of >> 10, 000 coupled non-linear equations. Because of the computation expense, NLTE calculations have so far only been attempted for 1D models, however in the future these will be extended to 2D and 3D.

The following changes to codes, mathematical methods and/or algorithms will be needed to achieve this project's scientific objectives over the next five years.

- For the ignition problem and for full star explosion models, level-set front tracking will need to be developed. That has mostly been done, but needs testing.

- The turbulent subgrid model for burning needs further development, though much of that will occur through smaller off-line calculations. This will be done in one year.

- Detonation physics needs to be included in the code to follow the models after a transition to supersonic burning occurs. The nucleosynthesis and nuclear energy generation rates need more off-line study. This is a few-month project.

- For spectroscopic and light curve studies, multi-dimensional non-LTE transport has not yet been implemented, although no fundamental roadblocks are foreseen. The team is working on load balancing optimization in order to increase efficiency of the domain-decomposed version of the code. Processes that have heavy packet loads, due to either high values of the radiation field or the opacity, will be replicated on additional processors.

References

[1] M. S. Day and J. B. Bell. Numerical simulation of laminar reacting flows with complex chemistry. Combust. Theory Modeling, 4(4):535–556, 2000.

[2] J. B. Bell, M. S. Day, C. A. Rendleman, S. E. Woosley, and M. A. Zingale. Adaptive low Mach number simulations of nuclear flame microphysics. Journal of Computational Physics, 195(2):677–694, 2004.

[3] A. S. Almgren, J. B. Bell, C. A. Rendleman, and M. Zingale. Low Mach number modeling of type ia supernovae. I. Hydrodynamics. Astrophysical Journal, 637:922–936, February 2006. Paper I.

[4] A. S. Almgren, J. B. Bell, C. A. Rendleman, and M. Zingale. Low Mach number modeling of type ia supernovae. II. Energy evolution. Astrophysical Journal, 649:927–938, October 2006. Paper II.

[5] A. S. Almgren, J. B. Bell, A. Nonaka, and M. Zingale. Low Mach number modeling of type ia supernovae. III. Reactions. Astrophysical Journal, 684:449, 2008. Paper III.

[6] P. Colella. Multidimensional Upwind Methods for Hyperbolic Conservation Laws. Journal of Computational Physics, 87:171–200, 1990.

[7] G. H. Miller and P. Colella. A Conservative Three-Dimensional Eulerian Method for Coupled Solid-Fluid Shock Capturing. Journal of Computational Physics, 183:26–82, 2002.

[8] P. Colella and M. D. Sekora. A limiter for PPM that preserves accuracy at smooth extrema. Journal of Computational Physics, submitted.

[9] M. J. Berger and P. Colella. Local adaptive mesh refinement for shock hydrodynamics. Journal of Computational Physics, 82(1):64–84, May 1989.

[10] J. Bell, M. Berger, J. Saltzman, and M. Welcome. A three-dimensional adaptive mesh refinement for hyperbolic conservation laws. SIAM J. Sci. Statist. Comput., 15(1):127–138, 1994.

[11] A. S. Almgren, J. B. Bell, P. Colella, L. H. Howell, and M. L. Welcome. A conservative adaptive projection method for the variable density incompressible Navier-Stokes equations. Journal of Computational Physics, 142:1–46, May 1998.

[12] F. Miniati and P. Colella. Block structured adaptive mesh and time refinement for hybrid, hyperbolic + n-body systems. Journal of Computational Physics, 227:400–430, 2007. 3

[13] D. Kasen, R. C. Thomas, and P. E. Nugent. Time Dependent Monte Carlo Radiative Transfer Calculations for Three-Dimensional Supernova Spectra, Light Curves and Polarization. Astrophysical Journal, 651:366, 2006.

10.2.2.3. HPC Requirements: Today

Following the runaway during the last hour leading up to ignition in a SN Ia requires a special code and a large computer. Convection can only be properly described in 3D and previous studies have shown that the convective flow takes on a dipole-like character. Matter rises from the center in a jet-like (but highly subsonic) outflow, moves around the outer perimeter and comes back in from the other side. Consequently, when ignition finally occurs, it does not happen in the exact center of the star, but displaced to one side, in the outflow. The magnitude of the displacement and the size of the initial burning region determine the final explosion energy and brightness in a major way. Thus, following this highly subsonic (M = 0.001 – 0.01) convection coupled to nuclear burning is critical. Yet the buoyancy driving the convection is very small, Dr/r ~ 10-4, and following the flow with an ordinary compressible code is impractical, as well as expensive.

This is why we have developed the new low-Mach number code, MAESTRO (see above). This new code is optimized for following highly subsonic convection. While applicable to many other problems in astrophysics where convection plays a role, its initial application is to the SN Ia problem.

Our studies have already shown that the answer will depend on the numerical Reynolds number that can be achieved. The actual value in the star is unachievable, but it should suffice to go well into the turbulent regime, Re > 1,000. The team's most recent calculations with 3,843 zones had a Re number of only a few hundred. Each calculation took 600,000 processor hours per run on 1,728 processors on Jaguar at ORNL. Next year, the team plans to run with 7,683 zones and estimates that a typical run then will take 4.8 M processor hours. MAESTRO currently scales well to 10,000 cores and efforts are continuing to improve overall performance and scaling for the algorithm. Eventually, two dozen runs will be needed to explore the dependence on ignition density, white dwarf rotation rate, and carbon mass fraction.

Two other groups – the Chicago FLASH Center and the Max Planck Institut für Astrophysik (MPA) in Garching – have carried out three-dimensional studies of SN Ia systems following ignition. These two groups use different approaches and reach different conclusions. The largest simulations so far use about 1,0003 zones and barely resolve the integral scale of the turbulence, which is about 10 km when the star has expanded to a radius of 5,000 km. The differences between Chicago and the MPA have largely to do with the treatment of the flame (thick flame vs. level set) and the turbulent subgrid model. We plan to explore both approaches, but it is thought that the results will become less sensitive to the flame model when the turbulence is well resolved. A realistic goal might be a volume of 4,0963 cells. CASTRO, without radiation transport, has been demonstrated to scale well to 60,000 cores. Fortunately, only simple conduction is needed to run SN Ia explosion models. A run using 20,000 time steps might take 3.2M processor hours and run on 16,000 cores. This scaling has not yet been demonstrated for a calculation with burning, diffusion and multiple species, so there is still work to be done.

Studying the transition to detonation sub-problem once again requires 3D to adequately represent the turbulence, and high resolution to study the small-scale mixing that can blend fuel and ash. It is best carried out in a compressible code so that the transition can actually be observed. Test simulations in this regime so far have used approximately 2 million processor hours but are inadequate to resolve the mixed regions that might serve as potential detonators. Again, a large numerical Reynolds number may be necessary to correctly represent the mixing. The code and scaling are the same as for case 3) but the overall resources should be significantly less.

The computational needs for SEDONA radiation transport calculations have been determined from previous runs on the Jaguar machine at ORNL. Our experience is that 2D models assuming local thermodynamic equilibrium (LTE) require 4,000 cores for three hours each. Exploratory studies of a single 3D model, and extrapolations based on 2D models tell us that a 3D radiation transport calculation will require 20,000 cores for 20 hours (400,000 processor hours per run) for a 512^3 explosion model. Because wavelength adds another dimension to the problem, very large hydrodynamic calculations may need to be coarsened to perform the radiative transfer. Next year we plan to do 3D radiative transfer for several CASTRO models. Ten LTE calculations would require about 4M processor hours. Inclusion of the non-local thermodynamic equilibrium (NLTE) physics into the radiation transport greatly increases the computational expense and is a problem for the future. Our previous calculations of 1-D NLTE models required nearly 50 times more execution time than the LTE calculations. Thus, to calculate 2-D NLTE models will require of order 1 million processor-hours per run. 3-D NLTE models would require of order 10 million processor-hours per run. Since many runs are required, this level of physical realism may have to wait for later years at NERSC. Fortunately, the scaling properties of the code are very good and the team has routinely done calculations on 2,000 – 5,000 cores on Jaguar and scaling to 10,000 cores for 3D runs should be good. Planned load balancing optimizations should further increase the efficiency of the domain-decomposed version of the code. Processors that have heavy packet loads, due to high values of either the opacity or radiation field, will be replicated on additional processors.

10.2.2.4. HPC Requirements: Three to Five Years

The following sections outline significant scientific progress we could achieve over the next five years with access to 50 times our current NERSC allocation of resources. Note: Much of what we want to do right now requires more resources than NERSC alone can provide. So the additional resources are needed today.

Ignition: To map out the allowed distribution of possible ignition points requires running a large number of simulations with differing initial models and perturbations—more than 25 are envisioned. Only then will the ignition process be fully understood. An uncertainty here is the minimum resolution required for such a survey. This will not be known until the completion of the first year's calculations at 7683, at which point a comparison can be made to the recently completed 3843 calculations. This means an individual calculation will require anywhere from 500,000 processor-hours to 6 million (or more) processor-hours. And performing 20 of these will need between 10 million and 40 million processor-hours. We may also want to push to higher resolution still (e.g. 12803) and using our recently implemented adaptive meshing.

Full star models: As noted before, a realistic simulation requires a minimum zoning of 40963 to fully resolve the integral scale. A run using 20,000 time steps might take 3.2 M processor hours and run on 16,000 cores. But many models must be run to explore the ignition densities and compositions and the multiple ignition possibilities revealed by the studies with MAESTRO. Thus realistically this is at least a 50M processor-hour task.

Detonation transition modeling: The future requirements here are difficult to estimate because the problem is just starting to be explored. The same code will be used as for the full-star models, but only small patches of the star will be studied in highly resolved simulations. Off-line studies with a code from the chemical combustion community – "the Linear Eddy Method" – suggest that a numerical Reynolds number of 10,000 may be needed to see the transient well-mixed regions that might serve as detonation sites. This is beyond what can presently be achieved in 3D with CASTRO but might be feasible with an order-of-magnitude larger computational resource. A rough estimate for these sorts of studies calls for 20M processor hours.

