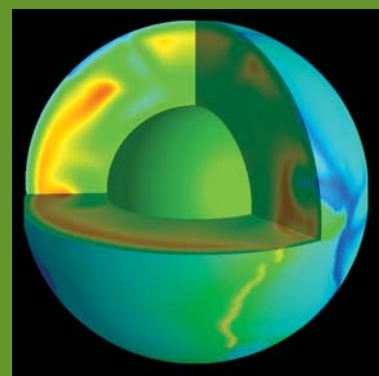
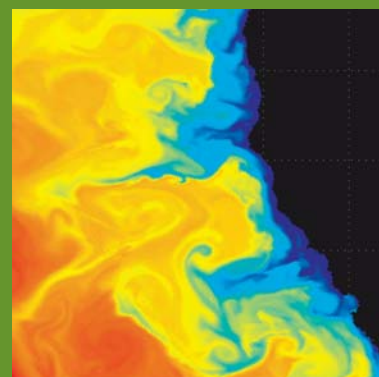
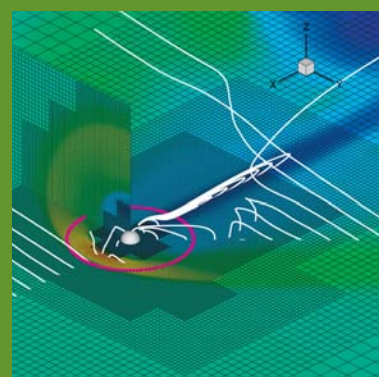
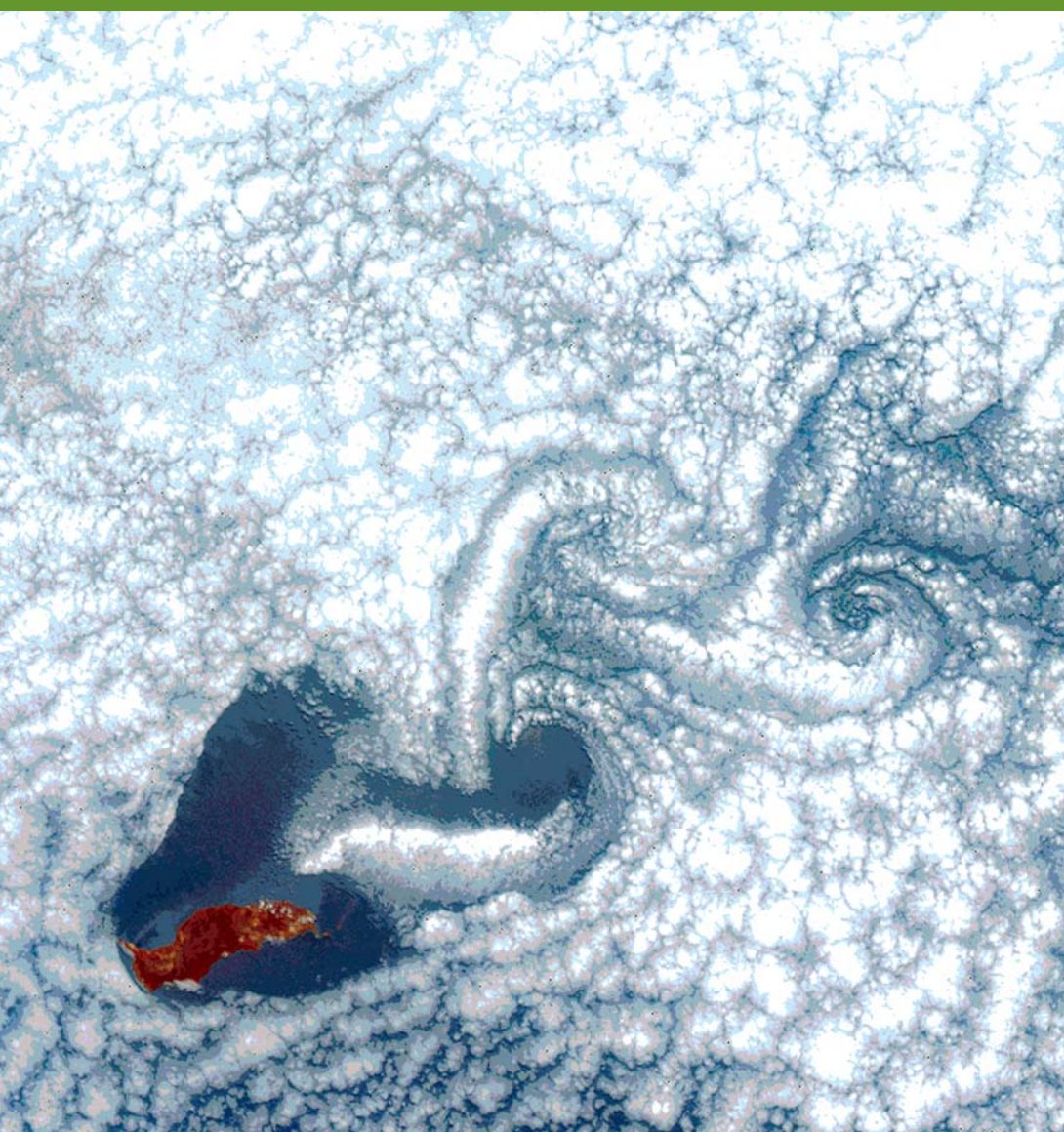


ESTABLISHING A PETASCALE COLLABORATORY FOR THE GEOSCIENCES



Technical and Budgetary Prospectus

PREFERRED CITATION

Technical Working Group and Ad Hoc Committee for a Petascale Collaboratory for the Geosciences. 2005. *Establishing a Petascale Collaboratory for the Geosciences: Technical and Budgetary Prospectus*. A Report to the Geosciences Community, UCAR/JOSS. 56 pp.



This MISR image from June 11, 2000 (Terra orbit 2569) demonstrates a turbulent atmospheric flow pattern. The alternating double row of vortices (made visible by clouds) were formed in the wake of the eastern Pacific volcanic island of Guadalupe. The image was provided by NASA/GSFC/JPL and the MISR Team.

**ESTABLISHING
A PETASCALE
COLLABORATORY
FOR THE GEOSCIENCES**

Technical and Budgetary Prospectus

A Report to the Geosciences Community

October 2005

Report Prepared By

The Technical Working Group for a Petascale Collaboratory for the Geosciences

Richard Loft (chair)..... National Center for Atmospheric Research
Chaitan Baru San Diego Supercomputer Center
Maurice Blackmon National Center for Atmospheric Research
John Boisseau..... Texas Advanced Computing Center
Jack Dongarra University of Tennessee, Knoxville
John Drake Oak Ridge National Laboratory
John Helly..... San Diego Supercomputer Center
James Kasdorf Pittsburgh Supercomputing Center
Tommy Minyard..... Texas Advanced Computing Center
Rob Pennington..... National Center for Supercomputer Applications
Avi Purkayastha Texas Advanced Computing Center
Allan Snavelly..... San Diego Supercomputer Center
Bob Wilhelmson..... National Center for Supercomputer Applications
Larry Winter National Center for Atmospheric Research

The Ad Hoc Committee for a Petascale Collaboratory for the Geosciences

Frank Bryan (chair)..... National Center for Atmospheric Research
Ronald Cohen Carnegie Institution of Washington
Inez Fung University of California, Berkeley
Tamas Gombosi University of Michigan
Jim Kinter Center for Ocean-Land-Atmosphere Studies
Bill Smyth Oregon State University
LuAnne Thompson University of Washington
Jeroen Tromp California Institute of Technology

Acknowledgements

In addition to the members of the committees listed above, many members of the geoscience community contributed to this report. We thank the following for their assistance: J. Anderson, J. Antoun, J. Arblaster, H. Batchelder, A. Bennett, C. Bitz, P. Bunge, G. Bonan, A. Busalacchi, G. Carmichael, D. Devenyi, T. Donato, A. Donnellan, L. Donner, G. Erlebacher, G. Fox, I. Fukumori, T. Fuller-Rowell, W. Geyer, G. Gisler, G. Glatzmeyer, C. Goodrich, T. Fuller-Rowell, J. Helly, E. Hunke, D. Jacob, L. Jaegle, D. Keyser, S. King, J. Klemp, B. Kirtman, E. Kostelich, W. Kuang, J.-F. Lamarque, S. Legg, D. Lettenmaier, M. Li, P. Lichtner, Y. Lvov, J. McWilliams, G. Meehl, J. Michalakes, M. Moncrieff, T. Pugh, D. Randall, J. Rustad, A. Sandu, M. Spiegelman, P. Sullivan, I. Szunyogh, Z. Toth, B. Travis, C. Williams, D. Yuen, and C. Zender. We also gratefully acknowledge the administrative support provided by Cathy Clark and Tara Jay at the UCAR Joint Office of Science Support, and Lisa Butler in the NCAR Climate and Global Dynamics Division.

CONTENTS

Executive Summary	1
Architectural Trends	8
Memory Latency and Bandwidth.....	8
Parallel Computing	9
Technology Forecast.....	10
Best Practices for System Design and Procurements	12
Application Analysis	14
Ocean Science Applications	16
Atmospheric Science Applications.....	21
Earth Science Applications.....	30
Space Science Applications	38
Facilities	41
Collaboratory Project Description	42
PCG Cost Model.....	42
PCG Disk Subsystem	44
PCG Mass Storage Systems	44
PCG Data Analysis and Visualization Systems	44
Networking and PCG Grid Services	45
Data Center Issues.....	45
Staff Costs	46
Total Cost: A PCG Project Scenario	48
References	49
Appendix 1. Chronology and Methods	50
Appendix 2. Charge to Technical Working Group	51
Appendix 3. Petascale Application Questionnaire	53
Acronyms	55

GLOSSARY OF TERMS

Bandwidth. The amount of data that is able to be sent over a network, measured in Kilobytes and Megabytes per second (Kbps and Mbps).

Bisection bandwidth. The bidirectional capacity of a network between two equal-sized partitions of nodes. The cut across the network is taken at the narrowest point in each bisection of the network.

Capability computing. A type of large-scale computing in which one wants to accommodate very large and time-consuming computing tasks. This requires that systems be managed with the highest priority for this type of computing—possibly with the consequence that the computing resources in the system are not always used with the greatest efficiency.

Capacity computing. A type of large-scale computing in which one attempts to achieve the highest possible throughput capacity using the machine resources as efficiently as possible. Achieving this goal may have adverse effects on the performance of individual computing.

Latency. In a network, latency, a synonym for delay, is an expression of how much time it takes for a packet of data to get from one designated point to another.

Leadership-class computer. DOE terminology for very large computing systems with unequaled performance and/or unique capabilities.

Petascale computer. A supercomputer with a realizable peak floating point performance of one PetaFLOPS.

Spatial locality. A program is said to have spatial locality if the memory reference stream it issues tends to access adjacent words of memory.

Temporal locality. A program has temporal locality if memory references close to one another in the stream it issues repeatedly access the same pieces of data.

EXECUTIVE SUMMARY

Establishing a Petascale Collaboratory for the Geosciences: Technical and Budgetary Prospectus (this document) is a companion document to *Establishing a Petascale Collaboratory for the Geosciences: Scientific Frontiers* (Ad Hoc Committee and Technical Working Group for a Petascale Collaboratory for the Geosciences, 2005), which presents the scientific justification for creating a Petascale Collaboratory for the Geosciences (PCG). That justification is based on evidence that making dramatically increased computational resources available to the geoscience community will enable scientists to conduct transformative research on the fundamental processes operating in the Earth system, their interaction across scales, and the feedback mechanisms among them. Specifically, the vast increase of computational power will allow scientists, across a number of disciplines, to perform simulations at much higher resolutions, take into account previously neglected phenomena, or employ more accurate, but more computationally costly, formulations of important phenomena. This in turn will improve model realism, reducing uncertainty derived from these computer models, for example, in predicting the variability of the response of the Earth system to human activity. Because of the opportunities for new scientific discovery and the importance of predictions of Earth system processes to society, the *Scientific Frontiers* document makes the overarching recommendation to “Establish a Petascale Collaboratory for the Geosciences with the mission to provide leadership-class computational resources that will make it possible to address, and minimize the time to solution of, the most challenging problems facing the geosciences.”

The petascale system for the geosciences will also allow U.S. geoscience research to remain competitive internationally in these critical areas. Media reports indicate that the Japanese government plans to form a collaboration with university, national research labs, and industrial partners to develop an Earth Simulator follow-on system. The new system is projected to be a 10 PetaFLOPS-class machine operational by

2011. The estimated cost of the Earth Simulator follow-on is projected to reach 100 billion Japanese Yen (approximately \$875M). It is not clear from reports what proportion of these costs constitute system, facility construction, staff, or operating costs. Additionally, unlike the Earth Simulator, this follow-on system would attack a broad set of computational challenges; geoscience would be allocated only a fraction of this capability. Europe is contemplating a similar petascale project for geoscience.

The technical feasibility of the PCG’s scientific vision is determined by the architectural trends in the supercomputing industry, the current and projected computational requirements of the applications, and the overall cost of the project. This document evaluates each of these aspects and discusses, in general terms, how best to construct the PCG within a five-year time frame, starting in 2007. Two important budgetary constraints were placed on this feasibility study: (1) the total project cost should fall between \$100-500M and (2) the funds would be made available in relatively fixed annual installments over five to six fiscal years.

The scientific requirements of the geoscience community have been translated into quantifiable computational requirements for the PCG: the results of this process are discussed in detail in the Applications section of this report. In turn, these requirements can be understood in terms of the impact of the PCG computational capability on scientific capabilities (Table 1). This table highlights the following scientific impacts from the *Scientific Frontiers* report:

- To be truly useful in decision-making, climate models must make credible predictions on a regional scale. To do so requires understanding the influence of important features such as individual mountain ranges or ocean western boundary currents on the regional climate response.
- Most current global ocean simulations have resolutions too coarse to represent the ubiquitous ocean mesoscale

Table 1. Capability Enhancement of Petascale System by Discipline

Discipline	Requirement	Current Capability	PCG Capability
Climate Modeling ¹	5 simulated yrs/day	Resolve atmosphere and ocean at 110 km and larger, parameterize mesoscale processes	Directly resolve mesoscale structure of ocean (10 km) and atmosphere (20 km)
Oceanography ²	40 simulated yrs/month	10-20 km eddy-permitting global circulation models	5-10 km eddy-resolving global circulation models coupled to ecosystem models with 10-20 biological constituents
Weather Research ³	2 simulated hrs/day	3 km thunderstorm simulation	10 m tornado simulation
Seismology: Earthquake Simulation ⁴	10 global earthquake simulations per month of waves above 1 Hz	O(10 billion) grid points: global seismic wave analysis limit 0.3 Hz	O(500 billion) grid points: global seismic wave resolution at 1 Hz resolution
Seismology: Imaging Earth's Interior ⁵	1 global assimilation of thousands of earthquakes per month	1000 km resolution of Earth's interior	Imaging at 100 km resolution of core-mantle boundary in Earth's interior
Hydrology ⁶	1 decadal basin-scale simulation per week	1-year simulation of 1 km Rio Grande River Basin	Decadal 1 km Columbia River Basin (100 times larger than Rio Grande River Basin)
Space Weather ⁷	Coronal mass ejection faster than real time	Resolve magnetic configuration associated with large sunspots: 1/40 solar radius	Resolve fine structure of corona magnetic field inside active regions: 1/320 solar radius

¹Scientific Frontiers, p. 18-21; ²Scientific Frontiers, p. 21-22; ³Scientific Frontiers, p. 28-29; ⁴Scientific Frontiers, p.44; ⁵Scientific Frontiers, p. 46; ⁶Scientific Frontiers, p. 38-39; ⁷Scientific Frontiers, p. 14-15.

eddies, requiring their effects on the large-scale circulation and climate to be parameterized. The PCG will open the possibility of representing the global ocean as the turbulent fluid that it really is, and will permit investigation of the interaction of mesoscale eddies with the large-scale circulation and with marine biogeochemical cycles.

- A large-scale atmospheric model that resolves individual clouds will revolutionize atmospheric science by making it possible, for the first time, to simulate the myriad dynamical, microphysical, and radiative processes that make up the complex interactions of clouds and weather systems.
- Currently, the largest supercomputers are capable of doing global earthquake simulations with on the order of 10 billion grid points, which captures seismic waves that have periods of about 3.5 seconds. However, observational seismologists routinely analyze seismic signals in the 1-Hz

range (waves with periods of 1 second) to study the structure of the inner core as well as the fine structure of the lowermost mantle, in particular the D" layer and ultra-low velocity layer at the core-mantle boundary. The PCG will permit computational seismologists to reach resolutions of 500 billion grid points and periods of less than 1 second. This parity with observational analysis is important if the predictions made by seismic models are to be improved.

- Solving seismological inverse problems is how seismologists image the Earth's interior: current tomographic mantle models resolve 1000-km length scales; the PCG will enable adjoint calculations for thousands of earthquakes, improving the resolution by an order of magnitude, bringing the fine structure of the Earth's mantle and inner core into focus for the first time.
- The responses of soil moisture, evapotranspiration, and runoff to precipitation are controlled by spatial variabil-

ity of soil properties, topography, and vegetation down to sub-meter scales, and so must be parameterized in large-scale models. Yet, the physical bases for these parameterizations, for example, their relationship to soil grain size and the role of fractured porous media, are poorly understood, a situation that can only be improved by modeling experiments that explicitly resolve these scales.

- The PCG will allow space weather forecasts to resolve the fine structure of the corona magnetic field, which is thought to occur on scales smaller than 1/70th of the sun's radius. This capability will allow more accurate predictions of the damaging effects of such flares on satellites and power-distribution systems.

Simply put, the PCG represents to computational geoscience what the Hubble Space Telescope represents to the observational astronomer: sudden clarity and sharpness replacing a blurred view of nature.

The Technical Working Group was charged with developing a project plan to realize the vision of the PCG. The constraints on the PCG project, in terms of timing, overall budget, and funding profile have certain consequences for the layout of the project plan. Although a compelling scientific case has been made for an immediate need to provide significant computational capability to the geoscience community, it cannot be ignored that the cost of a given amount of computer performance is dropping at a dramatic rate—faster than Moore's Law. This fact argues strongly for a later deployment date for the petascale system. In balancing these two competing factors, 2010 emerged in the committee's thinking as a likely date for the petascale system's deployment. The reasons for this determination are manifold: 2010 is the first point at which projected price performance and power consumption projections for PetaFLOPS (PFLOPS) supercomputers reach feasible levels of power efficiency and cost performance. Also, from a computer technology perspective, the period around 2010 is significant, because there are numerous indications from computer manufacturers that supercomputer architectures must and will change dramatically because of the increasing gap between CPU and memory speed. Finally, novel systems developed under DARPA's High Productivity Computing Systems (HPCS) program should be available in this time frame.

A three-year lead time in the project before deployment of the petascale system allows certain components of the plan to unfold in a natural way. It will take time to hire and train staff and to procure and deploy critical supporting facilities and cyberinfrastructure. On the application side, time and effort is needed to prepare some models to run efficiently on highly parallel systems or to develop new numerical experiments, methods, or parameterization schemes; the existing data analysis and visualization tool chains must be scaled up for the torrent of data that will be produced by such systems.

As will be seen in the Facilities section (see p. 41), the working group envisions the first phase of PCG development to consist of the deployment of two to three "midrange" supercomputing systems, comprising, in aggregate, a total of 100-200 TeraFLOPS (TFLOPS) of peak floating point performance. Currently, the Japanese Earth Simulator has 40.96 TFLOPS of peak performance, which U.S. researchers have access to in collaboration with Japanese scientists. The terascale systems proposed in the project's first phase will allow geoscientists access to Earth Simulator-scale systems for application development and testing as well as for simulations, data analysis, and visualization not possible today. Problems with facilities, grid technologies, and software tools for data analysis and visualization will be all "shaken out" during this first phase. Phase one will also give the collaborative engineers and computer scientists a chance to evaluate the viability of different platforms, setting the stage for the 2010 petascale system procurement. Phase two will then focus on providing a robust PCG that includes the petascale system coupled to the phase one terascale systems via a high-speed wide-area network.

The cost model for this two-phase deployment and operation is discussed in the Facilities section. It captures costs for the supercomputers and related systems, accounts for data archival costs, and takes into consideration operating costs such as utilities, payroll, and lease payments. To deploy the petascale system by 2010, the cost model indicates that about 60% of the project budget will need to be spent on computers, the rest going to staff and operating and data-archive costs. An overall PCG budget of \$390M over six years is projected for this scenario.

RECOMMENDATIONS

The committee has developed the following specific recommendations for the PCG project, which are grouped into categories of Computing Services, Data Analysis and Visualization Services, Interface and Collaboration Services, Physical Infrastructure, Software Infrastructure, and Planning. These recommendations were designed to help optimize scientific productivity, minimize technological risk, and maximize the cost effectiveness of resource deployment within the scope of the project.

Computing Services

Recommendation 1: Procure computing systems in a three-year technical refresh cycle with the first procurement in 2007, a second in 2010, and with a project renewal review in 2012. This time frame permits the PCG computing equipment to remain current and track the latest technological developments.

The period 2007-2012 was chosen as the planning window taking into consideration the current pent-up demand for high-end computing resources, the technology trends for high-performance computing systems, and the timetable associated with the budgetary mechanism that we expect to be pursued in funding this activity (i.e., an NSF Major Research Equipment and Facilities Construction [MREFC] proposal).

In the following recommendations, we define a terascale system as one that delivers 1-10 TFLOPS sustained on a broad spectrum of geofluids codes, and a petascale system as one that delivers 100 or more TFLOPS sustained.

Recommendation 2: During phase-one deployment in 2007, the project should procure two to three terascale systems (each 50-100 TFLOPS peak, totaling perhaps 100-200 TFLOPS in aggregate). Some or all of these machines could be located at computing facilities that will become nodes of the distributed collaboratory. This will not only reduce risk by exploring different architectural approaches, but also has the advantage of allowing systems to be procured with different designs and different balances of FLOPS/local memory bandwidth/global memory bandwidths so that applications can be run on the most appropriate architecture. Efforts should be

made to select and configure one of these systems to deliver leadership-class, sustained performance to geosciences applications, and serve as the “centerpiece” of the collaboratory.

Recommendation 3: In 2010, procure a new “centerpiece” system with 1 PFLOPS peak processing speed. Locate this system at a single facility because the computing platform must be physically tightly coupled in order to achieve sustained performance of 100 TFLOPS and beyond on grand-challenge Earth system simulations.

The phase one terascale systems will give geoscientists access to computing resources better than Japan's Earth Simulator while the PCG staff, physical infrastructure, and cyberinfrastructure necessary to support the phase two petascale system are brought online. These terascale systems will also provide PCG engineers and computer scientists with the opportunity to evaluate the viability of different platforms, setting the stage for the 2010 petascale procurement. An advantage of the timing of the second procurement in 2010 is that it admits the possibility of leveraging technological breakthroughs derived from DARPA's HPCS program, for example.

Based on current technology trends, the petascale computing system in 2010 is expected to consist of between 10,000 and 100,000 processors arranged, perhaps, in hundreds of cabinets. To achieve adequate scalability on geosciences applications, the parallel computing systems should be configured with careful attention to interprocessor communication performance.

Recommendation 4: Based on our analysis, we find that many geoscience applications will perform well on systems with a ratio of per processor bisection bandwidth to sustained FLOPS of at least 0.25. Thus, terascale systems should have on the order of 1 TB/sec, and the petascale system will likely need 25 TB/sec, of bisection bandwidth. Based on a latency sensitivity analysis of several applications studied in this report, message-passing latency should be below six microseconds in the 2007 terascale systems, and below two microseconds in the 2010 petascale system. Both of these latency targets are readily achievable technologically. The systems should also be capable of scalable global operations, such as synchronizations and reductions, which may require special architectural and operating system features to deliver.

The terascale systems procured during phase one can continue to provide substantial resources to the geosciences research community during phase two of the project.

Recommendation 5: Maintain some of the terascale systems procured during the initial deployment in service through phase two to provide continuity, to help meet the capacity computing needs of the community, and to avoid diluting the capability of the petascale system.

Data and Visualization Services

Ground-breaking scientific results justify the PCG and can only be enabled in an environment in which scientific productivity is paramount. Therefore, the collaboratory must be equipped to provide data storage, access, analysis, and visualization capability commensurate with a petascale compute server. Based on experience in research computing facilities and analysis in this report, we have two recommendations.

Recommendation 6: Equip the phase two PCG with mass storage systems capable of archiving approximately 100 PB per year of operation.

Recommendation 7: Invest approximately 10% of the overall computing resource in specialized data analysis and visualization systems, collocated with each of the systems in the collaboratory.

