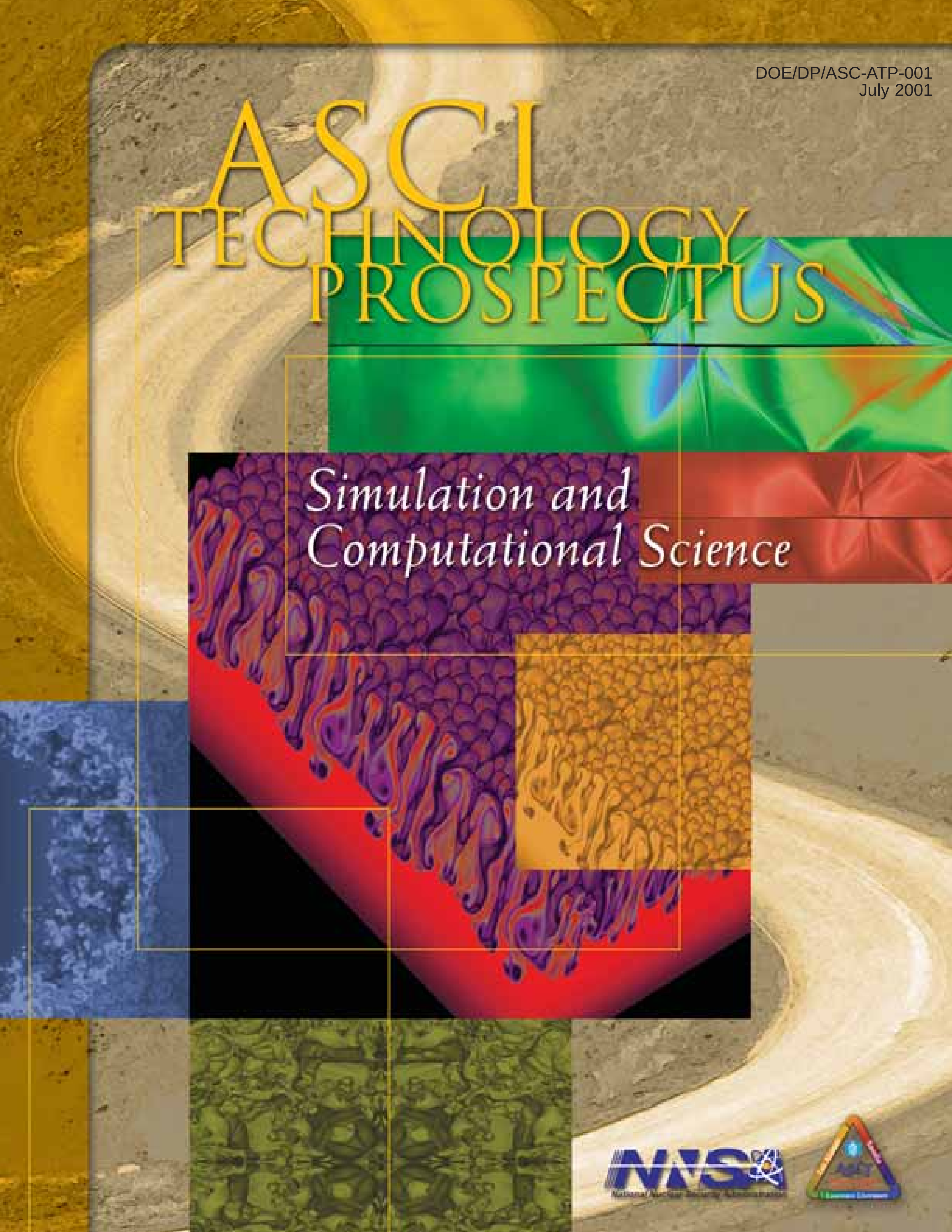
DOE/DP/ASC-ATP-001
July 2001

ASCI TECHNOLOGY PROSPECTUS

Simulation and Computational Science



ACKNOWLEDGEMENTS

The contributions of many individuals have been important to the production of this document, and many others have been instrumental in reviewing it. Without their dedicated efforts, the content would not be as authoritative and well organized as it is. Specifically, the ASCI program wishes to recognize Paul Messina and Dave Luginbuhl, both formerly with the National Nuclear Security Administration (NNSA) Defense Programs, for their vision of ASCI's future and their strong advocacy of this document.

The *Prospectus'* Planning Team members included Paul Smith, Steve Louis, Mike Koszykowski, Christie Morrison, and Tina Macaluso. Jo Sander (NNSA/Lawrence Livermore National Laboratory) successfully led this multi-lab/industry team and ensured that the project stayed on course.

The Review Panel members, Rick Stevens, Chuck Koelbel, Fred Johnson, José Muñoz, and John van Rosendale, contributed their considerable technical and scientific expertise on a complex subject. Alice Koniges from Lawrence Livermore National Laboratory generously donated her time and served ably as the computer science reviewer. Reeta Garber, at Sandia National Laboratories, helped manage all communications aspects of this project, served as the managing editor and writer, and handled the document's ultimate publication.

We are particularly grateful to the external domain-specific reviewers who provided critical insights, unique perspectives, and suggestions that greatly improved the quality of the work. They are by area:

Simulation Development Environment
Rusty Lusk / Argonne National Laboratory
John Reynders / Sun Microsystems

Scalable Solvers
Paul Saylor / University of Illinois
Jim Demmel / University of California
Homer Walker / Worcester Polytechnic Institute
Bill Gropp / Argonne National Laboratory

Data Storage and File Systems
Tom Ruwart / Ciprico
Darrell Long / University of California / Santa Cruz
Ben Kobler / NASA Goddard Space Flight Center

Frameworks and Software Integration
Dennis Gannon / Indiana University
Randy Bramley / Indiana University
Steve Parker / University of Utah

Visualization
Patrick Hanrahan / Stanford University
Kai Li / Princeton University

Scientific Data Management and Discovery
Alok Choudhary / Northwestern University
Joel Saltz / University of Maryland and Johns Hopkins University
Mike Folk / University of Illinois

Grid Services
Rich Wolski / University of Knoxville
Ian Foster / Argonne National Laboratory
Bill Johnston / Lawrence Berkeley National Laboratory

Networking
Charlie Catlett / Argonne National Laboratory

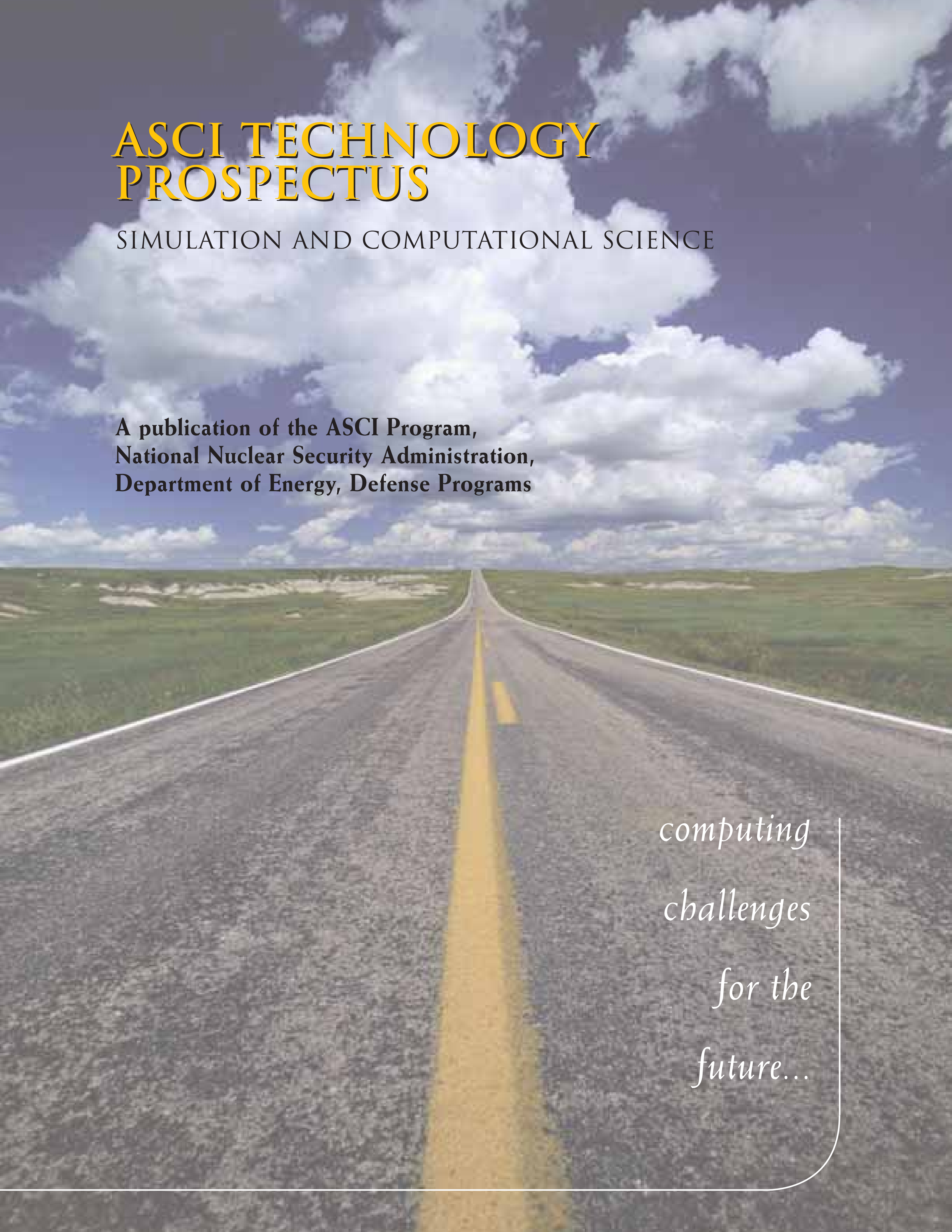
GRAPHIC DESIGN & PRODUCTION

Michael Vittitow - Creative Arts Department - 12620, Sandia National Laboratories, Albuquerque, NM

ASCI TECHNOLOGY PROSPECTUS

SIMULATION AND COMPUTATIONAL SCIENCE

A publication of the ASCI Program,
National Nuclear Security Administration,
Department of Energy, Defense Programs

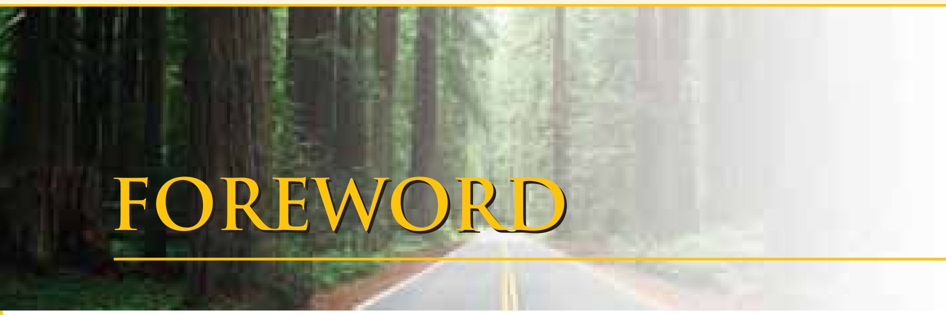


computing challenges for the future...

CONTENTS

TECHNOLOGY ROAD MAPS

FIGURES



FOREWORD

This *Prospectus* is an outgrowth of the ASCI "Curves and Barriers" effort whose goal was to identify and define ASCI's technology needs over the next five years and to compare them with road maps for those technologies that are driven by general market forces. The resultant information will help the ASCI team distinguish and identify (a) which technologies will evolve as a result of general market forces to meet ASCI's needs and schedule; (b) which technologies will acquire the needed capabilities but will require acceleration to match the ASCI road maps; and (c) which technology capabilities need to be developed by ASCI alone.

The success of this strategy of creating technology road maps to plan our development activities and investments depends on how accurately we understand development directions and schedules for all the relevant software and hardware technologies and methodologies. It is this current understanding of ASCI's needs and of technology trends that is described in this report. We invite the high-performance computing communities in the commercial sector and academia to critique our assessment and help us to improve it.

Background

Today's large-scale scientific and engineering applications are able to simulate and model extremely complex phenomena and devices with increasing fidelity. Multiple models need to be integrated into unified application programs to simulate complex phenomena. Furthermore, multiple, geographically distributed teams must share a succession of computing platforms, visualization servers, and archival storage systems. The hardware resources are similarly heterogeneous and geographically distributed, and are replaced frequently by more powerful ones.

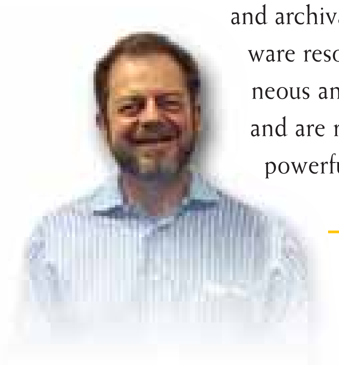
Almost all of today's technical computing is carried out on systems whose building blocks, often called nodes, are servers with 2 to 32 processors and are built with commodity parts. Large-scale scientific and engineering applications require closely coupled systems with hundreds to thousands of nodes.

The current ASCI computing platform and future development strategy is consistent with this trend. ASCI uses commercial building blocks. The difference lies in ASCI's scale and schedule. ASCI's mission requires systems with peak speeds of 100 teraOPS by 2005 (subject to budget constraints). In this time frame, on the order of 10,000 processors will be needed to achieve such speeds. Effective use and operation of so many processors presents challenges for scaling both the hardware and the system software.

The scale and complexity of ASCI's and other large-scale scientific and engineering computations require much more than big hardware platforms for evaluating data. Visualization, data storage and manipulation, and networking are also needed. Furthermore, effective use of systems with thousands of processors requires new or improved algorithms, software tools, compilers, run-time systems, debuggers, visualization systems, etc.

In fact, so much needs to be done to create the computing environments required by ASCI that it is extremely important that we analyze existing trends in technologies so that we can:

1. take advantage of the ones that will provide the needed capabilities without special stimulus from us,
2. adapt our approaches to utilize the broad existing trends as much as possible, and
3. invest in creating the tools we need that cannot be obtained through projected developments.



Dr. Paul C. Messina is the Director of the Center for Advanced Computer Research at the California Institute of Technology (CalTech) in Pasadena, CA.

Because ASCI's applications require huge computing power, their development schedule is perforce very ambitious. Therefore ASCI's computing platform needs to outpace the technologies' normal evolution as a result of general market forces. In this, ASCI is not alone. Most large-scale scientific and engineering applications have similar needs, although usually without the brutal mission-driven timetable. Regardless, such similarities provide many opportunities for collaborations with other high-performance communities. Ambitious schedules are not the only governing force for ASCI.

Other big needs are satisfied as much as possible with mass market commercial products — a technical community trend. General market forces dictate the development of many software and hardware technologies that benefit technical computing. Still, differences remain between high-end technical and commercial computing that require many software and some hardware products to be enhanced for ASCI. Such differences involve the scale of computations, their memory access patterns, the ratio of floating-point to other instructions, the type and volume of data produced by the simulation codes, the computing patterns of scientific and engineering simulations, and the analyses needed to interpret the output. Moreover, scaling software and hardware to work effectively on configurations with thousands of processors is a key requirement for ASCI and other high-end technical computing projects.

Hopefully, the realization that ASCI's needs are largely the same as those of other large-scale scientific and engineering computing applications will motivate researchers to develop technologies that will meet our needs and those of high-end technical computing in general. Equally important, by describing its technology requirements over the next few years, ASCI might motivate commercial companies to develop products that it needs. Many companies have noticed that the needs of high-end scientific computing are often harbingers of mass market requirements.

The mere exercise of identifying what technologies we need should help us sort out what is really important and what we need to do. For example, brute force extrapolation of network bandwidth needs, based on current usage models, might lead to infeasible requirements. This tells us that we should look for different ways to approach the task.

We understand that the ASCI program and its targets are influencing the rest of the world. But the resources available to us are not sufficient to meet our goals. Developing the technology curves and identifying hurdles and barriers each faces allows us to concentrate our resources where we need them most. Such objective analysis reduces the widely recognized "Not Invented Here" syndrome and helps us generate milestones for measuring progress towards our goals.

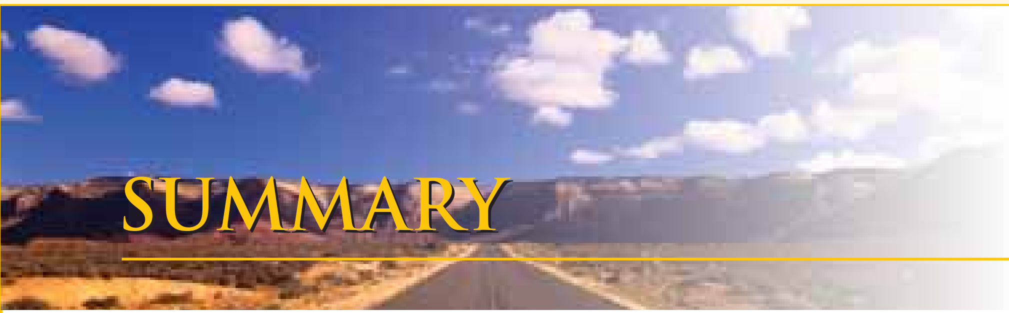
In summary, this report records ASCI's attempt to:

1. define computing capabilities needed to meet its mission goals;
2. derive corresponding technology road maps;
3. chart the evolution in relevant software technologies that is expected over the next few years because of general market demands;
4. map those trends against its road maps and identify the areas that will need targeted efforts to meet its needs.

Knowing the technology road maps, ASCI will be able to optimize its investments in research, development, and deployment.

To develop these technology road maps, we involved as many experts as possible from industry, academia, and research laboratories, but this is a difficult task and we seek additional input by publishing this report. I invite your thoughts on how we can best work together to achieve the considerable technical milestones described in this document.

Finally, it is our hope that presenting these challenges will motivate people in the research and industrial communities to try to solve them with us. We believe that defining these curves, hurdles, and barriers will prove to be an effective way to engage others. Delineating these requirements may motivate the private sector to improve its products. We can identify opportunities for coordinating, collaborating, and presenting a united front with other agencies, and we can work with the academic researchers to identify interesting issues.



SUMMARY

By the year 2005, the National Nuclear Security Adminstrations's (NNSA's) current Advanced Simulation and Computing (ASC)[1] supercomputers are expected to run large simulations producing hundreds of terabytes of complex information per simulation. Parallel computing is the only effective means of running these huge simulations, and the parallel computing environment must be efficient and easy to use. The massive data will dictate the need for developing platforms whose run-time systems operate effectively at the largest scale. Saving and retrieving this information will increase in importance and become even more demanding. In turn, the algorithms required for the modeling and simulation of the high-fidelity physics packages will need scalable solvers. Software interoperability issues will surface; a weapons designer's need to rapidly develop software will require software modules that can be easily and reliably integrated with each other. Similarly, the need for efficient, quick, and easy access to computing resources will become paramount, increasing the need for robust computing grids. Following closely will be the need for an ASCI networking architecture — one that would allow the timely movement of terascale data files through entire local and wide area networks at speeds of hundreds of gigabytes/s.

Emerging from these overarching needs is a host of specific capabilities needed by ASCI and visually described in five-year road maps. These capabilities are identified according to their current research and development status, i.e., whether they have been accomplished, are planned, or are faced with "hurdles" and "barriers." The capabilities' successful accomplishment will, in some measure, depend on collaborations with the rest of the high-performance computing community. The capabilities are categorized within eight critical technology areas: (1) simulation development environment, (2) scalable solvers, (3) software interoperability, (4) visualization, (5) scientific data management and discovery, (6) data storage and file systems, (7) grid services, and (8) networking.

- Simulation Development Environment: Of fundamental importance is the parallel computational environment complete with scalable communication libraries and diagnostic/debugging tools compatible with the needs of production codes. There is a need for a common set of standardized interfaces operable across multiple platforms, and effective resource allocation (including dynamic and distributed parallelism).

- Scalable Solvers: Algorithms, required for the modeling and simulation of the high-fidelity physics packages, rely on scalable solvers. Issues of scale, both in terms of problem size and number of processors are very important to ASCI — balancing workloads across thousands of processors in an often heterogeneous computing environment. Significant collaborative research is needed in integrating disparate codes with different structures and varying time scales.