Spectra and light curves: Inclusion of the non-local thermodynamic equilibrium (NLTE) physics into the radiation transport greatly increases the computational expense. Previous calculations of 1D NLTE models required nearly 50 times the execution time of the LTE calculations. Thus, to calculate 2D NLTE models will require on the order of 1M processor-hours per run. Three-dimensional NLTE models would require 10 million processor-hours per run. Ultimately, there is a need to calculate light curves and spectra for a large number of the models generated by CASTRO.

10.2.2.5. Computational and Storage Requirements Summary

	Current (2009 at NERSC)	In 3-5 Years
Computational Hours	2.5 M	250 M
Parallel Concurrency	1,500 - 5,000	20,000 - 100,000
Wall Hours per Run	150 - 300	100 - 400
Aggregate Memory	2,000 - 7,500 GB	10,000 - 50,000 GB
Memory per Core	2 GB	0.5 GB
I/O per Run	3,000 - 10,000 GB	6,000 - 40,000 GB
On-Line Storage Needed	1,000 GB in 10,000 Files	5,000 GB in 10,000 Files
Data Transfer	Typically small ~GB	Same
Size of Checkpoint File(s)	20 - 100 GB	200 - 1000 GB
Off-Line Archival Storage	50,000 GB in 2,500 Files	400,000 GB in 20,000 Files

10.2.2.6. Support Services and Software

Important software includes F90, C++, MPI, OpenMP, and htar (for creating a tar file archive directly on the HPSS archive storage system) or similar archiving software. There are two principal areas in which needs for software, services, etc. are anticipated. The first area is improved programming models to support hierarchical parallel approaches (see below). In a similar vein, tools are also needed for automatic performance tuning to improve overall node performance.

The other major area of significant needs is in the area of tools to facilitate archiving simulation data and rapid access to archived data for subsequent analysis. The data volume associated with the simulations discussed elsewhere in this document will be unmanageable without improved data-handling tools.

The priorities with regard to expanded HPC resources such as more processor hours, more memory, more storage, better throughput for small jobs, ability to handle very large jobs, etc., vary depending on the status of the problem. For several years now, including the present, this team has been in a stage of intensive code development and testing. For these purposes, rapid turnaround using a moderate number of processors (500 – 2,000) is most important. However, even these test runs often take hours. The codes typically run well, i.e., don’t crash, but developing new flame physics, nuclear networks, opacities, etc. take time to test.

However, we are rapidly moving, at least with some of the problems, to a production and publication stage where what we want is the largest possible number of processor hours. The codes scale well but not perfectly, especially when complex physics (multi-group radiation transport, non-trivial reaction networks, complex equation of state, etc.) is included. For some time to come, our needs would be optimally served by a large amount of time on 10,000 – 20,000 cores.

In general, we find 2 GB/core to be adequate memory for the three codes being fielded. As they start to deploy hierarchical models, we anticipate that this requirement will be reduced.

File storage is an increasing concern for all. It is not feasible to transfer many files to remote sites so the capability and software for efficient visualization must exist at NERSC. VisiT is used a lot. There is a need to archive checkpoint files for all major runs.

A rough estimate of the data we will produce is 200 TB/yr for just the ignition studies and about 500 TB/yr for everything. For online storage, an estimated 30 TB is needed per run.

10.2.2.7. Emerging HPC Architectures and Programming Models

We are currently pursuing development of a hybrid programming model for use in MAESTRO and CASTRO. This is based on the hierarchical parallelism model within the codes. The adaptive mesh refinement uses a patch-based approach in which the region requiring refinement at each level is covered by a collection of patches, each of which is a logically rectangular structured grid. Typical grid patches are 163 at a minimum and often larger. A coarse-grained parallelization strategy is used to distribute these patches to nodes (processing elements) using MPI. Within each node there is an additional fine-grained, loop-based parallelism over operations performed on the patches. OpenMP is currently used as the model for loop-level parallelization. Preliminary results suggest that this will be an effective strategy, at least up to a modest number of cores per node. It would potentially be helpful to have a better “light-weight” approach for starting threads than that used by OpenMP. These codes could potentially use GPUs or other accelerators effectively, but the system would need to be configured to move data between into and out of the accelerator quickly; a huge latency in getting data into a GPU, for example, could make it difficult to use effectively.

	Ignition studies with MAESTRO	Full star explosion models with CASTRO	Studies of transition to detonation with CASTRO	Light curve and spectra calculations with SEDONA
Current year (2009)				
Number of runs	5 at 384^3	10 at 1024^3 deflagration stage	6	10
M-Computational Hours – all runs	3	2	2	1
Cores	1,728	1,024	1,024	4,096
Size check point file (GB)	10	20	20	0.01
Total output (TB)/run	6	6	2	0.01
On-line storage (TB)	3	3	1	0.01
Off-line storage (TB)	30	20	10	0.2
Next Year (2010)				
Number of runs	10 at 768^3	10 at 1024^3; 10 at 4096^3 include detonation	10	30 3D-LTE
M-Computational Hours – all runs	60	40	10	12
Cores	5K-10K	4K-16K	4K-16 K	20 K
Size check point file (GB)	80	50	50	10
Total output (TB)/run	30	50	15	0.1
On-line storage (TB)	10	10	3	0.3
Off-line storage (TB)	300	500	150	3
Three years from now per year				
Number of runs	10 (higher resolution)	20 at $4,096^3$ include detonation	10	30 3D – LTE 2 x 3D -NLTE
M-Computational Hours – all runs	75	50	15	32
Cores	10K-15K	10K-40K	10 K-40 K	20 K
Size check point file (GB)	100	150	100	10-100
Total output (TB)/run	40	100	40	0.1
On-line storage (TB)	10	20	10	0.3
Off-line storage (TB)	400	1,000	300	4

10.2.3. Core-Collapse Supernovae

Prepared by: Stan E Woosley, UC Santa Cruz, John Bell, LBNL: Adam Burrows, Princeton University
Contributors: Ann Almgren, LBNL; Luke Roberts, Candace Church, UC Santa Cruz; Louis Howell, LLNL; Adam Burrows, Jason Nordhaus, Princeton University; A. Heger (University of Minnesota
NERSC Repository: m106

10.2.3.1. Summary and Scientific Objectives

Few events in the cosmos are as energetic as the explosion of a massive star in a supernova. For several seconds, the neutrino luminosity of a single event exceeds that of the rest of the visible universe. The output in light can rival that of a galaxy. If rotation and magnetic fields channel the explosion into a narrow relativistic jet, making a gamma-ray burst, the luminosity can be a billion times greater. An event on the other side of our galaxy would be as bright as the sun. Core-collapse supernovae are also responsible for making most of the elements heavier than helium in nature. In their interiors, conditions exist that test our understanding of novel particle physics and the behavior of matter at high density. They are also poorly understood. No supernova has even been completely modeled from first principles.

For these reasons, the numerical simulation of supernovae in massive stars has been a forefront problem in computational astrophysics for 40 years. It is a difficult problem. Hydrodynamics, nuclear reactions, and — eventually — general relativity and magnetism, must be coupled to the non-thermal transport of six kinds of neutrinos (electron, muon and tauon and their antiparticles). Neutrino energy deposition outside the collapsing neutronized core drives convection that affects the efficiency of energy deposition and transport. Simulating this convection requires high spatial resolution and three dimensions. It is also agreed by all major groups working on the problem that major advances will require a careful treatment of the neutrino transport in three dimensions.

Thus far, the only “credible” simulations have used two-dimensional models and these have provided mixed results. The group at ORNL (Mezzacappa) claims mild neutrino-powered explosions for the kinds of stars that would give the most common events (15 solar masses). Groups in Munich (Janka) and this group (Burrows) disagree and find failures via the neutrino mechanism for all but the lightest stars (10 solar masses). Given the current situation, some researchers now claim that the traditional “neutrino-transport” model without rotation and magnetic fields is inadequate to explain what we see in nature. They may be right for some range of stellar masses and rotation rates, but we believe that 3D effects are crucial for the neutrino mechanism and may be the missing ingredient. Magnetic fields, rotation and general relativity may, however, play a major role in that smaller set of massive stellar deaths, especially those that lead to gamma-ray bursts. The current frontier is determining whether the most common supernovae (15 solar masses) and those thought chiefly responsible for making heavy elements (25 solar masses) can explode as a consequence of neutrino transport acting alone in non-rotating stars.

We are in the late stages of developing of a code optimized for this problem. CASTRO is a 3D radiation, hydrodynamics code that uses adaptive mesh refinement in space and time. CASTRO is also already being applied to the Type Ia supernova problem, but for core collapse, “radiation-” (i.e., neutrino-) transport is a key component. Thus, for the core-collapse problem we are developing a multi-group flux limited diffusion capability targeted at treating neutrino transport. To date, the hydrodynamics in CASTRO has been extensively tested and 3D core-collapse problems have been run without neutrino transport. The radiation component is also essentially complete, but has not been extensively tested. We expect to have our first low-resolution neutrino transport model in 3D in the next year. Realistic resolution and the exploration of a variety of masses of progenitors will occur in later years and will take significantly more resources.

10.2.3.2. Methods of Solution

CASTRO is based on an Eulerian radiation hydrodynamics code that includes self gravity and reaction networks. The radiation module currently supports a multigroup diffusion approximation for neutrino radiation. CASTRO incorporates hierarchical block-structured adaptive mesh refinement and supports 3D Cartesian, 2D Cartesian and cylindrical, and 1D Cartesian and spherical coordinates. The hydrodynamics in CASTRO is based on the unsplit methodology introduced by [1]. The code has options for the piecewise linear method in [2] and the unsplit piecewise parabolic method (PPM) in [3]. The unsplit PPM has the option to use the less restrictive limiters introduced in [4]. All of the hydrodynamics options are designed to work with a general convex equation of state. CASTRO supports two different methods for including Newtonian self-gravitational forces. One approach uses a monopole approximation to compute a radial gravity consistent with the mass distribution. The second approach is based on solving the Poisson equation for the gravitational field. The Poisson equation is discretized using standard finite difference approximations and the resulting linear system is solved using geometric multigrid techniques. A third approach in which gravity is externally specified is also available.