This ratio of data analysis and visualization to supercomputing systems is based on that observed at the NSF computing facilities at the National Center for Supercomputing Applications (NCSA) and the National Center for Atmospheric Research (NCAR).

Interface and Collaboration Services

Production-quality distributed and grid-based services and user services are critical to maximizing scientific productivity.

Recommendation 8: Leverage the technical expertise and cyberinfrastructure available in existing national research computing facilities and universities to support the effective scientific use of PCG computing resources. Leveraging re-

sources can best be accomplished by creating collaboratory nodes at appropriate centers where specialized domain expertise resides.

Recommendation 9: Staff the built-out PCG with approximately 75 new hires, distributed among the petascale facility itself and the collaboratory nodes, and embedded within science teams, to provide robust operations and to provide adequate support to PCG users.

Physical Infrastructure

Existing facilities can be used to house some or all of the grid-based elements of the collaboratory, such as the phase one terascale systems and the data analysis and visualization systems. The power, space, and cooling infrastructure requirements for the petascale facility envisioned for 2010 are unprecedented at current national academic computing facilities, and are likely most cost-effectively met by new construction.

Recommendation 10: Equip the petascale computing facility with the all the attendant power, cooling, and redundancy features typical of a modern best practices (tier 2+) IT center.

The tier-level nomenclature for data centers summarizes the design reliability and fault tolerance of a data center or data center design. A tier 2 center is defined by the Uptime Institute white paper (available at <http://www.uptime.com/TUIpages/whitepapers/tuitiers.html>) as one composed of a single path for power and cooling distribution, with redundant components, providing 99.741% availability. A tier 2+ data center refers to the concept of planning for the addition of multiple active power and cooling distribution paths should higher levels of reliability be desired.

Recommendation 11: Consider leasing a data center of the appropriate scale. Leasing lowers the cost of the overall project and allows increased planning flexibility regarding the disposition of the computing facility at the end of the project.

Software Infrastructure

Recommendation 12: Leverage software infrastructure development at existing supercomputing centers and within various computer and software projects supported by NSF,

DOE, NASA, and DARPA. These include efforts at NCAR (<http://www.scd.ucar.edu/>), NCSA (<http://www.ncsa.uiuc.edu/>), and SDSC (<http://www.sdsc.edu/>) as well as grid developments such as those undertaken by the TeraGrid (<http://www.teragrid.org/>).

Because the petascale collaboratory will involve computer systems at multiple sites (most likely one petascale “center-piece” system and several midrange terascale systems), it is important to leverage grid developments in middleware, security, data, and gateways to integrate these sites into a useful grid environment that extends to the user’s desktop.

One significant advantage of a PCG is that it would catalyze broader collaborations in geoscience. One will not only need a general grid infrastructure, but also a geoscience-specific infrastructure of data repositories, data management and analysis tools, scalable computational frameworks, and applications and analysis tools that all interoperate. Two questions then need to be addressed: Who will develop and maintain, during the next decades, this common geoscience software infrastructure? What will be the standard languages and libraries that will be supported by PCG? Such software infrastructure requires fairly large software development teams—tens of people working over many years. However, the funding life cycles of many extant projects are too short, disjointed, and modeled on research activities rather than a sustained infrastructure support model. Although the hardware platform will change every 3-5 years, the software infrastructure will perhaps persist for two decades. Its proper design and implementation is much more important than a proper choice of hardware platform. Many software aspects are idiosyncratic to the geoscience domain they serve.

Recommendation 13: Investment in the PCG must be accompanied by commensurate and substantial investment in the supporting geoscience software infrastructure.

Planning

A system of the size and power contemplated by this study will necessarily push the technology envelope in many dimensions and will require close collaboration between the

geosciences research community and computer vendors. The PCG will also serve as a strong driver for promoting emerging supercomputing technologies and computational science, and it will also provide basic economic impetus for the advancement of computing technology in general. Further, the data links between the PCG and other computational resources will require attention to network and grid technologies that support scientific inquiry.

Recommendation 14: To address the technical challenges of creating a petascale system for the Earth System sciences, an in-depth dialog must be developed immediately among scientific researchers, computational scientists with domain expertise, and key system vendors. This dialog will allow computer vendors and the user community to further refine the computational requirements sketched out in this report. It may possible to positively influence the design of end-of-the-decade supercomputers relative to the needs of geoscience applications, but this window of opportunity will soon close. Finally, to be maximally effective, the scope of this dialog may need to ultimately be broadened beyond geoscience to include other scientific communities.

Developing the PCG, therefore, represents an opportunity for focused communication and cooperation among various sectors of the nation’s supercomputing community. This opportunity can best be exploited if all parties work together to improve general understanding of the computational characteristics of geoscience applications as they relate to computer architectures. One methodology for characterizing application performance, detailed performance modeling, is further discussed in the section on architectural trends (see p. 8). In the committee’s view, the best way to promote this dialogue is by creating a series of workshops on the subject.

New geosciences frontiers and challenges will emerge, and computer technology will evolve substantially, in the 2007-2012 time frame used as the planning window for this report. The community should begin planning for follow-on capability well before the end of the useful life of the PCG.

Recommendation 15: By the time of the second procurement, undertake a study to explore avenues for continuing to provide petascale and higher computational capability to the

geosciences. In the study, include an assessment of lessons learned and planning for the transition to a new management paradigm.

FEASIBILITY

A review of the numerous geoscience applications across a variety of sub-disciplines shows that many geoscience applications both need and can effectively exploit a petascale computational facility. It is also clear that the required system characteristics (such as bandwidth and latency) emerging from the applications analyses performed in this study are technically realizable during the time frame of the PCG project (2007-2012). A review of architectural trends and future projections indicate that, by FY2007, a variety of capability systems with 200 TFLOPS peak will be priced between \$50M and \$150M. In 2010, one PFLOPS peak is estimated to cost \$50M to \$150M. The overall cost model for the project shows that, in the most probable scenario, about 40% of the total budget will go to supporting facility operations and personnel, and providing adequate networking, data archiving, data analysis, and visualization resources for the project.

Thus, it appears to be technically feasible to construct a highly capable petascale computing collaboratory that can deliver 200 TFLOPS aggregate peak in 2007 and achieve 1 PFLOPS peak by 2010, at a six-year total project cost of approximately \$390M.

These costs are beyond what the NSF Directorate for Geosciences alone can accommodate, but are within the scope of project costs supported by NSF's MREFC account. PCG costs are reasonable when considered in the context of a balanced national investment in the geosciences research infrastructure including global observing systems. The National Academy of Sciences Committee on the Future of Supercomputing estimated that a total procurement of high-end computing resources of \$800M per year is required to support the national research agenda and to assure a viable U.S. research and development effort to maintain leadership in supercomputing technology (Graham et al., 2004; President's Information Technology Advisory Committee, 2005). Establishing the PCG would amount to approximately 10% of that recommended total investment.

REPORT METHODOLOGY

The Technical Working Group was charged with making a technical and budgetary prospectus for the PCG (Appendix 2). The methodology that the working group followed in responding to this charge was to pursue three interrelated lines of inquiry. One team examined the architectural and technological trends that will determine what system capabilities will be available from 2007-2012. A second team looked at a suite of nine geoscience applications and attempted to develop performance models for some of them that could inform requirements for balanced system design in terms of memory requirements, interconnect performance, and mass storage. A third team examined aspects of facility design and developed a cost model for it based on the inputs of the other two groups. The detailed discussion of the results of these lines of inquiry is to be found in the subsections that follow, which mirror the working groups' three-fold methodology.

To make a reasonable assessment of the cost of the PCG, it was necessary to gather as much detailed information about the future geoscience applications that would use it. As discussed in the Application Analysis section (see p. 14), this was difficult for several reasons: the applications are complex, being composed of many components; they are expected to change in significant ways by the end of the decade; and few have had detailed performance models constructed for them. Nevertheless, a detailed questionnaire was sent to geoscience researchers to obtain information about the computational, memory, and data-storage requirements of their applications (Appendix 3). This information was used in many cases to build simple performance models and to assess the configuration requirements for the petascale supercomputing environment based on current model frameworks and on characteristics of current high-performance computers.

The responses to the questionnaire came from three main areas of geoscience: physical oceanography, atmospheric science, and solid Earth science. In each case, a grand-challenge problem associated with a particular application was identified. In most cases, sufficient information was obtained to completely specify the problem size, scientific throughput desired, computational complexity, memory footprint, and dataset sizes and sampling frequencies. The results of this survey are discussed further in the Application Analysis section.

ARCHITECTURAL TRENDS

Modern supercomputers are composed of a hierarchy of sub-systems. Working outward from the central processing unit, there are registers, perhaps multiple levels of cache, local memory, system interconnect to other processors and memory, high-performance attached disk systems, and ultimately robotic tape systems for mass storage. Latencies to each of these components currently range across an incredible eleven orders of magnitude, from nanoseconds in the case of a CPU register to perhaps minutes for a robotic tape access. Likewise, the bandwidths taper as one works outward. Cache bandwidths generally are at least four times larger than local memory bandwidth, and this in turn exceeds interconnect bandwidth by a comparable amount, and so on outward to the disk and archive system.

The central issue of computer architectures lies in the following two observations. First, while the absolute speed of all computer subcomponents have been changing rapidly, they have not all been changing at the same rate. For example, the analysis in *Getting up to Speed: The Future of Supercomputing* (Committee on the Future of Supercomputing, 2004) indicates that, while peak processor speeds have been increasing at 59% between 1988 and 2004, memory speeds of commodity processors have been increasing at only 23% per year since 1995, and DRAM latency has only been improving at a pitiful rate of 5.5% per year. The fact that memory latency has not been keeping up with CPU speeds has produced a crisis of sorts for computer architects, who have developed a variety of complex latency-hiding mechanisms to compensate. As a result, with parts of the memory subsystem logically virtualized, but physically hierarchical and dramatically disparate in their performance characteristics, penalties for misuse of the memory subsystem are often both draconian and opaque to the applications programmer. The performance of two apparently sensibly written versions of an application can vary dramatically based on the number, frequency, and remoteness of data-access requests in each.

Looking towards the future, tolerance for remote (bad) memory access patterns will likely get worse, and computers will continue to become more unforgiving, unless something is done to drastically reorganize computer architectures and provide more intuitive programming models. Fortunately, a response to this challenge is emerging: boundaries among architectural paradigms are blurring in new designs, radical new computer architectures are on the horizon, and some government efforts to promote them are in place.

In the next subsection, the origin of this problem, driven by Moore's Law, is discussed in more detail. Some of the emerging architectural alternatives designed to tackle them are also touched upon there. A discussion of parallel computing challenges leads into the technology forecast in which a general outline of future large-scale computers is presented. This is followed by a discussion of best practices for benchmarking and procurement of large systems.

MEMORY LATENCY AND BANDWIDTH

Scientific calculations involve operations upon large amounts of data. Trouble begins when moving data around within the computer. As has been noted, the rates of improvement in memory bandwidth have not in fact been keeping up with Moore's Law for some time (Committee on the Future of Supercomputing, 2004). With this trend in place, eventually the time required to complete a calculation becomes dominated by memory load store times, where the rate of improvement tracks memory (not CPU speeds) and the fraction of peak sustained steadily declines.

As computer speeds have increased they have been able to tackle larger problems, requiring, in turn, larger amounts of memory and storage. Because the latency to memory is not decreasing as quickly as processor speed is increasing, the relative distance of a processor from its memory, as measured in CPU cycles, essentially doubles when the processor's speed

doubles. For example, the latency to main memory of a 1980s vintage Cray system was one cycle, while the latency to RAM of a modern microprocessor is now typically measured in hundreds of cycles.

Because memory bandwidth is also increasing faster than is memory latency, there is a slower, but important, overall trend towards an increase in the number of words of memory bandwidth that equate to memory latency. For a processor to avoid stalling and to realize peak bandwidths under these conditions, it must be designed to operate on multiple outstanding memory requests. The number of outstanding requests that need to be processed to saturate memory bandwidth is now approaching 100 to 1000 (Committee on the Future of Supercomputing, 2004).

It is clear that latency tolerance mechanisms are an essential component of future system design.

Designers of commodity microprocessors have attempted to mitigate the memory latency problem via two mechanisms: creating small amounts of high-speed, lower-latency memory near the CPU, called cache; and increasing the number of outstanding memory requests in flight that the processor can handle. Caching works well when calculations reuse values or, because caches bring in chunks of neighboring data when one is accessed, when neighboring values are used in subsequent calculations. However, cache does not work when memory access patterns fail, are irregular, or fail to reuse values, that is, when the memory access patterns fail to have good spatial and temporal locality. Applications exhibiting high spatial and/or temporal locality tend to run at a high percentage of theoretical peak speed because they can use cache effectively.

Another approach to latency hiding is vector processing. In a vector processor each load instruction is executed on a $O(100)$ long vector of values: a deeply pipe-lined processor

architecture allows data parallelism to hide latency in a vector system while issuing a small number of instructions. Vector processors in the high-performance computing (HPC) market are typically supplied with ample memory bandwidth, which in part contributes to their overall higher cost in terms of peak performance. For example, the Earth Simulator's SX-6 processor has a peak memory bandwidth of 0.5 words per peak FLOPS, approximately four to five times the ratio for an equivalent commodity microprocessor. For these reasons, vector processing is a proven custom technology in the scientific arena. It is also showing signs of revival domestically, a trend driven largely by national security and scientific leadership concerns in the government sector and the burgeoning video entertainment markets in the commercial space.

Multi-threaded processors provide a third approach to latency hiding. Such architectures typically provide hardware support for very fast context switching between outstanding threads of execution. This allows the processor to quickly move on to another thread of execution when a current thread stalls waiting for a memory request to complete. For such a system to keep busy, the system must support hundreds of concurrent threads of execution. Tera's MTA supercomputer was an example of this architecture, although hints of multithreading architecture are beginning to show up in some new microprocessor designs.

PARALLEL COMPUTING

Because many scientific applications require computing power beyond that which can be mustered on a single chip, the HPC community has turned to using large arrays of processors, working in concert, to solve the largest computational problems. Ideally, such scientific calculations can be decomposed into a set of independent work steps, with the work evenly divided among processors. In this "embarrassingly" parallel case, speed-up is equal to the number of independent steps. Embarrassingly parallel calculations do arise in

geoscience applications, for example, in ensemble forecasting or in parameter space studies. However, such tasks don't really require a single, large parallel computer, which is the object of this discussion. Instead, most important grand challenge calculations require frequent interprocessor communications, have workload balancing issues, and must do I/O to get the job done. Any one of these activities can impact efficiency by effectively serializing the application.

This is of particular concern the PCG petascale computer, which will certainly contain at least 10,000 processors at the end of the decade. Indeed, as processor count goes up, an increasingly stringent restriction is placed on parallel application developers by Amdahl's Law, which states

$$\text{Speed-up}(P) = \text{Time}(1)/\text{Time}(P) \leq 1/(s + (1-s)/P) < 1/s,$$

where P is the number of processors and s is the fraction of work done sequentially. For example, on a system with 10,000 processors, if $s = 0.0001$ for an application (i.e., if only 0.01% of the work cannot be parallelized), the speed-up will be cut in half to roughly 5,000. Anything that causes processors to briefly go idle contributes to the value of s , including network latency, contention, or load imbalances.

An application's communication pattern stresses the system's global interconnect in ways strongly analogous to the manner that local memory access patterns stress the processor's memory subsystem. Consider three different techniques for solving an elliptic problem, a common situation in geofluids applications. A Jacobi relaxation method involving finite difference approximations to the derivatives requires only localized network bandwidth and is not very demanding on the system interconnect. Unfortunately, it also may not be the fastest in terms of time to solution. A spectral solution technique transforms the problem into an appropriate wave number space in which the elliptic problem can be solved locally. However, the spectral method will require global all-to-all communications, which are highly non-local and will saturate the bisection bandwidth of the network. An iterative solver can be used to invert the problem to a specified degree of accuracy, and requires global sums to compute inner products. To perform well, these global sums will require very low network latencies and specialized hardware that can directly access local RAM memory to perform well on large systems.

An important observation about the relative cost and importance of the interconnect comes from *Getting up to Speed: The Future of Supercomputing* (Committee on the Future of Supercomputing, 2004). There, the authors point out that the cost of providing global bandwidth will gradually grow to dominate the costs of providing local bandwidth and processing power. This fact is made self-evident by counting cabinets devoted to interconnect hardware in large, highly capable parallel computers. In the largest configurations, interconnect cabinetry can account for up to half the machine.

TECHNOLOGY FORECAST

The Next Five Years: 2005-2010

There is general industry agreement that the overall trends and issues in HPC described above will likely continue over the next five years. Design plans are already in place and largely solidified for the next two generations of supercomputer systems. Thus, peak performance of these systems can be projected with a reasonable degree of confidence, although vendor proprietary concerns prevent a great deal of specificity about these plans here.

It is likely that Moore's Law will continue through 2010; however, the gains will not come entirely from increased clock speeds, but from chip multiprocessors (CMPs). As sources of processor speed-up, such as pipeline depth and instruction level parallelism, have become mined out, concerns about power density have begun to slow increases in clock frequency. In response, chipmakers have begun to turn to CMPs to maintain Moore's Law. First two, and then four, cores per chip are scheduled to appear in the next few years. Through this mechanism it is expected that Moore's Law will be maintained and that peak speed will increase by a factor of eight in the next five years. Explicitly, this means that by 2010, one can expect multiprocessor chips with speeds of 30-50 GFLOPS and vector chips with peak speeds ranging from 60-120 GFLOPS to be available. More speculatively, there are also developments in computer node architecture that will employ co-processors, such as FPGAs (field-programmable gate array) and GPUs (graphics processing unit), to speed up memory access and other operations. A serious study would be needed in order to know whether this is relevant to the PCG application portfolio.

There is every indication that current improvements in price performance are likely to continue through the end of the decade as well. Indeed, all vendors contacted for this report predict that a PFLOPS peak system will be available in the 2010 time frame for roughly \$100M. Power consumption is expected to become an increasingly important technical and cost-of-ownership issue. The competition between conventional clusters and systems composed of tightly integrated, low-power processors, such as IBM's Blue Gene/L, bears watching.

Projecting the overall efficiency of these systems is much more difficult. It is encouraging to note that recent generations of IBM POWER, Itanium, and AMD Opteron processors have managed to hold the line at 10% efficiency for many geofluids codes. For example, the Weather Research Forecast (WRF) model achieves this rate on a number of parallel systems. Throughout the next five years, the traditional paradigms of scalar and vector processors are expected to persist, but multithreading, and even more exotic features, will begin to blur the traditional definitions of the processor by the end of the decade. There are also many ways to organize the CMP architecture, and the winners and losers are unclear at this time. The net impact of CMPs on application performance is difficult to assess as well, but it is likely that memory bandwidth will not increase at nearly the rate of CMP floating-point performance.

2010 and Beyond: High Productivity Computing Systems

A series of recent reports, including the *Federal Plan for High-End Computing* (National Science and Technology Council, Committee on Technology, 2004), the *Science-Based Case for Large-Scale Simulation: Volume I* (Colella et al., 2003), and *Getting up to Speed: The Future of Supercomputing* (Committee on the Future of Supercomputing, 2004) have pointed out the gap between the apparent promise of Moore's Law and the actual difficulties in achieving the required time-to-solution (TTS) on real applications. It is now widely recognized that:

1. Peak theoretical performance as measured by Top500 (<http://www.top500.org>) is not a useful metric for HPC.
2. TTS on real science problems is a more viable metric.
3. TTS includes time to program, which can be extensive on

systems with new architectures.

4. The data decomposition problem and Amdahl's Law makes parallel programming difficult.
5. Machines with better balance between arithmetic and memory operations would be more efficient and also easier to program and thus would partially address points 2-4.
6. The definition of "balance" depends on the application (arguably today's machines are balanced for high-performance LINPACK [HPL].)
7. Future machines should be designed and procured with the requirements of the target applications in mind.

To address the issues of low efficiency, scalability, and the lack of software tools and environments, DARPA recently created the HPCS program in cooperation with several other agencies, including DOE and NSF. HPCS is a research and development program focused on creating new generations of high-end supercomputing architectures, programming environments, and software tools to realize a new vision of high productivity, HPC systems. HPCS goals include:

- Fill the high-end computing gap between today's late 1980s-based technology and the promise of quantum computing.
- Provide economically viable, high-productivity computing systems for the national-security and industrial user communities with the following design attributes in the latter part of this decade:
 - Performance: Improve the computational efficiency and performance of critical national security applications.
 - Programmability: Reduce cost and time of developing HPCS application solutions.
 - Portability: Insulate research and operational HPCS application software from system specifics.
 - Robustness: Deliver improved reliability to HPCS users and reduce risk of malicious activities.

Currently, three vendors—Cray Inc., IBM, and Sun Microsystems—are designing high-productivity supercomputer systems with significantly better balance and programming environments that are easier to use than those available on today's machines. All three designs leverage significant technological advances to decrease memory latencies, tolerate memory latencies (by supporting more parallelism), and assist the programmer with data decomposition and finding

parallelism via better hardware support and more intuitive programming languages. Although details of each design are proprietary, it can be said that the technologies being explored in these futuristic architectures include:

1. Multi-threaded architectures (i.e., having many threads of execution available per processor to mask memory latencies, as described earlier).
2. Processor In Memory (PIM) systems that integrate processors near or even on the memory chips themselves, thus allowing data to be operated upon locally, in memory.
3. Use of co-processors such as FPGAs and GPUs that can dramatically speed up certain types of operations and calculations
4. Logically flat and physically hierarchical memory with hardware assisted data movement.

These novel system architectures bear watching, as the first production versions are scheduled to become available in 2010. FY 2010 is in the middle of the PCG project, and so these exotic systems may come into play.

BEST PRACTICES FOR SYSTEM DESIGN AND PROCUREMENTS

The primary goal of supercomputing systems is to enable rapid execution of scientific simulations. Thus, one must design, configure, and procure systems that are well matched to application resource demands. Historically, the factors that most determine achieved application performance have not been well understood. Consequently, many systems have been configured or purchased based on simplistic metrics such as processor peak speed or expected HPL performance that had little or no direct bearing on TTS.

Today, best configuration and procurement practices rely on a set of strategic applications (National Science and Technology Council, Committee on Technology, 2004) and an associated set of input data to define a set of benchmarks. These applications and data are chosen to cover the space of expected application behaviors and resource demands. Ideally, one seeks to obtain performance data for the benchmarks on each of the potential platforms. Unfortunately, vendors may not have the time, or even the available systems, to run all the benchmarks, especially on very large systems.

Accurate performance models can reduce the time and cost of full benchmarking by highlighting the causal performance factors in the benchmark suite. For example, the Department of Defense High Performance Computing Modernization Program (DOD HPCMO) Benchmark Team defines their benchmarks and also uses models to predict the performance of the benchmarks on target platforms (Davis et al., 2004).

Figure 1 illustrates how a performance model of an application, such as the POP (Parallel Ocean Program), can be used to understand the sensitivity of its performance to changes in system attributes (Snaveley et al., 2004; Bailey and Snaveley, in press). Indicated is the performance impact on the 128-CPU POP x1 program of quadrupling the speed of the CPU-memory subsystem (lumped together we call this the processor), quadrupling the network bandwidth, reducing network latency by four, and various combinations of these four-fold hardware improvements. The data values are plotted in a logarithmic scale and normalized to one, so that the solid black quadrilateral represents the execution time, network bandwidth, network latency, CPU and memory subsystem speed of the baseline machine, a POWER3 system. At this size, POP x1 is quite sensitive to processor speed (a faster CPU and memory subsystem), somewhat sensitive to latency (because of the barotropic portion of the code is communications-bound, with small-messages), and fairly insensitive to bandwidth. In a similar way, we can “zoom in” on the processor performance factor to understand the impact of, for example, faster floating-point units, and faster or bigger caches. In the model, the processor axis shows modeled execution time decreasing from a four-times faster CPU with respect to clock rate (implying a 4X floating-point issue rate), but also quadruple bandwidth and one-quarter latency to all levels of the memory hierarchy (unfortunately this may be hard or expensive to achieve architecturally).

At this problem size, POP performance would be improved more by faster processors and memory subsystem rather than by a faster or higher bandwidth inter-processor network. Information like this can be used to choose the best machine for an application or set of applications.