- Software Interoperability: As software becomes more complex, the interoperability and reusability of software become essential. Building frameworks — a set of reusable design patterns expressed as clearly defined software abstractions — is one challenging approach. Acquiring the latest component technology is another approach. For ASCI to stay on an upward curve, more research is needed into exploring both these approaches.

- Visualization: This is an essential means for weapons designers who need to evaluate simulation results in minute detail. Although the best visualization hardware and software are being leveraged as appropriate, "terascale"-level scientific visualization tools are either not commercially available or viable at this time.

[1] Historically known as the Accelerated Strategic Computing Initiative (ASCI), and used as such throughout this document.

- Scientific Data Management and Discovery: Weapons designers face major challenges of managing multi-terabyte datasets in petabyte archives and understanding and evaluating mesh-based simulation datasets. These challenges become even more daunting because some of the data are generated and stored remotely. Delays in addressing data management issues of this magnitude can significantly impact ASCI's future commitments.

- Data Storage and File Systems: Another requirement is to build many storage devices per compute node and provide parallel access to all of them, thereby maintaining balance between computational speeds and I/O rates. ASCI can benefit from more research and development into improving existing, and building new, storage and archival systems technologies.

- Grid Services: Current work on developing a network of computational grid services with a well-defined set of interfaces aims to simplify remote access, make usability efficient, provide needed security, and coordinate scheduling of disparate resources. However, ASCI can still benefit from continued and new collaborations with academia and industry in the building of a computational Grid infrastructure.

- Networking: Efficient movement of terascale data files through local and wide area entire networks at speeds of hundreds of gigabytes/s involves many challenges: greater parallel bandwidth, increased speeds for network interface cards, and a reconsideration of existing end-systems such as computer architectures and I/O devices. In general, a robust end-to-end parallel data transport is essential to ASCI's future success.

DEFINITIONS & URLS

ACL	access control list
AF&F	arming, fuzing and firing
AMG	algebraic multigrid
AMR	adaptive mesh refinement
ANL	Argonne National Laboratory (http://www.anl.gov/)
API	Application Programming Interface
ASC	Advanced Simulation and Computing
ASCI	Accelerated Strategic Computing Initiative (http://www.asci.doe.gov/)
ATM	Asynchronous Transfer Mode

CCA	Common Component Architecture
CCM	CORBA component model
CORBA	Common Object Request Broker Architecture
COTS	commercial off-the-shelf
CPU	central processing unit

DCE	distributed computing environment
DFS	distributed file system
DMF	data models and format
DoD	Department of Defense (http://www.defenselink.mil/)
DOE	Department of Energy (http://www.energy.gov/)
DOF	degree of freedom
DRM	distributed resource management
DVC	Data and Visualization Corridors
DWDM	dense wavelength division multiplexing

EJB	Enterprise Java Beans
ESI	Equation Solver Interface

FCP	Fibre Channel Protocol
FTP	File Transfer Protocol

Gb	gigabit
GB	gigabyte
GPFS	General Parallel File System
GSF	Generalized Security Framework
GSSAPI	Generalized Security Services Application Programming Interface

HPC	high-performance computing
HPSS	high-performance storage system

iFCP	Internet Fibre Channel Protocol
IETF	Internet Engineering Task Force
I/O	input/output
IP	Internet Protocol
IPG	Information Power Grid
IPS	IP storage
ISO	International organization responsible for the setting of standards for similar technologies worldwide (http://www.iso.ch/)

KAI	Kuck and Associates, Inc., a division of Intel (http://www.kai.com/)

LAN	Local Area Network
LANL	Los Alamos National Laboratory (http://www.lanl.gov/)
LLNL	Lawrence Livermore National Laboratory (http://www.llnl.gov/)
LOTS	LOTS Technology, Inc. (http://www.lotstech.com/)

Mb	megabit
MB	megabyte
MEMS	micro-electro-mechanical systems

MG	multigrid
MIMD	multiple instruction/multiple data, or Maison de l'Informatique et des Mathématiques Discrètes (Society for Computer Science and Discrete Mathematics)
MIT	Massachusetts Institute of Technology (http://www.mit.edu/)
MPI	Message Passing Interface
MPX	a wireless data infrastructure company (http://www.mpxnet.com/)
MSTI	MPI Software Technology (http://www.mpi-softtech.com/)
MTBF	mean time between failure
MUTT	Memory Utilization Tracking Tool

NAS	Network-Attached Storage
NFS	network file system
NIC	network interface cards
NNSA	National Nuclear Security Administration
NPACI	National Partnership for Advanced Computational Infrastructure (http://www.npaci.edu/)
NSA	National Security Agency (http://www.nsa.gov/)
NSIC	National Storage Industry Consortium (http://www.nsic.org/)
NTK	need to know
NUMA	nonuniform memory access
NWC	Nuclear Weapons Complex

OLAP	online application processing
OMG	Object Management Group
OPS	operations per second
OS	operating system
OSD	Object-based Storage Devices

PDE	partial differential equation
PFTP	Parallel File Transfer Protocol
POOMA	Parallel Object-Oriented Methods and Applications (http://www.acl.lanl.gov/pooma)
POS	packet over SONET
PSE	Problem Solving Environment

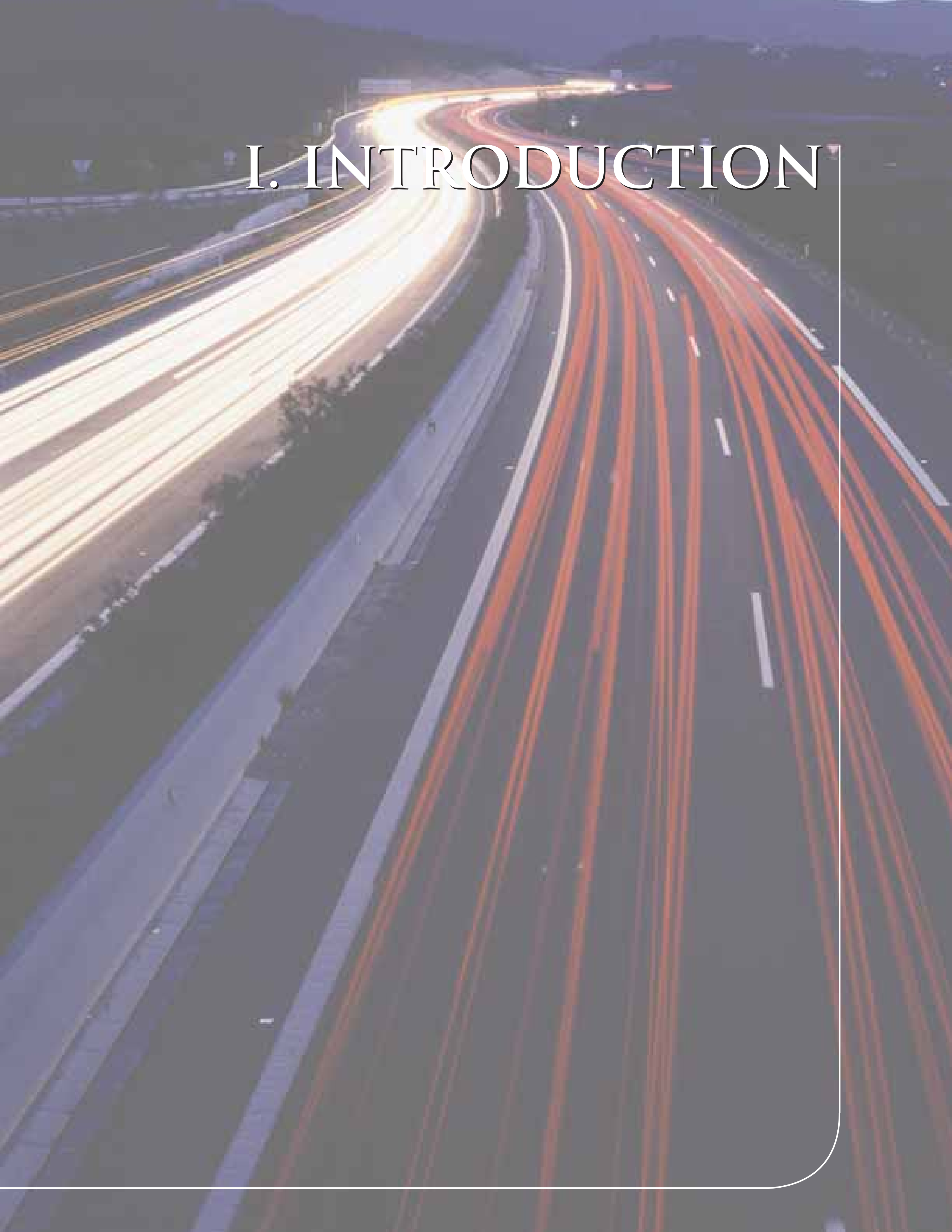
R&D	research and development
RAID	Redundant Array of Independent Disk
RAIT	Redundant Array of Independent Tape
RFP	request for proposal
RTS	Run-Time System

S&CS	Simulation and Computer Science
SAN	Storage-Area Network
SASL	Simple Authentication and Security Layer
SDE	Simulation Development Environment
SDM	Scientific Data Management
SESAME	a database
SGI	Silicon Graphics, Inc. (http://www.sgi.com/)
SGS-FS	scalable, global, secure file system
SMARTS	Scalable Multithreaded Asynchronous Run Time System
SMP	symmetric multiprocessor
SNIA	Storage Networking Industry Association (http://www.snia.org/)
SNL	Sandia National Laboratories (http://www.sandia.gov/)
SPMD	single program multiple data
SSP	Stockpile Stewardship Program
ST	scheduled transfer

VI	virtual interface
VIEWS	Visual Interactive Environment for Weapons Simulation
vtk	Visualization Tool Kit

WAN	wide area network

I. INTRODUCTION



The Accelerated Strategic Computing Initiative (ASCI) Computer Technology Prospectus: Vol. 1, Simulation and Computational Science — 2000 to 2005 contains a set of comprehensive, strategic road maps (previously known as the "ASCI curves") that govern the Advanced Simulation and Computing (ASC)[2] Program's research and development in computing technology and simulation and computer science. These road maps — visualized in the accompanying foldout — depict planned progress in a given area and identify critical challenges yet to be overcome for ASCI to meet the overall objectives of the Department of Energy, National Nuclear Security Administration (NNSA) Stockpile Stewardship Program.

Over the last two years, three NNSA defense laboratories — Sandia National Laboratories (SNL), Lawrence Livermore National Laboratory (LLNL), and Los Alamos National Laboratory (LANL) — have jointly developed these technology road maps. Their collective ASCI experience in code applications and in advanced computing platforms has led to a clearer understanding of new requirements for computational technology. The addition of several new programs within ASCI (such as distance computing, visualization and data management) have added a new dimension to this understanding, thereby requiring the development of a comprehensive, integrated approach to achieve ASCI's objectives. The *Prospectus* verbalizes an essential part of this integration strategy.

This volume describes ASCI's vision for computing technology and its approach for achieving some remarkable, but necessary, capabilities between the years 2000 and 2005. Eight technologies are described here, categorized within three broad functional areas: (1) Computational and Software Environment, (2) Data Management, Visualization, and Storage, and (3) Distributed and Distance Computing. Each description includes ASCI's overall drivers, their correlation with requirements for simulation and computer science research and development, technical issues and challenges, a visual road map accompanied by an annotated timeline, and the current state of ASCI. The volume concludes by summarizing the challenges still facing each technology area.

Collaborations with academia and industry can be particularly and mutually beneficial, since these technologies are not unique to ASCI. The goal of this *Prospectus* is to invite the entire high-performance computing community to participate in addressing these computing technology challenges.

ASCI Formation. For nearly a half century, confidence in the U.S. nuclear deterrent was a product of computation, experimental science, and weapons physics. The final judgments about the safety, performance, and reliability of the country's nuclear stockpile were confirmed by nuclear test results. Because of this, computer models much simpler than those needed today could be used with the best available computers to help design, modernize, and maintain the stockpile. Now, without nuclear testing as the final arbiter of scientific judgment, weapons scientists must rely much more heavily on sophisticated computers to simulate the complex aging process of the weapons components and the weapons systems as a whole, and determine the impact on the nuclear weapons stockpile.

The NNSA's Stockpile Stewardship Program was established to develop new means of assessing the performance of nuclear weapon systems, predict their safety and reliability, and certify their functionality. The program must not only fulfill its responsibilities without nuclear testing, but must also address constraints on non-nuclear testing, the downsizing of production capability, and the cessation of developing new weapon systems to replace existing weapons. Further complicating matters, weapon components are exceeding their design lifetimes, and manufacturing issues and environmental concerns will force changes in fabrication processes and materials of weapon components.

[2] Historically known as the Accelerated Strategic Computing Initiative (ASCI), and used as such throughout this document.

All of these issues must be resolved within a fast-approaching deadline. Not only are the weapons aging, but so are the scientists and engineers experienced in underground nuclear tests and in nuclear weapon design. Since the U.S. conducted its last nuclear test in 1992, its experience base has been declining; those who participated in the design of our enduring stockpile are fast approaching retirement. As a result, ASCI was established to be the focus of the NNSA's simulation and modeling work. ASCI's aim is to provide high-fidelity computer simulations of weapon systems that will enable scientists to continue making the necessary judgments for maintaining the credibility of the nuclear deterrent. The 2004 to 2010 timeframe is crucial for having usable, working ASCI computer systems and codes available, so a smooth transition from "test-based" to simulation-based certification and assessment can be made. To achieve this goal, experimental data from aboveground test facilities must be linked with archival nuclear test data from 50 years of nuclear tests, and with improved scientific understanding to provide high-confidence predictive simulation capabilities to support national decisions about the enduring stockpile.

Technology Impetus. To achieve its vision, ASCI is accelerating the development of simulation capabilities needed to analyze and predict the performance, safety, and reliability of nuclear weapons and certify their functionality — far exceeding what might have been achieved in the absence of such a focused initiative. These simulation capabilities are based on advanced weapon codes and high-performance computing that incorporate complete scientific models based on experimental results, past tests, and theory. The computing power required by these simulations is much greater than that normally provided by industry. Therefore, ASCI researchers are collaborating with their computer industry counterparts to accelerate development of much more powerful computing systems and to invest in creating the necessary software environment.

Computational and simulation capabilities developed through ASCI will help scientists understand aging weapons, assess when components will have to be replaced, and evaluate the implications of changes in materials and fabrication processes to the design life of the aging weapon systems. This science-based understanding is essential to ensure that changes brought about through aging or remanufacturing will not adversely affect the enduring stockpile.

To meet the needs of stockpile stewardship in the year 2005 and beyond, ASCI must solve progressively more difficult problems as we move away from nuclear testing. To do this, code applications must achieve higher resolution, higher fidelity, three-dimensional physics, and full-system modeling capabilities to reduce reliance on empirical judgments. This level of simulation requires high-performance computing far beyond our current level of performance. A powerful problem-solving environment must be established to support application development and enable efficient and productive use of the new computing systems. Therefore, by 2005, ASCI is responsible for ensuring the:

- Development of high-performance, full-system, high-fidelity-physics predictive codes to support weapon assessments, renewal process analyses, accident analyses, and certification.
- Stimulation of the U.S. computer manufacturing industry to create the powerful high-end computing capability required by ASCI applications.
- Creation of a computational infrastructure and operating environment that makes these capabilities accessible and usable.

ASCI recognizes that the creation of simulation capabilities needed for performance simulation and virtual prototyping is a significant challenge. To meet this challenge, there needs to be cooperation between the science and technology resources available at the national laboratories and the computer industry to accelerate their business plans to provide the computational platforms needed to support ASCI applications. Academia must also play a critical role in developing the computational tools and scientific understanding needed for this unprecedented level of simulation.

ASCI Components. There are three major components of ASCI: (1) applications, (2) platforms, and (3) simulation and computer science. The software applications implement three-dimensional, high-fidelity-physics simulation, and support efforts in modeling, and verification and validation. The powerful ASCI computing platforms run these codes, and the simulation and computer science research and development support development, deployment, and integration of the applications software and the platforms. ASCI is ultimately driven by the need for advances in the software applications. The required, increasingly complex applications mileposts leading to three-dimensional working simulation codes by 2005 are shown in Figure 1.

For the applications area to progress, work on a variety of computer science technologies such as the ASCI platforms needs to be "accelerated." In 1996, ASCI unveiled the 1-teraOPS Intel "Red" machine — the world's first teraOPS computer — at Sandia. This machine was followed in 1998 by two 3-teraOPS machines: the Silicon Graphics, Inc. (SGI) "Blue Mountain" at Los Alamos and the IBM "Blue Pacific" at Lawrence Livermore. More recently, Lawrence Livermore has acquired a 12-teraOPS IBM machine, known as "ASCI White," and Los Alamos has signed a contract with Compaq to deliver and bring online a 30-teraOPS machine, "ASCI Q," in 2002. Each of these machines represents a milepost, illustrated in Figure 2.

There is, of course, a great deal of simulation and computer science underlying both the applications and platforms road maps. Progress towards the ASCI goals, needed by 2005, requires coordinated advancement in applications, platforms, and underlying simulation and computer science infrastructure.

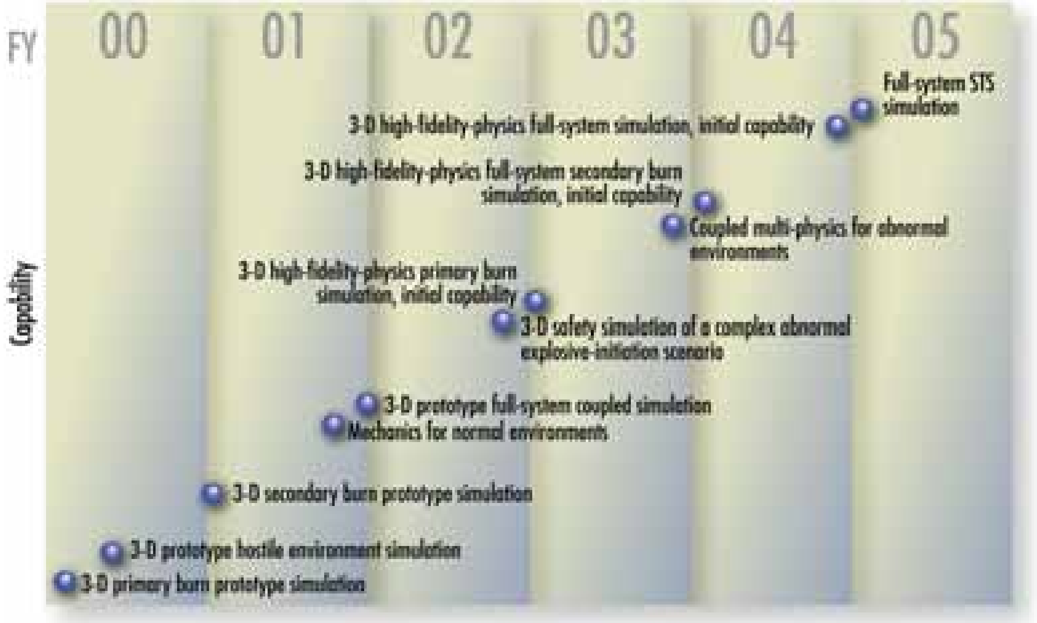


Figure 1. ASCI Applications mileposts.

The simulation and computer science visual (Figure 3) is conceptual. It is in fact abstracted from several individual technology road maps described in detail within the text. These road maps illustrate the advanced capability necessary for ASCI's mission.

It is understood that these technologies are not unique to ASCI. A number of agencies, organizations, companies, and researchers are working toward advancements in these areas. However, it is critical to the overall success that the broader research and industrial communities participate in reaching the goals outlined in this *Prospectus*.