CASTRO uses a diffusion approximation for neutrino radiation transport. The radiation model is based on a two-moment system for each of multiple energy groups and multiple species of neutrinos, using a mixed-frame formulation as presented in Hubeny & Burrows [5]. Since we are not using a full angle-dependent transport solution to derive appropriate Eddington factors for the moment equations, we intend instead to support flux limiters and their related Eddington factors along the lines of Levermore & Pomraning [6] and Levermore [7]. The primary computational expense of the radiation model is that it requires the solution of diffusion equations with non-symmetric perturbations both on single refinement levels and on multiple coupled levels of the AMR mesh. The code relies on the hypre library [8] for solving these systems on large parallel machines.

The adaptive refinement approach in CASTRO is based on the BoxLib package developed at Berkeley Lab by CCSE. It uses a nested hierarchy of logically rectangular grids with simultaneous refinement of the grids in both space and time [9,10]. The integration algorithm on the grid hierarchy is a recursive procedure in which coarse grids are advanced in time, fine grids are advanced multiple steps to reach the same time as the coarse grids and the data at different levels are then synchronized. During the regridding step, increasingly finer grids are recursively embedded in coarse grids until the solution is sufficiently resolved. An error estimation procedure based on user-specified criteria evaluates where additional refinement is needed and grid generation procedures dynamically create or remove rectangular fine grid patches as resolution requirements change.

CASTRO uses the same interface to equations of state and thermonuclear reaction networks as MAESTRO. This general interface allows different approximations for nucleosynthesis to be easily incorporated. It also enables switching from a low Mach number simulation with MAESTRO to a fully compressible simulation using CASTRO without changing the underlying physics. The parallelization strategy for CASTRO is to distribute grids to processors using MPI.

This provides a natural coarse-grained approach to distributing the computational work. When AMR is used a dynamic load balancing technique is needed to adjust the load. Both a heuristic knapsack algorithm and a space-filling curve algorithm are used for load balancing. Criteria based on the ratio of the number of grids at a level to the number of processors dynamically switches between these strategies.

There are several additional physics and computational capabilities that will potentially be added to the code over the next five years:

- Magnetohydrodynamics
- More complex radiation packages (possibly using Monte Carlo algorithms)
- Non-Newtonian gravity – this could range from implementing simple leading order corrections to incorporating a full general relativity solution.
- A hierarchical parallelism scheme will need to be developed that will use less memory per core based on having all cores on a node work on a single grid. Otherwise 2 GB/core could be a limiting constraint.
- Finally, improvements in both performance and scalability are continually being made to the underlying software. A particular focus for CASTRO will be improvements in the linear algebra routines used for the radiation, which is likely to involve continued collaboration with the Hypre group at LLNL.

References

[1] P. Colella. Multidimensional Upwind Methods for Hyperbolic Conservation Laws. Journal of Computational Physics, 87:171–200, 1990.

[2] J. Saltzman. An unsplit 3D upwind method for hyperbolic conservation laws.

[3] G. H. Miller and P. Colella. A Conservative Three-Dimensional Eulerian Method for Coupled Solid-Fluid Shock Capturing. Journal of Computational Physics, 183:26–82, 2002.

[4] P. Colella and M. D. Sekora. A limiter for PPM that preserves accuracy at smooth extrema. Journal of Computational Physics, submitted.

[5] Hubeny, I., and Burrows, A., “A New Algorithm for 2-D Transport for Astrophysical Simulations: I. General Formulation and Tests for the 1-D Spherical Case”, Astrophys. J., 659, 1458 - 1487, (2007).

[6] Levermore, C. D. and Pomraning, G. C., Astrophys. J., 248, 321, (1981).

[7] Levermore, C. D., J. Quant. Spectrosc. Radiat. Transfer, 31, 149, (1984).
Lin, D.

[8] Falgout, R. D., Jones, J. E. and Yang, U. M., Lecture Notes in Computational Science and Engineering, 51, 267, (2006).

[9] M. J. Berger and P. Colella. Local adaptive mesh refinement for shock hydrodynamics. Journal of Computational Physics, 82(1):64–84, May 1989.

[10] J. Bell, M. Berger, J. Saltzman, and M. Welcome. A three-dimensional adaptive mesh refinement for hyperbolic conservation laws. SIAM J. Sci. Statist. Comput., 15(1):127–138, 1994.

10.2.3.3. HPC Requirements

Currently, we have inadequate resources at NERSC to make major headway on this problem. A single low-resolution, full-star, 3D model would use more than our entire allocation. We have thus been restricted to 2D models with neutrino transport and exploratory 3D models without neutrino transport.

The radiation-hydro simulations are very computationally intensive. The MGFLD scheme requires 10-20 energy groups of neutrinos. The radiation solver is an iterative algorithm, coupling the groups together through their nonlinear interactions with the fluid state. The number of "equivalent diffusion solves" depends on the tolerances and on the size of the time step, but may be of order a few. In 2D, for a 128^2 coarse grid, with six levels of refinement, each simulation will be advanced to ~0.5 s after bounce. For a single coarse-grid time step, the wallclock time is about 10 s. Therefore, for 4,000 coarse time steps, a single simulation requires 2-3 million processor hours. For 3D, the spatial resolution will initially be reduced to compensate for the increase in computational requirements due to the resolution. Each run is estimated to take 2-3 M processor hours. In three to five years they will need a minimum of several 100 M processor hours per year to do realistic 3D models for a variety of masses. This is about 150 x the current usage.

Within five years, we will also need to include magnetic fields and at least first-order post-Newtonian corrections. We are also exploring now, and by then may have moved to, Monte Carlo neutrino transport instead of a multi-group flux-limited model. We also need to explore a large range of progenitor masses and rotation rates. These improvements in physics will, to first order, probably not greatly affect the computer resources needed, except those dealing with the neutrino transport.

The jobs are very processor-intensive due to the need to carry at least 20 energy groups of neutrinos. Two-dimensional, full-star models (1282 with six levels of AMR) take about 3M processor hours each with moderate resolution. The current frontier, though, is 3D runs, where even a coarsely zoned single calculation is estimated to take 2-3M processor hours. Faster processors, then, is the greatest need, and memory and storage will scale proportionately.

10.2.3.4. Computational and Storage Requirements Summary

	Current (2009 at NERSC)	In 3-5 Years
Computational Hours	3.0M	150M
Parallel Concurrency	2K-16K	20K-100K depending on development
Wall Hours per Run	150	500-1,000
Aggregate Memory	3,000 GB – 25,000 GB	8,000 – 50,000 GB
Memory per Core	2 GB	0.5 GB
I/O per Run	2,000 GB	4,000- 400,000 GB
On-Line Storage Needed	1,000 GB in 2,000 files	40,000 GB in ? files
Data Transfer Needed	minimal	minimal
Size of Checkpoint File(s)	20-200 GB	2,000- 10,000 GB
Off-Line Archival Storage	50,000 GB in 2,500 files	1 PB in 10,000 files

10.2.3.5. Support Services and Software

The needs for this project are identical to those outlined above for SN Ia: improved programming models to support hierarchical parallel approaches, tools for automatic performance tuning, and tools to facilitate archive and rapid access to simulation data for subsequent analysis.

10.2.3.6. Emerging HPC Architectures and Programming Models

The project team is currently pursuing the development of a hybrid programming model for use in CASTRO, as discussed above. The radiation module in CASTRO uses the hypre multigrid package from LLNL. Effective use of a hybrid parallelization strategy for radiation will require a suitable extension of hypre or other similar iterative linear solver package.

11. Astrophysics Data Analysis

11.1. Astrophysics Data Analysis Overview

The Office of High Energy Physics promotes broad, long-term research at three interrelated frontiers of physics, one of which is the Cosmic Frontier. The Cosmic Frontier explores the nature of dark matter and dark energy by using particles and radiation from space to explore new phenomena. Cosmic rays in the Earth's atmosphere, neutrinos from the Sun, and gamma rays from deep space are some of the known natural sources. Searches are also underway for alternate explanations of dark matter and energy. Observations of the Cosmic Frontier reveal a universe far stranger than ever thought possible.

Among the facilities and experiments that HEP supports to research the Cosmic Frontier are the Cryogenic Dark Matter Search, Sloan Digital Sky Survey, Pierre Auger, VERITAS, GLAST, Axion Dark Matter Experiment, Alpha Magnetic Spectrometer, Dark Energy Survey, Palomar Transient Factory, and the Baryon Oscillation Spectroscopic Survey. Two future instruments include the Joint Dark Energy Mission (JDEM) and the Large Synoptic Sky Survey (LSSS). These instruments are all intended to image a large fraction of the sky and they therefore generate massive amounts of data at high rates (hundreds of Mb/sec). These data typically require complex analysis in real time to assess both the quality of the data as well as to discover transients such as supernovae. The computing infrastructure involved in such data management and analysis is typically distributed among several collaborating sites, but in several cases NERSC plays a vital role in the processing and analysis pipelines, in archiving the data, and in providing mechanisms for making the data readily available to the entire community. NERSC resources used range from large clusters (like PDSF) to Franklin to the NERSC Global File System to HPSS.

The goals of these programs are often multi-fold, spanning cosmological measurements from a variety of methods to pure astrophysics and involve the creation of large catalogues of data both of a temporal and cumulative static nature. These observations are then typically confronted directly with simulation. Another key attribute is that they involve large international collaborations distributed over many institutions and sites.