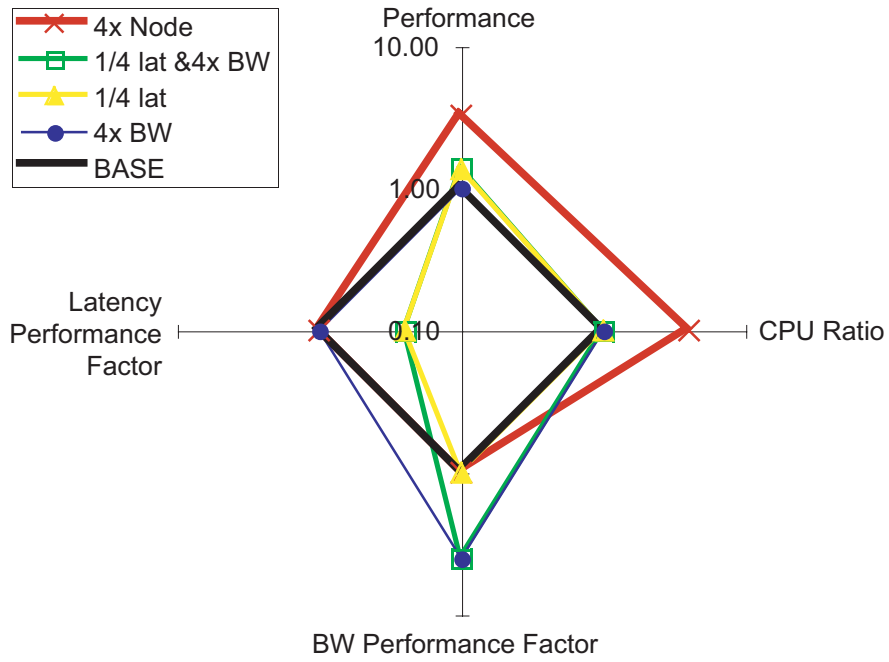


Figure 1. Sensitivity of POP performance (top vertical axis) to four-fold improvements to Node performance, interconnect latency, bandwidth, and both latency and bandwidth. The plot is logarithmic and normalized to system characteristics of an IBM POWER3 cluster (BASE). Effects shown are for one degree resolution POP on 128 processors, and are rather more sensitive to node performance improvements.

APPLICATION ANALYSIS

In this section, we discuss the overarching question of whether a petascale computer can deliver new geoscience insights and discoveries. This question breaks down into two related ones about geoscience applications and the petascale computers that will run them: (1) which, if any, scientifically interesting applications truly require simulation capabilities that can be called “petascale” and (2) what computing system characteristics will be required, and available, to deliver that performance in the time frame between 2007 and 2010?

Applications are a collection of algorithmic components that work together to solve a scientific problem. Each algorithm has a unique signature in terms of computational complexity, memory and network access patterns, and I/O requirements that collectively determine the performance of that algorithm on a particular computing platform. The performance of each component in turn determines the application’s overall performance.

Ideally, the issue of what system will run a suite of applications best is settled by benchmarking real applications on real equipment. Unfortunately, the task here is more difficult: it is not possible to benchmark systems that have not yet been constructed. Of course, one can rely on previous benchmark experience. This is certainly a worthwhile source of data to inform our opinions, but this will only provide an understanding of what past technology has provided to earlier versions of applications. At the same time, one must allow for the development and deployment of new algorithms in important applications or, at the very least, changes in the relative importance of existing algorithmic components in future versions of geoscience applications that attempt to capture new phenomena.

In the absence of real benchmarking, one can construct performance models of applications that take information about the algorithms in the applications and combine them with parametric information about anticipated characteristics of future computers to understand what attributes of the computer system are most critical. Creating such performance

models requires a great deal of detailed information about these algorithms if one is to construct even a simple qualitative model of the scaling and performance characteristics of the overall application.

In practice, creating performance models is a very difficult thing to do, as a single, modern geoscience application may have dozens, if not hundreds of individual algorithmic components. This is increasingly true as many geoscience applications add complexity in an attempt to capture previously neglected phenomena. Nevertheless, this pessimistic view of efforts to model applications is counterbalanced by several factors. First, a certain degree of universality exists in the underlying numerical methods employed in many scientific applications. Second, unlike computing platforms, scientific models generally change quite slowly (i.e., on a time scale of about 5-10 years). Third, we are only after a qualitative understanding of what the words “capability system” mean in terms of geoscience applications and whether systems will exist that will be able to run them with some degree of efficiency in the time frame of interest.

For this reason, a detailed questionnaire (Appendix A) was sent out to geoscience researchers to obtain information about the computational, memory, and data storage requirements of these applications. This information was used to build simple performance models and to assess the configuration requirements for the petascale supercomputing environment.

The responses to the questionnaire came from four main areas of geoscience: physical oceanography, atmospheric science, space science, and Earth science. In each case, a grand challenge problem associated with a particular application was studied. In most cases, sufficient information was obtained to completely specify the problem size, scientific throughput desired, computational complexity, memory footprint, data-set sizes, and sampling frequencies. Disk-bandwidth requirements were determined from this information by requiring that synchronous output of history files would result in no more than a 5% overhead relative to computa-

tional costs. A possible source of uncertainty in disk-bandwidth requirements relates to the need to write checkpoint/restart files. Currently, restart files represent a small percentage of the I/O traffic for most scientific simulations. However for very large systems, the mean time between failures (MTBF) will go down as the number of nodes per system increases—even if technology evolution does not decrease the MTBF of individual nodes. Thus, in the future, more frequent checkpoints may be needed in petascale systems, and the importance of restart files in the I/O mix may increase.

A summary of the requirements derived from an analysis of the twelve questionnaire responses can be found in Table 2. The table indicates that 11 applications require a range of sustained performance from 2-150 TFLOPS, with a “geoscience average” requirement of 38.6 TFLOPS sustained. The maximum system memory required by an application in the survey is 20 TB, with an average requirement of 3.8 TB. The aggregate mass storage archive rate for the nine applications for which this could be determined is in the range of 12 to 66 PB/year. Although the variance is large, this range of values

Table 2. Application computational, memory, data storage, and disk bandwidth requirements for various geoscience applications.

Application Name/ Discipline	Problem	Max Required Sustained TFLOPS	System Memory (TBytes)	Mass Storage Archive Rate (PBytes/ year)	Disk Bandwidth for 5% overhead (GBytes/sec)
flow_solve/ oceanography	3-D turbulence	2.5	6.5	0.14	1.1
POP/ oceanography	10 km global mesoscale eddy	6	0.15	0.32-3.2	0.2-2.0
POP/ oceanography	5 km global mesoscale	120	1.5	3.2-32	2-20
MITgcm/ocean data assimilation	15 km global ocean	7.3	0.82	0.66	0.4
WRF/meteorology	10 m tornado simulation	150	20	2-24	25-300
	5 years of 3 km global nonhydro- static simulation	66	1.75	1.0	8
CAM/climate modeling	Five instances of T341L52	13	0.5	4.6	1.1
CRCP/climate modeling	2 km global sub-grid scale model	22	-	-	-
ABINIT/minerology	DFT calculation	1.6	-	-	-
inverse problem/regional seismology	100M point inverse problem	17	0.01	0.12	0.07
forward problem/global seismology	36.6 billion degrees of freedom	10.4	7.3	0.01	0.00002
LADHS/regional hydrology	100 m Columbia river basin	10	0.3	20.8	0.66

agrees well with the value obtained if one uses the ratio observed for NCAR's mass storage system of 30 bytes archived per million floating-point operation performed. Using that rule of thumb, 36.5 PB/year would be expected to be produced from an average 38.6 TFLOPS sustained.

It is worth noting that these requirements represent current thinking about the next round of experimentation in grand challenge problems. Almost certainly, as applications that make use of this realm of computing power are explored, more complex and demanding questions will be posed and requirements will increase. This is the nature of computational science.

In a few cases, the information provided was detailed enough to create a performance model that could answer questions about interconnect performance characteristics. This analysis resulted in certain constraints on local and bisection bandwidths and system latency. It appears that it is both required and feasible that a ratio of per-processor, sustained bisection bandwidth of the interconnect to sustained FLOPS of approximately 0.25 B/s/FLOPS is sufficient for the most demanding applications. An interconnect latency below 6 μ sec in 2007 and 1.5-2 μ sec in 2010 seems both adequate for the applications and feasible given technology trends. Finally, the need for scalable cooperative communication operations, such as global reductions or barriers, seems indicated by the use of iterative solvers in many applications.

Finally, one area of concern that was not well characterized by this study was the application requirement for local memory latency and bandwidth. Doing so for each application domain would have required more detailed profiling than was possible given the time allowed for this study.

OCEAN SCIENCE APPLICATIONS

Seawater Turbulence Simulation: “flow_solve”

An application called “flow_solve” (Winters et al., 2003) is representative of a large class of three-dimensional direct numerical turbulence simulation (DNS) models based on the Fourier pseudospectral discretization method. The specific

scientific objective of “flow_solve” is to understand the intricate mixing properties of heat and salt in seawater that result from the very different chemical properties of those two scalar quantities. The application simulates rotating, stratified flow. It assumes periodic boundary conditions with an option for free-slip boundaries in one of the three dimensions. Discretization is Fourier pseudospectral in space, third order Adams-Bashforth in time. Salinity is resolved on a fine grid whose spacing is one-half that of the other fields, thus efficiently resolving the small-scale structure that results from low saline diffusivity. This optimizes the code for use in simulating turbulence in seawater.

To simulate fully developed turbulence in seawater requires “flow_solve” to be run on a 4096-cubed salinity grid, which involves simulating 400,000 pseudospectral time steps. Each time step consists of performing a three-dimensional sine/cosine Fast Fourier Transforms (FFTs) on a single, 4096-cubed salinity field as well as on the four, 2048-cubed fields representing 3-D velocity and temperature. The 3-D FFT calculation consists of two array transpositions (involving global all-to-all communications) and three on-processor discrete FFT operations. The researchers desire to complete one such run in a month using the petascale facility and to output the salinity, velocity, and temperature field for analysis every 2400 time steps.

Performance Model

A simple performance model of the transposition-based 3-D discrete FFT algorithm (Figure 2) can be created using the information provided and it can be used to estimate the computational complexity, the necessary sustained performance rate, and the size of the data set created. The performance model indicates that the 4096-cubed “flow_solve” simulation requires about 2.5 TFLOPS sustained to produce one 400,000 time step simulation per month. Assuming a 4096 processor, circa 2007 computing system with a network latency of 6 μ sec and capable of about 40 TFLOPS peak, one finds that an overall sustained efficiency of about 7% will be sufficient to achieve the performance goal. Optimized FFTs on current single processor systems routinely achieve 20% of peak, so the performance requirement translates into a scaling efficiency of 35% on 4096 processors.

"flow solve" Scaling Efficiency vs.
Petascale System Interconnect Strength

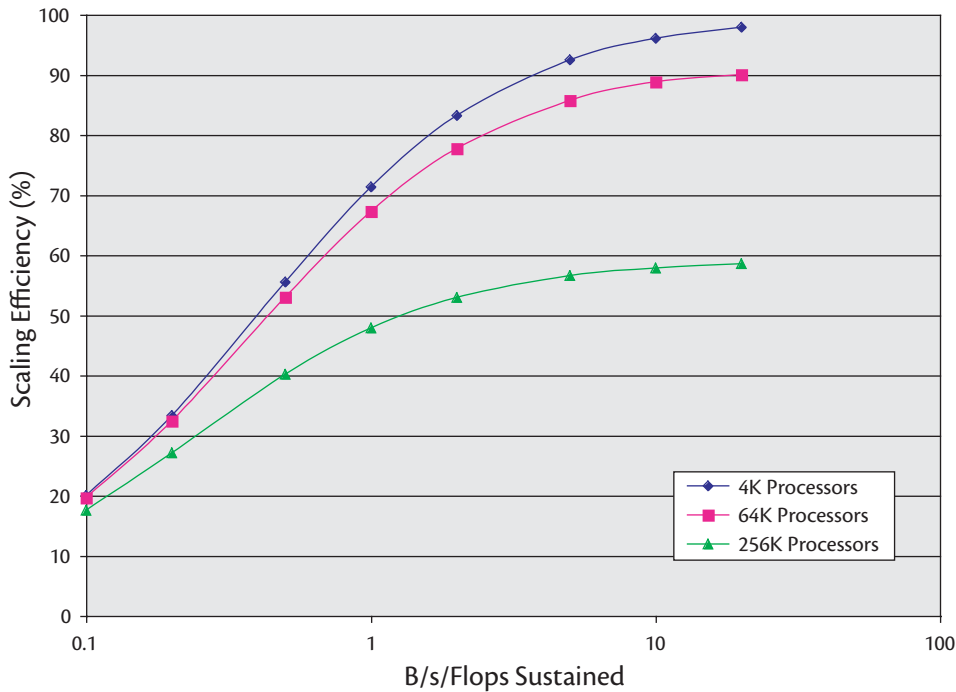


Figure 2. A performance model of a 2048-cubed discrete 3-D FFT in "flow_solve" illustrates the sensitivity of scaling efficiency (%) to the ratio of sustained per-processor bisection bandwidth to FLOPS for various size computing systems. Given a sustained FFT performance of 1.8 GFLOPS and a 6 μ sec network latency, the deficiency of 6 μ sec latency for very large (256K processor) systems is clear.

The system scaling efficiency can be modeled for different size systems as a function of the ratio of sustained per processor bisection bandwidth, expressed in units of bytes/sec/proc to sustained FLOPS. The results indicate that the "flow_solve" simulations require a minimum ratio of 0.25 bytes/sec/proc of sustained bisection bandwidth per sustained FLOPS to achieve a scaling efficiency of 35% for "flow_solve" for the 4096 processor case. Only when processor counts reach 256K does the 6 μ sec latency of the network become significant in determining the asymptotic scaling efficiency of "flow_solve."

Finally, the output data-set size of a single 4096-cubed "flow_solve" simulation is estimated to be approximately 70 TB.

Ocean Mesoscale Eddy Simulation: POP

POP is a widely used global ocean model primarily used for climate simulation. It was developed principally at Los Alamos National Laboratory (LANL) (for more information go to <http://climate.lanl.gov/Models/POP/>). The POP model solves the three-dimensional hydrostatic primitive equa-

tions for fluid motions on the sphere. POP uses finite-difference discretizations in orthogonal curvilinear coordinates to compute derivatives. The logically rectangular meshes used by POP are distorted polar grids called "dipole" or "tripole" grids. These grids shift the pole points of the polar grid to a point located over a convenient land mass. Time integration in POP is split into fast (barotropic) and slow (baroclinic) parts. The slow baroclinic modes are integrated explicitly using a leapfrog scheme. The very fast barotropic modes are currently integrated implicitly using a preconditioned conjugate gradient solver to iteratively solve an elliptic problem for the surface pressure.

Figure 3 shows performance data collected for a variety of current computing systems for POP running at one-degree (100 km) resolution (Worley and Dunigan, 2003). As can be seen, many microprocessor systems are severely limited in scalability. When one digs deeper, one finds that it is primarily the barotropic solver that limits scalability, and ultimately, the integration rate of the POP model (ESPOP, 2002).

Parallel Application Performance LANL Parallel Ocean Program, x1 benchmark

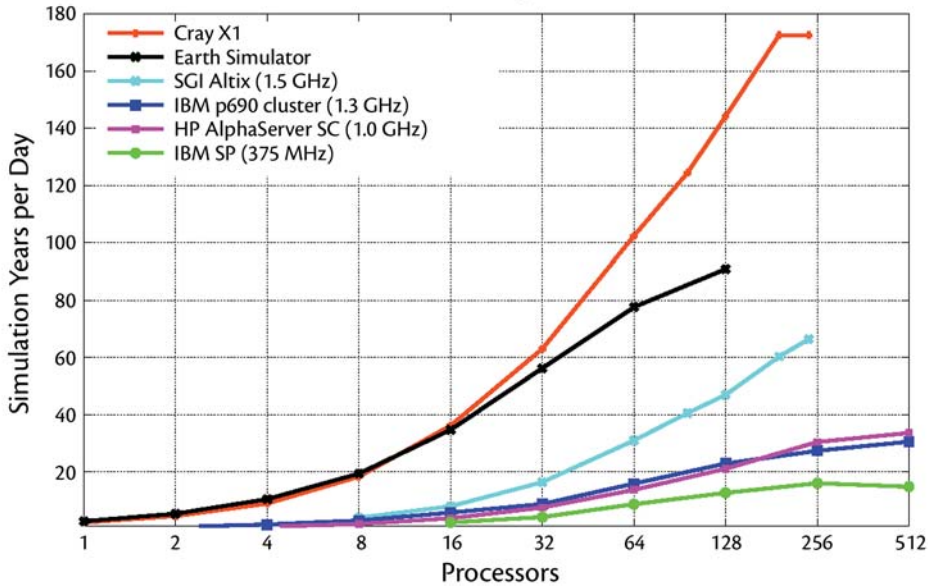


Figure 3. Scalability of the POP application at 1° resolution on various contemporary computing platforms (from Worley and Dunigan, 2003). Notice the wide variability in the scalability of different systems. This is due, in large part, to the scalability of the global sum operations in POP's barotropic conjugate gradient solver.

Because computational cost scales as the cube of horizontal resolution, the ten-fold increase in horizontal resolution desired by oceanographers to permit mesoscale eddies in climate models requires at least a thousand-fold increase in sustained application performance to hold the integration rate constant. Thus, achieving an integration rate of 5 years per day at 1/10° resolution requires a substantial portion of a terascale computer, such as the Earth Simulator. For example, a 1/10° (10 km) POP model has run in excess of 6 years/day on the Earth Simulator in Japan using 1600 vector processors. It is projected that several research groups will want to run such high-resolution global ocean simulations simultaneously on the PCG's computers.

Performance Model: POP 10 km Implicit Barotropic Solver

To make significant scientific progress on the ocean mesoscale eddy problem, researchers desire to routinely integrate ocean general circulation models (OGCMs) at 10 km (1/10°). An integration rate of approximately 40 years of simulation per wall clock month is considered acceptable for these experiments. Ultimately, researchers would like to press on to 5 km OGCMs to reach the eddy-resolving threshold.

In the current version of the POP ocean model in CCSM3 (Community Climate System Model-3), the barotropic fast modes in the ocean are solved using an iterative conjugate gradient solver performing about 200 iterations per baroclinic time step. For the 10 km (1/10°) case there are 250 baroclinic time steps/day. At 10-km resolution, the CG (conjugate gradient) solver operates on a 3200 x 3840 2-D grid. Because the CG solver is generally network-latency limited on parallel computers, the FLOPS required per CG iteration can be safely neglected. The dominant, latency-sensitive piece is the one MPI_ALL_REDUCE per iteration of a three vector that is required to compute the norms in the CG algorithm. To attain the stated goal of performing 40 years/wall clock month of simulation (~500 simulated days/ wall clock day), the implicit solver for the barotropic mode must perform about 300 CG iterations per second, or about 3.46 msec/CG iteration (or 3-vector reduction).

Although the other (baroclinic) calculations in the POP model are finite difference and scale well, realistically one should still expect the baroclinic portion of the model to take some time to execute: the system will have to execute the MPI_ALL_REDUCE summations far faster than 3460 µsec.

This appears straightforward on paper. The dominant latency portion of the time to calculate a reduce-sum operation on P processors should be given by $2 \cdot t \cdot \log(P)$, where t is the point-to-point latency of the interconnect. Taking $P=4096$ and t to be a nominal 6 μsec , the time to do the reduce-sum should be about 144 μsec . In practice, however, global reduction operations of this kind require special-purpose hardware and software to achieve anything like idealized logarithmic scaling in P . Clusters composed of multiprocessor nodes with full, heavyweight kernels and network cards installed on I/O buses are susceptible to various kernel interrupt. If processors spend a fraction off of their time in system activities, and these activities are spread randomly, then the odds that some processor is doing system time when the computation reaches a barrier are $1 - (1-f)^P \sim 1 - (1/e)^{(fP)}$ (i.e., when the number of processors gets closer to $1/f$ than the chance of completing a barrier with no system interference goes rapidly to zero). For this reason, such systems have reduce-sum scaling characteristics that are much worse than the idealized predictions.

In summary, the network latency required to perform a satisfactory reduction operation for POP is readily available from commercial interconnect technology. What is needed to realize this capability is customized hardware and software capable of achieving logarithmic scaling in P . Because many other geoscience applications employ iterative solvers, this is a critical requirement for “capability” supercomputing systems.

POP Explicit, Sub-Cycled Barotropic Model Component

Future implementations of the POP barotropic component may contain a sub-cycled explicit barotropic model, perhaps similar to that found in the Geophysical Fluid Dynamics Laboratory’s Modular Ocean Model (GFDL MOM). In this case, the fast modes are integrated explicitly with a very short time step, typically 50 barotropic steps per baroclinic time step. Using the time stepping information from the 10 km POP model, this means that a 40-year integration of 10 km POP requires calling the explicit barotropic time stepping routine $50 \times 250 \times 365 \times 40 = 182.5$ million times.

The details of the sub-cycled barotropic component of the MOM ocean model can provide input for a performance model of the explicit barotropic. In the explicit, subcycled barotropic model, nearest-neighbor communications of three

variables occur in a 2-D slab. Computations for the explicit model amounts to 60 FLOPS/site/barotropic step. Using these data, the performance of a computer for the explicit barotropic can be modeled. The 10 km POP grid (3200 x 3840) is assumed. Figure 4 shows the trade-offs between latency and the ratio of sustained bandwidth to sustained FLOPS in the explicit barotropic. For a 2007 system with a size of 4096 processors and a latency of less than 6 μsec and a ratio greater than 0.25 B/s/FLOPS sustained, the model indicates scaling efficiency in excess of 75%. For faster systems around the year 2010, a latency of 2 μsec will likely be required.

POP Baroclinic Model Component

To model the baroclinic component of POP, one needs to know the number of baroclinic FLOPS per site, the number of layers, the number of boundary exchanges, and the number of fields exchanged for each communication. The baroclinic communications are performed in a non-blocking north-south, east-west pattern. Two 3-D tracers are exchanged twice this way, one layer at a time, and two 2-D forcing terms are exchanged together once during a baroclinic time step. Using POP 1.4.3 as a guide, hardware performance counters indicate that 1-degree POP requires 2335 FLOPS/site/time step. It is then simple to create a baroclinic performance model. One finds that, because of the large number of FLOPS per site per time step, the baroclinic is much more dominated by computation than the barotropic.

POP Memory and Data Set Sizes

POP stores three prognostic 3-D fields (horizontal velocity and density) and several tracers; the number of tracers is taken for this estimate to be five. Each field is stored at three time levels, and there are approximately 30 additional 3-D work arrays. Thus, an estimate of the memory footprint for POP is given by computing the size of 54 3-D fields: for a 40-level POP at 10 km resolution, this works out to be 150 GB; for a 100-level POP model at 5 km resolution this is 1.5 TB.

As for as data-storage requirements, a simulation typically would save a file with 20 3-D fields every 3-30 simulated days, depending on the experiment. This translates into a data archive requirement range of 0.32 PB/wall clock year for monthly 10 km POP simulations to 32 PB/wall clock year for

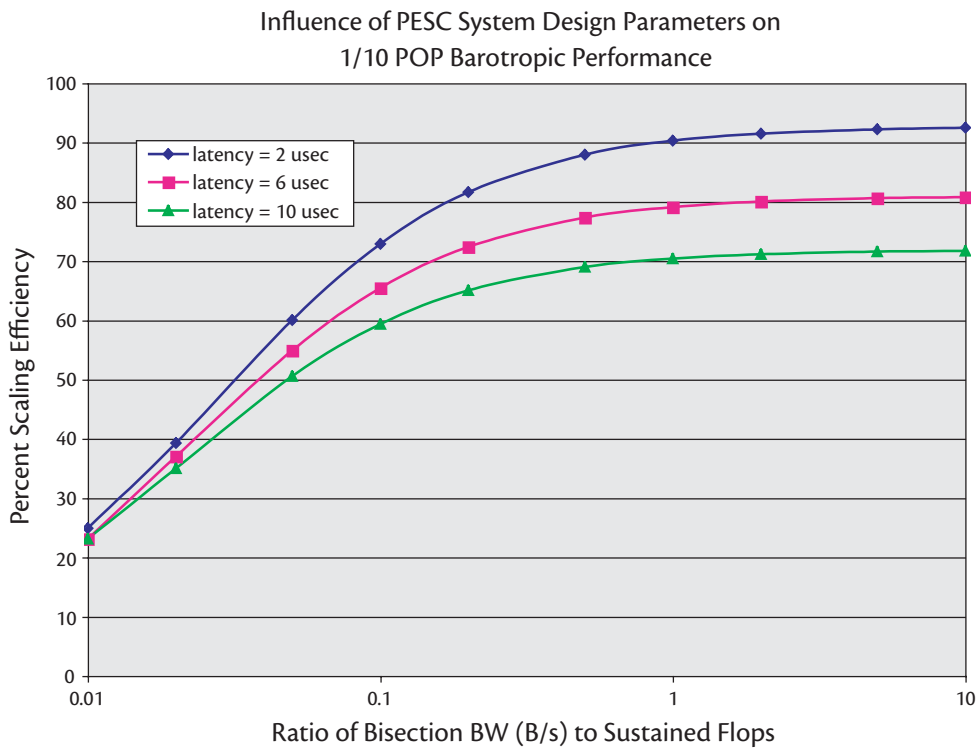


Figure 4. Plot of the trade off between latency and the ratio of sustained bandwidth to sustained FLOPS in the explicit barotropic component in the MOM model.