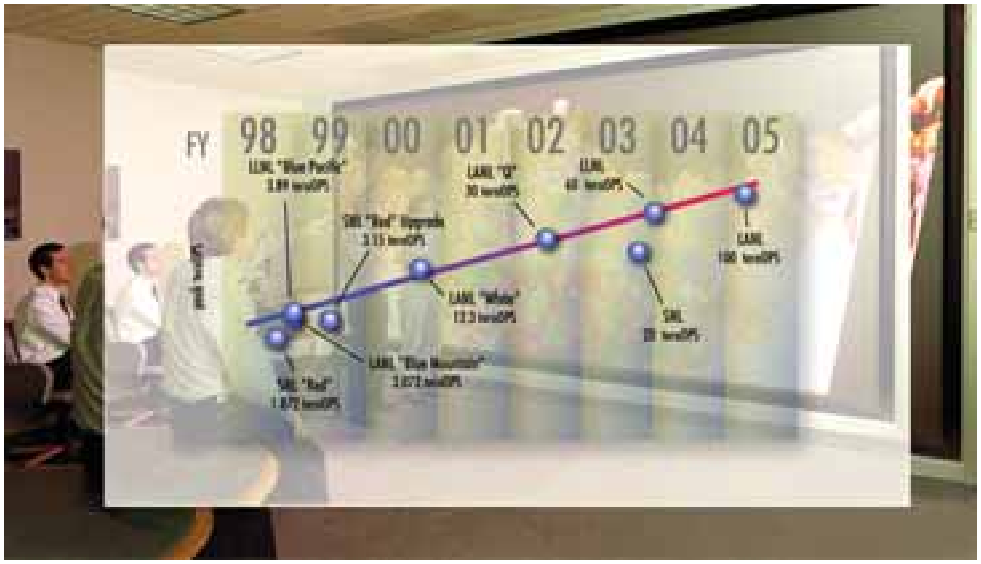


Figure 2. ASCI Platforms mileposts.

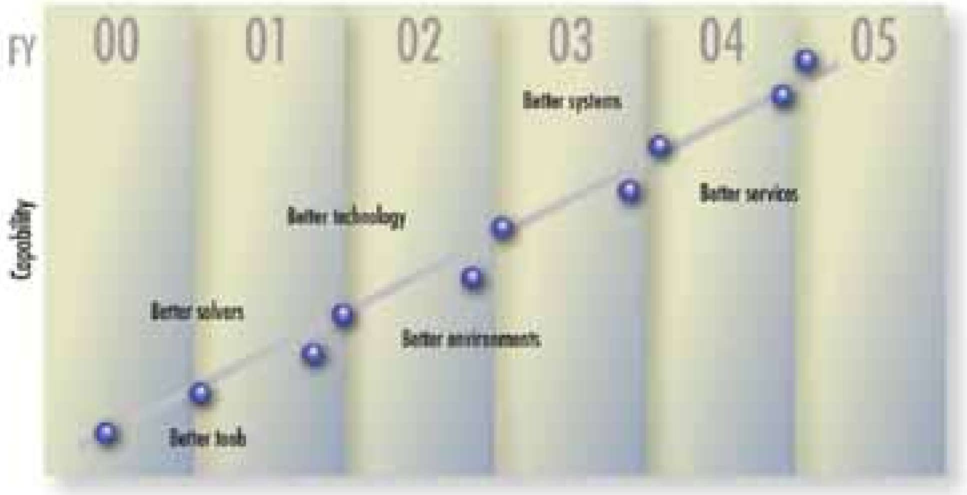


Figure 3. Simulation and Computer Science conceptual road map.

Road Map for SIMULATION AND COMPUTER SCIENCE REQUIREMENTS

R&D Effort Indicator ● ACCOMPLISHED ● PLANNED ● HURDLE ● BARRIER

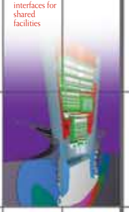
Accelerated Strategic Computing Initiative National Nuclear Security Administration

U.S. Department of Energy

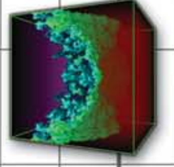
A Road Map to Success

2004 to 2010 is the target period for having Accelerated Strategic Computing Initiative (ASCI) computer systems and codes contributing to science- and simulation-based confidence in our nuclear weapons stockpile. To that end, a major goal of the National Nuclear Security Administration ASCI program is to develop three-dimensional, full-system, high-fidelity physics codes by 2004 to support assessment of stockpile safety, performance, and reliability.

Advancement towards this goal requires an integrated and systematic approach to the research and development of the applications, computer platforms, and underlying computer science infrastructure. A rigorous road map of increasingly complex applications mileposts tightly coupled to simulation and computer science technology development is the key to reaching and moving beyond the 2004 threshold.

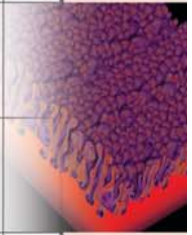
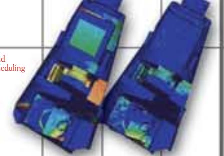
	Visualization				
	Data Handling	Scalable Rendering	Distance Visualization	Environments	Data Exploration
CY 2000		Scalable rendering system driving a 16M pixel tiled display	Interactive distance visualization with lossy image delivery	Shared immersive environments	
CY 2001					Data browsing/ multi-resolution data
CY 2002	Real-time interaction with 300GB timedump	Scalable rendering system driving a 64M pixel tiled display	Interactive distance vizualization solutions Digital approaches for desktop delivery	Display intensive offices	
CY 2003	Data subsetting/ manipulation services integrated with interactive visualization solutions		Collaborative interactive distance visualization	Improved user interfaces for shared facilities	
CY 2004	Real-time interaction with 2TB timedump	Scalable rendering system driving a 16M pixel display in offices	Integrated distance visualization and data services available across the WAN		
CY 2005		Scalable rendering system driving a visual acuity tiled display (100M pixel) in shared facility		Collaborative office environments	Data mining/ discovery integrated with visualization

	Scientific Data Management and Discovery			
	Data Access & Prep	Meta-data Infrastructure and Applications	Data Discovery	Data Models & Formats
CY 2000	Simple ad hoc queries on small simulation datasets	Automatic simulation data archiving and retrieval for high-demand codes	Scalable pattern recognition for medium-sized mesh-oriented data	Robust high-level data model for ASCI simulation data Application-oriented parallel IO capabilities on ASCI platforms
CY 2001			Scalable geometrical feature extraction for mesh-oriented data Smart comparison of small simulation datasets Guided feature detection on medium simulation	Advanced user-directed content-based subsetting of simulation results
CY 2002	Optimized data access for improved storage interaction		Scalable tool kit for extraction of common features	Data sharability across high-demand ASCI codes
CY 2003	Complicated ad hoc queries on large datasets Automated speculative data access DOE complex-wide integrated data access across wide range of data sources		Smart simulation/ experimental data comparison tools on large datasets	
CY 2004		Automatic simulation data archiving and retrieval for all ASCI codes	Scalable pattern recognition for massive mesh-oriented data Feature extraction and analysis simultaneous with simulation Interactive-example-based discovery in simulation	Application-oriented data manipulation operators
CY 2005	ASCI-pertinent legacy information integrated into data access infrastructure		Postprocessing and discovery operations integrated with uncertainty quantification analysis	Ubiquitous data infrastructure across all ASCI codes

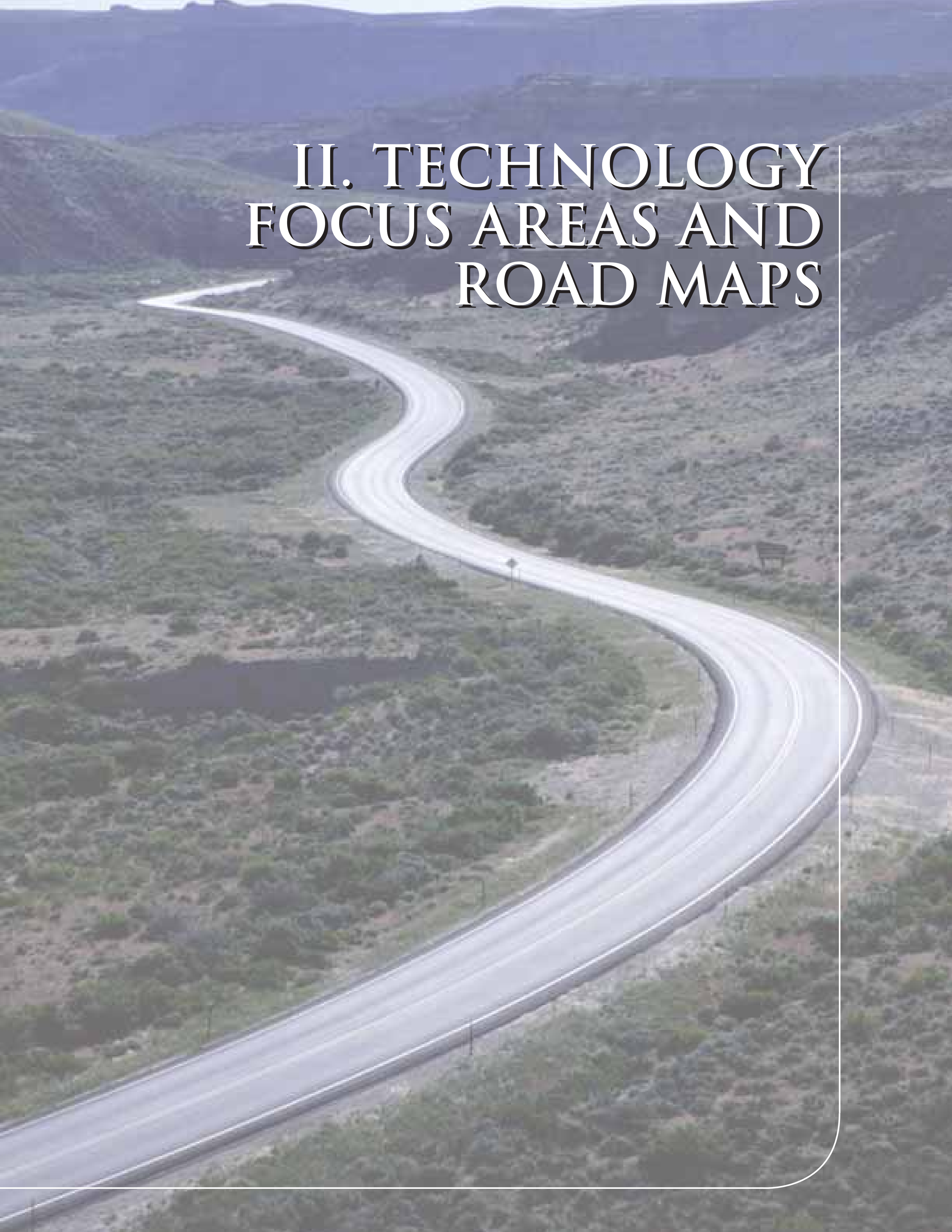
	Simulation Developments Environments				Scalable Solvers	
	Development Tools	Parallel RTS	Programming Models	SDE Deployment	Linear Solvers	Nonlinear Solvers/ Optimization
CY 2000	Debug and tune to 2000 CPUs Memory measurement tools Open- source tool infrastructure	Thread-aware MPI scales to 4000 tasks		10 teraOPS SDE	Parallel algebraic multigrid codes	
CY 2001	Debug and tune to 4000 CPUs Scalable measurement and diagnosis components Scheduling for thread/ memory affinity					Nonlinear optimization library for 8000+ processors
CY 2002	Mixed parallel model performance tools Multi-platform open-source tool infrastructure	MPI2 available	Scalable OpenMP	SDE scaling tests to 10,000 CPUs 30 teraOPS SDE	Eigensolver package for systems with 30M DOF Scalable algebraic MG for 1B unstructured mesh	
CY 2003	Debug and tune to 10,000 CPUs Scalable measurement and diagnosis components Tools support new language/ parallel models	MPI2 scaling to 10,000 tasks Dynamic parallelism support RTS for distributed parallel applications	High Performance Java Compiler/ language technology for serial performance			Constrained optimization tool kit for systems with 100M DOF
CY 2004	Large-scale open-source tools Distributed platform performance testing		Compilers support for current standards	SDE scaling tests to 15,000 CPUs	Eigensolver package for systems with 300M DOF Linear solver package for 10B unstructured mesh	
CY 2005				Production scalable SDE for milepost 100 teraOPS SDE		Optimization under uncertainty tool kit

	Data Storage and File Systems			
	Security	Accessibility	Aggregate Speed	Storage Capacity
CY 2000	Access Control Lists in HPSS	Layered tri-laboratory applications I/O architecture RFI issued for SGS-FS	40 MB/s Tape Prototype optical tape and high-end RAIT systems	500 TB Archive
CY 2001		Prototype tri-laboratory federated namespace RFP issued for SGS-FS	5 GB/s Disk COTS RAIT (tape)	1 PB Archive
CY 2002		Production tri-laboratory federated namespace Demonstration of SGS-FS on heterogeneous clusters	1 GB/s Tape COTS optical tape	5 PB Archive
CY 2003		Distributed Resource Management (DRM) support for storage systems Demonstration of SGS-FS on heterogeneous clusters	20 GB/s Disk Next-generation HPSS systems (post-Release 5)	10 PB Archive
CY 2004		Scientific Data Management support for storage systems	4 GB/s Tape HPSS secure access for Object-based Storage Devices	25 PB Archive
CY 2005	Fine-grained need-to-know access control		100+ GB/s Disk Evaluation of alternative mass storage systems (e.g., MEMS holographic)	50+ PB Archive Evaluation of alternative mass storage systems (e.g., MEMS holographic)

	Software Interoperability			
	Core Technology	Generic Service	Process Infrastructure	Domain Service
CY 2000				
CY 2001	Large-scale, distributed container specs Large-scale, distributed object persistence specs Parallel performance optimization components specifications Specs for large-scale, parallel testing		OMG-style working group and process for standard-ization	
CY 2002	Large-scale, distributed container implementations Large-scale, distributed object persistence implementations Parallel performance optimization components implementations Implementation of large-scale, parallel testing components Core parallelcomponent architecture implementation	Large dataset management and database connectivity specifications Gridded dataset interface specifications Numerical solvers interface specifications		
CY 2003		Large dataset management and database connectivity implementations Gridded dataset implementations Numerical solvers interface implementations		Grid generation and problem setup specs Material properties interface specs Visualization component specs
CY 2004				Grid generation and problem setup components implementations Material properties interface implementations Visualization components implementations
CY 2005	Mature parallel component architecture implementation	Mature parallel component architecture implementation	Mature parallel component architecture implementation	Mature parallel component architecture implementation

	Grid Services				Networking	
	Grid Accessibility	Core Grid Services	Grid Resource Interface	Local Resource Manager	Band Width	Security
CY 2000	Kerberos-secured grid access services				OC-48 (2.5 Gb/s) parallel network	Encryptors operating at OC-12—created a parallel network of four stripes of OC-12
CY 2001	Grid administration Resource coordination via dependencies	Grid monitoring service	Grid interface for 10 teraOPS ASCI platform Grid interface for HPSS	Gang scheduler for 10 teraOPS ASCI platform		OC-48 ATM encryptor accredited Provide support for development of OC-192 IP encryptor
CY 2002	Kerberos-secured web access	Grid instrumentation	Grid interface for 30 teraOPS ASCI platform		OC-192 (10 Gb/s) parallel network	
CY 2003	Grid-aware programming components Personalization of grid environment	Limited delegation of credentials			Three OC-192 (30 bits) parallel network	
CY 2004			Grid interface for 100 teraOPS ASCI platform	On-demand scheduling	Six OC-192 (60 Gb/s) parallel network	
CY 2005	Grid scheduling				Ten OC-192 (100 Gb/s) parallel network	OC-192 encryptor accredited

II. TECHNOLOGY FOCUS AREAS AND ROAD MAPS



1. COMPUTATIONAL AND SOFTWARE ENVIRONMENT

ASCI scientists have been developing simulation codes for many years, but only in the last few years have advances in computational platforms paved the way for new, complex three-dimensional modeling and simulation viewed as the backbone of stockpile stewardship. The modeling codes themselves are useful only if the computational and software environment also keeps pace with the raw terascale computing power. In this section, we consider first the simulation development environment. This includes message-passing libraries such as MPI, thread constructs such as OpenMP, and related tools that allow the terascale platforms to handle the thousands of processors that comprise the hardware. Next, we discuss scalable solvers, which often form the heart of a computational simulation. Finally, we discuss the interaction between various pieces of software that allow for reuse and reliability.

Mary Zosel
(lead writer and tri-lab curve owner)
Lawrence Livermore National Laboratory
Livermore, California
Zosel1@llnl.gov
(925) 422-400

Jeffrey S. Brown
(tri-lab curve owner)
Los Alamos National Laboratory
Los Alamos, New Mexico
jeffb@lanl.gov
(505) 665 4655

Curtis Janssen
(tri-lab curve owner)
Sandia National Laboratories
Livermore, California
cljanss@sandia.gov
(925) 294-1509

Juan Meza
(lead writer and tri-lab curve owner)
Sandia National Laboratories
Livermore, California
meza@sandia.gov
(925) 294-2234

Joel Dendy
(tri-lab curve owner)
Los Alamos National Laboratory
Los Alamos, New Mexico
jed@lanl.gov
(505) 667-5929

Rob Falgout
(tri-lab curve owner)
Lawrence Livermore National Laboratory
Livermore, California
falgout2@llnl.gov
(925) 422-4377

John Ambrosiano
(lead writer and tri-lab curve owner)
Los Alamos National Laboratory
Los Alamos, New Mexico
ambro@lanl.gov
(505) 665-0457

Dan Quinlan
(tri-lab curve owner)
Lawrence Livermore National Laboratory
Livermore, California
quinlan1@llnl.gov
(925) 423-2668

Robert C. Armstrong
(tri-lab curve owner)
Sandia National Laboratories
Albuquerque, New Mexico
ROB@sandia.gov
(925) 294-2470

1.1 SIMULATION DEVELOPMENT ENVIRONMENT

As the size and complexity of ASCI simulations grow, so does the need for platforms with run-time systems that operate effectively at the largest scale. Current development environment systems, including parallel programming languages and diagnostic/debugging tools, need continual enhancement to serve the needs of production application codes. Also needed are a common set of standardized interfaces operable across multiple platforms and more resource allocation (including dynamic and distributed parallelism). Without these, ASCI researchers will be hampered in developing large-scale applications and optimizing their performance.

ASCI's computer platforms must have run-time systems with robust scalable parallel programming models and functional diagnostic/development tools to support applications that utilize systems with as many as 20,000 processors. A common set of interfaces is needed across multiple ASCI platforms to support portable parallel programming models that underlie ASCI physics simulation codes. An effective set of compilation, debugging, diagnostic, and performance tools must operate across a wide spectrum of program sizes and varying numbers of processors. Detailed insight as to total program behavior is needed to determine appropriate scalable algorithms.

The Simulation Development Environment (SDE) technology area (Figure 1.1-1) supports the run-time parallel library interfaces and tool set needed for code development. The run-time systems must operate effectively at the largest scale needed by ASCI high-fidelity physics simulation codes in order to take advantage of the ASCI hardware. Many current hardware platforms experience difficulty in reliability and scaling when applications run on more than a few hundred processors. Most current tools are not usable on more than tens of processors. Such tools must scale hand in hand with the application to diagnose issues related to code performance on many processors. It is not sufficient, however, to simply scale-up current tools. New paradigms are required for adapting the environment systems and tools to production codes that use literally thousands of processors. At the same time, the application developers will likely be adopting

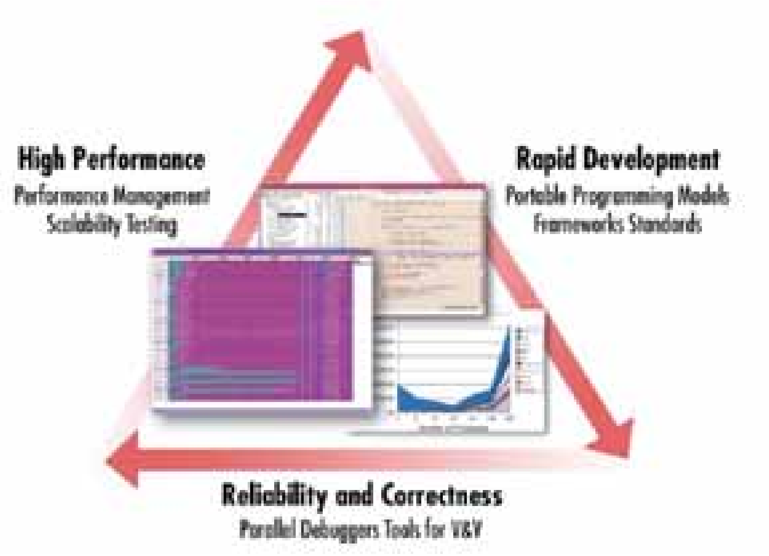


Figure 1.1–1. Simulation Development Environment.

new programming models and need to move between multiple platforms — adding more requirements to the development environment. Many government organizations are following the ASCI lead toward higher scale parallel platforms. All will benefit from the improvement of the development environment software.