11.2. Astrophysics Data Analysis Case Studies

11.2.1. Palomar Transient Factory & La Silla Supernova Search

Principal Investigators: Peter Nugent, LBNL; Shri Kulkarni, Caltech; Charles Baltay, Yale
NERSC Repository: m779

11.2.1.1. Summary and Scientific Objectives

The Palomar Transient Factory (PTF) and La Silla Supernovae Search (LSSN) are wide-field experiments designed to investigate the optical transient and variable sky on time scales from minutes to days. These experiments are designed to fill gaps in the observational data and to search for theoretically predicted, but not yet detected, phenomena, such as fallback supernovae, macronovae, Type Ia supernovae or the orphan afterglows of gamma-ray bursts.

These programs will also discover many new members of known source classes, from cataclysmic variables in their various avatars to supernovae and active galactic nuclei, and will provide important insights into understanding galactic dynamics (through RR Lyrae stars) and the solar system (asteroids and near-earth objects). The lessons that can be learned from PTF & LSSN will be essential for the preparation of future large synoptic sky surveys like DES, JDEM and LSST.

11.2.1.2. Methods of Solution

The code hotpants (High Order Transform of PSF ANd Template Subtraction) is used to perform image differences. This package is designed to photometrically align one input image with another, after they have been astrometrically aligned. This is an implementation of the Alard (1999) algorithm for image subtraction and is is very similar to the algorithm used in the ISIS package, but with a variety of improvements. The software has been used in real-time data pipelines with great success (eg. SDSS and DES), and much work has gone into making this a robust package.

The crux of the difference-imaging problem is to find a convolution kernel K that matches the point spread functions (PSFs) of two astronomical images, I (referred to as the image) and T (referred to as the template). These images in general are taken under different conditions, including atmospheric transparency, atmospheric seeing or exposure times. One may even difference data taken from different sites and equipment entirely. Mathematically, the equation solved is the minimization of the function

$$([T \oplus K](x,y) - I(x,y))^2$$

by solving for the kernel K. If K can be decomposed onto basis functions, then this is a linear least-squares problem, and can be solved uniquely by matrix inversion.

11.2.1.3. HPC Requirements

NERSC resources are used to process and archive the data (through the NERSC data transfer nodes), detect new transients through the subtraction pipeline and, in conjunction with the DeepSky database, aid and enhance the classification of new transients by providing an interface for scientists to annotate detections using NERSC's science gateway nodes.

Computational experiments today require approximately 50-100 GB of space each day to store the raw data. This then turns into 100-200 GB of processed data and subtracted images and approximately 200,000 candidate detections are loaded into a database from each program. Data analysis and manipulation occurs in as close to real-time as possible, which requires access to 32 processors to keep up with the load.

In the future, the need to generate large catalogs of static objects (through co-addition of a large number of individual images to make a single deep image) will require codes like SCAMP and SWARP (http://www.astromatic.net) that have threaded capabilities appropriate for new architectures.

Next-generation experiments like JDEM and LSST will generate more than an order of magnitude more data. Real-time computing will be a priority.

Access to hundreds of terabytes of file storage on a shared system like the NERSC Global Filesystem (NGF), which is accessible from a variety of computational resources, is crucial for both experimental and simulation scientists.

For some work, 2 GB of memory per node is often enough, although for several applications, having machines like NERSC's SGI Altix (Davinci) with ~100 GB of memory seen by several processors is a much more effective computational route. Compute tasks typically involve a wide variety of methods and libraries for FFTs, linear solvers, and Monte Carlo methods.

11.2.1.4. Computational and Storage Requirements Summary

	Current (2009 at NERSC)	**In 3-5 Years**
Computational Hours	75,000	150,000–1,000,000
Parallel Concurrency	32 processors	64-128 processors
Wall Hours per Run	8	8
Aggregate Memory	Up to 64 GB	Up to 64GB
Memory per Core	2GB minimum	64 GB
I/O per Run Needed	1 TB	10 TB
On-Line Storage Needed	50 TB/yr	150 TB/yr
Data Transfer Needed	100 GB/day	250 GB/day

11.2.1.5. Support Services and Software

This project needs support for large databases, which will contain information on both the observations and simulations. Other needs include access to simulation data through the science gateway nodes and much more real-time-accessible disk space. More analytics support to help visualize the results would also be beneficial.

11.2.1.6. Emerging HPC Architectures and Programming Models

A strategy for dealing with multi?core/many?core architectures is already in place, although smaller per-node memory will be an obstacle as image sizes grow in the future.

We are testing some of these codes (especially those that use FFTs) on GPUs. As winner of the SC09 Storage Challenge, we have demonstrated how flash memory disks can be used to greatly speed up codes that have large I/O requirements.

11.2.2. Cosmic Microwave Background Data Analysis for the Planck Satellite Mission

Principal Investigator: Julian Borrill, LBNL
Contributors: Christopher Cantalupo and Theodore Kisner, LBNL; Radoslaw Stompor, University of Paris
NERSC Repository: planck

11.2.2.1. Summary and Scientific Objectives

The goal of this work is to analyze the Cosmic Microwave Background (CMB) data being gathered by the joint ESA/NASA Planck satellite mission. The CMB consists of primordial photons, last scattered when the Universe first became electrically neutral —80,000 years after the Big Bang. The statistics of the tiny fluctuations in the temperature and polarization of the CMB provide powerful constraints on cosmology and physics at the highest energies. Launched in May 2009, Planck will scan the entire sky for up to 2.5 years, making the most precise measurements yet of its temperature and polarization at 9 frequencies between 30 and 857GHz. Extracting science from these O(1013) observations will proceed in 4 steps: 1) combining the time-ordered data to make maps of the sky (CMB+foregrounds) at each frequency; 2) using this set of maps to derive a single map separating the CMB from the foreground signals; 3) estimating the angular auto- and cross-spectra of the CMB temperature and polarization modes from this map; and 4) deriving constraints on the fundamental parameters of cosmology from these power spectra. The projected Planck results complement, and are routinely assumed by, all dark energy experiments.

11.2.2.2. Methods of Solution

The most computationally challenging elements of the analysis of a CMB dataset are those that involve manipulations of the time-ordered data. These include the map-making step 1) and, in particular, the power spectrum estimation step 3) which requires simulating and mapping tens of thousands of Monte Carlo realizations of the entire mission. Furthermore, each realization's data are correlated — indeed it is precisely the correlations in the CMB that researchers wish to estimate — so they cannot simply divide-and-conquer the data, but instead have to treat it as a single data object.

The noise in a CMB dataset such as Planck is not white, but instead includes low-frequency correlations, so the time-ordered data cannot simply be binned into pixels to make maps. Instead, the Gaussian, piecewise stationary nature of the noise is used to derive a maximum-likelihood map, which can be solved for using preconditioned conjugate gradient techniques as implemented in the MADmap code. Alternatively, it can be assumed that the noise is white plus an offset over some interval and use the so-called destriping approach, in which the data are binned within the intervals and solved for the offsets between them, as implemented for Planck in the Springtide and MADAM codes.

Simulating the Planck data involves generating both noise and signal realizations. Given the power spectral density of the noise for each detector in each stationary interval, simple Fourier methods can be used to generate noise realizations - although care has to be taken to avoid introducing spurious correlations when generating O(1017) independent pseudo-random numbers. Given a realization of the sky being observed, the signal in each detector depends on its detailed characteristics, including the shapes of its beam and band-pass. For Planck, such simulations can be generated using the proprietary LevelS package, although the traditional simulate/write/read/map cycle is increasingly IO bound. However, since the IO in this cycle is redundant, we are increasingly ingesting the critical elements of LevelS into the On-The-Fly Simulation (OTFS) capability being developed within the M3 data abstraction layer, using which a simulation is only performed when its data are requested by the map-making code.

11.2.2.3. HPC Requirements

Making maps of one year of simulated single-frequency Planck data on disk typically uses O(10^3) Franklin cores for a few minutes (destriping) to an hour (maximum likelihood). Simulating and mapping 100 Monte Carlo realizations of one year of the entire mission using OTFS uses O(10^4) Franklin cores for a few hours (destriping) to a few days (maximum likelihood). Generating a simulation of one year of the entire mission with LevelS uses O(10^3) Franklin cores for a few minutes to a few hours depending on its complexity.

All of the map-making codes have tunable memory footprints, in which recalculation of intermediate data vectors can replace their storage. Both the time-ordered data and the maps are distributed over the cores, and the gathering of the contributions to the map from each core requires an MPI reduction over all the cores, which is becoming the limiting factor in runs making very-high-resolution maps. Using the M3 layer all the map-making codes can minimize their I/O bandwidth and disk requirements, with both simulated time-ordered data and individual detector pointings being generated on the fly.

The LevelS simulation codes are inherently serial, and some parts can generate extremely large data objects (especially associated with asymmetric beam convolutions) that require many GB of memory and can only be run on the Planck cluster at NERSC, which has 32 GB of memory per node. Such a simulation is embarrassingly parallel over stationary intervals, so has no communication. However each core writes out its data interval by interval using the cfitsio library, leading to a significant I/O bottleneck at high concurrencies (hence the restriction to O(10^3) cores). Using the OTFS capabilities breaks this limitation, and indeed the team believes that — subject to being able to reduce and write 100s of maps simultaneously — they will be able to scale this code up to O(10^5) cores when Hopper (NERSC’s next-generation Cray XE6, scheduled for the second half of 2010) is fully operational.

Ultimately, we will need to simulate and map O(104) realizations of the full Planck mission. Projecting from current run times, this will require O(106-8) core-hours for destriping and maximum likelihood methods respectively. Scaling to a concurrency at which this is a reasonable wall-clock time will require addressing the existing map-reducing and anticipated map-writing bottlenecks identified above, and this is work in progress. We also hope to reduce the number of iterations required by the maximum likelihood PCG code by building better pre-conditioners and first-guesses.