5 km POP at a sampling frequency of three days. Likewise, minimum disk bandwidths required in order to maintain I/O at a 5% overhead level relative to computation time, range between 0.2-20 GB/sec for the two POP simulation scenarios.

POP Summary

In summary, 6 TFLOPS of sustained performance could achieve the desired integration rates for POP at the 10 km eddy-permitting resolution. It is likely that 120 TFLOPS sustained would be required to achieve eddy-resolving 5 km simulations at the same integration rate. The most restrictive requirement derived from POP is for a high bandwidth (> 0.25 B/s/FLOPS sustained) and low latency (< 6 usec) interconnect to allow scaling of the POP barotropic component. Hardware that permits near-ideal scaling of global reduction operations is highly desired if the implicit method of solving the barotropic equations with an iterative solver is to be used in this or any other application.

Ocean Data Assimilation: ECCO/MITgcm

ECCO (Estimating the Circulation and Climate of the Ocean) is a consortium formed by scientists at the Jet Propulsion Laboratory (JPL), the Massachusetts Institute of Technology (MIT), and the Scripps Institution of Oceanography (SIO) under the National Ocean Partnership Program (NOPP) (for more information, see <http://www.ecco-group.org>). More recently, ECCO has added partners at NOAA's National Centers for Environmental Prediction (NCEP), NASA's Goddard Space Flight Center (GSFC), NOAA's Geophysical Fluid Dynamics Laboratory (GFDL), and the Atmospheric and Environmental Research Inc. (AER). ECCO scientists wish to assimilate ocean observations into an OGCM in order to interpolate and extrapolate the observational information into a complete description of the ocean for process studies and for climate forecasting. Data-assimilation methods used in this system are based upon exploitation of techniques developed in both sequential methods (Kalman filter and subsequent smoothers) and in Lagrange multiplier methods (adjoints).

The computational requirements of data assimilation are proportional to that of integrating the geophysical model. For instance, compared with a forward integration of the model, typical implementation of the adjoint method requires order 300 to 400 times the processor hours and order 3 to 5 times the amount of processor memory. Thus, this analysis will focus on estimating the costs and performance of the OGCM.

The first generation of ECCO analyses have been based on the MITgcm model (for more information on this model, see <http://mitgcm.org>). MITgcm is a GCM created to study both oceanic and atmospheric problems on a wide range of scales. Spatial discretization in MITgcm is carried out using the finite volume method on the Arakawa C grid. Computationally, this looks analogous to a second-order finite difference method. The MITgcm code vectorizes and has been deployed on large-scale MPP systems (for example T3E systems and IBM SP systems). It is reported by the MITgcm group that in all known cases, the MITgcm model performs and scales very competitively with equivalent numerical code that has been optimized for a particular system.

The rate of scientific throughput data assimilation researchers desire to achieve on a petascale computer has been characterized as 1 model year of integration every three hours for an OGCM operating at 15-km resolution (1/6 degree) on a global grid with a commensurate vertical resolution of between 5 and 10 m near the surface. This resolution equates to a grid of approximately 650 x 650 x 6 (for a cubed sphere configuration) with 75 vertical levels.

Not much has been published in the literature that quantitatively characterizes the MITgcm performance characteristics on parallel computers. In the absence of detailed data, a rough estimate of the performance required to achieve this throughput goal can be achieved by calculating the computational cost of the equivalently sized POP ocean model. This analysis, using a simplified POP performance model, yields an estimate of 26 PFLOP/simulated year. Thus, one can achieve the desired integration rate of one simulated year every three hours for a single ensemble instance with about 7.3 TFLOPS of sustained performance. Such performance levels can be achieved through dedicated use of the postulated 2007 system, or embarrassingly parallel use of the 2010 petascale

system to simultaneously generate multiple ensemble members. The performance levels would allow rates of scientific throughput on these ocean data assimilation problems to be realized, in a practical sense, for the first time.

The system memory requirements for the MITgcm application are reported to be 50 times the size of one 3-D variable. For the 15 km MITgcm problem, this amounts to just 76 GB. This memory footprint is far smaller and less restrictive than other applications, such as the 3-D turbulence problem sited in this section. In terms of data output, the largest volume of data that system produces are files containing the four 3-D fields (horizontal velocity, temperature, and salinity) that are output every 10 simulated days. This produces roughly 220 GB/simulated year, or about 1.8 TB/wall clock day at the desired simulation rate.

ATMOSPHERIC SCIENCE APPLICATIONS

Non-Hydrostatic Numerical Weather Prediction (NWP) Model: Weather Research and Forecast (WRF)

The WRF model is the next-generation model and assimilation system being developed and used by a broad community of government and university researchers and educators (<http://www.wrf-model.org/>). It is being used to advance both the understanding and prediction of important non-hydrostatic mesoscale weather, particularly at the 1-10 km scale. It is also being used for regional climate modeling, chemistry and air-quality research and prediction, large-eddy simulations, cloud and storm modeling, and data assimilation. WRF features include several dynamical cores based on finite difference methodologies and many options for physical parameterizations (microphysics, cumulus parameterization, planetary boundary layer, turbulence, radiation, and surface model) that are being developed by various groups. It includes two-way moving nests and is, or will be, coupled with other models, including hydrology, land-surface, and ocean models. The current WRF community consists of over 2,200 registered users. The June 2004 WRF Users Workshop included 173 participants representing 93 institutions. WRF is currently in operational use at NCEP and AFWA (Air Force Weather Agency). In addition, WRF is run at a number of

universities and is the core model for the large, NSF-funded cyberinfrastructure effort, Linked Environments for Atmospheric Discovery (LEAD at <http://lead.ou.edu/>)

Key aspects of the WRF software design are: (1) single-source code methodology; (2) use of modern programming language constructs in Fortran90, including modules, dynamic memory allocation, derived data types (structures), and recursion (array syntax is avoided for performance reasons); (3) a layered software architecture with well-defined interfaces that separate computer-platform-specific concerns from model-specific concerns, enhancing modularity and reuse; (4) multi-level parallel decomposition to facilitate efficient execution on all foreseeable parallel computer architectures and processor types; (5) a code registry data base and tool that helps programmers manage data structures and interfaces across the software hierarchy; (6) application program interfaces (APIs) to insulate the WRF software from external packages for communication, I/O, and data formats that may vary by institution, model application, or computer architecture; (7) a moving nest and model-coupling infrastructure that is scalable and efficient; (8) choice of a storage order and loop nesting order that is optimally performance-portable for clusters and vector machines; (9) capability to add hundreds of new equations for complex representations of microphysical and chemical species interactions; and (10) portable and efficient on a variety of parallel computers.

WRF can be run with idealized initial conditions or with initial conditions based on assimilation strategies that use larger-scale model results and observations. Running a WRF forecast includes the collection of observational data, assimilation of the data, running a set of WRF simulations under slightly varying initial conditions and model physics (an ensemble), analyzing and synthesizing the results of all the simulations, and visualizing the forecast using two- and three-dimensional graphics, including animations. All of these steps need to be accomplished significantly faster than real time for the forecast to have any practical value. This fact limits, for a given computing resource, the operational forecast resolution and the number of ensemble members that can be employed.

WRF is also used for research that includes very-high-resolution simulations to study multi-scale phenomena such as a convective complex (e.g., a squall line), individual convective storms, and smaller-scale storm features such as downbursts and tornadoes. It is also used for carrying out extensive parameter studies that can be thought of as ensembles consisting of hundreds to thousands of simulations. Data mining of the results from these large ensembles then becomes crucial to finding features of interest within them and to synthesizing results by characterizing simulation behaviors and their relationship to the parameter changes. Interactive analysis and visualization of the results of single simulations or ensembles becomes challenging because of the large volume of data produced and benefits from parallel algorithms. Parallel I/O becomes vital to moving data quickly between disks and memory; non-sequential analysis is aided by the availability of large shared memory systems. Detailed analysis and visualization can take as many, or even more, computer cycles as the simulation itself.

WRF is a limited-area, non-hydrostatic, and conservative equation model that uses a regular grid and includes a mass-based vertical coordinate. Explicit, time-split numerics are used. A variety of parameterizations for microphysics, boundary and surface layer, and radiation are available. The base WRF model consists of approximately 165,000 lines of code of which about 40,000 are automatically generated. It runs on a variety of parallel distributed and shared memory systems, including Cray, IBM, HP, SGI, and Linux. Array layout was designed to efficiently use vector systems (i.e., taking into account that horizontal grid-point dimensions are almost always much greater than the number of vertical grid points). Model coupling is handled through an extension to the WRF I/O. Integration with aspects of the Earth Simulation Modeling Framework is underway. Data assimilation (e.g., WRF 4DVAR, ARPS ADAS), analysis, and visualization are typically considered separate activities. Their development is typically handled separately from the model infrastructure.

WRF performance has been documented on a variety of architectures. Figure 5 shows some of the benchmarking results from the WRF web page (<http://www.mmm.ucar.edu/wrf/bench/>). The fastest system displayed is the Itanium-based SGI Altix, which achieves just over 150 GFLOPS on

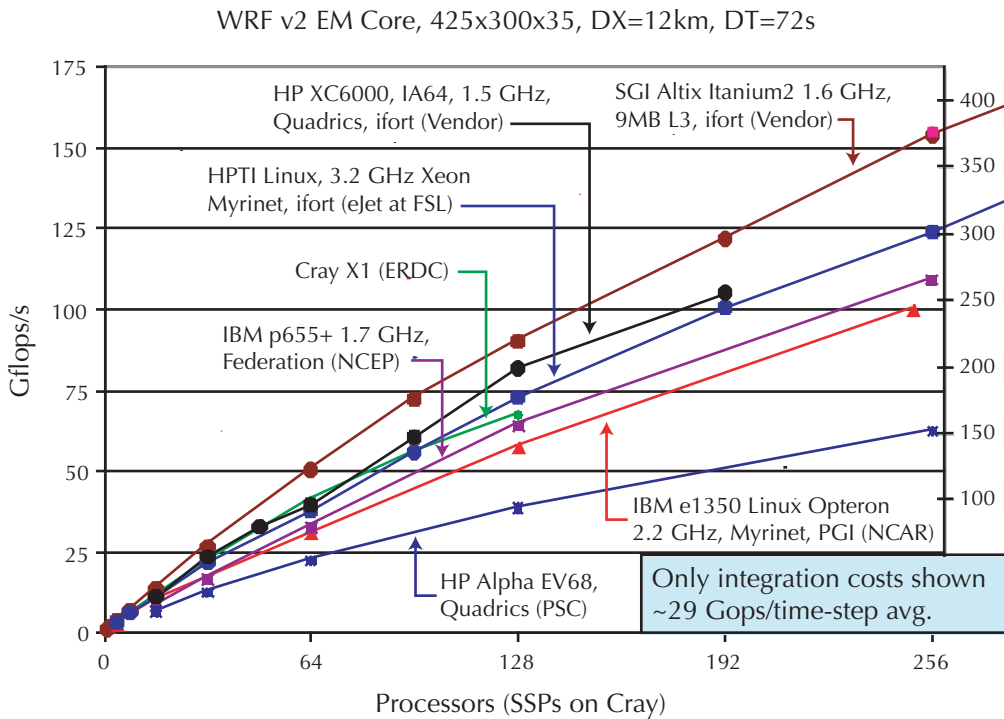


Figure 5. WRF performance on various systems with up to 256 processors (SSP's on the Cray X1 are shown). The fastest machine shown, the Itanium-based SGI Altix system, runs at just over 155 GFLOPS, or 9.5% of peak on 256 processors. (J. Michalakes, NCAR)

256 processors. If the subdomain patch size is maintained constant and the problem scaled up, a nearest neighbor, finite difference application such as WRF should scale well to thousands of processors.

Work is underway within the LEAD project to use grid technology to enable automatic use of multiple-grid resources in carrying out WRF simulations. More generally, it is a research project directed at the development of a comprehensive national cyberinfrastructure for mesoscale meteorology primarily for higher education and research communities. LEAD is addressing fundamental information technology research challenges needed to create an integrated, scalable environment for identifying, accessing, preparing, assimilating, predicting, managing, analyzing, mining, and visualizing a broad array of meteorological data and model output, independent of format and physical location. A transforming element of LEAD is Workflow Orchestration for On-Demand, Real-Time, Dynamically Adaptive Systems (WOORDS), which allows the use of analysis tools, forecast models, and data repositories not in fixed systems that can (a) change configuration rapidly and automatically in response to

weather; (b) continually be steered by new data; (c) respond to decision-driven inputs from users; (d) initiate other processes automatically; and (e) steer remote observing technologies to optimize data collection for the problem at hand.

WRF Performance Modeling

Based on benchmark simulations, John Michalakes (WRF architect) developed an estimator of needed computational resources. This provides a time estimate for a single-grid simulation with 2-category microphysics (not nested) given sustained computer parallel performance. Sustained performance will depend on the granularity of domain decomposition and communication costs, but in estimates below, it is assumed that subdomains with 40 x 40 horizontal grid points are sufficient to get good performance.

The real-time requirements of weather forecasting place a limit on the particular forecast interval, resolution, and simulation domain size that can be run. This is because the computational complexity of the simulation out-scales the parallelism that can be typically applied; quadrupling the number

of processors maintains a constant workload per processor if the horizontal resolution is doubled, but the time step must be reduced proportionately as well, thus increasing the overall run time. Also, the statistics obtained from using an ensemble prediction technique (running tens of forecasts with slight variations of the initial conditions) give a sense of the forecast uncertainty. For this reason, running embarrassingly parallel ensemble forecasts is considered preferable to running one very-high-resolution forecast across the same computing resource. It would also be quite useful to employ moderately embarrassingly parallel runs in carrying out hundreds to thousands of smaller simulations for research parameter studies.

The need for a petascale machine can be based simply on a single simulation with 2-category microphysics used in the WRF benchmark simulations. Consider an illustrative example in which a petascale system runs at 15% efficiency (typical for many fluid dynamics simulations run today on microprocessor clusters) yielding 150 TFLOPS sustained. A two-hour simulation with 10-m resolution for following the evolution of a tornado embedded within a severe storm can be used to study turbulent and fine-scale behavior. Consider an integration domain of 100 x 100 x 20 km represented using 10,000 x 10,000 x 2,000 grid points. This simulation will take approximately 220 hours using a 0.06 sec time step. This estimate is based on extrapolating current benchmark simulations that use a very simple 2-category microphysical parameterization. Data volumes from a single run will range between 1-12 PB depending on post-analysis and visualization needs. The memory required for the simulations exceeds 20 TB. One or two simulations per year would be reasonable. Analysis and visualization will require 20-200% of the actual run time.

It is known from performance modeling that a 40 x 40 2-D horizontal subdomain is typically big enough for the computational costs on this size subdomain to dominate the costs of perimeter communications with neighboring subdomains. For this tornado-formation problem there would be $250 \times 250 = 62,500$ such subdomains implying that the code should be efficient for a PetaFLOPS machine with up to 62,500 processors. If such a system could sustain 2.4 GFLOPS/processor, the performance goal of 150 TFLOPS could be achieved. As Figure 5 illustrates, this is approximately a factor of four faster than the current WRF benchmark figures, on a per-processor basis.

Research simulations often include more sophisticated parameterizations that involve the use of hundreds of additional equations for representing detailed aerosol, microphysical, or chemical components. Using a simple but more realistic microphysical parameterization with five, rather than two, categories will approximately double the computation time for the above simulation. Further, microphysical parameterizations being explored today contain up to several hundred microphysical variables. Consider a severe-storm simulation using 48 microphysical variables with 50-m resolution in the horizontal, a stretched vertical grid with 400 points, and a 0.3 second time step that is run for four hours. Use of horizontal subdomains of 40 x 40 points leads to 2500 subdomains. Thus, the calculation easily maps onto 2,500 processors. The estimated 8 hours falls within the 10-hour soft limit considered reasonable by some for making tens to hundreds of simulations in a year. Each simulation would produce nearly 84 TB of output if all variables are saved every minute over the four-hour simulation, with a memory footprint of approximately one TB.

The above calculations of execution time do not take into account I/O time, which can be substantial.

Towards High-Resolution Global Models

There is now ample evidence that the best, and perhaps only, way to correctly represent convective processes is to simulate them directly. In weather forecast experiments, researchers have found that at 4 or 5 km, when the convective parameterization is turned off, the simulation of convective systems, especially in the summer, suddenly looks realistic: in particular, the diurnal cycle of precipitation is improved. This has much to do with the fact that, at these resolutions, the latent heat in these systems can be released at more realistic scales. Instead of releasing it on scales where the model responds by producing balanced systems, systems appear that are driven by unbalanced dynamics. For slightly coarser resolutions, for example, between 5-20 km, these processes are not captured properly by either convective parameterization or by direct simulation. A similar breakdown in physics modeling approaches occurs both for radiative transport models and planetary boundary layer (PBL) parameterizations. In the case of PBL parameterizations, these breakdown at similar resolutions and must ultimately be replaced by large-eddy

simulations (LES) of turbulent processes near the surface. Likewise, column-wise radiative transport models must be replaced with fully 3-D models of radiation when resolution gets sufficiently fine. As a result, a kind of “physics no man’s land” exists, as shown in Figure 6.

Thus, there are excellent reasons for trying to achieve the computer power to push NWP models down below “convection resolving” resolutions and ultimately climate simulations as well. This threshold resolution below the physics “no man’s land” is not known precisely, but is taken here to be 5 km.

It is also likely that the global climate simulations would also benefit from simulations at these resolutions, though the need to go to much longer simulation time periods means that we may not be able to reach these resolutions as soon as one can for weather forecasting. Climate simulations have an advantage over weather predictions in that they don’t have to be produced in real time.

The computational costs and achievable integration rates for a global, nonhydrostatic numerical weather prediction and climate simulation version of WRF running at or below 10-km resolution can be estimated. It will be assumed that 5 simulated years/wall clock day is acceptable for climate

simulation and 60 simulated days/day will be acceptable for numerical weather prediction. The details of a data assimilation system for the numerical weather prediction model will be ignored, as the computational cost of the weather model typically dominates. The “performance estimator” application of Michalakes is once again used for these estimates. However, new models of land surface and chemistry processes (WRF-CHEM) are currently being added to WRF and details of a global version of the WRF model, such as the grid and underlying numerical methods, have yet to be worked out. Thus, performance estimates derived for an idealized GCM version of WRF, neglecting land-surface and chemistry model costs, are almost certainly underestimates. These estimates are still valuable, if a bit optimistic, in determining the transition point at which such global simulations at current mesoscale resolutions become feasible.

WRF Computational, Memory, and Data Requirements

The computational costs of WRF running with 100 levels between 10 km and 1 km have been calculated and are shown in Table 3. Memory costs for WRF are derived based on the estimates from profiling the running model; there are the equivalent of 77 3-D variables in the computational core. The table of results shows that, for NWP purposes, global-con-

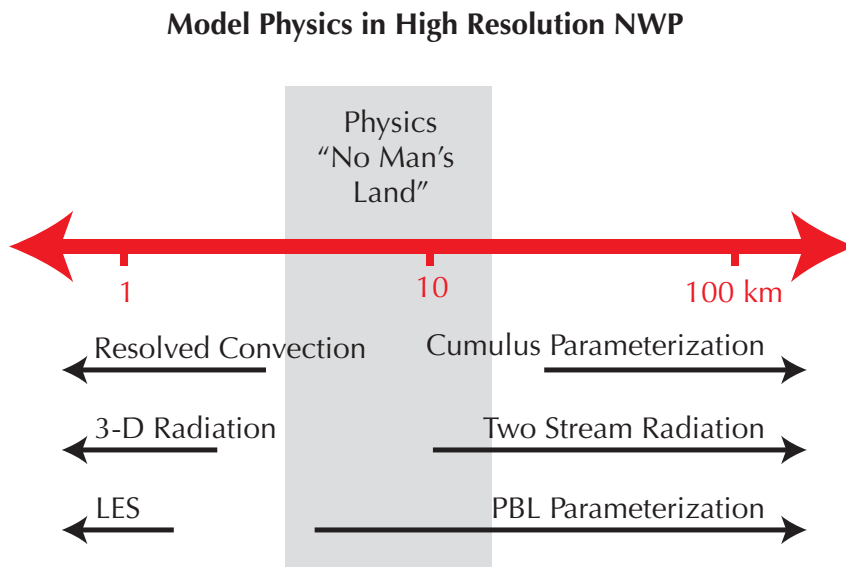


Figure 6. As one pushes through the physics “no man’s land” between 5 km and 20 km resolution, much more realistic, direct simulation of atmospheric physics phenomena becomes possible. Large-eddy simulation (LES) replaces parameterizations of the planetary boundary layer (PBL), resolved convective processes in clouds provide more realistic descriptions than convective parameterizations, and a model of 3-D radiative transport becomes necessary. (J. Klemm, NCAR)

vection-resolving simulations at 5 km, as well as experimentation with 1 and 2 km regional models, should be feasible in 2007. A global 3-km WRF-based model with 100 layers that would resolve convective processes with unprecedented accuracy on a global scale should become feasible in 2010 if a system capable of 100 sustained TFLOPS is available. The 3-km simulation would require 66 TFLOPS sustained, would occupy roughly 1.75 TB of main memory, and will generate 206 TB of data per simulated year. Some experimentation with global nonhydrostatic climate simulations at 5 km become feasible, on a limited basis, on a petascale system capable of 100 TFLOPS in 2010.

Hydrostatic Climate Simulation: CAM/CCSM

The Community Climate System Model (CCSM) consists of four component models—atmosphere, ocean, sea ice, and land—and a coupler that regrids data for the different spatial grids and time-stepping schemes. The current mode of execution places each component on a distinct set of processors with the atmosphere requiring roughly half the processors and constituting the coupled simulation’s dominant cost. So, to first order, a performance model of the coupled system can be developed by considering only the atmospheric model.

The Community Atmosphere Model (CAM) has two major computations: (1) the dynamics, representing the fluid flow calculation of atmospheric winds; and (2) the physics, representing column radiation balances, moist convection, and clouds. The dynamics employs a Eulerian spectral transform algorithm for the approximation of all terms in the momentum, mass, and energy-conservation equations, and a semi-Lagrangian approximation for the transport of moisture and atmospheric trace gases. The physics is embarrassingly parallel, though a significant load imbalance would exist if left in the natural parallel decomposition. Data transposition using message passing as well as shared memory parallelism is employed.

The dynamics calculation is dominated by the spectral transform. A performance model of the spectral transform can be developed to estimate the time for a multi-level calculation. The computational operation counts and communication-cost estimates are shown below for a one-dimensional decomposition and modified by Rich Loft (NCAR) to reflect a simple transpose between Fast Fourier Transform (FFT) and Legendre Transform (LT) phases including levels. The time for the FFT, Legendre transform, and communication overhead are estimated using machine-dependant rate constants a, b, d, and e.

Table 3: Integration rate requirements for a hypothetical NWP (60 simulated days/wall clock day) and climate-simulation (5 simulated years/wall clock day) versions of a WRF GCM, yields these sustained performance level requirements and data-set sizes for the PCG.

Resolution (km)	TFLOPS sustained to achieve 60 days/day	TFLOPS sustained to achieve 5 years/day	Global WRF Data volume TB/sim year
1	1609	48260	1892
2	212	6350	466
3	66	1975	206
4	29	875	116
5	15	467.5	74
8	4.3	129	29
10	2.4	71	18.5

$$\text{Time for FFT} = a \cdot 5 \cdot (6L+1) \cdot J \cdot I \cdot \log_2(I)$$

$$\text{Time for LT} = b \cdot 2 \cdot (6L+1) \cdot J \cdot M^2$$

$$\text{Time in COMM} = d \cdot P + e \cdot 2 \cdot (6L+1) \cdot J \cdot (2M+1)$$

Nomenclature:

- M wave number resolution
 - I number of longitudes ($I \geq 3M+1$)
 - J number of latitudes ($J=I/2$)
 - L number of vertical levels
 - P number of nodes (computational unit doing FFT or LT)
 - a computational rate of FFT in FLOPS/node
 - b computational rate for LT in FLOPS/node
 - d latency factor
 - e bandwidth factor
-

Using this model with estimates of network bandwidth and the speed of a node in computing FFTs and LTs, we can determine the overall computational rate of the computer for performing spherical harmonic transforms as well as parallel efficiencies.