TECHNICAL ISSUES AND CHALLENGES

The ASCI Problem Solving Environment (PSE) depends on working closely with the entire high-performance computing (HPC) community to meet the needs of a quality SDE. The overall strategy is (a) to promote standards; (b) to request software from the platform partners; (c) to work with independent software developers for portable second sources; (d) to encourage academic research; (e) to support local prototypes where appropriate; and (f) to fund full local development only in special cases. This approach and the associated technical challenges are detailed below.

The SDE strategy starts with promoting standards and standard compliance for development tools and application program interfaces. Portability across the ASCI platforms is very important to ASCI applications. Use of standard languages and interfaces also contributes to the quality of the development environment [1,2,3,4]. Furthermore, these standards provide design targets for more research and development. A prime example of a standard for ASCI-scale environments is the Message Passing Interface (MPI) and its successor MPI-2 [5,6]. MPI has become a de facto standard allowing portability of applications across a variety of architectures and vendors.

Building on the foundation of standards, ASCI works closely with our platforms partner developers to get high-quality supported products, thereby encouraging them to be innovative with regard to scalability, functionality, and performance. Where appropriate, ASCI also works closely with independent software vendors to provide portable tools that may be unavailable from the platform vendors or where significant improvement in functionality or portability can be found. The UltraScale Tools Initiative PathForward contracts with Etnus (debugging), Kuck and Associates (KAI)/Pallas (thread and MPI performance), and MPI Software Technology (MSTI) for MPI are an important part of this strategy. We also provide tools developed by independent software developers as backup to vendor-provided products for comparison purposes, error checking, and risk coverage. This is becoming an increasingly important part of the software strategy — especially as we observe the increased use of Linux-based systems.

In addition to the commercial-based solutions, there are local ASCI development and related academic research. In areas where ASCI has unique needs, or where timely commercial solutions are unlikely, we leverage these external efforts with local prototype development to provide example implementations and head-start development that may result in future commercial products. In some cases, the local development is so targeted to the needs of the users or environment that fully supported local products are required. PSE also funds academic developments that have a direct relationship to the longer term goals and/or offer the prospect of providing future employees in an area where hiring is difficult.

Finally, to complement all of the above approaches, PSE works very closely with ASCI users. We try to anticipate their needs without forcing a particular solution and, where feasible, provide them with backup alternatives in instances of product failure. A representative sampling of codes from all three labs is used to test the functionality of the SDE. The results of these tests are used to provide feedback to platform partners and third-party software developers, as well as direct local and academic research and development (R&D).

Universities and industry have been actively developing parallel machines and associated software for well over ten years now, and as outlined above, their contributions are very important [7]. Basic compiler technology is relatively mature, with the possible exceptions of full standards compliance, memory access optimization, and high performance implementations of Java. MPI 1.2 implementations are available from academia, government laboratories, and industry, and progress is being made toward a full MPI 2. Universities are conducting research in performance analysis tools and operating system (OS) bypass network drivers, but most of this work is targeted towards machines with tens of processors to a few hundred processors.

The factors limiting external development targeting machines required by ASCI are lack of access to both the latest platforms of sufficient size, access to the large target codes/data, and complications associated with proprietary software interfaces. These issues are difficult to overcome, even inside the laboratories — and are much more problematic for external R&D — but to deploy production-quality results, they must be addressed.

In general, the parallel run time (message and thread support) is the first component of the parallel system that must meet the ASCI challenge. If the run time cannot support ASCI-scale execution, then all other parts of the support (input/output, visualization, resource management, etc.) are moot. Message systems for a few hundred (and, in some cases, a few thousand) processors are relatively mature. However, message systems face new problems when using new forms of system interconnect. The situation will be further complicated in machines with more than 10,000 processors. Message system scaling and optimization will continue to be a challenge. Multi-threading is another feature that has recently emerged in scientific codes, and standards controlling the treads such as OpenMP are just now being put in place [8]. On symmetric multiprocessor (SMP) platform nodes, threading used together with messaging is one approach to address message scaling and code memory requirements. Full integration of the threaded environment is not yet complete, and the effects of how threads interact with process scheduling and nonuniform memory access (NMUA) memory are new challenges for SDE.

Running applications on 10,000 and more processors presents challenges to both application programmers and system software developers. The traditional technique of rerunning a calculation after a system failure may no longer be practical on machines with so many processors that several node failures would be expected in the course of a run. System developers must provide new tools to make fault recovery as simple and efficient as possible, and new programming paradigms must be developed to let application programmers take advantage of these tools. Applications are also expected to need increasingly more complex forms of resource allocation, including dynamic and distributed parallelism.

The associated parallel program development tools face challenges of a different sort. The problems of debugging and tuning at high scale are still research areas, with a number of issues in common with feature detection faced by the Visual Interactive Environment for Weapons Simulation (VIEWS) area. Debugging optimized code will continue to be an issue. The tools have to track the new platforms and any unusual system features. Dynamic instrumentation [9] is a known approach to adapting performance tools to scalable data collection, but has not yet been deployed, except in debuggers. To be most useful, these development tools and techniques need to be available on multiple parallel platforms. They also need to track the new programming models and features adopted by the code development teams. Use of cutting-edge or nonstandard technology in either of these areas (platforms or applications) presents new challenges for the tools (e.g., as the code development teams start using Java, then new interfaces are needed in all run-time libraries and development tools).

A continuing important challenge comes from optimal use of memory systems. Even with larger caches and aggressive memory architectures, new tools and techniques for understanding memory behavior and continuing research into compiler optimization and/or language directives for memory latency are needed.

The final challenges come from addressing the rate of change in the ASCI requirements. In addition to the requirements from the accelerated schedule for production platforms, there are possible advanced architecture research platforms, the requirements from the code developers are still changing, and the needs related to code verification and validation are not yet well understood.

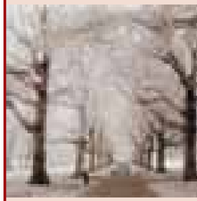
Simulation Development Environment Road Map

The associated technology road map visually depicts the five-year status (calendar year 2000 to 2005) of desired capabilities/activities within a functional area. Each capability is color coded to show what level of R&D effort it requires or anticipates.

R&D Effort Indicator:

Accomplished = completed
Planned = ASCI will accomplish even with slight budget fluctuations
Hurdle = ASCI will need some help
Barrier = ASCI will need significant help from the high-performance computing (HPC) community

NOTE: Both hurdles and barriers represent research opportunities for the HPC community.



Road Map for SIMULATION DEVELOPMENT ENVIRONMENTS CAPABILITIES

Functional Area	CY 2000	CY 2001	CY 2002	CY 2003	CY 2004	CY 2005
Development Tools	Debug and tune to 2000 CPUs Memory measurement tools Open-source tool infrastructure	Debug and tune to 4000 CPUs Scalable measurement and diagnosis components Scheduling for thread/memory affinity	Mixed parallel model performance tools Multi-platform open-source tool infrastructure	Debug and tune to 10,000 CPUs Scalable measurement and diagnosis components Tools support new language/parallel models	Large-scale open-source tools Distributed platform performance testing	
Parallel RTS	Thread-aware MPI scales to 4000 tasks		MPI2 available	MPI2 scaling to 10,000 tasks Dynamic parallelism support RTS for distributed parallel applications		
Programming Models			Scalable OpenMP	High Performance Java Compiler/language technology for serial performance	Compilers support for current standards	
SDE Deployment	10 teraOPS SDE		SDE scaling tests to 10,000 CPUs 30 teraOPS SDE		SDE scaling tests to 15,000 CPUs	Production scalable SDE for milepost 100 teraOPS SDE

R&D Effort Indicator ● ACCOMPLISHED ● PLANNED ● HURDLE ● BARRIER

ACCOMPLISHED—Completed
PLANNED—ASCI will accomplish even with slight budget fluctuations
HURDLE—ASCI will need some help from the HPC community
BARRIER—ASCI will need significant help from the HPC community

TIMELINE

The timeline elaborates the preceding road map and is a snapshot in a dynamic area. The requirements change based on both new application needs and new architecture developments. Scaling requirements depend on configuration and can be expected to track about half of the total processors available on new production ASCI platforms. Some items such as standards and compiler serial efficiency are ongoing requirements and are listed in a specific year, simply as a reminder that these issues are a continuing requirement.

Calendar Year **Description and Status**

2000 **Debug and tune to 2000 Central Processing Units (CPUs)** – The scalability of the TotalView effort continues, including new interfaces to sub-select processors of a large run. In addition, new statistical tools for measuring MPI scalability and improved MPI tracing functionality are deployed. **Accomplished**

Memory measurement tools – Memory tools are a high-priority concern of the user community. A variety of memory tools are now available and more are needed. On the IBM systems, two commercial tools are now deployed, and a cache performance tool MPX is available. On SGI systems, Rice University also has a cache performance tool available. **Hurdle**

Open-source tool infrastructure – The first components of what is expected to become an open-source tools infrastructure based on dynamic instrumentation are deployed on the IBM systems. The components include a source click-back interface and multiple performance measurements built on top of an IBM instrumentation system. **Hurdle**

Thread-aware MPI scales to 4000 tasks – The thread-safe MPI from IBM is extended to support jobs up to 4096 tasks. **Hurdle**

10-teraOPS SDE – The ASCI White platform (actually 12 teraOPS) is delivered to Lawrence Livermore and the PSE major Level 1 milestone and its tasks demonstrate that the development environment is ready for applications development activities. **Planned**

2001 **Debug and tune to 4000 CPUs** – In support of ASCI application milepost requirements, the tools for debugging and tuning applications will be tested with larger applications. Additional scaling techniques are planned. **Planned**

Scheduling for thread/memory affinity – Cache management with dynamic threads is an issue on any SMP system, but it is even more important on a shared memory NUMA system. The location of threads with respect to data is critical to performance. Techniques for scheduling are needed for current and future systems. **Hurdle**

Scalable measurement and diagnosis components – Additional performance components will be added to the infrastructure with emphasis on better techniques to track back to the part of the program or system that is contributing to performance problems. **Planned**

2002 **Mixed parallel model performance tools** – New products from the ASCI PathForward efforts are expected that will support a mix of OpenMP and MPI at large scale. **Planned**

Multiplatform open-source tool infrastructure – In order to leverage performance tool development, the underlying infrastructure components need to provide a portable Application Programming Interface (API) that can be used across multiple platforms. The availability across multiple platforms is considered vital to making research and development in scalable tools practical in both the academic and commercial arena. The interfaces for dynamic instrumentation must be extended, and the interface to the parallel systems must be made portable. **Hurdle**

MPI 2 available – Working with Argonne National Laboratory (ANL) and the platform partners, support for the MPI 2 standard will be made available on ASCI platforms. Portions of the standard are already deployed. **Planned**

Scalable OpenMP – OpenMP compiler implementations that minimize overhead and scale to larger thread counts efficiently are needed. The associated tools to identify the barriers to better thread efficiency are also important. **Hurdle**

SDE scaling tests to 10,000 CPUs – With the new 30-teraOPS Compaq platform, application scaling tests will extend to even larger processor counts. The software deployment initial tests of early technology from the 100-teraOPS platform are also expected. **Planned**

SDE scaling tests to 10,000 CPUs – With the new 30-teraOPS platform, application scaling tests will extend to even larger processor counts. The software deployment initial tests of early technology from the 100-teraOPS platform are also expected. **Planned**

30-teraOPS SDE – The application development environment on the 30-teraOPS platform will be completed and demonstrated. This will include a combination of Compaq software, third-party software, and ASCI-developed interfaces and libraries. **Planned**

2003 **Scalable measurement and diagnosis components** – The set of measurement and diagnostic tools available is expected to continue to grow as the infrastructure becomes more widely available and the system architectures change. **Hurdle**

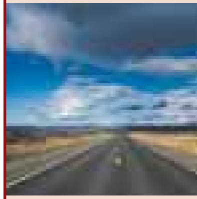
Debug and tune to 10,000 CPUs – The need for debugging and tuning techniques will continue to grow with the number of platform processors and the size of the ASCI application runs. **Hurdle**

Tools to support new language/parallel models – Experience with the ASCI environment shows that periodically one can expect developers to change to the use of a new "standard" language, and/or to a parallel run-time model. This change, in turn, introduces new requirements on the rest of the environment: library interfaces, tools, etc. This timeline entry is therefore temporary since it anticipates such requirements. Possible examples are new experimental architecture support, extended memory support, Java support, fault-tolerance support, and/or dynamic and distributed support. **Hurdle**

Dynamic parallelism support – Currently, MPI and the various components of the parallel environment do static resource allocation, but as the applications grow more into adaptive methods and load-balancing techniques, many of the ASCI code teams expect that they would benefit from resource partitions that could grow or shrink with the applications requirements. Issues of fault tolerance also require more dynamic resource allocation. **Hurdle**

MPI 2 scaling to 10,000 tasks – MPI 2 scaling to thousands of tasks has many challenges. Efficient mechanisms to implement collectives and input/output (I/O) are needed, as well as a way to adapt to new memory and switch architectures. **Hurdle**

Run-Time System (RTS) interfaces for distributed parallel applications – The need for coupled applications is anticipated. This may come from either an application using multiple runs to do calculations on disjoint parts of the problem or from separate but coordinated processing of visualization and I/O management. The run time and tools will have to be extended beyond single program multiple data (SPMD) support. Existing grid and component research is expected to contribute to development of support in this area. **Hurdle**



High-performance Java – Prototype and problem setup activities are starting to move to Java, and this trend is expected to continue. As more numerically intensive work is done, there will be a need for Java compilers that approximate C and C++ in performance. This is an area where ASCI requirements are trailing the requirements of the general community, so ASCI will depend on the commercial market to produce these compilers. **Barrier**

Compiler/language technology for serial performance – The relative efficiency of an application on a single CPU is already a recognized issue and will become more challenging as faster processors with more complicated memory systems become available. Continued research and development in this area is critical to the effective use of the large ASCI platforms. This technology item is really an ongoing challenge that needs to improve every year. **Barrier**

2004 **Large-scale open-source tools** – On the larger systems, tool development will need to consider issues related to the entire system load, configuration, and environment, rather than just an individual application. The most feasible development for such tools is expected to include integration of open-source components from a number of R&D projects. **Hurdle**

Distributed platform performance testing – As the complexity of the coupling of ASCI platforms continues, there will be a need for tools that correlate the performance on one platform with that on another (e.g., coupled visualization and storage, or systems involved in remote site calculations and transfers). **Hurdle**

2004 cont. **Compiler support for current standards** – There is an ongoing need for compilers to track the latest ISO language standards as well as the de facto standards for parallel applications. This entry on the SDE chart is a reminder that this effort will not stop with the vintage 2000 Fortran, C, C++, and OpenMP standard implementations. The compilers must continue to evolve and track new standards. **Barrier**

SDE scaling tests to 15,000 CPUs – With the newest ASCI platforms, application scaling tests will extend to even larger processor counts. **Planned**

2005 **Production scalable SDE for milepost** – ASCI Problem Solving Environment has a programmatic major milestone to deliver production-quality uniform user and application interfaces on all ASCI platforms. Work toward this goal will emphasize "uniform" and "production-quality." By this time in the evolution of the ASCI platforms, there should be stability in the parallel software environment for the production ASCI platforms. **Hurdle**

100T SDE – The application development environment on the 100 teraOPS platform will be completed and demonstrated. This will include a combination of platform software, third party software, and ASCI-developed interfaces and libraries. **Hurdle**

CURRENT STATE OF ASCI SDE

To date, a variety of accomplishments have contributed to improvement in the software development environment on the ASCI platforms. The specification of software requirements in procurement contracts has helped engage the platform partners in the solution. The ASCI UltraScale PathForward contracts with Etnus (debugging), KAI (thread and performance tools), and MSTI for MPI are accelerating functionality in independent software vendors' products that are portable across platforms. ASCI is a key driver in pushing the cross-platform OpenMP standard for thread access. SDE research and development have contributed significantly to the viability of the debugging tools. Techniques are emerging for both debugging and performance tools that will facilitate usability with thousands of processors. A new contract with ANL is in place to expedite the availability of a modular and portable MPI 2 reference implementation. Finally, a close, continuing SDE association with the ASCI code development teams is mutually beneficial because SDE learns first-hand of new requirements while providing help with their problems. But any ASCI code developer will say that the tools and interfaces are still far from the complete, easy-to-use parallel environment that they would like to see. And the coming architectures present new challenges, and the complexity of understanding code performance still leaves many unanswered questions.

SUMMARY, CONCLUSIONS, RECOMMENDATIONS

The development environment software for the future ASCI platforms presents a challenge that can benefit from substantial help from the external community. Specifically, the following areas are identified where the HPC community can help accelerate the software needed by ASCI:

- Compiler technology: serial performance, with special emphasis on debugging of optimized codes, optimization of memory access for NUMA systems and also for codes with unstructured memory accesses that do not fit in cache.

- Programming models: maturing of the Java environment so that competitive compilers exist and it is a full partner with Fortran/C/C++ in run-time interfaces (e.g., MPI bindings) and development tools.

- Parallel models: extension of environment for dynamic parallelism and distributed applications, and to other environments introduced by high-end experimental systems.

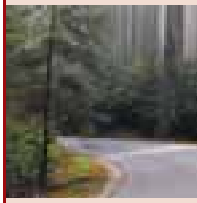
- Message interfaces: continued optimization of the communication layer to eliminate data copies and scale to thousands of tasks.

- Debugging: debugging of optimized code, improved understanding of memory errors and usage, and parallel (threaded and distributed) code verification techniques.

- Performance tools: a fully scalable, multiplatform instrumentation system that can be configured/extended for multiple kinds of performance measurement.

- Portability: to protect the code developer's investment and flexibility, all models, interfaces, and tools are most useful when they exist on multiple platforms of a variety of sizes.

The ASCI PSE Tools and Run-Time System Software project is working toward a robust and integrated simulation development environment that will cross all ASCI platforms. Together with the HPC community, we can create a truly usable environment that will in turn benefit a wide community of users of the high-end platforms.



REFERENCES

1. Pancake, C. M., Blaylock, B., and Ferraro, R. *Guidelines for Writing System Software and Tools Requirements for Parallel and Clustered Computers*, Technical Report 95-80-11, Department of Computer Science, Oregon State University, November, 1995. http://www.nero.net/~pancake/SSTguidelines/index.html

2. Pancake, C. M., and McDonald, C., eds., *Guidelines for Specifying HPC Software*, Northwest Alliance for Computational Science and Engineering's Task Force on Requirements for HPC Software and Tools. http://www.nacse.org/projects/HPCreqts

3. Simple, Portable, Scalable SMP Programming: OpenMP organization and documents. http://www.openmp.org/

4. MPI Forum documentation. http://www.mpi-forum.org/

5. Snir, M., et al., *MPI — The Complete Reference: Volume 1, The MPI Core* (second edition), MIT Press, 1998.

6. Gropp, W., et al., *MPI — The Complete Reference: Volume 2, The MPI Extensions*, MIT Press, 1998.

7. Koniges, A. E., *Industrial Strength Parallel Computing*, Morgan Kaufmann Publishers, 2000.

8. Chandra, R., et al., *Programming in OpenMP*, Morgan Kaufmann Publishers, 2000.

9. *The Dynamic Probe Class Library (DPCL) reference document.* http://www.rs6000.ibm.com/doc_link/en_US/a_doc_lib/sp32/pe/html/d3d60mst.html

1. 2 SCALABLE SOLVERS

Numerical algorithm research is the investigation of numerical methods to evaluate accuracy, convergence rates, robustness, computational performance, and scalability on ASCI-level problems. The algorithms required for the modeling and simulation of the high-fidelity physics packages rely on scalable solvers. Issues of scale, both in terms of problem size and number of processors, are critically important to ASCI — balancing workloads across thousands of processors in a heterogeneous computing environment. To meet ASCI's goals, significant collaborative research is needed to integrate disparate codes over a variety of time scales and solution methods.