11.2.2.4. Computational and Storage Requirements Summary

	Current	Next 3-5 Years
Computational Hours	1 million	5 - 50 million
Parallel Concurrency	1,000 – 10,000	1,000 – 100,000+
Wall Hours per Run	1	1 - 10
Aggregate Memory	1 - 10 TB	1 - 100 TB
Memory per Core	1 GB	1 GB
I/O per Run	Up to 1 TB	Up to 10 TB
On-Line Storage Needed	100 TB	500 TB
Data Transfer	10 TB/year	10 TB/year
Archival Storage	50 TB	1 PB

11.2.2.5. Support Services and Software

The project needs a center-wide high-speed, high-capacity file system, like the one the center supplies with its NERSC Global Filesystem (NGF). While the above discussion has focused on the most computationally challenging elements of the analysis, there are many other steps that require 100s of users accessing data from all of the NERSC systems, including the Planck cluster; synchronizing the Planck data sets across NGF and Franklin scratch has been extremely painful. We are confident that NERSC can make NGF perform sufficiently well across all of its systems (for example, no worse than half of the I/O performance of a system’s native scratch space) provided the commitment to, and resources for, this are present.

11.2.2.6. Emerging HPC Architectures and Programming Models

With funding from the NSF PetaApps program, we are just beginning a major revamping of our core simulation and map-making code base in order to enable them to take advantage of emerging petascale architectures. In particular, we are addressing

- The ability to take advantage of next generation nodes, including many-core, accelerator-augmented, and heterogeneous ones.
- The ability to tune (and ideally auto-tune) the code to make trade-offs between cycles, memory, communication and I/O loads to match the capacities and capabilities of an architecture’s sub-systems as well as the requirements of any particular analysis.

12. Lattice QCD

12.1. Lattice QCD Overview

Quantum Chromodynamics (QCD) is the theory of the strong interaction between quarks, mediated by gluons, expressed by the Dirac action for quarks. Strong interactions are responsible for binding quarks into protons and neutrons and holding them all together in the atomic nucleus. Gluons are in some ways analogous to photons, the crucial difference being that gluons couple to each other (whereas photons do not). Lattice QCD is the numerical simulation of QCD and it involves evaluating field variables on sites and links of a regular hypercube lattice in discretized four-dimensional space-time.

Research using Lattice QCD addresses fundamental problems in high energy and nuclear physics, and is directly related to experimental programs in these fields. It includes studies of the mass spectrum of strongly interacting particles, the weak interactions of these particles, and the behavior of strongly interacting matter at high temperatures. The goal of understanding the strong dynamics of quarks and gluons to extract fundamental parameters of the Standard Model of particle physics is beyond the reach of the traditional perturbative methods of quantum field theory.

The U.S. Department of Energy's Scientific Discovery through Advanced Computing (SciDAC) program supports lattice QCD research through the National Computational Infrastructure for Lattice Gauge Theory and other projects. The relevance of this work to DOE’s mission is illustrated in the Executive Summary of the SciDAC-2 proposal, namely that this work intends to calculate the masses of strongly interacting particles and obtain a quantitative understanding of their internal structure.

QCD involves integrating an equation of motion for hundreds or thousands of time steps that requires inverting a large, sparse matrix at each step of the integration. The sparse matrix problem is solved using a conjugate gradient method but because the linear system is nearly singular, many CG iterations are required for convergence. Within a processor the four-dimensional nature of the problem requires gathers from widely separated locations in memory. The matrix in the linear system being solved contains sets of complex three-dimensional “link” matrices, one per 4D lattice link, but only links between odd sites and even sites are non-zero. The inversion by CG requires repeated three-dimensional complex matrix-vector multiplications, which reduces to a dot product of three pairs of three-dimensional complex vectors. The code separates the real and imaginary parts, producing six dot product pairs of six-dimensional real vectors. Each such dot product consists of five multiply-add operations and one multiply.

12.2. Lattice QCD Case Studies

12.2.1. MIMD Lattice Computation (MILC) Collaboration

Prepared By: Doug Toussaint, University of Arizona

Contributor: Paul Mackenzie, Fermilab
NERSC Repository: mp13

12.2.1.1. Summary and Scientific Objectives

The MILC Collaboration is engaged in a broad research program in Quantum Chromodynamics (QCD). This research addresses fundamental questions in high energy and nuclear physics, and is directly related to major experimental programs in these fields. It includes studies of the mass spectrum of strongly interacting particles, the weak interactions of these particles, and the behavior of strongly interacting matter under extreme conditions.

The specific goals of lattice gauge simulations are:

1. Calculate hadronic matrix elements needed to relate experimental results to fundamental parameters of the standard model.

2. Understand the structure and interactions of hadrons, and how QCD works to confine quarks into hadrons.

3. Calculate QCD in extreme environments such as those in RHIC (Relativistic Heavy Ion Collider) collisions, neutron star interiors, and the early universe.

4. Explore other strongly interacting field theories as candidates for "“beyond the Standard Model” physics

The MILC project at NERSC addresses the first of these goals, as does the HPQCD project. Other NERSC HEP projects address the third and fourth goals. Projects aimed at the second goal are split between high-energy physics and nuclear physics. The MILC project simulations generate samples of QCD gluon field configurations, or “lattices,” and use these lattices to calculate pseudoscalar meson masses, decay constants and form factors and other hadronic properties. This is part of a larger effort that uses resources at DOE centers, NSF centers and USQCD collaboration computing facilities.

Although there is little doubt that QCD is the correct theory of the strong interactions, non-perturbative QCD calculations are crucial for testing the weak interaction part of the Standard Model: In the absence of such calculations, the strong effects completely obscure the weak physics one is trying to study. At present the only means of carrying out non-perturbative QCD calculations from first principles and with controlled errors is through large-scale numerical simulations. These simulations are needed to obtain a quantitative understanding of the physical phenomena controlled by the strong interactions, to determine a number of the basic parameters of the Standard Model, and to make precise tests of the Standard Model’s range of validity.

12.2.1.2. Methods of Solution

The project uses the MILC collaboration’s suite of QCD codes, which run on almost all parallel machines. These codes are typically accelerated by use of modules developed by the lattice gauge theory SciDAC project. Other lattice gauge projects may use this same code suite or another one with similar structure.

The basic structure of the problem is a four-dimensional rectangular grid, which is divided into equal domains for each processor. Field configurations are generated by a molecular dynamics evolution of the fields through an artificial simulation time. The time-consuming part of the simulation is the computation of the force coming from the dynamical quark fields, which is a non-local force on the gluon fields. Computation of this force requires solution of a Hermitian positive-definite sparse matrix problem, which is done with the conjugate-gradient algorithm.

Grid sizes in use now range up to 64x64x64x144, and the variables at each grid site include several dozen 3x3 complex matrices and three-component complex vectors.

On all machines currently in use, the communication is done using MPI, although other message passing implementations can be, and have been, used by replacing one file in the source code.

Once the field configurations are generated and stored, they are processed several times to calculate physical observables. Most of these analysis computations are completely dominated by conjugate gradient or, in some cases, biconjugate gradient, solutions of a sparse matrix problem.

12.2.1.3. HPC Requirements

Current simulations at NERSC are being run on up to 6,144 processors, and a molecular dynamics trajectory may take up to six hours at the lightest quark masses that are used. The number of processors is determined by a tradeoff between turnaround time and efficiency, where the main cause of lower efficiency on large numbers of processors is the time needed for the global reductions in the conjugate gradient algorithm.

A full simulation at one set of parameters requires around 5,000 molecular dynamics trajectories. The total time requirement is heavily dependent on the quark masses, and approaches 100 million Cray XT4 core hours for two of the simulations that the project team would like to do in the next three years.

The table below includes only the part of the collaboration's computation provided by NERSC. In the current year, this amounts to about 20 percent of the core hours used by the collaboration. However, the NERSC machine, Franklin, is one of only three machines that we have access to that can run our largest simulations.

In the table below, the numbers for the next three to five years are based on a scenario for reducing theoretical errors in some matrix elements needed to relate fundamental parameters of the standard model to accelerator experiments to about 1 percent.

12.2.1.4. Computational and Storage Requirements Summary

	Current (2009 at NERSC)	Next 3-5 Years
Computational Hours	19.5 M	280 M
Parallel Concurrency	6,144	16,384 (flexible)
Wall Hours per Run	11	
Aggregate Memory Needed	70GB	300 GB
I/O per Run Needed	30 GB	120 GB
On-Line Storage Needed	500 GB	2000 GB
Data Transfer		1 TB/week

12.2.1.5. Support Services and Software

The most important requirement for lattice gauge simulations is computational performance. Compared to many other fields, our I/O and storage requirements are quite modest, and I/O abilities have generally not been a serious bottleneck with the current NERSC machines. The limiting factors for computational performance appear to be processor-to-memory bandwidth and inter-processor communication times. Since inversion algorithms require global reductions, the latency time for small messages is very important.

Because the data types used in the simulations are somewhat special (3x3 unitary matrices, for example) almost all of our higher-level software is specialized for lattice gauge simulations. Thus our software support requirements are for reliable efficient compilers and standard message passing or other parallelization interfaces.

A stable and convenient development environment is also creucial. In particular, a debug queue with fast turnaround time for modest size jobs is important.

12.2.1.6. Emerging HPC Architectures and Programming Models

We are implementing parts of the code suite on GPUs, with the idea of using them as accelerators for the compute intensive sections. At this time significant speedups have been demonstrated for the sparse matrix solution. It is not yet known how well this will work on large simulations that require many processors.

We have experimented with mixed parallelization models, such as OpenMP with MPI, several times in the past. These past experiments have not resulted in any gains relative to straight MPI, but it is expected that this will need to be revisited in the future.

13. HEP Data Analysis and Detector Simulation

Prepared By: Craig Tull, Lawrence Berkeley National Laboratory

13.1. Overview

The goal of this work is to use powerful particle accelerators and neutrino sources to investigate the constituents and architecture of the Universe, its core constituents and forces. This section covers several distinct experiments, specifically ATLAS, Daya Bay, and CDF.