Each process in the physics can be modeled based on the relevant process time step and the number of floating point operations per column. The operation counts employed were measured from real simulations with 26 levels at 160-km

(T85) resolution using hardware performance counters. Here it is assumed that the number of levels increases proportionately with increased resolution, according to the values given in Table 4. It is assumed that the computational cost of physics routines varies linearly with vertical resolution, with the exception of the long-wavelength radiative transport component for which the CAM algorithm is known to vary quadratically. This assumption may overestimate the cost of physics in future, higher-resolution versions of CAM as new, less costly long-wavelength radiation algorithms come on line. This underestimate is mitigated by that fact that as more computing power becomes available, modelers have historically added more physics processes, driving up the computational cost per column. For example, it is currently planned that tropospheric chemistry will be added to CAM. Thus, operation breakdown with these assumptions is shown in Table 4. The physics costs for T85/26 levels resolution has been used as the basis of estimates of the overall scaled computational costs at higher resolutions.

Performance of the dynamics is combined with the performance of the physics by calculating the required time step for a given horizontal resolution (subject to CFL limits) and the frequency of the process updates. The single processor efficiency for the physics is an input parameter to the model as it depends on the level of vectorization and cache utilization

Table 4: Physics component operation counts for CAM at 26 level T85 resolution.

Physics Component	Flops per column per day	Flops per column per time step
LW Radiation	4.50E+06	-
SW Radiation	4.90E+06	-
Other Physics	-	1.00E+05
Sulfur Chemistry	-	4.30E+04
Trop. Chemistry	-	-
Strat. Chemistry	-	-
Total Physical Ops	9.40E+06	1.43E+05

The zeros for chemistry operations reflect that the current standard simulation does not include these processes. LW=long wavelength; SW=short wavelength

achieved. In this way, one can avoid trying to predict the actual memory performance of a processor based on hardware specifications.

The number of time steps required for a century-long “simulation” is given for the standard horizontal resolutions in Table 5. It is a fundamental feature of climate simulations that the long time integrations limit the spatial resolution that may be applied. Faster processors, fast memory (e.g., vector), and low-latency interconnects are key to fast time stepping.

Typically, production simulations are only performed if throughput greater than five simulated years per wall clock day can be achieved. The model of computational complexity outlined above can be used to compute the number of FLOPS in each part of the calculation, physics and dynamics. These results, shown in Table 6, can in turn be used to

compute the sustained FLOPS rates required for the different resolutions to achieve the threshold integration rate of five years per day. One sees immediately that useful climate simulations are probably out of the question at T1279 on the system envisioned in the 2010 collaboratory.

Historically, a 1-D decomposition along the latitude direction has been used in production CAM simulations. This is because of application scalability issues associated with attempting finer grain parallelism. Table 6 shows the required sustained, per processor computation rate in GFLOPS for various resolutions if a latitude-based, 1-D decomposition is employed. Based on computer technology projections for microprocessors, the data in Table 6 suggest that any resolution higher than T170L40 would be impractical during the lifetime of the petascale project if microprocessors-based systems were employed. Vector systems will be able to go a bit

Table 5: Dimensions and time steps of the spectral CAM model as a function of resolution.

Wave number	Latitude	Longitude	Time step (sec)	Steps/year
42	64	128	1200	26280
85	129	258	600	52560
170	256	512	300	105120
341	513	1026	150	210240
682	1024	2048	72	438000
1279	1920	3840	48	657000

Table 6. Computational complexity of spectral CAM at different resolution, and the simulation rates required to achieve five simulated years per wall clock day.

Wave Number	Latitudes	Vertical Levels	Dynamics PFLOPS/yr	Physics PFLOPS/yr	Total PFLOPS/yr	TFLOPS to sustain 5 yr/day	Required GFLOPS/PE (1-D decomp)
85	128	26	0.018	0.362	0.38	0.022	0.17
170	256	40	0.416	3.9	4.316	0.25	0.98
341	512	52	8.34	37	45.34	2.62	5.12
682	1024	80	208	448	656	38	37.1
1279	1920	104	3020	5650	8670	502	261

farther. It seems certain that a spectral dynamical core version of CAM can integrate on vector machines at 5 years/day at T341L52 resolution in the 2007-2010 time frame.

These conclusions depend on the ability of CAM to scale well beyond 1-D decompositions. CAM currently has the capability to decompose dynamics and load balance the embarrassingly parallel physics across larger numbers processes. To work effectively, these approaches require a certain system balance between processor speed and interconnect bandwidth (i.e., there must exist a sufficiently large bytes/sec/FLOPS ratio in the system such that load balancing the physics workload actually improves CAM's scalability and performance).

A simple model will illustrate the point. Assume that N state variables must be packed together and passed back and forth between the respective dynamics and physics decompositions each time step. Also assume, pessimistically, that every vertical column must be passed between processors in this way, and that the communication is non-local in nature. In reality, some traffic will stay on the local processor or its neighbors. One finds that the interconnect bisection bandwidth-to-processor balance that must exist for the overall load balancing communication overhead to remain < 5% is approximately $N \cdot 0.04$ bytes/sec/FLOPS sustained. For a nominal value of $N=6$, this implies 0.24 bytes/sec/FLOPS. This value is quite close to the requirements of the pseudospectral 3-D turbulence model "flow_solve" that was studied earlier in this report. Thus, T341L52 and even T682 might be possible.

Finite Volume Version of CAM

Currently, CCSM long-range modeling plans call for using a Lin-Rood finite volume dynamical (FV) core option in CAM. The finite volume dynamical core has a pseudo-Lagrangian algorithm in the zonal direction, but has none in the meridional. Thus, the CAM FV option must take short time steps. For example, the 1 x 1.5 degree FV dynamics takes ten time steps for each 1800-sec physics time step. This adds computational cost to the FV dynamics. Table 7 illustrates these costs for a series of FV resolutions roughly analogous to the spectral T85, T170, T341, T682, and T1279, respectively. The computational costs of the FV dynamics used in the table are derived from a measured hardware operation count of 1246 FLOPS/site/FV time step. Although the computational costs of FV CAM are somewhat higher (~40%) when compared to the spectral core (see Table 7), they do not substantially alter the conclusion that it is feasible to run CAM at the equivalent of T341 or even T682 resolution on the petascale system.

Cloud Resolving Convection Parameterization

Climate models are relatively consistent in regard to the warming due to CO₂, but differ widely in prediction of the hydrological cycle (e.g., some suggest that average precipitation will increase sharply, others that it will not change much). Quantifying this aspect of the climate system requires knowledge of scale interaction ranging from cloud (~1 km)

Table 7. Computational complexity of finite volume (FV) version of CAM at different resolution, and the simulation rates required to achieve five simulated years per wall clock day.

FV Longitudes	FV Latitudes	Vertical Levels	Dynamics PFLOPS/yr	Physics PFLOPS/yr	Total PFLOPS/yr	TFLOPS to sustain 5 yr/day
288	181	26	0.30	0.31	0.61	0.035
576	362	40	3.64	2.99	6.63	0.384
1152	724	52	37.9	25.2	63.1	3.65
2304	1448	80	466	276	742	42.9
4340	2715	104	3994	2160	6154	356

to global. Understanding interactions among convection, radiative transfer, and surface processes across a wide range of scales is vital.

Cloud-system-resolving models are an approach of choice for problems of this type because they couple the physical processes at scales where instabilities and attendant transport mechanisms develop spontaneously. Therefore, scientists have developed an approach called Cloud Resolving Convection Parameterization (CRCP) by some researchers, and superparameterization by others, that incorporates sub-grid-scale land-surface processes, topography, improved microphysics, and boundary layer processes, and couples them to an atmospheric general circulation model that captures the motions of the atmosphere at large scales. In the superparameterization approach, atmospheric convection is represented explicitly, thereby mitigating several problems associated with traditional convective parameterization. This approach originated in the Cloud Systems Group of the Mesoscale and Microscale Meteorology Division of NCAR (Grabowski and Smolarkiewicz, 1999; Grabowski, 2001; Smolarkiewicz et al., 2001) and has been successfully applied in an “aquaplanet” model as well as in the more realistic setting of the Community Atmospheric Model (Ziemianski et al., submitted).

Estimate of Computer Power

Both traditional parameterized and CRCP physics schemes are embarrassingly parallel. On the one hand, CRCP has good parallel load balancing characteristics: the execution time in empty cells and cells with clouds is found to only differ by 30% due to faster convergence of the generalized conjugate residual (GCR) Krylov solver. The traditional 1-D column model approach suffers from severe load imbalance because only a fraction of model columns may have active clouds. On the other hand, CRCP is computationally expensive, roughly a factor of 100 times more so than traditional parameterizations. Finally, future improvements to these schemes may introduce local communication between columns. In the case of the 2-D CRCP models, the boundary conditions are periodic within each 2-D slice. We anticipate that inflow/outflow type conditions may be implemented in the future, implying some form of local communication between neighboring columns.

Benchmark tests of CRCP kernel provided by Grabowski et al. (1999) indicate that the physics package performs 158.5 GFLOPS/dynamics column/simulated day for a 2-km sub-grid-scale model coupled to T85 resolution GCM dynamics. CRCP involves numerous calls to mathematical intrinsic functions, such as EXP and LOG: if vectorizing versions of these intrinsic functions are employed, it is known that optimized versions of the CRCP physics can run in excess of 10% of peak on microprocessor system. There is no reason to think that the calls could not be optimized for very efficient execution on vector systems as well. These data can be used to estimate the computational costs of running CRCP physics: one finds that 22 TFLOPS sustained would be required to achieve even a 1 simulated year/wall clock day integration rate in the case of a CAM-like GCM coupled to CRCP physics. As expensive as it is, CRCP is still two orders of magnitude cheaper than direct numerical simulation of global models at 1-2-km resolution.

The large increase in computer power required for CRCP is best put into perspective by the following comparison: a 1-day run of a global cloud-system-resolving model (perhaps with a 3-km grid spacing) is equivalent to a 1-year run of a global model operating superparameterization and to a 1-millennium run of a contemporary climate model operating traditional parameterizations of convection and clouds.

EARTH SCIENCE APPLICATIONS

Mineral Physics

Mineralogists wish to understand the structural, elastic, and thermodynamic properties of the major minerals present in the Earth's interior, their phase diagrams, and the properties of melts; however, the Earth's interior harbors physical conditions that are often difficult or impossible to reproduce in the laboratory. Thus, numerical simulation on the atomic scale is a critical technique for understanding the physical and chemical properties of minerals under such conditions. In the lower mantle, over a large range of pressures (0-130 GPa) and temperatures (up to 3000 K), the important phases are (Mg,Fe,Al) (Fe,Al,Si)O₃ perovskite and post-perovskite, (Mg,Fe)O magnesiowustite, and CaSiO₃ perovskite. In the upper mantle, the most important phase is (Fe,Mg)₂ SiO₄

olivine. The Earth's transition zone contains complex phases such as garnets, modified spinels, and silicate ilmenite. Modeling the physical properties of these minerals requires a method for extracting their ground state electronic structure and atomic configuration, and a method for converting that structure into a knowledge of the macroscopic physical properties of the mineral.

Computing the Electronic Structure

Two basic physical methods are used for modeling the electronic structure of materials at the atomic scale. In both approaches, only the electrons, which are much lighter than atomic nuclei, are treated quantum mechanically. In the first approach, a stochastically generated approximation of the N-body electron wave function in a potential of atomic nuclei is used to calculate the energy of the system using Quantum Monte Carlo (QMC) methods. The second approach, density functional theory (DFT), simplifies the N-body quantum mechanical problem, effectively replacing it with a set of single particle problems with an interaction potential that contains the many-body forces. DFT is a theoretically (but not practically) exact way to extract the ground-state properties of the system without constructing the full many-body quantum mechanical wave function. In DFT, the atoms are moved in such a way as to lower the total energy of the system, at zero temperature for ground state properties, or to sample the ensemble appropriate for a given temperature and pressure. QMC is generally too slow on current computers for this kind of optimization or simulation, but through advances in theory and computation, QMC will allow relaxation and simulations at finite temperatures representative of conditions in the Earth's interior.

Although the more exact solutions to the many-body quantum mechanical problem obtainable using QMC techniques are becoming feasible, most current computations in mineralogy are based on DFT. Various basis sets are used in solving the DFT equations, depending on the problem at hand, and the most computationally optimal method may depend on the computer architecture. Current computations carried out with common community codes use plane wave bases and

pseudo-potentials, and typically scale to tens of processors. More computationally optimal methods are not as widely available in well-supported codes and less often used.

All electronic structure codes, whether DFT- or QMC-based, ultimately depend on fast implementations of linear algebra algorithms. QMC requires fast solution of changes in large determinants, and DFT requires solving the (usually generalized) eigen-problem. Methods using plane wave bases give dense matrices, whereas local bases can give sparse matrices for large enough systems. In both methods, the atoms are moved in such a way as to lower the total energy of the system, with the goal of iteratively minimizing the total ground state energy.

Calculating Physical Properties

For geoscience applications, the ground state energy is not the primary goal, but rather high-order properties such as elastic constants as functions of pressure, temperature, and composition; densities and equations of state of crystals and melts; thermodynamic properties; and physical properties such as electric resistivity and thermal conductivity. These macroscopic properties of minerals can be extracted from atomic-level simulations using two general approaches. In first method, the calculations are performed in two major steps:

1. Perform static calculations based on first-principles data:
 - crystal structure
 - elastic constants
 - electronic properties: electronic band structure and the corresponding density of states
 - electrical and magnetic properties
2. Perform dynamical property calculations based on static properties:
 - zone-center dynamical properties and dielectric properties for non-metals
 - phonon band dispersion in the whole Brillouin zone
 - thermodynamic properties within the quasi-harmonic approximation

Alternatively, in first principles molecular dynamics (MD), a supercell of atoms is simulated. A supercell is a periodic cell of hundreds, thousands, or in some cases (using a potential

model rather than from first principles), up to a billion atoms. Forces between atoms are calculated at each time step and the positions of the atoms are propagated forward in time using Newton's Law, $F=ma$. These calculations are extremely computationally intensive, but have been shown to scale well on a variety of parallel supercomputers. Such MD calculations already have been done for some Earth minerals, at least for small supercells and relatively short run times. The integration time steps of these models are typically a fraction of a femtosecond, and total run times required are tens of picoseconds.

A DFT-based Calculation: Computational Complexity

Examples of DFT codes used in computational mineralogy are the ABINIT (<http://www.abinit.org>) and PWscf (<http://www.pwscf.org>). ABINIT is a GNU public-licensed package whose main program allows one to find the total energy, charge density, and electronic structure of systems made of electrons and nuclei (molecules and periodic solids) within DFT, using pseudopotentials and a plane-wave basis. PWscf is another DFT-based code that works on a variety of architectures, including microprocessors and vector processors, and can execute in parallel on distributed memory systems using MPI.

Both the ABINIT and PWscf codes are parallelized in terms of the number of k points (wave numbers) in the plane-wave basis of the problem and require little communication. The ABINIT code is parallelized on k points for the static calculations and on both electronic bands and k points for the dynamical calculations. Consequently, the static calculations (like crystal structure determination) of the small systems with large number of k points may be performed in a highly parallel run, while the large systems with a small number of k points will not scale as well. As noted earlier, many hundreds of such runs may be required in the course of a mineralogical study. An ensemble of such calculations could consume substantial resources.

Previous ABINIT runs can be used to get a sense of the amount of computation required in this phase of the calculations. For example, a relaxation at a given pressure of a perovskite structure with 20 atoms per unit cell and a grid of

$4 \times 4 \times 4$ k points in the whole Brillouin zone (BZ), folded to 8 special k points in the irreducible part of the BZ, performed in parallel on the k points (8 CPUs) may require up to one week on an 8 CPU AMD Opteron cluster. The determination of the electronic density of states on the same structure with about 3500 k points in the whole BZ in a spin-polarized system may require up to 48 hours on a AMD 12 CPU Opteron cluster. The determination of the dynamical matrix in one high-symmetry q point (needed to calculate the phonons) may require up to 72 hours on a 12 CPU Opteron cluster. Using a nominal, observed efficiency of 8% of peak for scientific codes, a project of this magnitude is estimated to require 2×10^{15} floating-point operations. Note that there is both sequential and parallel work in this sequence of runs. For example, once the crystal structures are determined, the computation of the phonon band dispersion requires a large number of embarrassingly parallel calculations. Researchers typically make hundreds of such simulations in the course of a scientific study. Thus, the computational requirement of a single scientific study is estimated to fall in the range of 10^{18} floating point operations. Allowing one such complete study to be completed per week, a reasonable rate of scientific output for a group of researchers, would require 1.6 TFLOPS sustained.

It may appear that, for mineralogy, the resource requirements of current individual DFT-based calculations are not that substantial. However, computational mineralogical calculations dealing with solid solutions must be repeated for each end-member mineral, such as MgSiO_3 , FeSiO_3 , Al_2O_3 , MgO , FeO , CaSiO_3 and for some intermediate members, such as $(\text{Mg}_{0.5}\text{Fe}_{0.5})\text{SiO}_3$, $(\text{Mg}_{0.75}\text{Al}_{0.25})(\text{Al}_{0.25}\text{Si}_{0.75})\text{O}_3$, $(\text{Mg}_{0.75}\text{Fe}_{0.25})\text{O}$, and $(\text{Mg}_{0.875}\text{Fe}_{0.125})\text{O}$. Thus, the overall computational resource requirements of a complete scientific characterization of a mixture of minerals can be several orders of magnitude greater.

Alternatives to DFT-based methods

In current practice, calculation of ground state electronic structure of minerals using present DFT codes are limited by the speed of computer systems to a periodic cell with about 500 atoms. However, other algorithms for DFT-based methods that are more scaleable than those presently used, such as multigrid and finite elements (see, for example, [32](http://www.</p></div><div data-bbox=)

tcm.phy.cam.ac.uk/LocalOrbital/report.pdf), are emerging and will be more appropriate on a petascale platform. Further, more accurate DFT functionals used with greater computational cost and more exact calculations using quantum Monte Carlo (QMC) methods will require orders of magnitude greater computer time than current DFT-based calculations.

Hydrology

The computational requirements of the hydrological community are presented here in the context of the Los Alamos Distributed Hydrologic System (LADHS) (Winter et al., 2004). This model's software design consists of separate groundwater, land surface, and atmospheric (RAMS) modules linked by a coupler. The coupler supports a common workspace consisting of a uniform image of distributed memory and of methods for synchronization. The components in LADHS consist of the following:

Regional Atmosphere

The LADHS regional climate component uses RAMS (Regional Atmospheric Modeling System) (Pielke, 1992; Cotton, 2003). RAMS provides precipitation, temperature, humidity, and wind data to the hydrology component.

Land Surface

The Land Surface Hydrology (LaSH) component uses a grid-based discretization to partition precipitation or snowmelt into evaporation, transpiration, soil water storage, surface runoff, lateral subsurface flow, and subsurface recharge. Lateral subsurface flow is routed among grid elements using Darcy's equation.

Subsurface Hydrology

Groundwater is a major water resource that is not considered in current climate and most regional models. LADHS uses the Finite Element Heat and Mass code (FEHM), a three-dimensional multi-phase flow model (Zyvoloski et al., 1997).

River Routing

The channel-routing component is an important element of regional assessments in a river basin. More complex routing algorithms are being evaluated.

In such hydrological models, high resolution is required to account for the effect that spatial and temporal heterogeneities in the parameters have on critical hydrologic variables like spatially distributed soil moisture. Table 8 (from Winter et al., 2004) estimates the computational demands of a highly resolved simulation of the flux of water through the Upper Rio Grande Basin during a typical year. The analysis indicates that a simulation of one year of the land surface-atmosphere system dynamics requires on the order of 10^{16} operations with the load fairly evenly balanced between the two subsystems. Computational experiments bear this out (Winter et al., 2004).

Although the computational cost of the Rio Grande Basin problem is too small to occupy an entire petascale system, the computational requirements of hydrological studies are increased by three factors. First, scientists in most cases will want to perform Monte Carlo simulations based on multiple realizations to account for parametric uncertainties. Second, most simulations will cover a period of many years. Third, the Upper Rio Grande is about 10 times smaller than the Colorado Basin and 100 times smaller than the Columbia Basin. Thus, the requirements for a scientific study based on simulating a large basin over a period of many years could easily be on the order of 10^{18} - 10^{20} FLOPs. Of the three factors, the first argues for many instances or ensembles of simulations; the second suggests the need for faster processors; and only the third factor (simulating larger basins) will demand a system with a high capability. This case will be examined more closely.

Developing a detailed performance model of LADHS is complicated by several factors. Although RAMS is well documented in the literature (Cotton, 2003), insufficient detail about its communication requirements and the overall coupled application was obtained to complete a rigorous performance analysis for this report. Further, the requirement estimates in Table 8 (Winter et al., 2004) are only for the land surface-atmosphere subsystems. Nevertheless, some simple estimates of the computational requirement can be extrapolated for large-basin problems.

It is assumed that the land surface and atmospheric model dominates the computational cost. Also, it is surmised that the RAMS model resembles other finite difference meteorolo-

logical models with nearest neighbor communications and thus should exhibit good scaling on parallel systems if the problem size is sufficiently large. The LANL surface hydrology model, with one hundred times as many grid points, should be even more scalable than RAMS because it is also appears to have nearest-neighbor communications. Thus, the 1-km RAMS simulation of the Columbia Basin, with approximately 10 million grid cells, should have enough horizontal grid points to scale to large terascale and petascale systems consisting of thousands of processors. Scaling the data in Table 8 to problems of this size one finds that, at 10 TFLOPS sustained, a year-long simulation of the Columbia Basin would require one wall-clock week, and is estimated to produce 400 TB of data. A 100 TFLOPS system could complete a decadal simulation of the Columbia Basin in the similar amount of time. This sort of turn around in numerical experiments would seem to make such high-resolution studies of large river basins feasible for the first time.

Seismology

This section will discuss the computational requirements and capabilities of large-scale forward and inverse modeling of earthquake ground motions for both regional and global domains. Seismic simulations break down into two distinct

types: (1) the forward problem (calculating the propagation of seismic waves through spatially inhomogeneous soil materials) and (2) the inverse problem (estimating the soil property distribution that results in a predicted response that most closely matches observed records of past earthquakes). The forward problem is well posed, but is characterized by sparse operators: numerically, the forward problem consists of the computation of thousands of very large sparse matrix-vector multiplies (SMVPs). The inverse problem is several orders of magnitude more difficult to solve than the associated forward wave-propagation problem. It is also ill-posed and characterized by multiple solutions that are discontinuous.

The discussion of regional seismic simulation largely relies on the findings of Akcelik et al. (2003). In that paper, researchers extended earlier capabilities to include aspects crucial to practical seismological simulation (i.e., larger basins, with softer soils, and for higher resolved frequencies, all of which add significant computational complexity). The earthquake simulation algorithms employed in Akcelik et al. (2003) are based on finite elements on multi-resolution hexahedral meshes with up to 300 degrees of freedom (100 million grid points). The solution technique for the forward problem in Akcelik et al. (2003) is a hexahedral trilinear finite element

Table 8. Computational complexity details of Rio Grande Basin Hydrology problem.

	RAMS	LANL surface hydrology model
Basin size (upper Rio Grande)	92,000 sq. km	-
Duration of simulation	One year	-
Resolution	1 km	100 m
Number of grid cells	92,000	9.2M
Number of vertical layers and themes	22	80
Floating point operations per grid cell	300	100
Time step	1 second	1 minute
Total number of operations	20 PFLOPs	40 PFLOPs

spatial discretization on spatially adapted, balanced octrees with explicit central differences in time. Work complexity grows according to $O(n^{4/3})$, with a smaller constant due to the spatial adaptivity (the complexity is optimal for high-frequency wave propagation). The forward solver is executed repeatedly within the inverse solver, which is described below, and is thus the critical component determining performance. As previously indicated, state-of-the-art forward calculations have involved ~ 100 million hexahedral elements, that is, ~ 300 million unknowns, with prototype simulations approaching 3 billion unknowns. Future plans for this seismic application include moving to production simulations with several billion unknowns, with prototype simulations approaching 10 billion unknowns.

The largest global seismic forward simulations to date, performed on the Earth Simulator, used 4056 processors, had 36.5 billion degrees of freedom (13.8 billion grid points), and used 7.3 TB of distributed memory and sustained 10.4 TFLOPS, or 33% of peak (Komatitsch et al., 2003). Unlike Akcelik et al. (2003), these researchers used a spectral-element method (SEM), whose main advantage is computational efficiency through an exactly diagonal mass matrix and a very simple and efficient explicit time integration scheme. This model includes the full complexity of the 3-D Earth (i.e., 3-D wave speed and density structure, 3-D crustal model, ellipticity, topography, and bathymetry).