The development of three-dimensional, high-fidelity applications that can be used to implement virtual testing and prototyping is fundamental to ASCI's goals. Early in ASCI's development, it was recognized that to meet ASCI's future goals, a minimum of a 10,000-fold speedup was required in computing capabilities. This speedup was to be achieved through a combination of a 100x speedup in the computing platforms and a 100x speedup in the algorithms. The algorithms used for ASCI physics modeling and simulation rely on scalable, efficient, and accurate solvers. At the heart of many of the algorithms lie solvers for systems of linear and nonlinear equations. In many cases, a substantial fraction of the computational time in a simulation is spent within either the nonlinear or linear solver. As such, it is critically important to the ASCI program to develop and implement solvers that scale to the size problems envisioned for the full-system simulations as well as to the tens of thousands of processors that ASCI future computer platforms will use. Without the expected speedup from the algorithms and solvers, the time to solution for any one of the simulations required for the Stockpile Stewardship Program (SSP) would become prohibitively long.

Scalable solvers are similarly needed in modeling and simulation efforts in programs other than ASCI. For example, the DOE Office of Science reports [1] the need for nonlinear solvers capable of solving systems of equations with 275 million unknowns in a computational chemistry application. In this same report, the Office of Science's investment plan calls for the development of mathematical "algorithms that scale to thousands, and eventually millions of processors."

TECHNICAL ISSUES AND CHALLENGES

The scalable solution of large complex systems on massively parallel computers is tightly coupled to achieving overall performance of simulation codes. By *scalable*, we generally mean the ability to use additional computational resources efficiently to solve increasingly larger problems. For solvers, it is useful to divide the notion of scalability into two basic components: *algorithmic scalability* and *implementation scalability*. Algorithmic scalability requires that the computational work grow linearly (ideally) with problem size. For example, in the context of iterative methods for the solution of linear systems of equations, scalability usually means that the number of iterations required for convergence does not increase with problem size. Implementation scalability (sometimes known as architectural scalability) requires that a single iteration take the same amount of time if both the problem size and the number of processors are increased by the same factor. Both are necessary to achieve overall scalability.

Because most of the problems that we are addressing arise from discretizations of coupled systems of nonlinear partial differential equations, we can split the solution of these systems, in the simplest case, into two phases: an outer nonlinear solver and an inner linear solver. More general forms of this framework, with multiple forms of nesting, are also possible [2].

Nonlinear Solvers

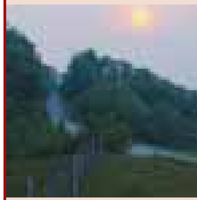
In the nonlinear phase of the solution, there are three major features of the types of problems that we must address. The first aspect deals with the problem decomposition. Modeling of nonlinear processes has usually been accomplished by decoupling and then linearizing the basic physics equations. By applying these techniques, scientists simplified their models sufficiently that problems of moderate size could be solved. However, this requires a careful balance between the nonlinear and the linear solver, and may lead to inefficiencies in the solution method. The second feature concerns the use of more complex higher fidelity physics models in the ASCI simulations. The third feature deals with the goal of coupling single discipline simulations to study complete system responses. Both of the last two requirements change the fundamental characteristics of the nonlinear equations. Traditional methods, which have been developed to address single-discipline problems, are usually neither efficient nor scalable.

Traditionally, variations of Newton's method have been the most efficient and robust algorithms for nonlinear problems. Improvements in performance have been achieved through a variety of modifications to the basic algorithm including an approach known as the Inexact-Newton method. Here, one is able to reduce the amount of work required at each step, and often improve robustness as well, by recognizing that in the earlier iterations of the algorithm, when one is far away from the solution, exact linear solutions are not necessary. Thus, one can relax the linear solution tolerance (assuming an iterative method), thereby saving on computational expense, and only "tightening up" the linear solution accuracy as convergence is approached in the nonlinear portion. *While there has been extensive research on this approach, it tends to be problem-specific, and more and better options for doing this are needed.*

Presently, Jacobian-free Newton-Krylov methods may offer the best general-purpose tool for solving a wide variety of multidisciplinary problems. However, the viability of these methods for large-scale multiphysics simulations depends upon the efficient preconditioning of the Newton-Krylov iteration. The development of preconditioners in this setting is viewed as a barrier to the overall approach. *More research is needed in constructing physics-based preconditioners for multiphysics systems such as radiation hydrodynamics, magnetohydrodynamics, and solidifying flows.*

Further research is also needed in the area of multigrid-based algorithms for nonlinear equations. Currently, there is ongoing research into Newton-Krylov-Multigrid solvers that would allow the fully implicit formulation of a problem to be solved, thereby removing the need for very small time steps and providing more accuracy to the overall solution. An alternate method for solving nonlinear systems is full approximation schemes (FAS) or nonlinear multigrid. This method may require less storage than the Newton-Krylov-Multigrid methods, but may not be as robust. As in the linear solver case, these two approaches can be combined in various ways to improve scalability, robustness, and overall efficiency.

Added difficulties arise when two or more simulation codes are coupled to solve a full system problem. One approach is to formulate the problem as a system of nonlinear equations divided into a few large blocks, one for each of the sub-problems. The goal is to solve the coupled system of nonlinear equations simultaneously. For example, one type of surety study being done at Sandia requires the coupling of a structural analysis code to a thermal analysis code to study the full system. At Los Alamos, studies involved in the modeling of melt convection require the simulation of five coupled partial differential equations including simultaneously solving for fluid flow, heat transfer, and phase change. *Although, the nonlinear equations problem is interesting in its own right, the data transfer problem is equally important.* How does one get the output of one block into the form required as input by another block? One approach is to use spline fits to compress the data into the spline coefficients. Then only the coefficients are passed, and the splines are used to generate the data needed for the various receiving solvers. As the size of the problem increases, this method for solving the data transfer problem may not scale, and further research will be required.



A closely related set of problems involves the optimization of various types of model, e.g., the W76 AF&F (arming, fuzing and firing) shown in Figure 1.2-1. In engineering design optimization, for example, the function evaluations typically consist of large-scale simulation codes based on

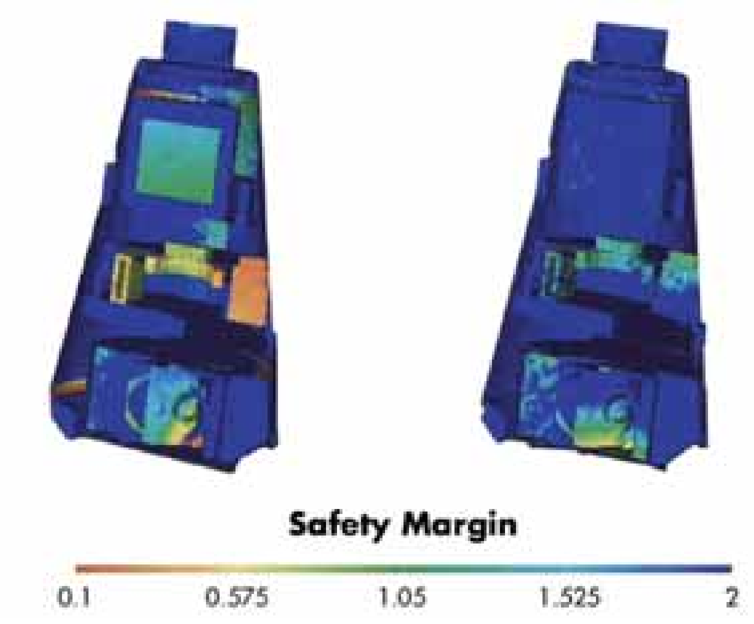


Figure 1.2–1. Dakota Optimization of W76 replacement AF&F.

partial differential equations (PDEs) that are computationally expensive and dominate the cost of the optimization algorithm. In addition, most of these simulations do not compute derivative information, which is a requirement of many traditional optimization methods. Finally, because the objective functions are based on the solution of a PDE, there is inherent noise in the evaluation of the function. Traditional approaches to solving standard optimization problems have been to develop algorithms that require fewer function evaluations. These approaches also usually assume the availability of derivative information and noise-free function evaluations. As the size and complexity of our problems grow, these approaches have proven to be inadequate and have become a barrier to solving design optimization problems that arise in the ASCI program.

A new approach based on using approximation models of the function in order to solve the original optimization problem has recently received some attention. The general idea is to use a computationally cheaper model to approximate the original problem and only evaluate the more expensive model when absolutely necessary. Further research needs to be done with respect to adjusting the fidelity of models and using parallel methods to improve the efficiency of these methods.

The ultimate goal, however, is to develop algorithms for nonlinear equations and optimization within a framework that takes into account uncertainties in the design parameters and models used within the ASCI simulations. One must be able not only to assess the accuracy of the solutions but also to quantify the uncertainty in those solutions given real-world uncertainties in the simulations. Areas that will require future investigations include optimization under uncertainty, robust optimization, stochastic optimization, and sensitivity analysis. These areas represent some of the biggest barriers in the entire area of scalable solvers.

Linear Solvers

At the lowest level of this solver hierarchy is the linear system solver. Normally, a sparse linear system of equations results from the discretization and nonlinear solution step. Because of their superior scaling properties, iterative methods are typically the methods of choice for large problems on parallel computers. However, direct methods are still important in a variety of situations, for example, as a preconditioner for an iterative method or where the linear system results in a matrix that is not sparse, is highly ill-conditioned, or has little identifiable structure. When sparse direct methods are used to solve linear systems, the data usually reside completely within each processor. In some cases, however, where the linear system may be highly ill-conditioned, it may make sense to have a medium-coarse grid that spans several processors as the coarsest grid in a multigrid method in order to achieve a reasonable convergence rate. In both of these cases, it is critical that the solver implementation be tuned to the processor architecture with careful attention paid to cache and memory hierarchies, as well as the bandwidth between them. Some work already exists in automatically tuning preconditioners based on past history of solvers and problem characteristics, as well as automatic tuning of higher level choices [3], but further work is needed to address new problems and new computer architectures.

Robustness is also an important issue, especially for simulations that may run for days on an ASCI computer. Unfortunately, robustness and scalability issues compete with each other. For example, direct methods are extremely robust but not scalable. Krylov subspace methods are fairly robust, but robustness and, in addition, scalability depend heavily on the choice of a good preconditioner. Multilevel methods are highly scalable but tend to be problem-specific. For example, two different approaches, geometric multigrid and algebraic multigrid, are needed for the case of problems with structured meshes and unstructured meshes, respectively. Algebraic multigrid is particularly attractive in the unstructured mesh case since it does not require any geometric information to build the coarse grids. Another approach, which ASCI researchers have followed, is to use multilevel methods as preconditioners for Krylov subspace methods to combine the best features of both.

It is well known that for iterative methods to be effective, good preconditioners must be developed and implemented. Consequently, one of the main technical barriers to scalable linear solvers is the development of effective preconditioners. Currently, a popular preconditioning technique involves an additive Schwarz preconditioner based on varying levels of incomplete factorizations in processor subdomains with arbitrary levels of overlap between processors. However, there are many other methods, which may be used in combination, that promise improved convergence properties and need to be investigated. These include, but are not limited to, improved matrix reorderings and scalings, and multilevel and multigrid methods. It is quite likely that the most efficient and robust preconditioner for a multiphysics simulation will result from a solid understanding of the physics problem. A physics-based preconditioner for a Newton-Krylov method will be based on a divide-and-conquer approach, applied to simple linearizations, to construct a preconditioning process. This could be viewed as physics-based domain decomposition. Alternatively, this approach could be viewed as an outer Newton-Krylov iteration controlling the nonlinear convergence of an inner preconditioner based on operator splitting.

Robust, scalable preconditioners for solving problems on structured meshes already exist. However, these algorithms are expensive and have large memory requirements. Also, although these preconditioners are theoretically scalable, developing implementations that scale well has proven difficult. For problems on unstructured meshes, the development of robust, scalable preconditioners is even more difficult. Since multilevel methods currently offer the best hope for scalability, e.g., algebraic multigrid, the main research issue to address here is the selection of coarse grids in parallel. The traditional algorithms for doing this are inherently sequential, and so far, parallel variants of these algorithms produce unscalable solvers. Other options would include using some geometric information from the original grid to produce a coarse grid and applying a hybrid algebraic/geometric method.

Road Map for Scalable Solvers

The associated technology road map visually depicts the five-year status (calendar year 2000 to 2005) of desired capabilities/activities within a functional area. Each capability is color coded to show what level of R&D effort it requires or anticipates.

R&D Effort Indicator:

Accomplished = completed
Planned = ASCI will accomplish even with slight budget fluctuations
Hurdle = ASCI will need some help
Barrier = ASCI will need significant help from the high performance computing (HPC) community

NOTE: Both hurdles and barriers represent research opportunities for the HPC community.

Road Map for
SCALABLE SOLVERS CAPABILITIES

Functional Area	CY 2000	CY 2001	CY 2002	CY 2003	CY 2004	CY 2005
Linear Solvers	Parallel algebraic multigrid codes		Eigensolver package for systems with 30M DOF Scalable algebraic MG for 1B unstructured mesh		Eigensolver package for systems with 300M DOF Linear solver package for 10B unstructured mesh	
Nonlinear Solvers/ Optimization		Nonlinear optimization library for 8000+ processors		Constrained optimization tool kit for systems with 100M DOF		Optimization under uncertainty tool kit

R&D Effort Indicator 🔵 ACCOMPLISHED 🟢 PLANNED 🟡 HURDLE 🔴 BARRIER

ACCOMPLISHED—Completed
PLANNED—ASCI will accomplish even with slight budget fluctuations
HURDLE—ASCI will need some help from the HPC community
BARRIER—ASCI will need significant help from the HPC community

TIMELINE

This following timeline elaborates on the desired capabilities shown on the preceding road map. These capabilities are also goals that we have set within our program to be able to satisfy the requirements of the application codes. In all cases, we assume that scalability means that a particular solver scales (at a minimum) to one-half of the processors on the largest machine available at that date. In addition, the algorithms are assumed to be able to run on all of the ASCI computing platforms that are available within the given timeframe. The major areas that we have chosen to address are linear solvers, eigensolvers, nonlinear solvers, and optimization algorithms. These four areas encompass a majority of the numerical solvers that are needed within our modeling and simulation capabilities. In addition, breakthroughs in these areas would have both an immediate and a dramatic impact in our ability to achieve our goals.

Calendar Year — Description and Status

2000 **Parallel algebraic multigrid (MG) codes** – An algebraic multigrid solver for systems of linear equations arising from applications with unstructured meshes with scalability up to 8000 processors. **Accomplished**

2001 **Nonlinear optimization library for 8000+ processors** – A software package with a capability of running on 8000+ processors using multiple levels of parallelism. **Planned**

2002 **Eigensolver package for systems with 30M Degrees of Freedom (DOF)** – A nonsymmetric eigensolver package capable of handling at minimum 30 million equations. **Planned**

Scalable algebraic MG for 1B unstructured mesh – A linear solver package containing algebraic multigrid (AMG) solvers capable of handling up to 1 billion equations for the unstructured grid case. **Hurdle**

2003 **Constrained optimization tool kit for systems with 100M DOF** – A nonlinearly constrained optimization software package capable of handling coupled nonlinear systems with up to 100M degrees of freedom. **Hurdle**

2004 **Eigensolver package for systems with 300M+ DOF** – A nonsymmetric eigensolver package capable of handling at minimum 300 million equations. **Hurdle**

Linear solver package for 10B unstructured mesh systems – A linear equation solver package containing methods with the capability to solve systems of linear equations with up to 10 billion equations arising from application problems that use unstructured meshes. **Hurdle**

Optimization under uncertainty tool kit – A tool kit for performing optimization on problems with uncertain or stochastic design parameters and algorithms for robust optimization and sensitivity analysis. **Barrier**

CURRENT STATE OF ASCI SCALABLE SOLVERS

Two different variations of a geometric multigrid method for structured grid problems have been developed, incorporated into existing solver packages, and applied to several milestone calculations. The largest calculation we have run to date has involved the solution of a linear system of equations with 1 billion degrees of freedom on 3150 processors of ASCI Red. We have also been able to solve linear systems with up to 100 million equations on problems with unstructured meshes. In the area of eigensolvers, a recent Sandia calculation computed the 400 smallest eigenvalues from a system with 5 million unknowns using 2000 processors.

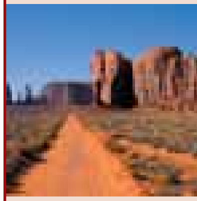
In the area of nonlinear equations and optimization, we have developed several tool kits with the capability of handling design optimization problems. The tool kits contain various algorithms that can be used to solve a number of simulation-based optimization problems. These algorithms are capable of running in parallel and can handle objective functions with and without analytic derivatives. In one design optimization problem that took four days using 2560 processors of the ASCI Red machine, the optimization algorithms were able to improve the solution by more than an order of magnitude.

All of the accomplishments mentioned here have come about because of previous strong research programs in algorithm research. This research, which has been in collaboration with university researchers, had developed the initial algorithmic ideas, and the ASCI program helped to support the development of the mathematical software needed to make these algorithms more robust and efficient for the large-scale applications required by ASCI.

SUMMARY, CONCLUSIONS, RECOMMENDATIONS

Algorithm research is traditionally a long lead-time activity and typically requires years before an algorithm is embodied in a robust, reliable, and scalable code. In addition to the research areas outlined above, there are several integrating issues that must be addressed.

A major barrier is the integration of disparate codes with different structures, discretizations, and methodologies of dynamically modeling problems. Large-scale simulations based on PDEs can include various geometric discretizations including regular meshes, block meshes, and adaptive mesh refinement. The PDE discretizations include finite difference, finite volume, finite element, spectral element, and particle methods. Finally, meshes can be Eulerian (fixed node), Lagrangian (moving node), or Arbitrary Lagrangian-Eulerian; furthermore, they may be adaptively refined. Developing codes that combine these disparate computational methodologies will require a careful analysis on the effects of communicating boundary conditions between codes. For example, certain boundary conditions can significantly deteriorate the convergence of multigrid methods. Obtaining boundary conditions dynamically from another code will likely only worsen the situation.



A second barrier to improving algorithms is the traditionally long lag times between the creation and prototyping of an algorithm and its efficient implementation in an application. One source of this delay is the absence of practical performance models for ASCI platforms. Present computer architecture performance modeling methods are generally too simplistic and fail to address the levels of complexity introduced by large-scale and multibox machines with deep memory hierarchies and heterogeneous network layers. Secondly, there is little work on mapping algorithms onto the architectures. The net effect is that given a set of algorithms, only rough heuristics exist to select among promising ones, leaving full implementation and testing the only option. Developing practical performance models may require a mixture of analytic and discrete event modeling, but the usefulness of such models in the first generations of parallel machines suggests they will be good tools for predicting performance. Particular examples of current work show that with small (8 to 64 processor) machines the time per iteration for Krylov solvers can be accurately modeled, and discrete event simulation models have provided a better cost function for domain decomposition methods.

Two other issues to be addressed are fault tolerance and load balancing. Fault tolerance and restart and recovery mechanisms will be crucial for ASCI platforms where the mean time between failure (MTBF) can be on the order of days. New algorithms and solvers that can adapt to these environments need to be developed for these platforms. Some early research is investigating the use of specific algorithm characteristics to recover or sometimes even ignore certain faults. Another possibility includes the use of asynchronous methods, ignored until recently, in an attempt to address fault tolerance. The other area, load balancing, can have a significant impact on underlying numerical algorithms. The numerical methods may provide local and predictive error measures for highly dynamically changing loads, and a normally scalable algorithm can be reduced to serial execution speeds by an inappropriate load balancer.