The ATLAS experiment is being constructed by 1,800 collaborators in 150 institutes around the world. It will study proton-proton interactions at the Large Hadron Collider (LHC) at the European Laboratory for Particle Physics (CERN). The detector was scheduled to begin operation in November 2009, but was delayed. The primary purpose of the detector will be studying the origin of mass at the electroweak scale; therefore the detector has been designed for sensitivity to the largest possible Higgs mass range. The detector will also be used for studies of top quark decays and supersymmetry searches. Currently and in the foreseeable future, ATLAS uses the PDSF cluster at NERSC for physics and detector simulations to understand detector behavior and to test physics hypotheses. After the LHC goes into production, ATLAS will use PDSF for data analysis to understand detector behavior and to pursue physics topics such as the search for the Higgs boson. ATLAS uses HPSS for backup, and online storage (e.g., NGF) for active data.

The Daya Bay Neutrino Experiment is a neutrino-oscillation experiment designed to measure the mixing angle ?13 using anti-neutrinos produced by the reactors of the Daya Bay and Ling Ao nuclear power plants. The experiment is being built by blasting three kilometers of tunnel through the granite rock under the mountains where the power plants are located. Data taking is scheduled to start in Winter 2010 and reach full configuration in Winter 2011. On PDSF, Daya Bay performs simulations of the detectors, reactors and surrounding mountains to help design and anticipate detector properties and behavior. This will continue for at least five years. Once real data is available, Daya Bay will be using PDSF to analyze data from commissioning and operation of the detectors. Daya Bay uses HPSS as the central U.S. repository for all data, information and backups. U.S. collaborators from 15 institutions will access data stored at NERSC.

CDF is a legacy project completing its physics analysis of Tevatron data from the Collider Detector at Fermilab (CDF). The work involves both the analysis of experimental data collected at Fermilab and also the generation, simulation and analysis of Monte Carlo data samples used to calculate detector acceptance. The analysis concentrates on the search for the Standard Model Higgs boson, study of properties of the top quark (mass and production cross section), and precision measurements of B hadron decays. The analysis performed during this project should result in several publication papers in refereed journals (i.e., Phys Rev Lett and/or Phys Rev D).

Other HEP projects, large and small, use the facilities at NERSC and PDSF to do feasibility study simulation and/or analysis, or to explore advanced techniques associated with simulation and analysis of HEP data.

13.2. Methods of Solution

There are too many codes in use within this NERSC community to list exhaustively, but of special note are:

- GEANT4: The object-oriented simulation framework in use for almost all HEP experiments. GEANT4 models extremely complex detector geometries, material composition and physics processes including radiation, particle capture, spallation, Cherenkov radiation, optical photons, electromagetics, etc. In essence, almost any experiment studying interactions of particles and matter.
- ROOT: An object-oriented toolkit and framework in use by most HEP experiments. ROOT forms the basis of many analysis and visualization systems in HEP.
- Gaudi: A general-purpose simulation and analysis framework in use by many HEP experiments, including ATLAS and Daya Bay.
- Python and PyROOT: Python is very actively used by HEP and especially ATLAS and Daya Bay. PyROOT is an introspection-driven interface between Python and ROOT that presents a full-featured python interface to any ROOT C++ class described in the ROOT dictionary.
- CERNVM (CERN Virtual Machine): A virtualization project being developed at CERN and by LBNL scientists to provide virtual appliances for deployment of LHC experiments' software.
- PAW and GEANT3 are legacy Fortran programs that predate ROOT and GEANT4. These programs provide much of the same functionality and are still in use.

Each of the programs above is built upon component frameworks into which physicists dynamically link their own scientific component codes. These science components are not constrained to any particular mathematical method or technique and in an experiment as large and complex as ATLAS run the gamut from simple data processing such as channel calibration to processor and memory intensive pattern recognition and track finding, to other sophisticated techniques involving neural networks/digital filters, Kalman filters and multi-variable optimization tasks.

By design, the range of computational components is limited only by the scientists' needs, imagination and programming capabilities. Though the mathematical techniques vary widely, there are certain characteristics of these applications that are common.

Current mathematical methods, data models and algorithms will only evolve within the constraints of the current software architectures for ATLAS, CDF, and DayaBay. These experiments will not make major changes in software design or approach as they are taking data and their software stacks are quite mature.

Future experiments such as Super-B or the DUSEL experiments or Big BOSS may have new approaches, but each of these experiments have time frames further than five years out.

13.3. HPC Requirements

Computing for HEP experiments is particularly data-intensive, distributed and long lived. Collaborations are large and distributed, typically involving many countries, institutions and collaborators. The code base is developed on many different individual systems, which leads to the constraint that it is typically not highly optimized for any one system. Thus, portability of software for these experiments is vital. This applies to NERSC's PDSF system as well, and can lead to delays in loading and running the most recent versions of software at NERSC. This is not a NERSC-specific problem, but virtualization technologies that will help with this issue should be deployed and supported by NERSC and PDSF.

ATLAS jobs require on the order of 2 GB of RAM per core and all experiments require high bandwidth, stable network connectivity to the wide-area network. In order to support HPSS transfers and data-intensive computing, PDSF must have high-performance, robust, and stable disk access from all batch nodes.

Interconnect latency is not important as there is little or no inter-process communication in the calculations. The one exception is for inter-process filesystem utilities such as xrootd. xrootd is an I/O protocol for high-concurrency, random I/O. It is in use by several experiments and is being investigated by others, including Daya Bay. It allows remote I/O from within an application framework such as Gaudi, and can allow local disk (e.g., on PDSF batch nodes) to be used as a distributed file system.

Each experiment must synchronize the code base across many institutions and computer resources. This implies computer architectures must follow a larger, collaboration-wide movement from which no single site can depart. For the foreseeable future, ATLAS, CDF, Daya Bay and others will continue to focus on Linux clusters running variants of the Scientific Linux (SLCx) release.

Here are some key characteristics of codes being used on PDSF:.

Trivial Parallelism: Because of the physics independence of sequential simulated or recorded detector events, HEP codes on PDSF are trivially parallelizable. These codes do not require inter-processor communication during execution and are run as separate, autonomous processes on separate nodes.

Large Per-Processor Memory Usage: Relatively large conditions data, complex analysis chains, and memory-hungry component algorithms lead to a high memory usage per processor. Though ATLAS and others work hard to reduce their memory footprint for any particular production application, the memory footprint for Athena jobs requires 2 GB per core.

Smaller experiments like Daya Bay have much smaller memory footprints. Projections of Daya Bay memory usage are below 1 GB per core.

Data Intensive Computing: The current generation of high-energy physics experiments takes large amounts of data. Projected data volumes for ATLAS will reach 5 PB per year during full luminosity running. This includes raw, simulated, reconstructed, derived and conditional data. Simulation and analysis done at PDSF will only handle a fraction of the full volume, but the I/O to processor ratio of a typical ATLAS, Daya Bay or other HEP experiment is large.

Daya Bay is a much smaller experiment than ATLAS. However, NERSC is the U.S. Tier 1 center for Daya Bay and is a Tier 3 center for ATLAS. This means the total yearly data volumes of Daya Bay will be higher than those for ATLAS. Daya Bay projects 150 TB per year of raw, simulated and processed data in steady state starting in calendar year 2011.

The data-intensive and distributed nature of HEP computing means that our production codes are particularly sensitive to disk I/O and network bandwidth problems or instabilities. NERSC and

PDSF network bandwidth are exemplary. When problems arise, they are almost always traced back to the remote site. Disk I/O on PDSF is less satisfactory. The NERSC NGF system is prone to problems when many small files are opened simultaneously by many concurrent jobs. This is a typical failure mode when hundreds of copies of the same production job are launched. The startup phase of most such jobs involves reading many configuration files.

Grid: High energy physics researchers are early adopters of the concept of Grid computing and have been developing and depending on Grid services and utilities. The Open Science Grid is widely used by both the ATLAS and CDF experiments and is in consideration for Daya Bay.

Virtualization: Virtualization is becoming increasingly important as a computational strategy for HEP experiments as this is proving to be the most reliable and robust way to fully reproduce the complex, quickly evolving computing environments used for development, production and analysis.

13.4. Access to 50X resources at NERSC

It should be noted that NERSC and PDSF are a relatively small component of the global ATLAS and CDF computing resources. They are proportionately much more important to local ATLAS scientists. A 50x increase over five years is approximately fivefold over the conventional Moore's law increase of doubling every 18 months. This would make NERSC-using ATLAS collaborators more efficient and competitive relative to their colleagues and would significantly increase their contributions to ATLAS discoveries. CDF is unlikely to benefit much from increases on the five-year timescale as their analysis is likely to wind down by then.

NERSC and PDSF are the largest computing resource for Daya Bay and some other HEP experiments. Daya Bay, however, would not greatly benefit from such an increase in compute power as the experiment will be in its final years by that time.

Such an increase in resources would certainly attract many new HEP experiments and scientists to NERSC and allow smaller experiments tremendous opportunities for science discovery.

13.5. Support Services and Software

HEP computing relies on a large number of common, open-source software packages and tools such as MySQL, Python, Subversion, gcc, etc. In addition, there are many HEP-specific open-source software packages and tools required to develop, build and run HEP simulations and analyses, e.g., ROOT, GEANT4, AIDA, and CLHEP. For the most part, experimental teams have been happy with the responsiveness of the NERSC PDSF management in supplying required and requested software packages. The modules utility at NERSC is a powerful and important utility for customizing collaboration-specific software environments.

Virtualization technologies are a special case that will require additional collaboration between the NERSC PDSF management and representatives of the HEP experiments. Products like CERNVM, FUSE (Filesystem in Userspace), and Squid are being tested and used by ATLAS and Daya Bay and others as foundational to the development of virtual appliances simulation and analysis of physics data.

Data science gateways are an important addition to NERSC service capabilities. The ability to serve large amounts of scientific data by an organized, high-performance gateway will permit scientists in large, distributed collaborations like those in HEP to more effectively collaborate on science and shorten time to discovery for important HEP questions.