Solving the inverse problem requires knowledge of the earthquake source, which means one must invert for the source model in the process of inverting for the material model. The system of equations is discretized by Akcelik et al. (2003) with Galerkin finite elements in space, and explicit central differences in time. The discretized system is solved using a multi-scale Gauss-Newton-conjugate gradient (CG) method built from components of the PETSc library. In every CG iteration, the state wave equation is solved forward in time for given material and fault properties and their Newton increments, and the adjoint wave equation is solved backward in time using the computed states and state increments. This is algorithmically analogous to the so-called 4DVAR data assimilation technique in numerical weather prediction. Because the majority of the work in the inverse algorithm involves repeated solution of forward or backward wave equations, and if that is made to scale, the only remaining

performance issue is algorithmic scalability of the inversion algorithm. Akcelik et al. (2003) have shown that the number of both linear and nonlinear iterations grows weakly with problem size, so this issue would seem to be addressed.

Seismic researchers surveyed for this report would like to execute the very largest seismic simulations at a throughput rate of one inverse or ten forward problems per month on the PCG system. Translating this requirement into system performance characteristics is tricky: the runtime required to complete a simulation is a complicated function of problem parameters. For the forward problem, it depends on the dimensions and material properties of the target regional or global model, highest resolved frequency, structure of the resulting adapted mesh, and number of processors. The inverse problem inherits the complexities of the forward solver, and adds to them the high degree of problem nonlinearity. This makes runtime prediction almost impossible. However, some current runtimes are available, and can be used to give rough estimates for future target problems.

Typical performance for a regional earthquake forward solver is documented in Akcelik et al. (2003). Memory-access patterns are pointer based and irregular, with locality typical of unstructured finite element meshes. This unstructured mesh application algorithm does not vectorize well. On microprocessors, the 100 million point forward problem code in Akcelik et al. (2003) sustained 500 Mflops/s on a single 2 GFLOPS/s peak Alpha EV68 processor (25% of peak). In parallel, the application achieved 80% scaling efficiency, and 1.21 TFLOPS sustained, on 3000 processors of the Pittsburgh AlphaServer system.

In contrast, the spectral element global seismic application of Komatitsch et al. (2003) does vectorize very well, achieving 99.3% vectorization of $5 \times 5 \times 5$ 3-D spectral elements, and approximately 33% of peak performance, on the Earth Simulator. One can conclude from this that different numerical methods, solving similar problems, can achieve high efficiency on drastically different computer architectures.

The parallel implementation of the unstructured grid communications in Akcelik et al. (2003) is pure MPI. Parallel load balance is achieved through mesh partitioning and is

straightforward to achieve since there is a direct, proportional relationship between elements on each processor and FLOPS executed. The algorithmic structure of communication in the forward model is as follows: at each time step and for each subdomain (i.e., mesh partition), volume matrix-vector products are computed, followed by communication of surface field values between neighboring partitions (and thus processors). The communication connectivity pattern between MPI processes follows the structure of the finite element mesh, and is therefore nearest-neighbor, but unstructured and thus quasi-local. The primary difference relative to a regular grid code, such as the SEM application in Komatitsch et al. (2003) is that this is a highly unstructured mesh, so the number of neighboring subdomains varies from processor to processor, and the message size can vary considerably as well. A global inner product is computed every linear conjugate gradient iteration of the inverse solver, but this is amortized over several full earthquake simulations and is thus negligible. The communication patterns change with every nonlinear iteration of the inverse problem, but the changes are amortized over multiple full earthquake simulations. So, the unstructured mesh's communication pattern can be regarded as fixed for long time periods.

The performance required of future calculations on the proposed computational facility is discussed here. The 100-million grid-point simulations in Akcelik et al. (2003) require approximately 40,000 time steps and 12 hours on 2048 Alpha EV68 processors of the system at PSC, running at approximately 1 TFLOPs sustained. This amounts to 43.2 peta floating point operations performed in the course of this calculation. A forward problem with ~1 billion elements would require roughly 20 times as much resource; perhaps 930 peta floating point operations. Completing such a simulation in three days would require a 3.4 TFLOPs sustained, a performance level readily achievable in the 2007 time frame. A 10-billion-element simulation of this kind would require another factor of 20 in sustained computational power, perhaps 62 TFLOPs sustained, to achieve the same scientific throughput. This is likely achievable within the budgetary parameters of this study (by 2010). By comparison, a 1944 processor SEM-based simulation of 60 minutes (50,000 time steps) of seismic wave propagation with 5.5 billion grid points (14.6 billion degrees of freedom), accurate down to a period of 5 seconds,

requires about 15 hours of wall-clock time on the Earth Simulator. Subsequent SEM-based seismic simulations with over twice as many degrees of freedom have already achieved over 10 TFLOPs on the Earth Simulator.

The output data sets for the forward problem are typically quite small. For example, for the global SEM seismic simulation, the output data consists of a time series of three degrees of freedom of displacement computed at each of 1000 seismic stations in the Global Seismographic Network (GSN). For 50,000 time steps, the size of the data set is 1.2 Gbyte.

The inverse problem is several orders of magnitude more difficult to solve than the associated forward problem, and therefore inverse problem sizes have typically lagged those of the forward problem. In particular, the inverse solver involves outer Newton (nonlinear) iterations combined with inner CG (linear) iterations. Each inner iteration requires as many forward/adjoint earthquake simulations as there are earthquake sources contributing data to the inverse problem. The inversion algorithm exhibits numbers of outer and inner iterations that are independent of mesh resolution, which is asymptotically optimal. Typical values of the inner and outer iteration count are 20-25 for each—although these values can vary widely depending on the choice of inversion fields and regularization parameter. Thus, the inverse problem typically requires the equivalent of approximately 1000 forward earthquake simulations. In summary, both the forward and inverse solver work complexities scale as $n^{4/3}$; however, the constant for the inverse solver is three orders of magnitude larger. Using this rule of thumb, one calculates that achieving one, 100-million element inverse problem solution per month would require a dedicated resource of approximately 17 TFLOPs sustained, again likely achievable for systems available in the 2007 time frame.

Thanks to spatial adaptivity and data structures that exploit the similarities in the structure of the octree-based hexahedral finite elements, the earthquake forward simulation code requires little memory relative to typical unstructured finite element implementations—its memory requirements are more like a regular grid finite difference code's: 12 double-precision floats per grid point, plus nearest-neighbor indices necessary to describe the unstructured mesh. For the

100-million element simulations, this amounts to ~10 GB memory total, and this scales linearly with problem size. The inverse solver formally requires the entire field-time history (i.e., $O(n^{4/3})$ memory) to compute the objective gradient. The time history is stored when enough memory is available; when it is not, a trade off of memory for work is made by algorithmic checkpointing, that is, reconstructing the time history from checkpointed (in memory) solutions as needed. In summary, memory is not an issue for the forward solver; for the inverse solver, it can be, but the algorithm can be adjusted to the available memory at the cost of an increase in work.

The inverse solver has modest I/O requirements: its goal is to update the (static) material field given observations at select points and/or to identify the seismic source. For the forward solver, on the other hand, the worst-case scenario is outputting the entire 4-D space-time field values. In this case, two (3-D) vector fields are output at every grid point every ~10 time steps (i.e., 6 floating point numbers per snapshot). For 100-million-grid-point simulations, this amounts to a total of approximately 10 TB. The output total scales like $n^{4/3}$, similar to the work complexity. Currently, each processor's computed field is written to a separate file.

However, because of I/O bottlenecks associated with large simulations, researchers are moving toward an environment in which all computations and data analysis are done on-line and in parallel, including mesh/model generation, wave propagation, and volume rendering. This will permit them to avoid writing and reading volume field information. Subsets of the computed data will still be written for subsequent analysis, including surface fields and field values at selected points. These will typically involve one to two orders of magnitude less output.

Interconnect Constraints from Seismic Simulations: The Quake Performance Analysis

Quake is a finite element application developed to predict ground motion in the San Fernando Valley of Southern California during earthquakes. The execution time of the Quake application under a variety of conditions has been studied in O'Hallaron et al. (1998). This seismic application also employs a three-dimensional unstructured mesh. Thus, the Quake seismic application is also dominated by a SMVP op-

eration that is repeated thousands of times, and the SMVP is the only operation, besides I/O, that requires the transfer of data between processors. The detailed characterizations and modeling of the Quake application's communications reveal that bisection bandwidth is not important for irregular finite element applications because the communications involved are quasi-local; bandwidth at each PE is what matters. O'Hallaron et al. (1998) also found that, while large SMVP problem indeed have reasonable computation/communication ratios, these ratios do not increase quickly with increasing problem size as they do for cubic problems like dense matrix multiply. Thus, one cannot rely on simply increasing the problem size to guarantee good computational efficiency. The irregular SMVP computations in Quake are quite inefficient, largely because of irregular memory reference patterns and because the data structures are too large to fit in cache. O'Hallaron et al. (1998) found that parallel computers needed a local communication to computation ratio about 1.5 bytes/sec/FLOPS sustained to run irregular SMVP codes with 90% efficiency. This is probably not as restrictive of a requirement in absolute terms as it appears because the sustained performance in Quake was so low to begin with. Perhaps the most troublesome conclusion of this study is that because the blocks transferred among MPI processes tend to be small even for large irregular applications, block latency costs cannot be amortized by large messages, and they conclude that achieving sub-microsecond communication latency will be necessary and need to be a central focus of future efforts to engineer effective communication networks and software for applications dominated by SMVP operations.

The interconnect performance requirements derived from SMVP applications can be compared to the results achieved by Akcelik et al. (2003) on the HP AlphaServer. The Quadrics Elan-3 interconnect has latencies in the 5.5 μ sec range and sustained, asymptotic local communication bandwidth of over 800 MB/sec. Thus, since the forward seismic application in Akcelik et al. (2003) sustained 500 MFLOPS, the sustained local bandwidth to FLOPS ratio is 1.6. This is in good agreement with the estimates for this ratio of 1.5, derived from the Quake performance model. However, the network latency requirements derived from the Quake analysis seem over restrictive compared to Akcelik et al. (2003): 5.5 μ sec provided by Quadrics Elan-3 seems quite adequate in that case.

Microprocessor systems available in 2007 will likely sustain up to two to four times the observed performance of the AlphaServer EV68's on a forward seismic problem. Thus, latencies and bandwidths should improve proportionately in order to ensure good scaling: a 1.5-2 μ sec network latency and 1.6-3.2 GB/sec of sustained local interconnect bandwidth would be required for SMVP-dominated applications in that time frame.

SPACE SCIENCE APPLICATIONS

The most computationally intensive space science calculations are those that attempt to unify a number of models into a single framework so that events that manifest themselves throughout the Sun-Earth environment can be simulated. One example of such a global framework for space-science applications is the Space-Weather Modeling Framework (SWMF). The SWMF combines numerical models of the Solar Corona (SC), Inner Heliosphere (IH), Solar Energetic Particles (SEP), Global Magnetosphere (GM), Inner Magnetosphere (IM), Radiation Belt (RB), Ionosphere Electrodynamics (IE), and Upper Atmosphere (UA) into a high-performance coupled model.

These models are coupled by the framework code, including a control module that determines the overall time stepping of the code, the parallel decomposition of the models, the initiation and termination of the model runs, and the saving of restart files of the models. This coupling involves code that determines when the coupling should occur, how it happens, grid interpolation, message passing between different components, and synchronization of the model runs to allow for a physically meaningful coupling.

SWMF uses a component architecture, shown in Figure 7, with each component created from a physics module by making some minimal required changes in the module and by adding two relatively small units of code: (1) a wrapper, which provides the standard interface to control the physics module; and (2) a coupling interface to perform the data exchange with other components. Both the wrapper and coupling interface are component interfaces constructed from building blocks provided by the framework.

The numerical methods used in the models vary, but generally are of four types: global plasmadynamic models, multi-component fluid models, generalized moment models, and particle-based kinetic models. The plasmadynamic and multi-component fluid models dominate the cost of the global calculations: the generalized moment models are typically computed on a lower-dimensional manifold (e.g., along magnetic field lines); the kinetic models, while inherently expensive, are typically used only in a very limited spatial region (e.g., to compute reconnection in the magnetopause).

The fluid and plasmadynamic models are based on solution-adaptive, domain-decomposition, high-resolution upwind finite-volume schemes, with a mix of implicit and explicit temporal integration. The explicit temporal integration is characterized by an excellent computation-to-communication ratio, leading to ideal or near ideal scaling up to thousands of processors across a broad range of architectures. The amount of system memory used in the explicit schemes is quite low, and bandwidth to disk is not an issue; the explicit scheme is almost entirely limited by sustained teraflops rate. The implicit temporal integration introduces two challenges: much higher memory demands (depending on the scheme used and the parameters chosen, roughly between one and two orders of magnitude more memory) and a degraded computation-to-communication ratio. On parallel machines with gigabit interconnects, the scaling starts to drop off above 256 processors.

The extent to which the implicit temporal integration can be used is a function of the computer architecture. The implicit scheme greatly enhances the stability of the numerical method, allowing much larger time steps. The Newton-Krylov-Schwarz technique employed in the implicit scheme scales very well on current architectures up to several hundred processors. Depending upon the petascale system's latency and bandwidth characteristics, there will be a performance trade-off between the larger time steps facilitated by the implicit scheme and the near-perfect scaling of the explicit scheme. The SWMF allows a mix of implicit and explicit blocks; the mix can be optimized to a given architecture.

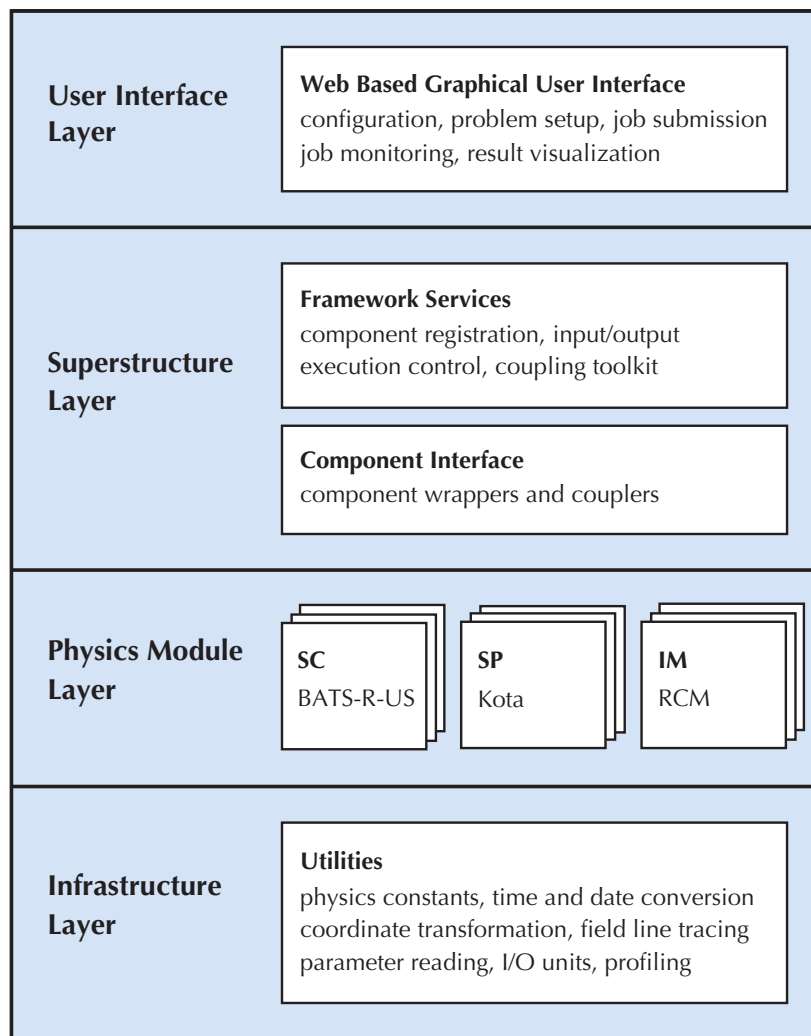


Figure 7. The layered hierarchy of SWMF.

A sense of the computational needs associated with these global space science models is given in Figure 8 and Table 9. The model can currently run faster than real time on the machines shown in the left part of the figure, at the temporal and spatial resolution shown in the table. This resolution is moderate; substantially more physics could be studied at higher levels of resolution. As can be seen from the scaling figure, the current state of the art is a global model that scales almost ideally up to 256 processors at this resolution.

For future needs, a reasonable goal is a coupled space-weather run that achieves a factor of four faster than real time (for predictive capability), with a resolution 1/8 that shown in Table 9. Due to the adaptive-grid capability, the 1/8 resolution does not lead to an 8 cubed multiplier on the number of cells, but “merely” to roughly an order of magnitude increase in number of cells. Scaling from current requirements, this gives a projected need of 52 TFLOPS sustained, with system memory requirements of 2 TB of system memory. Neither the mass storage rate nor the disk bandwidth rate is a limiting factor.

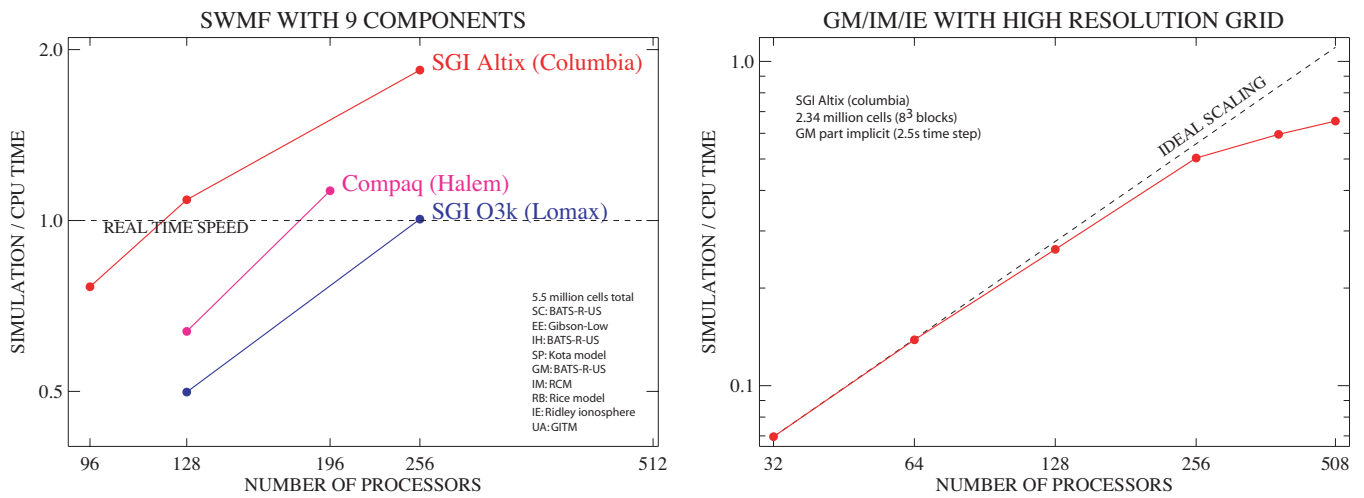


Figure 8. (Left) Parallel speed-up of the full SWMF simulating the Sun-Earth interaction. Note that with relatively low resolution, the SWMF runs faster than real time on >128 processors. (Right) Parallel speed-up of a medium-resolution, near-Earth space environment simulation. Note that this simulation cannot run in real time even on the top-of-the-line SGI Altix machine. About a factor of 100 higher resolution is needed for realistic space-weather simulations, which requires petascale computational resources.

Table 9: Spatial and Temporal Resolutions in the Test Run.

Component	Version	# of Cells	Smallest Cell	Var/Cell	Time Step
IH	BATS-R-US	2,500,000	1 R_S	8	60.0 s
SC	BATS-R-US	1,400,000	1/40 R_S	9	0.4 s
GM	BATS-R-US	1,300,000	1/4 R_E	8	4.0 s
UA	GITM	64,800	5° x 5° x 2 km	30	10.0 s
IE	Ridley	33,000	1.4° x 1.4°	1	—
SEP	Kota	10,000	0.1 R_S x 10°	150	varies
IM	RCM	3,800	7.5° x 0.5°	150	5.0 s
RB	RRBM	3,800	7.5° x 0.5°	1	60.0 s

FACILITIES

In this section, a breakdown of the facilities and services necessary to house and support a PCG, including infrastructure and manpower, will be discussed. The goal of the overall system is to maximize its scientific impact on the geoscience community over the longest possible time. To achieve this goal, the system must be deployed within a balanced and sustainable software and support infrastructure. To deploy a one-shot, hardware-oriented, stand-alone system would be counterproductive and wasteful.

The type, size, and costs of the proposed facilities are derived from an end-to-end design of a system that is both driven by the scientific requirements outlined in previous sections and constrained by the projected technical and economic realities of the next five to seven years. It is also clear that the system must begin successfully delivering breakthrough scientific results from the beginning and provide continuity as well as increased capability throughout successive generations of supercomputing systems. From a technical perspective, petascale systems between 2007 and 2012 are almost certain to contain tens to hundreds of thousands of processors. Thus, achieving breakthrough science on such systems will require extensive preparation work on applications, system software, and analysis tools. *The success of major geoscience initiatives on a petascale system depends critically on creating and sustaining the software environment capable of exploiting it.*

The three main options for building the proposed petascale system are:

- Construct one very large capability compute system at a single facility.
- Integrate a set of smaller, yet substantial, systems into a single petascale computational system. These systems may be collocated or, likely, distributed across a number of sites and integrated as a computational grid.
- Construct a very limited, very tightly directed set of mid-range resources that support and focus on the deployment of a single large system at a single facility.

The first option for a single petascale system has the major advantage of allowing much larger single jobs to be run relative to a distributed facility model, but it also has the primary disadvantages of a more substantial initial cost, a very long incubation period, and more limited upgrade possibilities over the projected initial six-year collaborative operation. This concentration of effort into a single system has been successful for the Earth Simulator Center in Japan. However, typical large-scale computing systems have about a three-year life span before the components become obsolete relative to the latest hardware, necessitating the periodic equivalent of the Apollo Moon program to deploy successor systems. A possible compromise plan, consisting of a phased upgrade to produce a single large system in a multi-step process through anything other than a “forklift” upgrade invariably, results in technological mix of new and old technologies, which results in decreased efficiency for the newer, more capable technologies. This type of phased plan has, of necessity, been performed at existing supercomputing centers and can be made to work, but is less than optimal for users.

The second approach has the serious disadvantage that, without a single, monolithic resource, very large jobs cannot be scheduled. However, within this deployment model, more than one type of computational platform architecture can be provided to users and evaluated. This is potentially a significant advantage; experience has shown that many applications perform well, or more importantly, exceptionally badly on particular architectures. With a variety of systems, PCG users will have more flexibility so that overall the most efficient and effective use of the provided systems is achieved. Risks related to relying on a single vendor-supplier are dissipated. The growing maturity of grid and web services technologies, as exemplified by the NSF TeraGrid and Supercomputing Centers programs, can be leveraged for the PCG and make this system model increasingly practical.

The third approach, which is recommended, integrates the most successful features of the first two strategies, but ultimately commits to supporting very large, single-system jobs. Yet, it provides immediate and much-needed access to a set of mid-range resources usable by the entire geoscience community. These initial systems will integrate and leverage existing facilities, cyberinfrastructure, and domain-specific human capital; fulfill the high-end system role during the initial phase; and lay the groundwork for the large-capability systems to follow in subsequent phases. This strategy provides important lead time for the construction of the necessary software, facilities, and support infrastructure while enabling scientific efforts in application development, as well as scale up and testing prior to the installation of the large, leadership-class systems.

In summary, with the increased development of grid services and improved networking infrastructure, such as the National Lambda-Rail coming online, a distributed PCG becomes feasible. The most cost-effective configuration of such a grid-based collaboratory is one in which one large-capability system is located at a central facility and a small set of mid-range satellite systems are deployed in strategic locations and dedicated to geoscience usage. Constructing the satellite systems first would leverage existing resources and would provide lead time for development of the necessary cyberinfrastructure for petascale computing. Such a facility would best match the needs of the geoscience community by providing mid-range computational, data analysis, and data management systems early on, while the grid-based collaboratory is being brought into existence. The result of this strategy will be that the large-capability system will be fully and effectively utilized for the largest and most important computations as soon as it is deployed.