Another issue that must be addressed is deciding when the computed solutions produced by both the nonlinear and linear solvers are accurate enough. Part of this issue falls under the area of uncertainty quantification. This issue might also be addressed by incorporating some form of sensitivity analysis into the major solver packages. In either case, this area will require a large effort to develop new algorithms and tools that can be used to answer these questions.

In summary, to stay on the curve will require a combination of a strong and active research program in algorithms, which would involve close collaborations with university researchers, and support for the implementation of new algorithms in the ASCI application codes.

REFERENCES

1. U. S. DOE Office of Science, *Scientific Discovery through Advanced Computing*, March 24, 2000. http://www.sc.doe.gov/images/news_photos/SDAC_Overview_000330.pdf

2. X.-C. Cai and Keyes, D. E., "Nonlinearly Preconditioned Inexact Newton Algorithms," submitted to *SIAM J. Sci. Comp*, 2000. http://www.math.odu.edu/~keyes/papers.html

3. Zhaojun, Bai, et al., *Templates for the Solution of Algebraic Eigenvalue Problems: A Practical Guide*, SIAM Press, 2000.

1.3 SOFTWARE INTEROPERABILITY

A growing need for ASCI scientists is to be able to rapidly develop software using interoperable software modules that can be easily and reliably integrated. Building computational frameworks — reusable design patterns expressed as clearly defined software abstractions — is one approach. However, their application, use, and maintenance remain a challenge. Exploiting component technology is another approach widely adopted in industry. For ASCI to stay on an upward curve, more research is needed to explore both these strategies, especially in the context of high-performance computing.

In the future, current methods of developing software for ASCI are likely to become too costly, time consuming, and unreliable to scale up to the level of effort required to fulfill the ASCI mission. Industry is facing similar problems. Concerted efforts to achieve code reusability and interoperability may serve to minimize these difficulties.

We envision a time when ASCI scientists will rapidly develop software by using a number of interoperable technologies and services ranging from core technologies such as parallel communication and I/O to domain-level services like grid generation and visualization. The resultant software will be used locally as well as widely on *computing grids* made up of the combined resources of many collaborating computing environments.

TECHNICAL ISSUES AND CHALLENGES

ASCI's code development projects constitute software engineering and scientific research activities that require disparate skills and experience. Major codes contain several hundred thousand to several million lines of code. Often the code pieces that fused together to create the ultimate simulation tool are based on disparate technical areas, differing computer languages, and a wide variety of time scales and solution methods. The problem of bringing these many software elements together, as well as improving and maintaining them, is itself a technical challenge. This problem becomes even more complex because scientific simulations are performed on advanced parallel computer architectures.

The complexity and scale of many modern scientific software development projects present a number of difficulties. Advanced simulations are typically composed of many collaborating subsystems, each representing the collective expertise of different development teams. The cost and the difficulty of developing and integrating software written in many languages, for varied purposes and by different development teams, are daunting.

Software Integration and Interoperability. We define *software integration* as the general problem of assembling software constituents into a robust and operational system. We define *interoperability* as the property possessed by software modules or subsystems that can be integrated easily and reliably. The notion of "plug and play," in which one constituent can be easily substituted for its equivalent, is consistent with this definition. Broadly speaking, an interoperable module or subsystem is one that "plays well with others."

Frameworks. ASCI and similar programs have addressed the interoperability problem by building frameworks. There is unfortunately no standard definition of "framework" although there is a reasonably broad understanding of the concept within the software development community. A good working definition is that a framework is a set of reusable design patterns, expressed as clearly defined software abstractions. This definition fits the practice of object-oriented programming well, so most frameworks are designed as object-oriented systems. Frameworks usually exist within a conceptual "frame" that defines a class of applications or software services. For example, there are frameworks for I/O, for database management, for GUIs, and so on. For high-performance scientific applications such as those in

ASCI, we can envision frameworks that support linear and nonlinear equation solvers, parallel I/O services, dynamic parallel data structures and parallel communication, performance optimization, and many other common services.

The most common approach to implementing a framework is to build a class library. Framework class libraries provide solutions to interoperability at "compile time," enabling client applications to use their services in application source code linked to the library. Although framework class libraries have been very useful in various ASCI projects and programs, their widespread application, use, and maintenance remain a challenge. There are numerous issues that contribute to this that are beyond the scope of this discussion. However, two common problems are that (1) class libraries are not language independent and (2) they are bound at compile time rather than at run time. The first problem means that users must accept a particular language interface unless additional language bindings are provided. The second means that interoperability is dynamic; distributed applications, on the other hand, are limited.

Components. Components are the main technical solution to interoperability in industry. The terms "component framework" and "component architecture" are used in this context. Like the term "framework," there is no standard definition of "component." But a good working definition is that a component is a reusable run-time entity with the following attributes:

- *Visible Properties* – Components having publicly accessible parameters can be used to customize their function.
- *Reflection* – Components can describe their own interfaces.
- *Persistence* – Components can be persistently stored, thereby retaining their properties.

One of the most attractive features of components is that they can be easily composed. That is, components can be easily plugged together, provided their interfaces are compatible. Components cannot exist by themselves. Rather, they exist within a component architecture. Components require the services of a run-time system to manage their existence cycle. Configuration tools provide the means of assembling components and customizing them for a given application.

There is a straightforward way to make library classes available within a component framework. The process involves wrapping the source code for the class within a component skeleton that provides access to class methods. The Common Object Request Broker Architecture (CORBA) is one way of wrapping classes to make components. Thus, the effort that has gone into the many useful libraries and tools developed by ASCI and others can still be exploited within component architectures.

Frameworks based on component architectures are important for ASCI to consider in meeting its requirements for interoperability. Component architectures are particularly attractive because the scenarios that ASCI must support are so complex and varied. The main challenge for ASCI is to acquire component technology that can support the extreme demands of ASCI application codes. Such capabilities are currently unavailable commercially, and industry is not driving requirements in this direction fast enough to meet ASCI needs.

Grid Computing. Another compelling reason to promote interoperability standards is related to what many consider the next important trend in large-scale computing. This trend, called "grid" computing, enables large-scale applications to run on the combined resources of many collaborating systems. In this complex computing environment, software components are assigned and reassigned to networked resources based on availability and performance characteristics. Such a computing environment is not only a severe test of software interoperability but, because of its inherent complexity, may depend on high-performance component architectures to become a reality.

Efforts Outside ASCI. Considerable effort has been expended on the development of component technology outside of ASCI. Industry makes heavy use of frameworks in software development. Many organizations rely heavily on component architectures and libraries of components supplied by third-party vendors. Microsoft's COM component system is the basis of nearly their entire product line, and many companies, especially those that are data driven, lean heavily on Enterprise Java Beans and similar component systems.

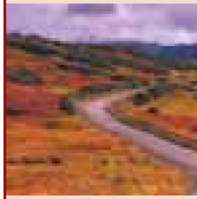
Universities also conduct a great deal of the cutting edge research in this area. Much work is currently under way to develop the Common Component Architecture (CCA) specifications intended to extend component technology to high-performance computing.

The Object Management Group (OMG), a private-public-sector consortium, actively promotes development of new interoperability technologies including high-performance CORBA implementations. A CORBA Component Model specification has very recently emerged.

The Java community has been a leader in component technology with the introduction of Java Beans and Enterprise Java Beans (EJBs). EJB components are "smart" about client-server applications and greatly facilitate the development of enterprise systems by integrating network security, namespace resolution, and many other services.

The drivers for framework and component technology outside of ASCI are much the same as those inside. Businesses need to manage complex and elaborate software systems and to develop them on time and on budget. Recently, the need to handle large-scale data for real-time, distributed applications and online application processing (OLAP) has spurred interest in high-performance modifications to the standard component models. Therefore, some of the high-performance computing goals of ASCI overlap with those of industry.

In the related area of Grid Computing, the high-performance computing community has launched several development efforts and has recently begun to demonstrate the feasibility of computing on large-scale, widely distributed computing resources [1].



Challenges. We anticipate continued advances in software interoperability largely based on industry-driven component technology. Our main objective here is to discern which areas of the technology needed by ASCI are unlikely to be developed, or to be developed fast enough to meet the ASCI mission. The answers to these questions are best considered in terms of the following categories:

1. Common interoperability standards, including common software abstractions, and processes for achieving them (discussed under Standardization Process Barriers below).

2. Development of core extensions to existing component technologies (discussed under Core Technology and Generic Service Barriers).

3. Development of additional tools and services, some of which are generic to high-performance computing, while others are specific to the ASCI problem domain (discussed under Domain Service Barriers).

The first of these categories introduces the important concept of process — something that is not often considered a technical challenge or barrier in the usual sense. However, the search for interoperability solutions leads to such considerations out of necessity. For example, the CORBA standard, and the many distributed computing solutions based on it, could not have come about without the OMG and the processes for consensus building and standardization that it devised. In planning for advances in software interoperability within ASCI, processes should be standardized; anything judged to be a process obstacle would implicitly be considered a technology barrier.

The second category refers to the extensions that will be necessary to make component technology accessible and workable in high-performance computing applications like ASCI's.

The third category identifies particular kinds of services needed by ASCI, some of which are generic to high-performance computing. For example, large, distributed, composite data structures, such as multidimensional arrays partitioned across thousands of processors, are not a mainstream industry concept. Nor are services based on ASCI main-line technologies like MPI parallel message passing, or high-performance gather-scatter. There are other generic, services in this category more specific to scientific computing, such as linear system solving. Many of the application domains must be covered in ASCI development are also not mainstream. Advanced algorithms for physics models like hydrodynamics and radiation are examples.

There are technologies required by ASCI in each of the above areas that we do not expect to come from industrial software technology development, or to come soon enough. Consequently, they constitute technology barriers for ASCI.

Standardization Process Barriers. Although standardizing processes (or developing them) is not typically considered a technology area in high-performance computing, the development of interoperability solutions makes it important. Currently, ASCI has no processes for achieving common interoperability standards, including common software abstractions. Conversely, other standards bodies like OMG have successful organizational mechanisms and processes in place but are not expected to address the extreme requirements of ASCI computing in the immediate future. Research interest groups like the Common Component Architecture Forum are indeed working to develop new standards as well as extensions to current standards in this area but are only peripherally associated with ASCI programs.

In general, a standards body:

- Identifies key technologies and service areas.
- Develops and standardizes interfaces, design patterns, and other abstractions necessary for interoperability in the key areas.
- Participates in the development of reference prototypes that can be used to integrate the technology in real applications and encourage commercial development.

Core Technology and Generic Service Barriers. The following is a list of key technologies that need to be made interoperable or that contribute to interoperability solutions:

- Large, Distributed, Composite Data Structures and Parallel Communication tools. An ASCI application is likely to depend on large data containers holding upwards of 10^8 to 10^{11} numerical values. This can translate to as much as 10^6 megabytes of data associated with a single container. In a typical scenario, these values could be distributed across tens of thousands of smaller data structures each holding hundreds of megabytes. Nothing on this scale is expected in industrial applications in the near term, yet some ASCI projects are already within an order of magnitude of this. Data movement and processor remapping are integral aspects of this problem. Projects like Parallel Object-Oriented Methods and Applications (POOMA), A++/P++, OVERTURE, PAWS, and Scalable Multithreaded Asynchronous Runtime System (SMARTS) have exploited a common model of multidimensional data structures to facilitate many important processes associated with accessing and transferring data structures in a high-performance setting. However, the lack of common interfaces and design abstractions limits the interoperability of these systems.

- *Parallel Performance Analysis, Optimization, and Testing*. Services related to analyzing parallel performance and optimizing parallel programs are strongly needed in ASCI. Tests for correctness and reliability are critical to high-consequence application development. In the commercial sector, there are typically automated tools in component form to assist in this process. While tools of this kind are currently available in the academic research community, interoperable solutions for exploiting them and integrating them within a production setting do not exist by and large, and we do not expect industry to provide such components in the near term.

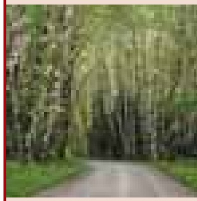
- *Parallel Component Architecture Extensions*. Standards for building applications from components in a massively parallel computing environment have not been addressed in the commercial sector. There is encouraging activity in this area within academia and DOE. The CCA effort [2] and insightful reports such as Sandia's "Advanced Software Interoperability Architecture and Strategy" [3] have contributed a great deal. However, we cannot expect the research community alone to develop robust production-quality solutions for ASCI. Some of the key technology issues in this area have to do with the efficiency of application servers for high-performance applications and how well the overall architecture can support parallel communication and synchronization in applications that mix asynchronous-task-parallel and synchronous-data-parallel components.

- *High-Performance Database Services*. Like industry, ASCI needs comprehensive data management and database tools that support both object and relational data models. However, neither the capacity required by ASCI, nor the support for high-throughput parallel I/O (10 to 100 Mb/s), is to be found in mainstream database products. Thus, this is a technology barrier, and is related to barriers in the Scientific Data Management (SDM) section.

- *Numerical Solvers*. ASCI's focus on numerical modeling emphasizes the importance of creating interoperable numerical tools. Industrial applications only marginally deal with this issue, and there is nothing we anticipate from industry that would address the scale and complexity of ASCI requirements for numerics. The Equation Solver Interface (ESI) effort, under way for some time, has attempted to address some of these issues. Specifications have emerged [4], but the status of the effort and its relationship with ongoing projects is uncertain.

- *Gridded Data Sets*. Industry occasionally works with gridded data in situations ranging from image analysis to geographic information processing. However, these areas are not widespread, are not expected to push development of common solutions, and do not address the requirements of scale anticipated in the ASCI program. This then presents another barrier, and one that again overlaps with SDM. SDM activities to address the management of large, gridded data sets have made considerable progress at all three laboratories. However, there is still a great deal to be done technically and a long way to go in achieving consensus and standardization.

Domain Service Barriers. There are many services that can be identified at the domain level. We list a few of these for which there is continued interest and concern:



- Grid generation and problem setup
- Interfaces to material properties data
- Common simulation input specifications
- Scientific visualization tools

Again, the challenge here is not the development of tools per se, but how to achieve interoperable solutions in each of these areas.

Road Map to Software Interoperability

We propose a road map by which ASCI may be able to keep ahead of the curve in interoperable technology and can promote solutions that will help overcome technology barriers.

An unusual feature of our road map involves initiating a process for promoting interoperability solutions and standards in this area. The OMG is a good model, and possibly a good vehicle for this. The consensus development process in the OMG was carefully designed to promote interoperability without requiring that software vendors reveal their source code or their methods. OMG standards address design patterns, interfaces, and design contracts — exactly the issues that are the foundations of interoperability.

Early steps in the road map focus on establishing acceptable standardization processes. These processes could be patterned roughly after the OMG, or after other organizations closer to this application area like the CCA. However this is done, it must be a deliberate and visible feature of the road map for achieving this technology.

In the following, we have assumed some process framework of this kind and have organized our key technology goals accordingly. We have also distinguished each as belonging to the core, generic, or domain-level service category.

The associated visual depicts the five-year status (calendar years 2000 to 2005) of desired capabilities/activities within a functional area. Each capability is color coded to show what level of R&D effort it requires or anticipates.

R&D Effort Indicator:

Accomplished = completed

Planned = ASCI will accomplish even with slight budget fluctuations

Hurdle = ASCI will need some help

Barrier = ASCI will need significant help from the high performance computing (HPC) community

NOTE: Both hurdles and barriers represent research opportunities for the HPC community.

Road Map for SOFTWARE INTEROPERABILITY CAPABILITIES

Functional Area	CY 2000	CY 2001	CY 2002	CY 2003	CY 2004	CY 2005
Core Technology		Large-scale, distributed container specifications Large-scale, distributed object persistence specifications Parallel performance optimization components specifications Specifications for large-scale, parallel testing	Large-scale, distributed container implementations Large-scale, distributed object persistence implementations Parallel performance optimization components implementations Implementation of large-scale, parallel testing components Core parallel component architecture implementation			Mature parallel component architecture implementation
Generic Service			Large dataset management and database connectivity specifications Gridded dataset interface specifications Numerical solvers interface specifications	Large dataset management and database connectivity implementations Gridded dataset implementations Numerical solvers interface implementations		Mature parallel component architecture implementation
Process Infrastructure		OMG-style working group and process for standardization				Mature parallel component architecture implementation
Domain Service				Grid generation and problem setup specifications Material properties interface specifications Visualization component specifications	Grid generation and problem setup components implementations Material properties interface implementations Visualization components implementations	Mature parallel component architecture implementation

R&D Effort Indicator ● ACCOMPLISHED ● PLANNED ● HURDLE ● BARRIER

ACCOMPLISHED—Completed
PLANNED—ASCI will accomplish even with slight budget fluctuations
HURDLE—ASCI will need some help from the HPC community
BARRIER—ASCI will need significant help from the HPC community

TIMELINE

This timeline elaborates on the activities that appear on the preceding road map.

Calendar Year **Description and Status**

2001 **An OMG-style working group and process** for consensus building and standardization. We believe one of the most important requirements is the establishment of an Interoperability Working Group and a process model. Without such a group there will be no way to bind the various ASCI and HPC community efforts together on an ongoing basis. This item is relevant to Simulation and Computer Science (S&CS) as well as Apps. **Planned**

Large-Scale, Distributed Container Specifications to standardize large distributed arrays and other basic containers. This is relevant to S&CS. **Hurdle**

Large-Scale, Distributed Object Persistence Specifications for general object persistence in applications that use large-scale distributed containers and objects that are derived from them. This is relevant to S&CS. It is a difficult challenge to standardize, especially if we wish to align our efforts with industry. **Barrier**

Parallel Performance Optimization Components Specifications for components and tools that analyze performance and contribute to performance optimization. Relevant to S&CS. **Barrier**

Specifications for Large-Scale, Parallel Testing for components and tools that facilitate validation and testing for reliability at the core technology level. Relevant to S&CS. **Barrier**

2002 **Large-Scale, Distributed Container Implementations** of the specifications given in the previous year. This is relevant to S&CS. While challenging, this is an area in which we can leverage existing (nonstandardized) prototypes. **Hurdle**

Large-Scale, Distributed Object Persistence Implementations that go with the specifications in the previous year. Relevant to S&CS. It is a difficult challenge to implement. **Barrier**

Parallel Performance Optimization Components Implementations of the previous year specification. Relevant to S&CS. **Barrier**

Implementation of Large-Scale Parallel Testing Components – Implementation of the previous year specification. Relevant to S&CS. **Barrier**

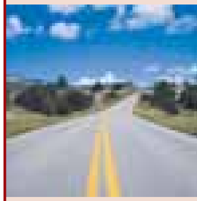
Core Parallel Component Architecture Implementation – Because preliminary specifications for a parallel component architecture have been developed within CCA in prior years, we believe that a prototype implementation could be available at this time. There may be some additional refinement of specifications along the way as a result of the Framework Working Group's activities. This is in the S&CS area and requires a great deal of coordination and community participation. **Barrier**

Large Dataset Management and Database Connectivity Specifications that is a challenging area because it aims to standardize database access to large data objects such as those that incorporate large containers. This area overlaps S&CS with Apps because of the combination of generic technology with application-driven access patterns. **Barrier**

Gridded Dataset Interface Specifications that attempt to standardize gridded dataset models and interfaces such as those developed under SDM. This overlaps S&CS with Apps because of the combination of generic technology with application-relevant data models. **Planned**

Numerical Solvers Interface Specifications – This would continue the ongoing efforts begun with ESI to establish interfaces for numerical solvers. This is another area of S&CS and Apps overlap. **Barrier**

2003 **Large Dataset Management and Database Connectivity Implementations** – The implementation of the previous year's specifications. Relevant to S&CS and Apps. **Barrier**



Gridded Dataset Interface Implementations – The implementation of the previous year's specifications. Relevant to S&CS and Apps. **Hurdle**

Numerical Solvers Interface Implementations – The implementation of the previous year's specifications. Relevant to S&CS and Apps. **Hurdle**

Grid Generation and Problem Setup Specifications – Specifications and abstractions for a number of capabilities related to grid generation and problem specification. One of the key areas here will be gridded dataset specifications developed under generic services and can be leveraged. Another is interpolation, grid re-mapping and various computational geometry services. This overlaps S&CS with Apps, but is mostly driven by Apps. **Hurdle**

Material Properties Interface Specifications – A long-standing problem is the development of good abstractions to interface these tabulated databases to physics codes. **Hurdle**

Visualization Component Specifications – There are a number of mature models within scientific visualization on which to base a specification. There is also an existing data-flow paradigm that can be leveraged. This area overlaps S&CS with Apps. **Hurdle**

2004 **Grid Generation and Problem Setup Components Implementations** – Implementation of the specifications developed in the previous year. Again overlaps S&CS with Apps. Still a challenge, but because a great deal of domain knowledge and source code can be leveraged. **Hurdle**

Material Properties Interface Implementations – Implementation of the specifications developed in the previous year. Again overlaps S&CS with Apps. The main challenge is to connect the interface with the data at the various labs. **Hurdle**

2004 cont. **Visualization Component Specifications** – Implementation of the specifications developed in the previous year. Again overlaps S&CS with Apps. Most of the difficulty will be in interfacing existing software, which is mature and effective, with the interfaces and other specifications. **Hurdle**

2005 **Mature Parallel Component Architecture Implementation** – The main requirement at this stage. This relies on successes in developing the required technologies in the prior years. Of necessity, it spans both S&CS and Apps and, because of its dependence on technology barriers, is itself a **Barrier**.