In addition, the PDSF interactive and batch model should be extended to explicitly support standing up of dedicated servers such as database servers, which are needed by most HEP experiments.

13.6. Emerging HPC Architectures and Programming Models

The HEP experimental community is already adopting software to take advantage of multi-core machines, and investigating coding for many-core architectures. These new architectures do pose a challenge and are recognized as an important issue to address in the medium term.

The code base of these experiments consists typically of millions of lines of C++, Python, Fortran, Java and C written by hundreds of scientists over tens of years. The prospect of rewriting such a system is so daunting as to be impossible. A dramatic change in computer architecture requiring such a rewrite would be prohibitively disruptive for any running experiment.

The community is currently researching concepts such as Viper for transparent optimization and scaling of scientific Python applications. This approach holds the possibility of insulating the typical scientist from the underlying hardware architecture and delegating expert knowledge of the hardware to computing science professionals. Viper would allow scientists to express their science intentions in a higher level language (Python in this case) which can be machine analyzed and factorized to take maximum advantage of underlying, changing computer architecture. Using such a system, would allow an experiment to rapidly adapt to new architectures. Without such a tool, HEP risks falling behind the rapidly changing hardware landscape in the long term.

Appendix A. Attendee Biographies

Amber Boehnlein is in the Physics Analysis Tool Group in the Fermilab Computing Division where she works on 3D graphics for the McFast detector simulation. She is currently an HEP program manager for Scientific Discovery through Advanced Computing (SciDAC) at DOE.

John Bell is a Senior Staff Mathematician at Lawrence Berkeley National Laboratory and leader of the Center for Computational Sciences and Engineering in LBNL's Computational Research Division. Prior to joining LBNL, he held research positions at Lawrence Livermore National Laboratory, Exxon Production Research and the Naval Surface Weapons Center. Bell's research focuses on the development and analysis of numerical methods for partial differential equations arising in science and engineering. He has made contributions in the areas of finite difference methods for hyperbolic conservation laws and low Mach number flows, discretization strategies for multiphysics applications, and parallel computing. He has also pioneered the development of adaptive mesh algorithms for multiphysics and multiscale problems. Bell's work has been applied in a broad range of fields, including aerodynamics, shock physics, seismology, flow in porous media and astrophysics. Bell is the author of more than 100 scientific publications. In 2005, he was awarded the 2005 Sidney Fernbach Award by the IEEE Computer Society. He was also co-recipient of the 2003 SIAM/ACM Prize in Computational Science and Engineering, awarded by the Society for Industrial and Applied Mathematics (SIAM) and the Association for Computing Machinery (ACM).

Julian Borrill is a computational cosmologist, specifically interested in the application of high performance computing (HPC) to the analysis of the most profound — and intractable — data sets in cosmology. His current work is focused on the coming generation of Cosmic Microwave Background (CMB) temperature and polarization measurements, including those of the Planck satellite, the EBEx balloon flights, and the PolarBear ground-based mission. While seeking general computational science solutions to the challenges of these data sets, much of this work is performed on the NERSC HPC systems. Other research areas range from simulations of the multi-dimensional energy knots expected in the first moments after the Big Bang to holistic performance evaluation of HPC systems using an application-derived benchmarking tool. He has previously worked at Dartmouth College, New Hampshire and Imperial College, London. He holds an M.A. in mathematics and political science from the University of Cambridge, an M.Sc. in Astrophysics from the University of London, an M.Sc. in Computer Science also from the University of London, and a D.Phil. in Physics from the University of Sussex.

David L. Bruhwiler is the Vice President for Accelerator Technology at Tech-X Corp. He has 20 years of experience in the development of algorithms and high-performance software for the design and simulation of particle accelerators and other beam and plasma devices. From 1992 through 1997, Bruhwiler designed RF photocathode electron guns and subsequent beamlines for the generation of high charge (>1 nC) sub-picosecond electron pulses. Upon joining Tech-X Corp., Bruhwiler co-developed a unique algorithm for modeling particle trajectories far from the accelerator axis. He also co-developed the electrostatic and electromagnetic PIC codes OOPIC Pro and VORPAL, including the use of MPI messaging and the parallel sparse matrix solver Aztec, as well as the co-development of multiscale algorithms. Bruhwiler has used these codes extensively to study plasma-based particle accelerator concepts. He is a member of the VORPAL development team and recent work has included implementation of a 4th-order electromagnetic update, participation in implementation of the Dey-Mittra algorithm for cut-cell boundaries and modeling photocathode electron sources. A recent focus has been the development, implementation and use of algorithms in VORPAL for modeling electron cooling physics.

Cameron Geddes is a physicist in the LOASIS program of Lawrence Berkeley National Laboratory. He received his Ph.D. in 2005 from the University of California, Berkeley. He is Principal Investigator computational projects modeling laser-driven accelerators, and also pursues experiments on laser guiding and control of particle injection in laser accelerators, and proton acceleration and x-ray production. Geddes received the 2006 American Physical Society Rosenbluth dissertation award in plasma physics and the 2005 Hertz Foundation dissertation prize, as well as an LBNL Outstanding Performance Award for his Ph.D. work demonstrating the plasma channel guided laser wakefield accelerator, in which laser pulse propagation was controlled by a pre-formed plasma channel resulting in production of monoenergetic beams for the first time in such an accelerator. Previous work includes Langmuir wave decay experiments in the Nova laser plasma physics group at Lawrence Livermore National Laboratory, ion wave mixing experiments at the Omega laser (Polymath research), small aspect Tokamaks (Princeton/U. of Wisconsin), and nonlinear optics (Swarthmore). He received the American Physical Society Apker Award for the outstanding undergraduate thesis, and the Swarthmore College Ellmore prize for outstanding work in physics (1997) for research on the equilibria of spheromak plasmas.

Chengkun Huang received his B.S. and M.S. degrees in engineering physics from Tsinghua University in 2000. After completing his Ph.D. degree in electrical engineering at UCLA in 2005, he was a post-doctoral researcher in the Plasma Simulation Group at UCLA and is currently a Technical Staff Member in the Applied Theoretical Research Division at Los Alamos National Laboratory. His research interests include plasma-based acceleration and high performance computing. He was the recipient of the 2007 Nicholas Metropolis Award for Outstanding Doctoral Thesis Work in Computational Physics for “for his innovative work in plasma physics that led to the development of the QuickPIC code that has revolutionizes the simulation of plasma-based accelerator research.”

Lie-Quan (Rich) Lee developed the first version of the Boost Graphics Library as a doctoral candidate at the University of Notre Dame. Now at SLAC National Accelerator Laboratory, his research interests include generic programming, scientific component libraries, and high performance computing. Lee is an active member of the Boost C++ Library Group.

Paul B. Mackenzie is a theoretical physicist at the Fermi National Accelerator Laboratory. He did graduate work in physics at Cornell University where he was a student of G. Peter LePage. He is an expert on Lattice Gauge Theory. He is the chair of the Executive Committee of USQCD, the U.S. collaboration for developing the necessary supercomputing hardware and software for QCD formulated on a lattice. Mackenzie has published 71 scientific papers listed in the SPIRES HEP Literature Database[1]. The most widely cited of them, "Viability of lattice perturbation theory" in Physical Review D 48 (5), pp. 2250–2264 (1993) had been cited 589 times as of March 2009. The second most widely cited, "On the elimination of scale ambiguities in perturbative quantum chromodynamics " Physical Review D 28 (1), pp. 228–235 (1983) has been cited 406 times.

Michael L. Norman, named San Diego Supercomputer Center (SDSC) director in 2009, is a distinguished professor of physics at UC San Diego and a globally recognized astrophysicist. Norman is a pioneer in using advanced computational methods to explore the universe and its beginnings. In this capacity, he has directed the Laboratory for Computational Astrophysics -- a collaborative effort between UC San Diego and SDSC resulting in the Computational Astrophysics Data Center (CADAC), a free service for the astrophysics community that hosts a public data collection of large astrophysical simulations and provides data-analysis resources worldwide. Following his appointment as SDSC's chief scientific officer in June 2008, Norman worked to foster collaborations across the UC San Diego campus for cyberinfrastructure-oriented research, development and education. He also serves as division director of SDSC's Cyberinfrastructure Research, Education and Development (CI-RED). Norman's work has earned him numerous honors, including Germany's prestigious Alexander von Humboldt Research Prize, the IEEE Sidney Fernbach Award, and several HPCC Challenge Awards. He also is a Fellow of the American Academy of Arts and Sciences, and the American Physical Society. He holds an M.S. and Ph.D. in engineering and applied sciences from UC Davis, and in 1984 completed his post-doctoral work at the Max Planck Institute for Astrophysics in Garching, Germany. From 1986 to 2000, Dr. Norman held numerous positions at the University of Illinois in Urbana, as an NCSA associate director and senior research scientist and as a professor of astronomy. From 1984 to 1986, Norman was a staff member at Los Alamos National Laboratory.

Peter Nugent is the co-leader of the Computational Cosmology Center (C3), a collaboration between Berkeley Lab's Physics Division and Computational Research Division (CRD), where Nugent is a staff scientist. His research interests include discovery and observation of supernovae of all types with the goal of understanding the physics of their explosions, their progenitor systems and nucleosynthesis products; Spectrum Synthesis of supernovae; Cosmology, specifically anything involving supernovae (Type Ia, IIP, etc.) to measure the cosmological parameters; and computational astrophysics. He is a member of the SciDAC Computational Astrophysics Consortium, the SNAP Collaboration GOSH, the SN Factory and, most recently, the Palomar Transient Factory and a former member of the Supernova Cosmology Project.