COLLABORATORY PROJECT DESCRIPTION

An outline of a six-year PCG project for the period 2007-2012 is presented in this subsection. The project is envisioned to unfold in two phases: the first phase in 2007 in which initial equipment is deployed to establish the collaboratory and its distributed aspects, and a second phase in 2010 in which a single very large computing system with a peak speed at or near 1 PFLOPS is deployed. The motivation for this plan is

derived from the application analyses of the previous section. These analyses clearly indicate that many applications exist that have immediate requirements for sustained performance levels in the range of 5-10 TFLOPS. It is also clear that many longer-range requirements for achieving performance in the range of 50-100 TFLOPS will likely require significant scientific, algorithmic, and software development efforts to realize. It is also clear that substantial effort will be required to design and construct the supporting cyberinfrastructure and software tool chain to support the hundreds of petabytes of data that will be produced by a petascale facility. Another attractive feature of procuring multiple systems in 2007 is that it allows some architectural diversity and system intercomparisons to be performed.

It is therefore recommended that the initial deployment in 2007 be of two to three “mid-sized” systems (each 50-100 TFLOPS peak speed), along with the requisite supporting data storage data analysis and visualization systems. The initial focus of the first two years of the collaboratory should be to bring staff on line, build the petascale data center, create necessary grid infrastructure, and begin production supercomputing with these initial mid-sized supercomputing systems.

Around 2010, a technological refresh will be in order, and the staff, facilities, and cyberinfrastructure to support a petascale computer will have been constructed; a 1 PFLOPS peak system will then be procured and installed at the PCG facility. This system will be in a position, at that time, to enable breakthrough geoscience at 100 TFLOPS sustained and beyond.

PCG COST MODEL

A cost model has also been developed that attempts to capture the costs of the collaboratory systems needed to support the plan outlined above, in particular, the costs of supercomputer and related high-performance disk systems; the facility to house the petascale computer; the data analysis, visualization, and data archive systems; and the annual operating costs, including such factors as utility costs and salaries for the collaboratory staff. A constant, annual inflation rate of 3% is assumed throughout.

Two important scale factors influence the model of the petascale system: the future cost of a peak TFLOPS of computing power, which is dropping roughly by a factor of three every two years; and the fuel efficiency of future supercomputers as measured in peak GFLOPS/Watt. The former determines the overall cost of the supercomputing system; the latter determines the size and cost of the facility for which the costs of power distribution and cooling equipment are critical determiners. The cost model assumes a current industry average cost performance projection for “capable” computers of \$1M per peak TFLOPS in 2006. The fuel efficiency of systems varies more widely and is therefore more difficult to estimate, as Figure 9 illustrates for contemporary processors. The values in Figure 9 tend to overestimate the fuel efficiency of the overall supercomputing system, as they neglect the

contributions from system memory and disk components. To compensate, a 2006 estimated value of 0.1 peak GFLOPS/Watt is used for the cost model. The peak fuel efficiency in the cost model is assumed, on a per socket basis, to track Moore’s Law (i.e., to double every 18 months). This is based on the observation that the power per socket is the real limiting factor, and that Moore’s Law will be increasingly maintained in future chips by increasing the number of processor cores on the chip. This trend is evident in the planned road maps of major chip suppliers, including AMD and Intel.

It is important to recognize that the overall cost effectiveness of a computing system is regulated by the sustained performance, generally expressed in terms of a fraction of peak. Ignoring this one might conclude, falsely, that a system with

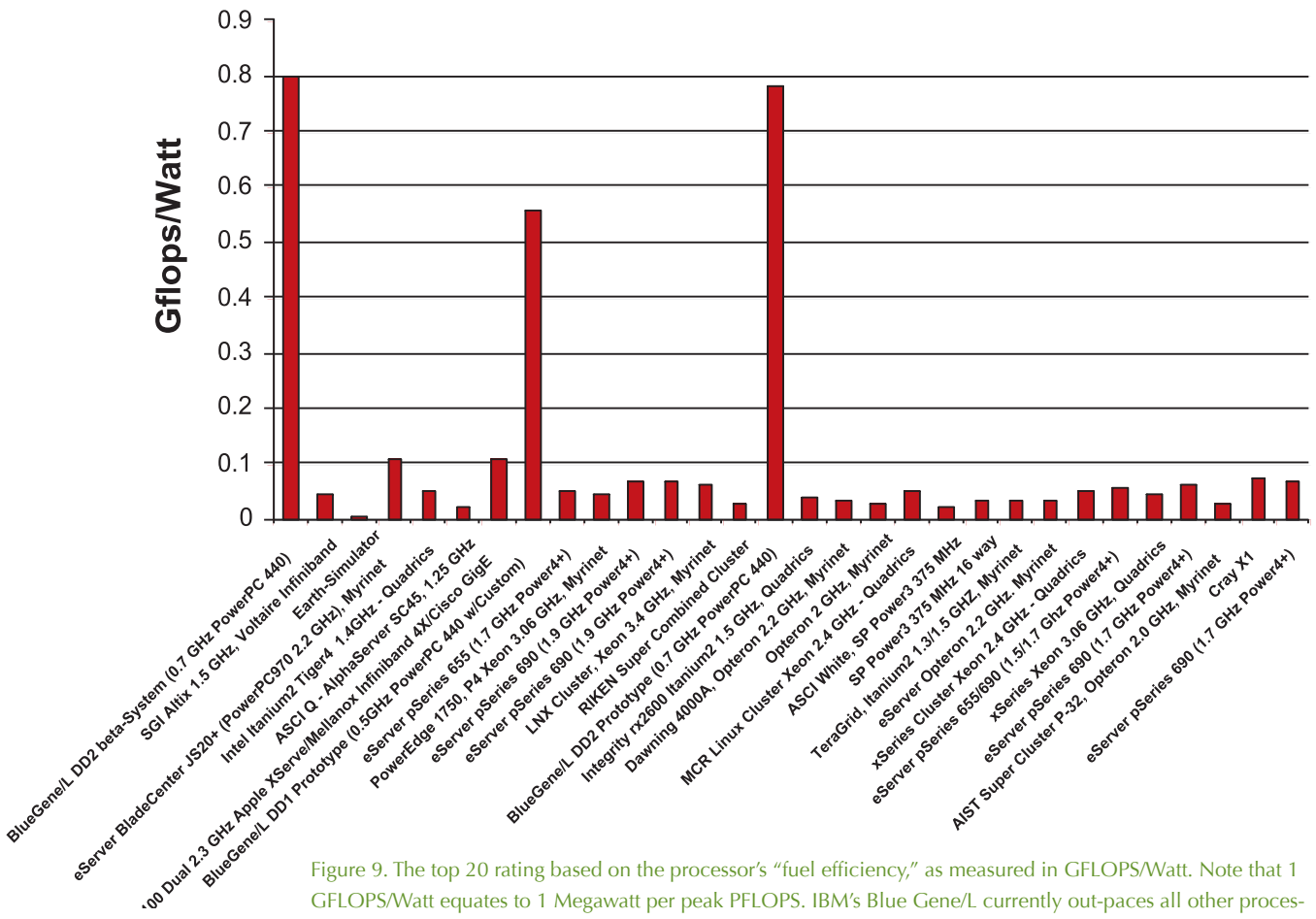


Figure 9. The top 20 rating based on the processor’s “fuel efficiency,” as measured in GFLOPS/Watt. Note that 1 GFLOPS/Watt equates to 1 Megawatt per peak PFLOPS. IBM’s Blue Gene/L currently out-paces all other processors by this metric. The effect is diluted by the power consumption of disk and memory and, of course, the fraction of peak sustained. (J. Dongarra, University of Tennessee)

a higher cost per peak TFLOPS is less cost effective. This issue is especially critical when one looks at capability systems, which are generally more expensive because of factors such as custom hardware components or other resource enhancements to system interconnect bandwidth local memory system performance characteristics. Unfortunately, as was seen in the preceding sections, values for sustained performance vary widely, not only among computer architectures and across applications, but also with the number of processors and with time. This latter temporal effect is directly attributable to the increasing distance of main memory from the CPU, as measured in CPU clock cycles, and is an underlying technological trend that quietly dominates most of the computer architecture decisions facing the industry for the immediate future. Thus, the most important factor, cost per sustained TFLOPS, is the most difficult to measure, determine, or predict.

Looking across the benchmark values obtained from geofluid applications such as WRF, CAM, and POP on various emerging computers systems, 10% of peak appears to be a reasonable value for sustained fraction of peak in such geofluid codes. Certainly some applications, such as the global seismology application of Komatitsch et al. (2003), achieve much higher fractions of peak (33%), particularly on vector systems. However, as noted before, these systems generally cost more, so the variations in terms of sustained performance tend to average out.

PCG DISK SUBSYSTEM

Application requirements for the petascale system suggest a minimum aggregate sustained bandwidth to disk of ~20-30 GB/sec. Given current (2005) disk capacities and the number of disks required to saturate controllers, it would require close to a petabyte of SATA disk to achieve this bandwidth today. A terabyte of such disk currently occupies about 1U of space in a standard 19" rack, so a petabyte currently would require about 20 racks. Because disk capacity is growing faster than I/O bandwidth to disk, it will likely require more total disk capacity to achieve the same aggregate bandwidth in the future. The rack density will likely increase, however, because disk densities will increase, perhaps a factor of four by 2010,

so a reasonable estimate is that the number of racks would roughly stay constant. The petascale system would have perhaps as much as 4 PB of disk.

PCG MASS STORAGE SYSTEMS

The cost of the collaboratory's mass storage systems scales with the amount of data produced by the petascale supercomputer. Using the NCAR mass storage rule of thumb that geoscientists archive 30 bytes for every MFLOPS performed, one computes an archival rate of approximately 1 PB/year/TFLOPS sustained. This empirical ratio has been shown to fall close to the stated future requirements of scientists in the Application Analysis section of this report. Using this figure, a PFLOPS system, operating at 100 TFLOPS sustained (10% of peak), will be expected to produce ~100 PB/year of data. The cost model specifically takes into account the cost of tape media, storage silos, and the necessary tape drives to support the archival rate. The price performance of robots, tape drives, and tape densities are assumed to improve according to Moore's Law over the period in question, starting from an estimated archival cost of about \$445,000/PB in 2007. A tape archive with a footprint of 225 square feet holding approximately 3-5 PB of data is assumed to be available in 2010.

PCG DATA ANALYSIS AND VISUALIZATION SYSTEMS

A key component of the collaboratory strategy should be to provision substantial data analysis and visualization systems. These systems are essential because they are the critical means by which scientific knowledge is extracted from the torrent of data that will be produced by the petascale system. A survey of data analysis and visualization systems at Lawrence Livermore National Laboratory, the National Center for Atmospheric Research, and the National Center for Supercomputing Applications indicates as much as 10% of the overall computing resource should be invested in these specialized systems. A model of operation that appears particularly promising and cost effective for data analysis and visualization is one in which a parallel, multi-TB shared-memory supercomputer (such as an SGI Altix) is used for data analysis and is connected, via shared, high-performance file system, to a commodity visualization cluster. This allows

data to be analyzed and rendered into visualization frames that can be distributed interactively to remote users via Gigabit Ethernet.

NETWORKING AND PCG GRID SERVICES

Remote researchers will perform much of the computational work run at the collaboratory, so it is critical that the appropriate wide area 10 Gigabit networking infrastructure and the associated grid software stack be deployed to deliver high bandwidth, secure, production-quality, distributed grid services to the PCG. With the complexity of geoscience applications increasing and the amount of data that must be processed growing accordingly, effective methods for cataloging and federating data holdings must be found, and efficient techniques for transferring data between collaboratory nodes must be created. Workflow systems for job submission and control must be put in place that are relatively transparent and easy to use. Some of the necessary grid services are a central job submission/monitor “meta-scheduler” utility, grid portal application gateways, and security and authentication mechanisms. These grid services should be phased in as they mature. The NSF TeraGrid is pioneering many of these capabilities and is already putting much of the networking infrastructure required by the PCG in place. This study assumes that much of this grid infrastructure can be leveraged in the construction and deployment of the PCG.

DATA CENTER ISSUES

Three options were considered for the data center housing the PCG supercomputer: expanding an existing facility, commissioning new construction, or acquiring a leased facility. Existing facilities can be used to house the grid-based elements of the collaboratory, such as the mid-range systems to be deployed in the first phase of the project, as well as the collaboratory’s data storage, visualization, and data analysis components. Retrofitting an existing data center to accommodate the very large petascale supercomputer system is not generally recommended, particularly if the center under consideration is old. Modern systems have very high power densities (approaching 25-30 KW per 19” rack), and older data centers, built in an era of cooler computing systems, were not

constructed to handle such power densities. Even if the existing center is relatively new, expansion may encounter infrastructure problems specific to the site under consideration.

Construction of a new petascale facility offers its own set of challenges. Perhaps the most serious is the long lead time required to complete building construction, estimated to be at least two to three years. New construction also implies a twenty to thirty-year commitment to the facility on the part of NSF, which may be a disadvantage, particularly if computer technology or NSF priorities change. According to the Uptime Institute, new construction costs for a data center are dominated by the costs of power/cooling infrastructure and floor space required. The latter is estimated to cost \$255 per square foot; the former at \$12,000 per kilowatt. The calculations of power requirements for the facility include 50% overhead for cooling and other mechanical systems.

These figures relate to a moderately redundant, so-called tier 2+, data center. A tier 2+ data center is designed in accordance with industry best practices with sufficient redundancy to cost-effectively minimize unplanned downtime and has sufficient pre-installed infrastructure to enable data center expansion or upgrade to higher levels of availability, should that become desirable at a later date. Minimizing disruptions of power and cooling to the data center’s equipment is a highly desirable feature, particularly important to modern systems with high power densities. Following industry guidelines, the mean power density design of the facility is assumed to be 150 Watts/sq ft.

Another possibility is engaging one of many companies that build and lease back large data centers. Leasing has several advantages, primarily the lower cost in the context of a six-year project, the generally shorter lead-time required for a lease, and the flexibility it provides. NSF could terminate the lease at the end of the lease period, or adjust it as necessary, and would not be locked into owning a fixed bricks-and-mortar solution at the end. Some lease solutions incorporate management services such as networking, utilities, and physical site security. Lease costs are scaled by the power required and are broken down into one-time setup and annual operating costs. Typical values for these are \$85,000/MW for setup and \$1.62M/Megawatt/year.

The cost advantages of a lease over a six-year period are easy to demonstrate. Consider a tier 2+ data center that can accommodate a 2 Megawatt data center. Given the assumptions of the cost model, building such a data center is estimated to cost \$27.4M. However, the five-year lease costs of the data center are \$16.4M. Obviously, a longer commitment on the part of NSF to the PCG facility would militate towards new construction as the most cost-effective solution.

STAFF COSTS

Ground-breaking scientific results are the justification for the PCG and can only be enabled in an environment in which scientific productivity is paramount. Creating a science-driven set of user services requires the ability work very closely with the science teams to facilitate their ability to port, optimize, and efficiently use the full range and scale of computational, storage, and visualization resources in the PCG as it grows to incorporate and leverage multiple sites and resources and evolves into the premier geosciences community platform over the life of the project.

To accomplish this, the collaboratory staff must be tasked to:

- Operate a robust, reliable, secure, predictable, and stable production hardware and software environment.
- Rapidly identify and solve problems whether they are local or affect multiple resources or sites.
- Provide direct support for scientific teams through embedded applications and system performance engineers.
- Provide high-end visualization support services.
- Track and resolve long-term issues and provide strong planning support for the carefully directed growth of the PCG.

Successful user support depends on a multi-level support model:

- Level 1: quick, accurate triage of reported problems across the PCG.
- Level 2: applications support in porting, tuning, and making efficient use of the PCG resources.
- Level 3: long-term and critical-situation response with access to knowledgeable resources to resolve complex issues.

An attempt has been made to estimate the personnel costs of the collaboratory, including the administrative staff. The analysis of the staffing requirement of each support level is presented in the discussion that follows, and Table 10 summarizes the costs, which can be seen to amount to about \$13.5M in 2007 dollars.

Level 1: Operational Support

A centralized 7 x 24 x 365 operations center is critical to the successful operation of the PCG. This center will monitor the status and security of all systems, and will provide the first point of contact for all problem reports related to PCG operations. Due to the distributed nature of the PCG, these problems are inherently more difficult to track and resolve, and will require substantial interaction and cooperation with the satellite data centers in order to operate effectively. Two front-line user consultants will be available around the clock to immediately answer user questions, resolve system-related problems, or escalate problems further with the application support or systems teams, as appropriate.

The operation of the petascale facility itself, because of its size and complexity, is estimated to require the staffing of three shifts of four operators per shift. The costs related to the 18 operations staff of the collaboratory are shown in Table 10.

Finally, the availability of up-to-date and easy-to-find online documentation is a key component of the Level 1 support plan. By providing extensive user guides and lists of frequently asked questions, many low-level user support issues can be resolved without requiring interaction with a consultant. It is recommended that the PCG leverage as much pre-existing documentation resources at other centers as possible.

Level 2: Applications Support

Depth and breadth of applications support talent is essential to the successful use of the collaboratory. To this end, the collaboratory's 21 applications support staff should be composed of both scientifically embedded and centrally pooled experts with geoscience domain experience, including applications support, performance tuning, visualization support, and ex-

Table 10: Suggested breakdown of the Petascale Collaboratory Staff

Position Description	Number	Est. annual cost (2007 dollars) (including salary, benefits and overhead)
Level 1 Support/Help desk		
Help Desk Consultants	6	\$0.8M
Operations staff	12	\$1.35M
Level 1 Support Staff Subtotal	18	\$2.15M
Level 2 Support/Applications		
Applications Engineers	8	\$1.8M
System Performance Engineers	4	\$0.9M
Data Product Consulting	4	\$0.72M
Visualization Support	5	\$1.13M
Level 2 Support Staff Subtotal	21	\$4.55M
Level 3 Support/Systems		
Supercomputer System Admins/Engineers	7	\$1.26M
Storage System Administrators	3	\$0.54M
Mass Storage Administrators/Engineers	3	\$0.54M
Grid/Web Systems	4	\$0.81M
Visualization Systems Engineers	3	\$0.54M
Network Engineers	6	\$1.08M
Level 3 Support Staff Subtotal	26	\$4.77M
Collaboratory Administrative Staff		
Director/Managers	5	\$1.53M
Administrative Assistance	5	\$0.45M
Collaboratory Administrative Subtotal	10	\$1.98M
Total Estimated Staff Costs	75	\$13.45M

expertise with geoscience data products. Front-line application engineers will be available to the community to directly assist research teams in porting and profiling applications on various collaboratory platforms. The applications engineers will be supported by, and will have direct access to, the internal pool of experts in performance tuning, data products and visualization systems. Performance experts will ensure that key

codes will perform and scale well on collaboratory systems. It is essential for reasons of encouraging information exchange and overall efficiency that the collaboratory's Level 2 applications support team be well integrated with similar personnel within the pre-existing data centers that form the collaboratory's nodes, as well as key scientific research teams.

Level 3: System Support

Petascale systems composed of tens to hundreds of thousands of processors will be more complex to maintain, and will experience failures more frequently. Massive amounts of data will be produced and moved and a complex hierarchy of data storage systems and high-performance networks will have to be maintained. The systems support staff of the collaboratory is therefore expected form the largest individual unit in the organization with an estimated 26 staff in six different functions, including supercomputing, data storage, mass storage, visualization, web/grid, and network systems support. Table 10 shows the suggested breakdown of Level 3 support functions.

TOTAL COST: A PCG PROJECT SCENARIO

Using all of the information about costs and technology trends from the previous sections, a cost breakdown of the two-phase project scenario can be constructed in which mid-range systems with 200 TFLOPS peak are purchased and

deployed in phase one in 2007, followed by the acquisition of a 1 PFLOPS peak system in 2010. The petascale system is projected to require 2.4 Megawatts of electrical power and occupy 15,800 square feet of floor space. It will likely consist of roughly 100 cabinets. A multiple petabyte high-speed disk subsystem will require about 20 additional cabinets. An amount of space (9,000-15,000 square feet) will be required for a mass storage system archiving hundreds of PB of data in roughly 20-30 silos. The annual electrical power bill for the petascale system and related mechanical equipment is projected to be \$1.8M. This breakdown is presented in Figure 10. The total cost of the project is estimated to be \$390M, or approximately \$65M/year. Of this amount, a total of \$234M (60%) is spent on computing equipment. Projected mass storage costs are \$38M. The money spent on computers is nearly equally divided between the 2007 and 2010 procurements. Operating costs, composed of facility lease costs, utility bills, and staff payroll account for \$118M. The average annual operating cost of the collaboratory is thus \$20M/year.

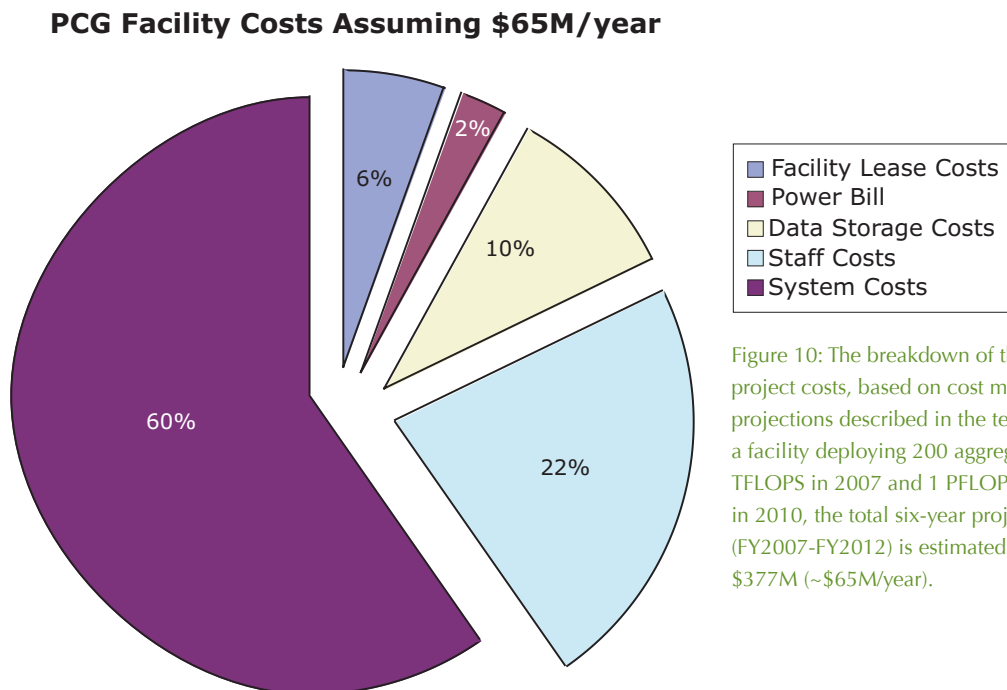


Figure 10: The breakdown of the PCG project costs, based on cost model projections described in the text. For a facility deploying 200 aggregate TFLOPS in 2007 and 1 PFLOPS peak in 2010, the total six-year project cost (FY2007-FY2012) is estimated to be \$377M (~\$65M/year).