CURRENT STATE OF ASCI SOFTWARE INTEROPERABILITY

There are many accomplishments in the area of frameworks within ASCI as well as outside. These include both the areas of Applications and Simulation & Computer Science. There is a good deal of existing technology and experience to draw upon. Below is a summary by technology. The list is not intended to be complete, but to serve as a means of placing a few recognized activities on the road map.

Framework Libraries – There are a number of general numerical framework libraries including POOMA and its physics-oriented sister framework Tecolote, Overture, A++/P++, and the Sierra framework.

Parallel Component Architectures – The CCA community has issued an initial specification, and there are some preliminary prototypes currently under development.

Large-Scale, Distributed Containers and Parallel Communication – Large, distributed containers are handled by POOMA and by PAWS. For parallel communication, there is, for example, Cheetah. A number of academic libraries also cover this area.

Large-Scale, Distributed Object Persistence Specifications – POOMA's recent 2.3 release contains a first attempt as part of its "object I/O" capability. A recently approved, internally funded project at Los Alamos is just getting under way to address the larger problem.

Parallel Performance Optimization Components – Currently, there are a number of profiling tools from the academic HPC community. These are considered separate tools at this time and have yet to be incorporated into a general framework architecture.

Large-Scale, Parallel Testing Components – The status of this area is unknown; however, it is beginning to be recognized as a key technology need.

Large Dataset Management and Database Connectivity – There is an effort just getting underway at Los Alamos to address this. There are also HPC community activities in this area such as the Storage Request Broker, a National Partnership for Advanced Computational Infrastructure (NPACI) alliance project.

Gridded Dataset Interfaces – The ASCI SDM program has made progress in this area and now has several prototypes. Most activity outside of ASCI is less sophisticated. A notable exception is the gridded data interface used by visualization projects at NASA's Ames Research Center.

Numerical Solvers Interfaces – This is currently an active area of research. The aforementioned ESI activity has issued some preliminary specifications. Its present status is unknown. Incidentally, there are a number of linear solver libraries in the academic community based on various abstractions that could be helpful in this area.

Grid Generation and Problem Setup Components – There is a great deal of effort within ASCI Apps to develop these capabilities. To date there have been no attempts to synthesize general abstractions. We believe there are efforts on the outside (e.g., in the mechanical engineering area) that could be helpful. There are also commercial tools available for core geometry modeling services that could be exploited.

Physical Properties and Similar Specifications – There has been no systematic effort in this area as far as we know. Each code team has an ad hoc solution. Attempts have been made to upgrade and standardize the condition of specific databases such as SESAME in order to improve the interface problem, but these do not represent a general approach to our knowledge. Linking simulation entities to externally archived data describing physical properties is a generic problem that should have an interoperable solution.

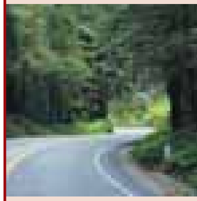
Visualization Components – There has been tremendous development of Scientific Visualization within ASCI and on the outside. ASCI program scientists and alliance partners are heavily engaged, as are various academic and commercial interests. This technology area is fairly mature. Here it is mainly standardization and interoperability that are the main issues.

SUMMARY, CONCLUSIONS, RECOMMENDATIONS

The discussion thus far may seem abstract. This is because much of the development in software operability is just getting under way. As a result, technology needs are not as clear as they might be in other areas such as high-capacity data storage. Many challenges are still to be overcome if the right technology is to be developed for ASCI. However, a few recommendations can be made:

1. A great deal can be gained from coordinating this activity and leveraging existing experience in developing specifications and in implementing the tools and components themselves.

2. ASCI and its partners in HPC can benefit from either establishing a working group patterned roughly after the OMG, or participating in the creation of a special interest group within OMG or similar organization. As industry requirements for high-performance computing grow and begin to match ASCI's, the standards developed in this setting could be merged into the OMG or similar body, similar to defense industry software standards that have been integrated within OMG and IEEE in the past.



In conclusion, the technology barriers described above can be overcome. The experience, ingenuity, and resources to meet these objectives exist within ASCI, the larger HPC community, and industry. Earnest outreach and coordination efforts could greatly enhance the chances for success.

Similarly, the tentative road map can be amended as requirements become clearer. We also think they should be extended to include strategies for integrating high-performance component technologies with grid computing. We have not addressed computing grids specifically in this section because we feel there are a number of important barriers to high-performance component technology that must be given priority. However, we view grid computing and component architectures as complementary in many ways, and feel that the two technologies should be part of an integrated computing strategy.

REFERENCES

1. The Globus Project; see http://www.globus.org.

2. Los Alamos National Laboratory, The Common Component Architecture Forum Web Site. http://www.acl.lanl.gov/cca-forum

3. Clay et al., *Advanced Software Interoperability* Architecture and Strategy, v.1.0a, SAND2000-8229, Sandia National Laboratories, Albuquerque, NM, April 2000. http://z.ca.sandia.gov/ASIA

4. Sandia National Laboratories, The Equation Solver Interface (ESI) Standards Multi-Lab Working Group & Design Effort. http://z.ca.sandia.gov/esi

2. DATA MANAGEMENT, VISUALIZATION, AND STORAGE

Terascale computing, i.e., parallel computers calculating at speeds of tera-operations per second (OPS), is only the first step in performing complex modeling. The next step in a simulation is organizing, understanding, and analyzing the data produced by (and during) the simulation. Connected in this general category of data, we have scientific visualization, management of large-scale data sets for scientific analysis, and software and hardware necessary for storage and processing of those data. All of these areas become more complex than those of the corresponding industrial simulations when the computational environment pushes technology to the limit. In this section, we examine what features of real data processing are necessary to support ASCI-scale simulations.

Constantine (Dino) Pavlakos
(lead author)
Sandia National Laboratories
Albuquerque, New Mexico
cjpavla@sandia.gov
(505) 844-9089

Robert D. Tomlinson
(tri-lab curve owner)
Los Alamos National Laboratory
Los Alamos, New Mexico
bob@lanl.gov
(505) 665-6599

Sam Uselton
(tri-lab curve owner)
Lawrence Livermore National Laboratory
Livermore, California
uselton1@llnl.gov
925-423-9055

Phil Heermann
(tri-lab curve owner)
Sandia National Laboratories
Albuquerque, New Mexico
pdheerm@sandia.gov
(505) 844-1527

Celeste Matarazzo
(lead writer and tri-lab curve owner)
Lawrence Livermore National Laboratory
Livermore, California
matarazzo1@llnl.gov
(925) 423-9838

Bill Spangenberg
(tri-lab curve owner)
Los Alamos National Laboratory
Los Alamos, New Mexico
whs@lanl.gov
(505) 667-4278

Philip Kegelmeyer
(tri-lab curve owner)
Sandia National Laboratories
Livermore, California
wpk@sandia.gov
(925) 294-3016

Steve Louis
(lead writer and tri-lab curve owner)
Lawrence Livermore National Laboratory
Livermore, California
stlouis@llnl.gov
(925) 422-1550

John Blaylock
(tri-lab curve owner)
Los Alamos National Laboratory
Los Alamos, New Mexico
jwb@lanl.gov
(505) 667-8970

Rena Haynes
(tri-lab curve owner)
Sandia National Laboratories
rahayne@sandia.gov
(505) 844-9149

2.1 VISUALIZATION

By the year 2005, ASCI calculations are expected to produce hundreds of terabytes of information per simulation. ASCI scientists will need to evaluate these simulation results in detail, thus creating a demand for state-of-the-art visualization tools. Although the best visualization hardware and software are being leveraged as appropriate, "terascale"-level scientific visualization tools are either not commercially available or not viable at this time. ASCI researchers are pursuing collaborative efforts to gain the additional functionality needed

A critical goal for ASCI is to explore, understand, and compare massive scientific datasets. The ASCI visualization program seeks to build a visualization environment, allowing weapons scientists and engineers to quantitatively and qualitatively understand and analyze these volumes of data.

As the weapons community becomes increasingly reliant on computation, the visualization program must ensure that weapons designers and analysts can efficiently gain maximum understanding from their calculations (see Figure 2.1-1). By the year 2005, ASCI calculations are expected to produce hundreds of terabytes of information per simulation. Thus, meaningful evaluation of these simulation results will require major advancements in data manipulation and visualization.

The challenges facing the Visualization program over the next few years are multifaceted, and it is the combination of these various challenges that creates an even larger challenge. The major drivers can be categorized into six functional areas: data handling, data exploration, environments, scalable rendering, displays, and distance visualization.

- Data Handling and Data Exploration. One primary driver pertains to the unprecedented size and complexity of the datasets we are expecting to visualize over the next several years (see Figures 2.1-2 and 2.1-3); this provides a very large data management challenge. Simply manipulating a 2-terabyte time-dump, which is part of a 200-terabyte dataset, by CY 2005 is daunting. Improved data exploration methods are required for rapid data browsing and selecting. For an analyst to peruse 200 terabytes of data, techniques for efficient data mining and data discovery are needed, and these tools must be integrated with visualization to provide the proper context of the results.

- Environments, Rendering, and Displays. The need to improve user environments is another major driver. The results of the calculations must be comprehensible to the analysts and engineers. Both shared facilities (Figure 2.1-4) and offices (Figure 2.1-5) must be enhanced with larger displays with higher pixel counts to enable a designer to fully comprehend the results (Figure 2.1-6).

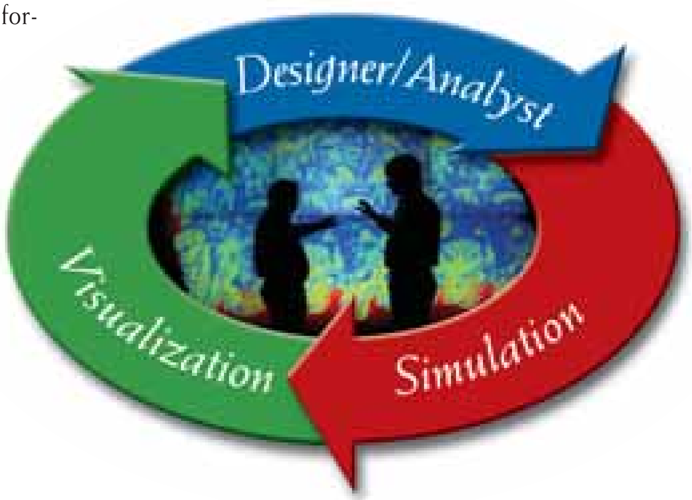


Figure 2.1-1. A cycle for ASCI compute-based simulation and analysis. Visualization capabilities are needed that enable efficient, yet maximum comprehension so that this iterative cycle can be executed as fast as possible.

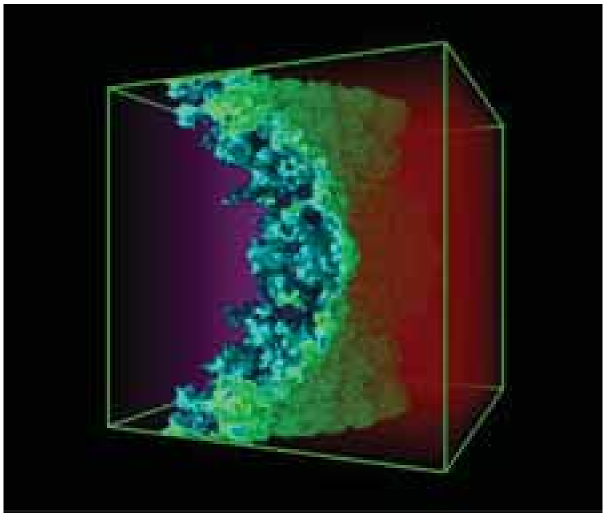


Figure 2.1-2. Visualization of a Richtmyer-Meshkov instability simulation involving two gases subjected to shock. The fidelity of ASCI simulations demands the need to be able to observe intricate detail. Surfaces extracted from such data have been as complex as 470 million triangles.

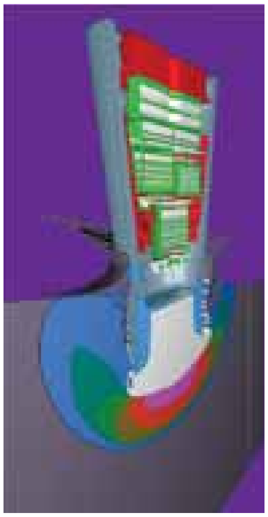


Figure 2.1-3. Visualization of a reentry-body impact showing full-system features needed to support ASCI data analysis requirements.

It is not feasible to explore a one-billion-cell, three-dimensional calculation with a one- or two-million-pixel display. Multiple simultaneous views of the data, at as high a resolution as possible, greatly accelerate an analyst's understanding of the results. Moreover, to support interactive visualization of these large three-dimensional datasets on high pixel count displays, significant improvements in rendering speeds are required. Thus, rendering rates also need to increase on the order of 1000 times by 2005.

- Distance Visualization. Remote use of large ASCI platforms is increasing — something that requires the use of a wide area network (WAN) to the largest extent possible. The major ASCI compute platforms reside at different laboratories, several miles away from each other. It is therefore necessary for analysts situated at a distant laboratory to work effectively across a WAN (see Figure 2.1-7).

Failure to recognize these drivers and meet the associated challenges could result in calculations being run but inefficiently or inadequately analyzed, allowing an important issue to go undetected. Computational results are only as good as the understanding gleaned from them. Visualization is a major tool for understanding these results. The scale of datasets presents the very real possibility of overwhelming current visualization tools and "closing" this key channel of information from the simulations.

Other industries also stand to benefit from the technologies required for ASCI visualization. Medical imaging

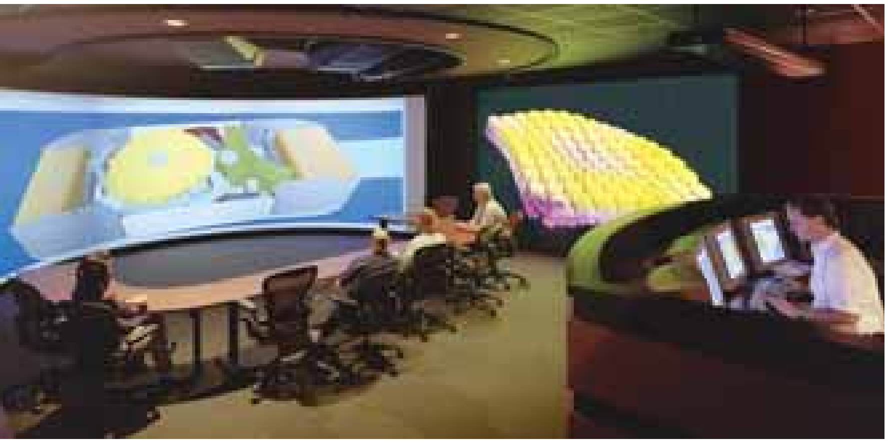


Figure 2.1-4. Visualization Corridors provide advanced facilities for high-end visualization and shared, collaborative data analysis.

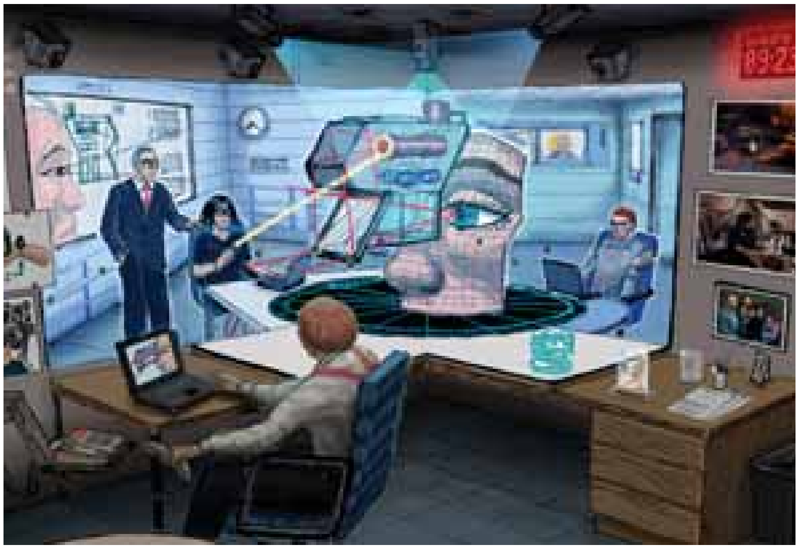


Figure 2.1-5. Office-of-the-future environments are being considered that would increase end-user productivity, the ability to collaborate with co-workers, and the ability to gain scientific insight (courtesy of University of North Carolina).

and the oil/gas industry both have ongoing needs for the visualization of large datasets. Advances in visualization technologies stand to benefit both industries. Other areas that could benefit from the technologies include global systems and climate modeling, basic science research, and computational biology.

TECHNICAL ISSUES AND CHALLENGES

The extremely aggressive ASCI visualization requirements, as with the aggressive ASCI computing requirements, have led to the development of a visualization architecture that depends on scalable hardware and software in order to attain the required performance. We are focusing our research and development efforts on applying and combining commodity graphics chips and boards, commodity computing cluster technology (see Figure 2.1-8), and tiling of commodity display technology in order to reach our goals.

We are collaborating with various university researchers in most of the technology challenge areas mentioned. We have engaged researchers in developing techniques for multiresolution data exploration, for using commodity PC clusters and commodity graphics cards for high-end scientific visualization, for utilizing tiled displays driven by such commodity systems, for developing immersive visualization facilities for scientific visualization, and for developing parallel and distributed software architectures for driving these systems.

We have engaged industry (e.g., IBM) in producing high pixel-density displays and are working with industry in the development of a commodity-based scalable rendering system. We have also engaged industry in scaling commercial scientific visualization software systems (e.g., CEI's EnSight Gold) to unprecedented capabilities and adapting a successful scientific visualization framework (the Visualization Tool Kit – "vtk") from a serial-distributed model to a parallel-distributed model. Commercial markets do not demand as much as we do from our visualization software. We have significant research

Figure 2.1-6. Large, tiled power-wall display technologies are being developed to support very high resolution and data immersion for analysis of high-fidelity data.

Figure 2.1-8. A PC-based visualization cluster. Cost-effective technologies are needed to enable highperformance visualization at levels well beyond today's traditional, high-end systems.

efforts to develop software tools that complement the commercial tools (see Figure 2.1-9).

Two major technical challenges for ASCI visualization are the size of the datasets and the dynamic changes taking place in the graphics hardware industry. The greatest impact of dataset sizes is in bandwidth to the data. Storage devices, especially disk drives, are advancing rapidly in capacity, but the bandwidth to access the disks is advancing at a much slower rate. This requires data handling and visualization systems to require striped pathways across many times the number of devices than is common in industry.