Alexander Szalay is the Alumni Centennial Professor of Astronomy at Johns Hopkins University. He is also professor in the Department of Computer Science. He is a cosmologist, working on the statistical measures of the spatial distribution of galaxies and galaxy formation. Born and educated in Hungary, after graduation Szalay spent postdoctoral periods at UC Berkeley and the University of Chicago, before accepting a faculty position at Johns Hopkins. He is the architect for the Science Archive of the Sloan Digital Sky Survey and collaborated with Jim Gray of Microsoft to design an efficient system to perform data mining on the SDSS terabyte-sized archive, based on innovative spatial indexing techniques. He is leading a grass-roots standardization effort to bring the next generation terabyte-sized databases in astronomy to a common basis, so that they will be interoperable – the Virtual Observatory. Szalay is project director of the NSF-funded National Virtual Observatory. He has written over 340 papers in various scientific journals, covering areas from theoretical cosmology to observational astronomy, spatial statistics and computer science. In 1990 Szalay was elected to the Hungarian Academy of Sciences as a Corresponding Member. In 2003 he was elected as a Fellow of the American Academy of Arts and Sciences. In 2004 he received one of the Alexander Von Humboldt Prizes in Physical Sciences.

Craig Tull is group leader of the Science Software Systems group in the Advanced Copmputing for Science Department at Lawrence Berkeley National Laboratory. Tull has a Ph.D. in Physics from University of California, Davis, and has been developing scientific software and managing software projects for more than 25 years. His interests are in component frameworks, generative programming, and using scripting languages to enhance the power and flexibility of scientific data exploration. He has worked on science frameworks for several experiments, including as framework architect in the STAR experiment, and as leader of the LBNL framework effort in ATLAS. Tull has worked on the PPDG (Particle Physics Data Grid) and the GUPFS (Global Unified Parallel File System) projects that aim to deliver innovative solutions to data-intensive computing in the distributed environment. He recently ended a three-year assignment in DOE headquarters as program manager for Computational High Energy Physics including HEP's SciDAC portfolio, and is currently the U.S. manager of Software and Computing for the Daya Bay neutrino experiment in China.

Stan Woosley's interests in the origin of the elements and the death of massive stars have led him to do theoretical work in diverse fields. On the one hand, he studies nucleosynthetic "processes," the nuclear physics and theoretical astrophysics whereby the jigsaw puzzle of abundances that we see in stars has been assembled. This requires a firm grounding in nuclear physics, but also a thorough understanding of the lives of stars and their deaths as supernovae. Since the latter is poorly understood, Woosley and his many collaborators also use supercomputers and develop the necessary software to study supernovae and gamma-ray bursts of all types. Woosley proposed the "collapsar" model for gamma-ray bursts and was a co-investigator on the High Energy Transient Explorer that studied them. He is currently the Principal Investigator for the nine-institution Computational Astrophysics Consortium funded by DOE's Scientific Discovery through Advanced Computing (SciDAC) Program. This consortium is dedicated to a better understanding of supernovae of all types achieved through a combination of analytic studies and supercomputer models. Woosley is a professor in the Physics Department at UC Santa Cruz and has a Ph.D. in Space Science from Rice University.

Panagiotis Spentzouris is a scientist in the Computing Division and the Accelerator Physics Center of the Fermi National Accelerator Laboratory. Since 2001, his main research interest has been computational accelerator physics. He serves as the head of the Accelerator and Detector Simulation and Support department in the Computing Division, and is the PI of the SciDAC2 ComPASS project.

Doug Toussaint's research involves the use of massively parallel computers to calculate some of the most fundamental quantities in high-energy physics. He employs lattice gauge theory to calculate the masses and lifetimes of strongly interacting particles, the weak interactions of these particles, the behavior of nuclear matter at very high temperatures, and the structure of the electroweak interactions. Toussaint is a professor in the Physics Department at the University of Arizona. He earned his Ph.D. in Physics from Princeton University in 1978.

Appendix B. Workshop Agenda

Thursday, November 12

8:00 am	Arrive, informal discussions	
8:30 am	Welcome, introductions, workshop goals, charge to committee	Yukiko Sekine, DOE-SC/ASCR
8:45 am	Workshop outline, logistics, format, procedures	Harvey Wasserman, NERSC
9:00 am	HEP Program Office Research Directions	Amber Boehnlein, DOE / HEP
9:30 am	NERSC Role in High Energy Physics Research	Kathy Yelick, NERSC Director
10:15 am	Break	
10:30 pm	Case Studies: Accelerator Physics	Panagiotis Spentzouris, Discussion Leader
Noon	Working Lunch	
1:00 pm	Case Studies: Astrophysics — Data Analysis	Julian Borrill, Discussion Leader
2:00 pm	Case Studies: Astrophysics — Modeling	Stan Woosley, Discussion Leader
3:00 pm	Break	
3:15 pm	Case Studies: Lattice QCD	Doug Toussaint, Discussion Leader
4:15 pm	Case Studies: HEP Detector Simulation and Data Analysis	Craig Tull, Discussion Leader
5:15 pm	Time for general discussions; overflow from previous sessions	
6:00 pm	Adjourn for the day, self-organize for dinner	

Friday, November 13

8:00 am	Arrive, informal discussions	
8:30 am	Summary of previous day's discussions	Richard Gerber, Harvey Wasserman, NERSC
9:00 am	Additional time for case studies if needed, final report planning	
10:00 am	Breakout sessions (in same room)	
11:00 am	Full group discussion	
11:30 am	If time permits, discussion of individual issues	
Noon	Adjourn	

Appendix C. Abbreviations and Acronyms

ALCF	Argonne Leadership Computing Facility
AMR	adaptive mesh refinement
ASCR	Advanced Scientific Computing Research
BAO	Baryon Acoustic Oscillations
BELLA	Berkeley Lab Laser Accelerator
CCSE	Center for Computational Sciences and Engineering at LBNL
CERN	European Organization for Nuclear Research
CLIC	Compact Linear Collider
ComPASS	Community Petascale Project for Accelerator Science and Simulation
CMB	Cosmic Microwave Background
CUDA	Compute Unified Device Architecture
EM	electromagnetic
ESnet	DOE's Energy Sciences Network
FACET	Facility for Accelerator Science and Experimental Tests
FDTD	finite-difference time domain
FNAL	FermiLab National Accelerator Laboratory
FFT	Fast Fourier Transform
GPU	Graphical Processing Unit
HDF	Hierarchical Data Format
HEP	High Energy Physics Office of Department of Energy
HPC	high-performance computing
HPSS	High Performance Storage System
I/O	input output
IDL	Interactive Data Language visualization software
ILC	International Linear Collider
INCITE	Innovative and Novel Computational Impact on Theory and Experiment
JDEM	Joint Dark Energy Mission
LANL	Los Alamos National Laboratory
LBNL	Lawrence Berkeley National Laboratory
LHC	Large Hadron Collider
LINAC	linear accelerator
LLNL	Lawrence Livermore National Laboratory
LOASIS	Lasers and Optical Accelerator Systems Integrated Studies program at LBNL
LSST	Large Synoptic Survey Telescope
LWFA	Laser Wakefield Acceleration
MPI	Message Passing Interface
MGFLD	Multi-group flux-limited diffusion
MILC	MIMD Lattice Computation Collaboration
MIMD	Multiple Instruction Multiple Data
NERSC	National Energy Research Scientific Computing Center
NetCDF	Network Common Data Format
NGF	NERSC Global Filesystem
NLTE	nonlocal thermodynamic equilibrium

ORNL	Oak Ridge National Laboratory
OS	operating system
PTF	Palomar Transient Factory
PDSF	NERSC's Parallel Distributed Systems Facility
PETSc	Portable, Extensible Toolkit for Scientific Computation
PIC	Particle In Cell
PWFA	Plasma Wakefield Acceleration
QCD	Quantum Chromodynamics
RF	radio frequency
SC	DOE's Office of Science
SciDAC	Scientific Discovery through Advanced Computing
SLAC	SLAC National Accelerator Laboratory
SN	supernova
SNL	Sandia National Laboratories
SRF	Superconducting Radio Frequency
UPC	Unified Parallel C programming language
USQCD	United States Lattice Quantum Chromodynamics Collaboration

Appendix D. About the Cover

Left top: Visualization of a 3-D laser plasma accelerator simulation using VORPAL. This simulation of LOASIS (LBNL) experiments models the acceleration of particles in a laser-plasma particle accelerator at very high gradients and the self-consistent evolution of the plasma wave (wake) driven by the radiation pressure of a laser pulse, which forms the accelerating structure. Shown is a volume rendering of the wake (blue) and a particle bunch (green for low energy, yellow for high energy). Simulations such as these are being used to improve future LOASIS experiments. Image courtesy of Cameron Geddes, LBNL.

Middle top: This image, created by Professor Claudio Rebbi of Boston University, visualizes a quark field after ten time units of propagation since the quark was created on a Quantum Chromodynamics lattice. Software from the USQCD collaboration was used. The figure shows one component of a quark propagator (since displaying all three components would have been difficult). Quarks are described by three complex numbers that represent their so-called "color" degree of freedom over all the space lattice coordinates. Since the quark component being displayed is a complex number, it has a magnitude and a phase, with the radius of the spheres representing the magnitude and the color representing the phase (going towards red for positive real, towards yellow-green for positive imaginary, towards cyan for negative real, etc.).

Right: CASTRO Adaptive grids superimposed on entropy in a Type II (core collapse) supernova. Image courtesy of Prof. Adam Burrows and Dr. Jason Nordhaus, Princeton University.

Middle bottom: Images of the microwave sky temperature and two polarization components from recent Cosmic Microwave Background simulation work at NERSC. Images courtesy of Julian Borrill, LBNL.

Left bottom: Simulation of plasma-based acceleration — in which electrons or positrons gain energy by surfing on a wave generated by a particle beam in an ionized gas. The image shows the density profile (blue color) and the laser pulse (red color) superimposed on the density map. This snapshot is at the end of the plasma (z = 8.5 mm), where the highest energy electrons have begun to dephase, producing a 700 MeV electron beam. Image courtesy of Warren Mori of the University of California, Los Angeles.

NERSC

U.S. DEPARTMENT OF
ENERGY
Office of Science

BERKELEY LAB
Lawrence Berkeley
National Laboratory