REFERENCES

- Ad Hoc Committee and Technical Working Group for a Petascale Collaboratory for the Geosciences. 2005. *Establishing a Petascale Collaboratory for the Geosciences: Scientific Frontiers*. A Report to the Geosciences Community. UCAR/JOSS. 80 pp.
- Akcelik, V., J. Bielak, G. Biros, I. Epanomeritakis, A. Fernandez, O. Ghattas, E.J. Kim, J. Lopez, D. O'Hallaron, T. Tu, and J. Urbanic, 2003. High-resolution forward and inverse earthquake modeling on terascale computers. *Proceedings of ACM/IEEE SC2003*, Phoenix, Arizona. [Online] Available from: <http://www-2.cs.cmu.edu/~oghattas/papers/sc2003.pdf>.
- Bailey, D.H. and A. Snively. In press. Performance modeling: Understanding the present and predicting the future. In: *Combinatorial Scientific Computing*, Electronic Transactions on Numerical Analysis (<http://etna.mcs.kent.edu/>), Proceedings of SIAM PP04. [Online] Available at <http://crd.lbl.gov/~dhbailey/dhbpapers/dhb-perf-model.pdf>.
- Colella, P., T.H. Dunning, Jr., W.D. Gropp, and D.E. Keyes, eds., 2003. A Science-Based Case for Large-Scale Simulation: Volume 1, Office of Science, U.S. Department of Energy, July 30, 2003, 51 pp. [Online] Available at <http://www.cs.odu.edu/~keyes/scales/>.
- Committee on the Future of Supercomputing. 2004. Getting up to Speed: The Future of Supercomputing. National Research Council, Computer Science and Telecommunications Research Board, National Academy Press, Washington, D.C., 308 pp. [Online] Available at: <http://www.nap.edu/books/0309095026/html/>.
- Cotton, W.R., R.A. Pielke Sr., R.L. Walko, G.E. Liston, C. Tremback, H. Jiang, R.L. McAnelly, J.Y. Harrington, M.E. Nicholls, G.G. Carrio, and J.P. McFadden. 2003. RAMS 2001: Current status and future directions. *Meteorology and Atmospheric Physics* 82:5–29.
- Davis, L.P. C.J. Henry, R.L. Campbell, W. Ward, A. Snively, and L. Carington. 2004. Making HPC System Acquisition Decisions is an HPC Application. Masterworks at Supercomputing 2004, Pittsburgh, PA. From a Powerpoint Presentation [Online] Available at: <http://www.sdsc.edu/pmac/Papers/papers.html>.
- ESPOP. 2002. Earth Simulator 2002 Annual Report, 1/10th degree model. [Online] Available at: <http://210.189.77.208/English/result/result.htm>.
- Grabowski, W.W., 2001: Coupling cloud processes with the large-scale dynamics using the Cloud-resolving Convection Parameterization (CRCP). *J. Atmos. Sci.*, 58, 978–997.
- Grabowski, W.W., and P.K. Smolarkiewicz, 1999: CRCP: A Cloud Resolving Convection Parameterization for Modeling the Tropical Convecting Atmosphere. *Predictability: Quantifying Uncertainty in Models of Complex Phenomena*, 18th Annual Conference of the Center for Nonlinear Studies, Los Alamos, NM, USA, 11–15 May 1998. 133, 171–178.
- Graham, S.L., M. Snir, and C.A. Patterson, eds. 2004. *Getting Up to Speed: The Future of Supercomputing*. Committee on the Future of Supercomputing, Computer Science and Telecommunications Board Commission on Physical Sciences, Mathematics, and Applications. National Research Council. National Academy Press. Washington, D.C. 308 pp. [Online] Available at: <http://www.nap.edu/books/0309095026/html/>.
- Komatitsch, D., S. Tsuboi, J. Chen, and J. Tromp. 2003. A 14.6 billion degrees of freedom, 5 teraflops, 2.5 terabyte earthquake simulation on the Earth Simulator. *Proceedings of the Supercomputing SC2003 Conference*, published on CD-ROM and at <http://www.sc-conference.org/sc2003>.
- National Science and Technology Council, Committee on Technology. 2004. *Federal Plan for High-End Computing, Report of the High-End Computing revitalization Task Force*, May 10, 2004, [Second printing July 2004] 72 pp. [Online] Available at: <http://www.itrd.gov/pubs/>.
- O'Hallaron, D.R., J.R. Shewchuk, and T. Gross. 1998. Architectural implications of a family of irregular applications. Pp. 80–89. In: *Proceedings of the Fourth International Symposium on High Performance Computer Architecture*. IEEE. [Online] Available from: <http://csdl.computer.org/comp/proceedings/hpca/1998/8323/00/8323toc.htm>.
- Pielke, R.A., W.R. Cotton, R.L. Walko, C.J. Tremback, M.E. Nicholls, M.D. Moran, D.A. Wesley, T.J. Lee, and J.H. Copeland. 1992. A comprehensive meteorological monitoring system—RAMS. *Meteorology and Atmospheric Physics*, 49:69–91.
- President's Information Technology Advisory Committee. 2005. *Computational Science: Ensuring America's Competitiveness*. Report to the President. National Coordination Office for Information Technology Research and Development, Arlington, Virginia, 104 pp. [Online] Available at: <http://www.nitrd.gov/>.
- Smolarkiewicz, P.K., L.G. Margolin, and A.A., Wyszogrodzki, 2001: A class of nonhydrostatic global models. *J. Atmos. Sci.*, 58, 349–364.
- Snively, A., X. Gao, C.B. Lee, N. Wolter, J. Labarta, J. Gimenez, and P. Jones. 2004. Performance Modeling of HPC Applications. Pp. 777–784. In: *Parallel Computing: Software Technology, Algorithms, Architectures and Applications*, G.R. Joubert, W.E. Nagel, F.J. Peters, W.V. Walter, Eds., PARCO 2003, Dresden, Germany. Advances in Parallel Computing 13 Elsevier, The Netherlands. ISBN 0-444-51689-1.
- Winter, C.L., E.S. Springer, K. Costigan, P. Fasel, S. Mniewski, G. Zyvoloski. 2004. Virtual watersheds: Simulating the water balance of the Rio Grande Basin. *IEEE Computing in Science and Engineering* 6 (3):18–26.
- Winters, K., J. MacKinnon, and B. Mills. 2003. A spectral model for process studies of rotating, density-stratified flows. *Journal of Atmospheric and Oceanic Technology* 21(1):69–94.
- Worley, P.H., and T.H. Dunigan, Jr. 2003. Early performance evaluation of the Cray X1 at Oak Ridge National Laboratory. *Proceedings of the 45th Cray User Group Conference*, Columbus, Ohio, May 12–16, 2003. [Online] Available at: <http://www.csm.ornl.gov/~worley/papers/recent.html>.
- Ziemiński, M.Z., W.W. Grabowski, and M.W. Moncrieff. submitted. Explicit convection over the Western Pacific warm pool in the Community Atmospheric Model. *Journal of Climate*.
- Zyvoloski, G.A., B.A. Robinson, Z.V. Dash, and L.L. Trease. 1997. Summary of models and methods for FEHM application—A finite element heat and mass transfer code. *Los Alamos National Laboratory Report LA-13307-MS*, Los Alamos, NM.

APPENDIX 1

CHRONOLOGY AND METHODS

As detailed in the body of this report, recent assessments of cyberinfrastructure requirements by each of the disciplines represented by the NSF Directorate for Geosciences (OITI Steering Committee, 2002; Ad Hoc Committee on Cyberinfrastructure Research, Development and Education in the Atmospheric Sciences, 2004; Cohen, 2005) have indicated that geosciences research in the United States is being impeded by an acute shortage of high-end capability class computing resources. In late summer 2004, an ad hoc committee, working on behalf of the atmospheric, solid Earth, ocean, and space science communities, with the encouragement of the National Science Foundation, was formed to develop a strategy to address this gap between the scientific requirements for, and the availability of, high-end computational resources.

The committee addressed two tasks. The first was to articulate those research problems in geosciences where progress depends on expanded access to capability-class computing. Building on the discipline-specific reports cited above, the committee conducted a survey of a cross section of the computational geosciences research community to assess the scientific requirements for high-end computing resources over the coming decade. A copy of the letter requesting input is included as Appendix 2 of *Establishing a Petascale Collaboratory for the Geosciences: Scientific Frontiers* (Ad Hoc Committee and Technical Working Group for a Petascale Collaboratory for the Geosciences, 2005).

Community forums conducted at the fall meeting of the American Geophysical Union in December 2004, and the annual meeting of the American Meteorological Society in January 2005 provided additional opportunities for community input into this process.

The second task was to develop a technical and budgetary prospectus for the deployment of a petascale computing resource to the geosciences community, using the period 2007 to 2012 as the proposed planning window (Appendix 2). This period was chosen taking into consideration the current pent-up demand for high-end computing resources (breakthrough research can be accomplished with petascale capability now), the technology trends for high-performance computing systems (it would be less expensive later), and the timetable associated with the budgetary mechanism that we expect to be pursued in funding this activity, namely an NSF MREFC proposal. The technical and budgetary prospectus is not meant to be detailed at the level that it specifies particular hardware components, but rather to assess the technical feasibility and to estimate the cost of deploying a highly productive, petascale computing environment to the geosciences community in the 2007-2012 time frame. To complete this task, a technical working group was formed with representatives from existing NSF supercomputing centers, national labs, and the academic computer science community.

APPENDIX 2

CHARGE TO TECHNICAL WORKING GROUP

Introduction

A consistent message is emerging from a number of reports on cyberinfrastructure that geosciences research in the United States is being impeded by lack of high-end computing resources. We represent a committee working on behalf of the atmospheric, oceanic, and Earth science communities, with the encouragement of the National Science Foundation's Directorate for Geosciences to address this concern.

We are preparing the way for an effort to establish a Petascale Collaboratory for the Geosciences. The mission will be to:

Provide leadership-class computational resources that will make it possible to address, and minimize the time to solution of, the most challenging large-scale problems facing the geosciences, thereby substantially advancing our understanding and predictive capability of the components of the Earth system and their interaction with one another and with human society.

Charge to the Technical Working Group

The immediate tasks at hand are twofold: first to develop a science plan clearly articulating the potential scientific pay-offs of such a resource, and second to prepare a technical and budgetary prospectus for the collaboratory. The charge to the technical working group is to provide the material for the latter.

The prospectus we are seeking from the technical working group should be a document describing how to build a petascale collaboratory within five years, using available technology. The document need not be detailed to the level that it could serve directly as the basis of a procurement, but should be detailed enough to indicate what is technically fea-

sible and to estimate the budget required. It might suggest a small number of options that could be evaluated against the science plan.

Additional Background

To set some boundary conditions, the funding mechanism we wish to pursue could potentially provide a budget on the order of \$100 to \$500 million, an initial award period of 5 to 6 years, and a start up date no earlier than 2007. Our target for completing these documents is the end of calendar year 2004.

The goal of the collaboratory is not to incrementally increase the resolution of existing modeling studies or data analyses. Rather, it is to make possible fundamental advances in computational geosciences by permitting the inclusion of additional physical, chemical, or biological mechanisms by providing access to previously unexplored parameter regimes, or by dramatically expanding the scale content and "mechanism content" of simulations of complex systems. The mission includes plans for algorithmic reformulation, coding, and optimization; data pre- and post-processing, and analysis and visualization.

Some of the characteristics of the collaboratory and the science that it should support that we, as scientists, envision include:

- The primary use of the collaboratory would be to serve a limited number of ambitious applications or experiments, but each experiment could involve quite a large scientific community in its design and analysis, thus requiring high-performance networking and data-access capabilities.
- The primary high-end compute server could be devoted in its entirety to a single "hero" application for periods of time, and thus should support application scalability to its full system size.

- The resources of the collaboratory must serve the needs of data assimilation as well as simulations, and thus must address the additional demands on storage and I/O performance that these applications require.
- That there would be minimum performance or efficiency criteria imposed on applications to assure the most effective utilization of the system.

Some of the issues that we feel need to be addressed in the technical prospectus include:

- What hardware resources will be required to deliver compute cycles, data storage and access, visualization, and analysis capabilities to the grand challenge scale problems posed by the GEO community in the science plan?
- What do the problems posed in the science plan, the characteristics of the codes used in these problem domains, and the scale of computing resources required imply for the architecture of the compute servers? Is more than one system architecture (or even facility) necessary or more economical to serve the needs of the breadth of the GEO community?
- What level of effort in software re-engineering of current codes will be required to exploit these architectures to their fullest capabilities?
- What is the appropriate balance between centralized vs. highly distributed facilities and services (e.g., grid) for delivering a full suite of resources to the collaboratory user base?
- What services should the collaboratory provide to support the user community?
- What auxiliary physical services (electrical and mechanical infrastructure), and human resources will be required to support the collaboratory? What leveraging of existing facilities is possible?

- What type of procurement model or phasing in of technology would best serve the collaboratory?
- How can the collaboratory help to position the GEO community to take best advantage of innovative high performance computer architectures that may emerge over the rest of this decade and the next?
- How can the GEO community leverage or influence federal or commercial sector R&D programs in high performance computing? What role could the collaboratory play in these programs?
- What is the scale of the budget required to establish and maintain the collaboratory?

We recognize that answering many of these questions requires extensive knowledge of the scientific problems to be undertaken. We will provide the technical working group with drafts of the science plan as it develops, and expect that there will need to be some iteration between the science plan and technical prospectus. Many of you already have backgrounds working with scientists in the GEO community on leading-edge computational research problems and a good understanding of some of the more widely used codes. This background and the information contained in the aforementioned reports should be enough to get your discussions started.

Further, if the technical working group finds that it needs to bring in additional expertise in certain areas it should feel free to do so. We would also appreciate input from you on how to best facilitate interaction between the technical working group and those of us developing the science plan and overseeing the process.

We look forward to working with you.

Ad Hoc Committee for a Petascale
Collaboratory for the Geosciences

APPENDIX 3

PETASCALE APPLICATION QUESTIONNAIRE

Applications can be viewed as a collection of algorithmic components that work together to solve a science problem. Unfortunately, it is not possible to benchmark systems that have not yet been constructed. Of course, one can rely on previous benchmark experience, and this is certainly a worthwhile source of data to inform our opinions, but this will only provide us with an understanding of what past technology has provided to previous versions of applications. In the absence of benchmarking, one can construct performance models of applications, which take information about the algorithms in the applications and combine this with parametric information about computer performance to understand what attributes of the computer system are most critical. Creating such performance models requires a great deal of detailed information about these algorithms if one is to construct even a simple qualitative model of the scaling and performance characteristics of the overall application.

The purpose of this questionnaire is to obtain this information from the domain expert (you), so that such simple performance models of your Earth Science application performance can be constructed. This will allow us to intelligently assess the requirements for a Petascale supercomputer environment (computing, storage, networking, ...) in terms of your future application requirements.

Please also supply any reports of code performance analysis that have been written up (particularly on dedicated resources as it is expected that much of the petascale facility will not be available interactively but in scheduled batch mode).

Application Characterization

Description. Provide name of application, discipline, science objective.

Application Workflow. Describe subcomponents, or phases of application workflow. Provide Application Data below for these subcomponents.

Community. What is the size of the user community for the application?

Portability. What computing platforms does the application run on? (Vector systems? Linux clusters? Technical Servers)

Grid Enablement. Do you plan to run your application(s) across long-haul networks in a distributed fashion using the Computation Grid?

Real Time Requirement. Do you conduct research that requires running codes and analyzing their output in real time?

Sensors/Instruments. Do you conduct research that requires management or control of remote devices or instruments?

Application Data

Target Resolution(s). For future, not current, science goals and ideally expressed as application grid sizes and array dimensionalities. Include number of levels in 3D grid.

Integration Time scale. Provide timestep and run length or express run length as number of timesteps/run. For iterative solvers, include iterations/timestep.

Capability Metric. Number of runs per day/month desired by researcher.

I/O Complexity. For the worst case and typical usage scenario, provide data that allows us to compute how many files are output and how large? How often in terms of timesteps, and how many fields are output?

Memory Requirements. Provide data that would allow us to estimate the memory requirements of your application, such as measured size of working set, number of state variables and time levels, etc.

Data Analysis/Visualization. Please express the resources used for analysis/mining/visualization of output generated by your application as a percentage of the resources required for the application itself.

- a. Do you conduct research that requires real time data acquisition and /or cataloging (and that requires extensive computer resources dedicated or in spurts)?

Algorithmic Data

It is expected that each application or application subcomponent may have several algorithms/components with different characteristics. Therefore, for each important algorithm in your application, list the following.

Characterization of Algorithm/Component. For example, Fourier/Legendre Transform, type of solvers used (PCG, GMRES, 1-, 2- or 3-D Multigrid), linear algebra operations employed (e.g., dgemm, the BLAS matrix multiply routine).

Algorithmic Computation

Computational Access Patterns. For example, are local memory access patterns stride 1, strided, or non-uniform in this algorithm? If they are non-uniform, can this be characterized further, e.g. quasi-local? Does the algorithm vectorize? Provide average vector length if known.

Computational Complexity. Estimated computational scaling properties with change in resolution (e.g., estimated number of operations performed per site (node, grid point) per timestep.

Computational Performance Data. Exclusive of communication, what is known about this algorithm's floating point performance on real systems? Is the computation load

balanced on parallel systems and how is load balancing dependent on availability of shared or distributed memory paradigms?

Algorithmic Communication

Communication Access Pattern. Characterize the communication patterns for the model in terms of locality (local, quasi-local or non-local), and persistence (dynamic/static) communication patterns. Provide more details that would allow message sizes and therefore communication costs to be estimated. For example, for nearest neighbor communications, how many ghost cells are there? Typically how many neighbors?.

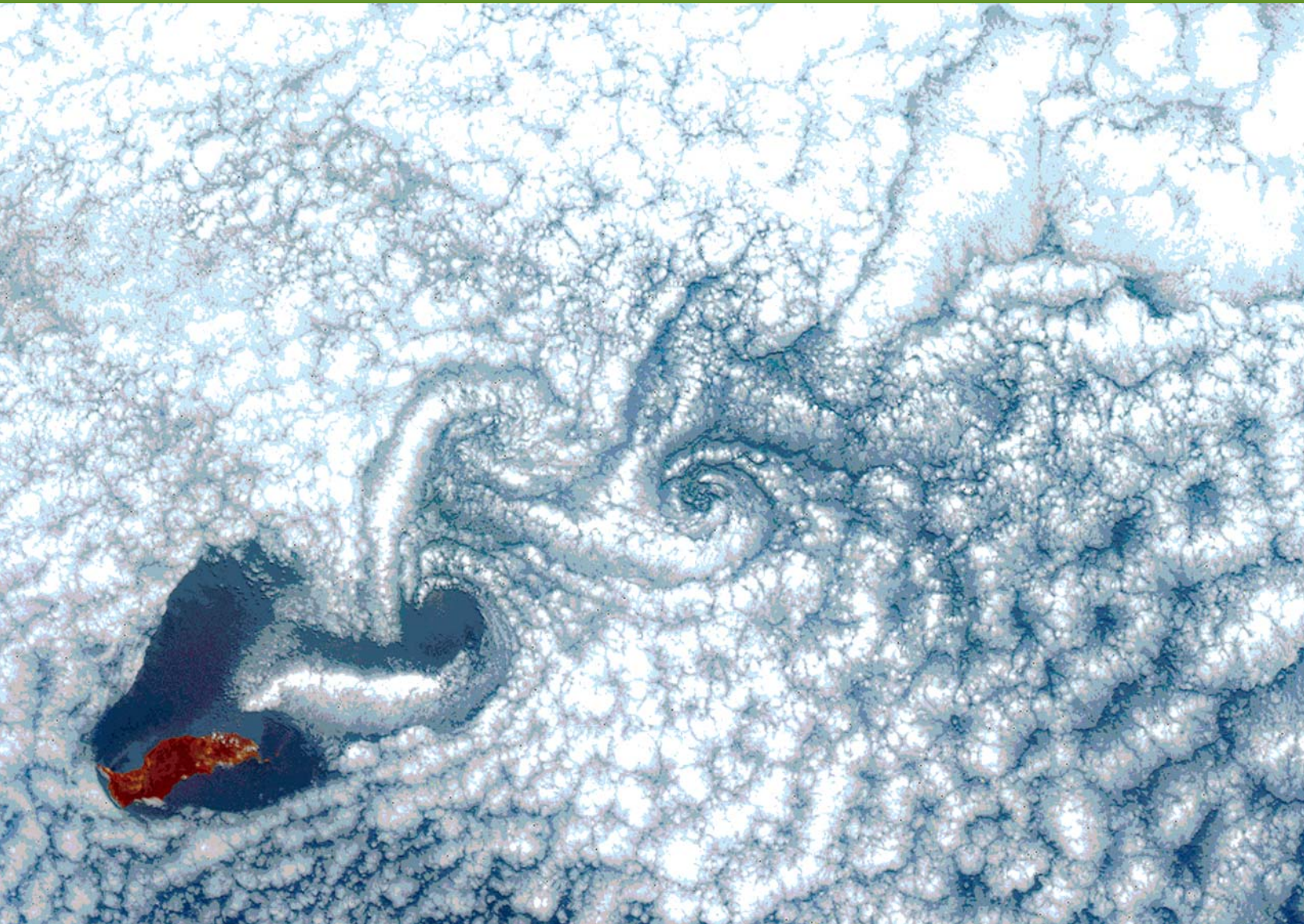
Communication Complexity. Include number of communication calls per timestep/iteration and number of fields communicated in each call.

Communication Performance Data. Exclusive of computations, what is known about the communication performance of this algorithm?

ACRONYMS

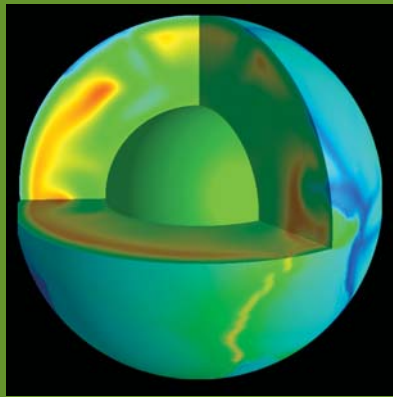
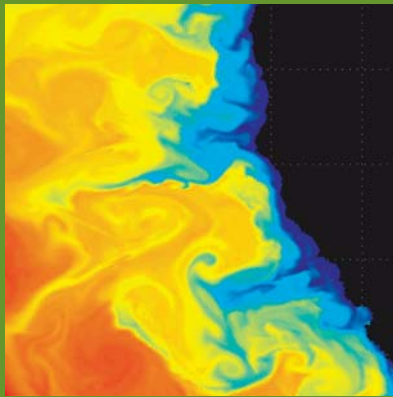
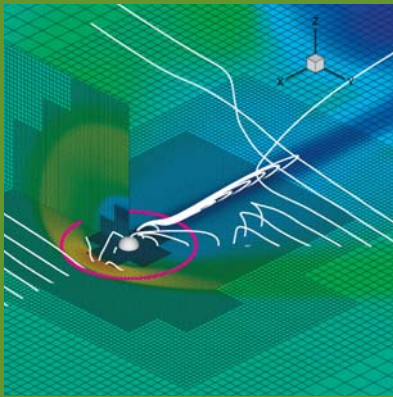
AAAS	American Association for the Advancement of Science
AMR.....	Adaptive mesh refinement
BATS-R-US	Block Adaptive-Tree Solar-wind Roe-type Upwind Scheme
CAM.....	Community Atmosphere Model
CAMR.....	Continuous adaptive mesh refinement
CCSM.....	Community Climate System Model
CG	Conjugate Gradient
CIDER.....	Cooperative Institute for Deep Earth Research
CMB	Core-Mantle Boundary
CME	Coronal mass ejections
CMP	Chip multiprocessors
CPU	Central processing unit
CRCP.....	Cloud Resolving Convection Parameterization
CRM.....	Cloud Resolving Model
CSEM	Center for Space Environment Modeling
DARPA.....	Defense Advanced Research Projects Agency
DFT	Density functional theory
DMC.....	Diffusion Monte Carlo
DMFT	Dynamical mean field theory
DNS	Direct numerical simulation
DOE	Department of Energy
ECCO.....	Estimating the Circulation and Climate of the Ocean
ECMWF.....	European Centre for Medium Range Weather Forecasting
ENSO	El Niño Southern Oscillation
ESM	Earth System Model
ESMF.....	Earth System Modeling Framework
FFT	Fast Fourier Transform
FLOPS	Floating Point Operations Per Second
GCM.....	General Circulation Model
GEWEX	Global Energy and Water Experiment
GGA	Generalized gradient approximation
GITM	Global Ionosphere-Thermosphere Model
HCP	Hexagonal closest packed
HPC.....	High-performance computing
HPCS.....	DARPA High Productivity Computing System program

HPL A portable implementation of the high-performance LINPACK benchmark
 for distributed-memory computers
 ICB..... Inner core boundary
 IPCC..... Intergovernmental Panel on Climate Change
 ITR..... Information Technology Research (NSF grant program)
 JAMSTEC Japan Agency for Marine-Earth Science and Technology
 JPL Jet Propulsion Laboratory
 LADHS Los Alamos Distributed Hydrologic System
 LAPW Linearized augmented plane wave
 LDA Local density approximation
 LES..... Large Eddy Simulation
 LMTO Linear-Muffin-Tin-Orbital
 LT..... Legendre Transform
 MHD Magnetohydrodynamics
 MISR Multi-angle Imaging SpectroRadiometer
 MPI..... Message passing interface
 MREFC NSF's Major Research Equipment and Facilities Construction account
 NASA National Aeronautics and Space Administration
 NCAR..... National Center for Atmospheric Research
 NCC Northern California Current
 NSF..... National Science Foundation
 NSWP National Space Weather Program
 NWP..... Numerical weather prediction
 OGCM Ocean General Circulation Model
 PCG Petascale Collaboratory for the Geosciences
 PDE Partial differential equation
 P.I. Principal Investigator
 PSC..... Pittsburgh Supercomputer Center
 QMC..... Quantum Monte Carlo
 SMVP Sparse matrix-vector problem
 SWMF Space Weather Modeling Framework
 TOPS Terascale Optimal PDE Solvers group
 TOMS..... Total Ozone Mapping Spectrometer
 TTS..... Time-to-solution
 UCAR/JOSS University Corporation for Atmospheric Research/Joint Office for Science Support
 WCRP World Climate Research Programme
 WRF Weather Research Forecast



This publication is funded by a subcontract with the University Corporation for Atmospheric Research (UCAR) under the sponsorship of the National Science Foundation. The views herein are those of the authors and not necessarily reflect the views of NSF or UCAR.

Editing and design by Geosciences Professional Services, Inc.



October 2005