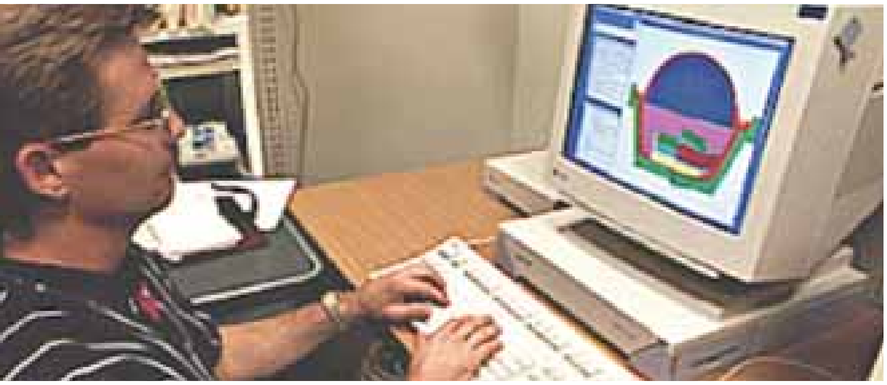


Figure 2.1-7. A critical objective is to enable ASCI-class visualization from the day-to-day office work environment, regardless of distance-separation from critical ASCI resources.

The second major challenge is that the PC 3D graphics market is so large that it is overtaking the much smaller workstation and high-end graphics server market. Raw performance of $300 graphics cards is exceeding the abilities of $100,000 workstations. Low cost cards, however, are tailored to the "computer-games" market, not the scientific visualization market. The technical challenge is to harness the features demanded by the games market to meet the visualization needs of ASCI datasets.

Road Map for Visualization

The associated technology road map visually depicts the five-year status (calendar year 2000 to 2005) of desired capabilities/activities within a functional area. Each capability is color coded to show what level of R&D effort it requires or anticipates.

R&D Effort Indicator:

Accomplished = completed
Planned = ASCI will accomplish even with slight
budget fluctuations
Hurdle = ASCI will need some help
Barrier = ASCI will need significant help from the high
performance computing (HPC) community

NOTE: Both hurdles and barriers represent research opportunities for the HPC community.

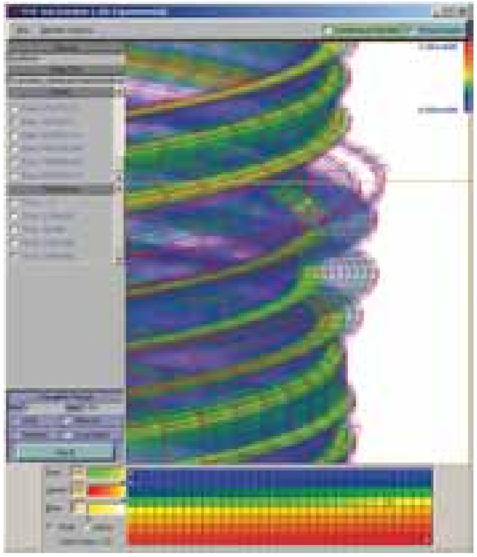


Figure 2.1-9. An advanced tool for volume rendering of unstructured mesh data. Parallel visualization tools, as well as new algorithms and techniques, are being developed to enable more effective exploration of ASCI-class data.

Road Map for
VISUALIZATION CAPABILITIES

Functional Area	CY 2000	CY 2001	CY 2002	CY 2003	CY 2004	CY 2005
Data Handling			Real-time interaction with 300GB timedump	Data subsetting/ manipulation services integrated with interactive visualization solutions	Real-time interaction with 2TB timedump	
Scalable Rendering	Scalable rendering system driving a 16M pixel tiled display		Scalable rendering system driving a 64M pixel tiled display		Scalable rendering system driving a 16M pixel display in offices	Scalable rendering system driving a visual acuity tiled display (100M pixel) in shared facility
Distance Visualization	Interactive distance visualization with lossy image delivery		Interactive distance vizualization solutions Digital approaches for desktop delivery	Collaborative interactive distance visualization	Integrated distance visualization and data services available across the WAN	
Environments	Shared immersive environments		Display intensive offices	Improved user interfaces for shared facilities		Collaborative office environments
Data Exploration		Data browsing/ multi-resolution data				Data mining/ discovery integrated with visualization

R&D Effort Indicator 🔵 ACCOMPLISHED 🟢 PLANNED 🟡 HURDLE 🔴 BARRIER

ACCOMPLISHED—Completed
PLANNED—ASCI will accomplish even with slight budget fluctuations
HURDLE—ASCI will need some help from the HPC community
BARRIER—ASCI will need significant help from the HPC community

CONSTANT ASSUMPTIONS (ACROSS TIME)

5–10 Hz Interaction/Rendering rate on largest displays (e.g., spinning around a view of an isosurface)

10-s Interaction/Processing rate on largest datasets (e.g., updating from one time step to another, moving a clip plane, etc.)

TIMELINE

The visualization technology timeline is based on two major underlying assumptions: (1) a 5–10-Hz interaction/rendering rate and (2) a 10-s interaction/processing rate. The interaction/rendering rate is the rendering rate for performing an activity like spinning around a view of a large isosurface on the largest display. There is a desire to have rendering rates sufficiently high to support head-tracked stereo frame rates (20+ Hz/per eye), but our assessment of future technology availability drove us to set a constant goal of 5 to 10 Hz. This is highly aggressive considering the dataset sizes.

The 10-s interaction/processing rate is the rate on the largest datasets to update the data to be rendered. This would include functions like moving from one timestep to another, moving a clip plane, or calculating an iso-surface. This is envisioned as the wait time between when a slider is moved or a button is pushed before the system can start rendering the new data. This is very aggressive. For example consider that a brute force approach to simply accessing 300 gigabytes in 2002 would require a 30 gigabytes/s storage system.

(Note: Many of the milestones below are for the same technology development path. For the sake of brevity, the first reference to the technology path contains the most detailed description of the technology path. Later milestone descriptions are brief.)

Calendar Year **Description and Status**

2000 **Scalable Rendering System driving a 16-megapixel tiled display** – In order to see important details present in data at image resolutions that are better matched with the data resolution of high-fidelity geometric grids, display capabilities are needed that enable data to be visualized at pixel-resolutions on the order of 16 megapixels. This effective resolution can be achieved via the integration of multiple displays. Meeting the long-term goals of the development track requires that a scalable rendering system be deployed. **Accomplished**

Shared immersive environments – Implies the ability of a group of people to share an immersive experience in which ASCI data are being explored by one of the people in the group. For this target, no independent capability to explore the data is required. **Accomplished**

Interactive Distance Visualization with Lossy Image Delivery – The capability to interact with, and visualize, data across distance, particularly inter-site, at rates that are comparable with sitting at the console of a high-performance visualization server. **Accomplished**

2001 **Data browsing/multiresolution data** – Multiresolution techniques should be mature enough for integration with the end-to-end visualization process for ASCI data visualization. The work is planned and is in process. The main challenge is to supply interactive rates (defined above) for datasets that will be at 20 terabytes (300-GB timedumps) by CY 2002. **Planned**

2002 **Real-time interaction with 300-gigabyte timedump** – Implies the ability to interact with a full timestep of result data whose raw size is approximately 300 GB, at frame rates that are defined at the beginning of this section. Sustaining high bandwidths and high-performance parallel tools, as well as effective data-exploration techniques, are the challenges here. **Hurdle**

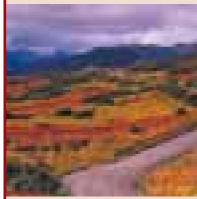
Scalable Rendering System driving a 64-megapixel tiled display – The scalable rendering system will be extended to drive display resolutions of 64 million pixels; effective resolution can be achieved via the integration of multiple displays. This is a barrier because the scalable rendering technology to scale to this number of pixels for very large datasets does not exist. **Barrier**

Interactive distance visualization solutions – Methods are required for loss-less interaction and visualization with very large datasets across the WAN. This is a hurdle that is limited mainly by the money available. The technology exists (e.g., chips sets and standards), but DOE funding and the current markets have not permitted the development into production equipment. A major challenge is supporting high-resolution displays. **Hurdle**

Digital approaches for desktop delivery – Various approaches will be explored for providing effective delivery of high-performance data visualization to remote displays. In this case, remote can be intra-site, but must also satisfy inter-site needs. Approaches may leverage methods such as image-based visualization/rendering as an alternative to data migration; advanced data services including multiresolution, feature detection/extraction and compression/decompression, and high performance networking/communications. Note that desktop delivery requirements will also demand high display resolutions and frame rates. Although the technology is under development, it still remains a challenge to determine which technology is most effective for the high-pixel screens and large datasets. **Hurdle**

Display-intensive offices – The goal here is to provide many wide screens to the offices of designers and analysts. University research and alpha users have demonstrated the utility of increasing screen size and pixel counts. The displays support both multiple views (many windows) and higher resolutions than are commonly found in offices. This deployment is limited by available funding. Low-cost, high-resolution displays or projectors are needed. There are also some hurdle aspects to this milestone relating to the development of the software to correct keystoning, and other practical issues resulting from the tight spaces in offices. **Hurdle**

2003 **Data subsetting/manipulation services integrated with interactive visualization solutions** – A complete set of data services, including those provided by Scientific Data Management project, will be integrated with visualization components to enable efficient, effective data exploration and comprehension. While most of the data-size performance targets identified on the ASCI visualization curve are specified for a single timedump, the resulting end-to-end environment will also support efficient processing of time-series data. The challenge is integrating tools developed by many teams into a seamless functional system as dataset sizes and hardware platforms are evolving rapidly. **Hurdle**



Collaborative interactive distance visualization – Technologies will be developed and deployed that enable the ASCI designer/analyst to extend the distance visualization capability to support

2003 cont. collaboration with users at remote locations. This will allow a work group with different locations to visualize and interact with a large dataset. Many tools today either share the display or replicate the data. Replicating the data is impractical at ASCI data scales, and at high display resolutions, the WAN bandwidth is insufficient for numerous users. The challenge is to develop the solution that leverages the high-performance parallel visualization tools. **Barrier**

I**mproved user interfaces for shared facilities** – Shared facilities supporting a group of people sharing large screens and high-performance visualization systems require means of interacting with the computer beyond a mouse and keyboard. University researchers have demonstrated technologies like hand-held Palm Pilot and gesture-recognition-based interfaces, but none are well developed and ready for deployment. The required technologies are still in the basic research phase, and their application on large datasets and for scientific visualization is unexplored. **Barrier**

2004 **Real-time interaction with 2-terabyte timedump** – Implies the ability to interact with a full timestep of result data whose raw size is approximately 2 terabytes, at frame rates that are defined at the beginning of this section. The basic computer science and hardware technology required to meet the high I/O and processing rates do not exist today. **Barrier**

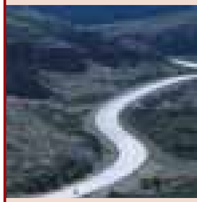
Scalable Rendering System driving a 16-megapixel display in offices – In order to see important details present in data at image resolutions that are better matched with the data resolution of high-fidelity geometric grids, display capabilities are needed that enable data to be visualized at higher pixel resolutions. The object is to take the capabilities available earlier in shared facilities and extend them to the offices of designers and analysts. The technology exists with the challenge being how to cost-effectively deploy it in the office setting. **Hurdle**

I**ntegrated distance visualization and data services available across the WAN** – This milestone is to extend the 2003 "Data sub-setting/manipulation services integrated with interactive visualization solutions" milestone to allow seamless access across a WAN. The challenge is integrating tools developed by many teams into a seamless functional system as dataset sizes and hardware platforms are evolving rapidly. This also is likely hampered by money as the WAN bandwidth may be a limiting factor. **Hurdle**

2005 **Data mining/discovery integrated with visualization** – Large 200 terabyte databases are too large to be visually explored; computer-augmented techniques to search the database and suggest points of interest are required. The main challenge is to develop the algorithms that locate "interesting" features. The integration of these tools into the parallel visualization tool set is secondary. **Barrier**

Scalable Rendering System driving a visual acuity tiled display (100M pixel) in shared facility – In order to see important details present in data at image resolutions that are better matched with the data resolution of high-fidelity geometric grids, display capabilities are needed that enable data to be visualized at high pixel resolutions.   These effective resolutions can be achieved via the integration of multiple displays, but this is a barrier that is limited by the scalability of the rendering system. **Barrier**

Collaborative office environments – Completing the 2002 milestone, "Display intensive offices," will deliver offices that more fully engage an analyst. Accomplishment of this milestone is also dependent on successful completion of the 2003 milestone, "Collaborative Interactive Distance Visualization" This milestone extends these capabilities and adds video conferencing and distance collaboration. This is a planned activity with largest challenges being limited budgets and security restrictions. **Planned**



CURRENT STATE OF ASCI VISUALIZATION

The Visualization work at the three laboratories has consisted of teaming with universities and industry for several years in efforts to address the technology needs. Progress has been made in data handling, data exploration, environments, displays, and distance visualization.

Data Handling and Data Exploration. The three laboratories are engaged in research with both universities and industry to explore cluster-based solutions to I/O bandwidth needs. In 1999, a cluster was used to set a world speed record in sorting. Advanced wavelet techniques have been demonstrated to permit rapid browsing of very large datasets. High-performance disk-to-frame-buffer volume rendering capabilities have also been demonstrated.

Environments, Rendering, and Displays. Continued collaborations with universities and industry have resulted in improved visualization environments. Large tiled displays, with as many as 20 megapixels, and shared immersive facilities are deployed at the laboratories. Prototype flat-panel displays have been developed that deliver 9 megapixels across a 21-inch diagonal display. The feasibility of cluster-based rendering has been demonstrated, achieving rates as high as 300 million triangles/s for polygonal rendering of a very large isosurface. Industry partners are improving graphics card drivers for Linux; ASCI funding directly supported the development of Linux drivers for nVIDIA and 3dfx graphics cards. Partnering with university researchers continues on the development of scalable rendering software and technologies for the conference room and offices of the future.

Distance Visualization. Distance visualization has been demonstrated across the WAN, between New Mexico and California, using special hardware compression/decompression and Asynchronous Transfer Mode (ATM) networking equipment. The three laboratories are continuing to collaborate with universities and industry to develop improved techniques for distance image delivery, tele-collaboration, video-conferencing, and other techniques for distance visualization.

SUMMARY, CONCLUSIONS, RECOMMENDATIONS

Plans for the next five years will focus on topics discussed here. The major challenges are in scalable visualization (rendering and data handling), scalable data exploration, and distance visualization. Addressing these areas requires advancing both hardware and software. A combined effort of the national laboratories, universities, and industry will continue to address ASCI's visualization needs.

2.2 SCIENTIFIC DATA MANAGEMENT AND DISCOVERY

ASCI's current computing platforms produce a huge volume of complex data, which are expected to increase within the next five years. Managing these multi-terabyte datasets in petabyte archives and understanding and evaluating mesh-based simulation datasets are major challenges for ASCI researchers — challenges compounded by the fact that some of the data are generated and stored remotely. A delay in addressing data management issues of this magnitude can significantly impact researchers' efforts to quickly access and analyze these data.

Scientific data management and discovery involves building tools to enable weapons scientists and engineers to explore and understand terascale datasets. Today's ASCI scientists are overwhelmed by the vast amount of data produced by applications using current computing platforms and are hindered in their efforts to analyze their results by inadequate data management and discovery tools. One major problem is understanding and comparing multi-terabyte datasets in petabyte archives. Although the management of massive datasets is a problem addressed in other scientific and experimental contexts (e.g., satellite images, high-energy physics), the problem of analyzing mesh-based simulation datasets of this magnitude has not been adequately addressed. A delay in addressing the terascale scientific data management issues will severely limit ASCI users' ability to find useful information and exploit their results.

Several industries also stand to benefit from the technologies required for ASCI data management and discovery. Medical imaging and the petroleum industry both have ongoing needs for understanding massive datasets. Advances in data technologies stand to benefit both industries. Other areas that could benefit from the technologies include environmental and climate modeling, computational engineering, basic science research, and computational biology.

TECHNICAL ISSUES AND CHALLENGES

Addressing the scale, complexity, and remoteness of data is the major challenge for the data management effort. Data complexity comes from the computational algorithms employed and the broad range of discrete representations used by the applications, from simple product meshes to grids of general polyhedra. In addition, ASCI simulation codes require models with a fidelity and spatial resolution far beyond the current state of the art, which greatly magnify the volume of the resultant data. Remote data access adds both technical and procedural challenges. In the tri-lab computing environment, simulation results may be computed on a remote resource and analyzed and stored in another location. Differences in the computing environments and the issues relating to managing remote data add significant complexity for the user as well as for the software tools developed to assist them.

If managing ASCI data is a challenge, then exploiting it is even more so. ASCI data discovery is a collection of techniques and tools for representing and extracting information from simulation data. It is true that the demands of the e-commerce industry have resulted in commercial products that can handle and "mine" data comparable in quantity to that of ASCI. *However, those tools most often require that the data be represented in a relational schema and already be prepared for the analysis, thereby rendering them inapplicable to ASCI's mesh-based physics data.*

Many important challenges and research opportunities exist in the ASCI data management and discovery area. One is the simple but essential requirement for scalable algorithms for computing low-level features, useful in characterizing and classifying the data. Scalability is crucial because these algorithms will be applied to terascale datasets. Also needed are query and pattern recognition schemes with various advanced properties. One property is the ability, in the interest of user interactivity, to trade off the accuracy of their responses against the time invested in them. Another is sufficient robustness to handle datasets that have not been "cleaned" because the sheer volume of ASCI data makes any preprocessing expensive. A third characteristic is parallel pattern recognition algorithms that scale to ASCI data sizes and are flexible enough to handle distributed data. An additional and long-term requirement is algorithms for analyzing the analysis. That is, there are myriad sources of uncertainty in physics simulation, and post-processing only adds more of them. It is crucial that there be a scalable framework for capturing and integrating these sources of uncertainty into the final conclusions drawn from an analysis. Additional technical issues exist because these algorithms and techniques can be costly in terms of both processor time and wall clock time, and they can generate substantial amounts of data that further stress system resources.

CURRENT STATE OF ASCI DATA MANAGEMENT

The ASCI data exploration solution for improving data organization and management leverages the creation and exploitation of meta-data. Meta-data can range from a scientist's notes on the relevance of a study, to system level meta-data that documents the size, type, and creation date of a dataset, to intermediate data files generated to support post-processing. The initial suite of data management and preparation tools was deployed in 2000. The SimTracker tool (see Figure 2.2-1) for generating Web-based summaries of calculations is being used with several applications codes within the three laboratories. Meta-data editing, searching, and browsing tools are in beta release, and additional tools for automatic meta-data creation are in the prototype stages. These tools are being integrated into a framework supporting a wide variety of query and exploration methods. In addition, at the beginning of CY2001, the first production release of an ASCI parallel data models library became available. This library collects and shares simulation data through the use of a mathematical-based common data model using principles from the field of topology.

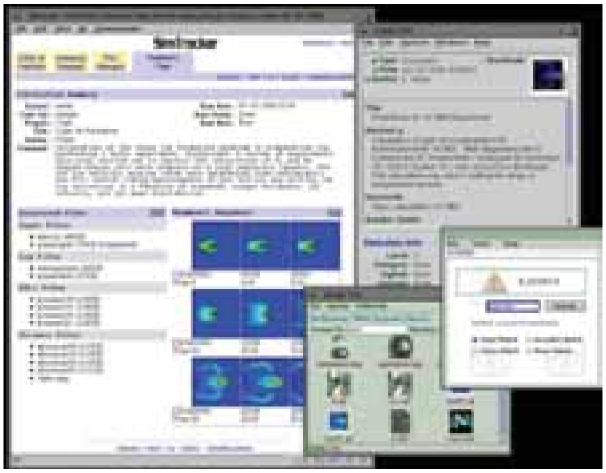


Figure 2.2-1. New software tools such as SimTracker (clockwise from upper left) and companion meta-data editor, searcher, and browser help scientists keep track of and access ASCI analyses.

The laboratories' data discovery effort has developed initial parallel and scalable pattern recognition algorithms. These algorithms are deployed to ASCI platforms and designed and tuned for the ASCI problem: densely sampled, spatially organized, mesh-based simulation data. In addition the team has developed new, faster, more accurate, and parallel decision-tree algorithms, prepared a prototype tool for user validation of data mining results, and demonstrated data discovery techniques with an astrophysics application using developed software. Furthermore, a prototype ad hoc query tool for selecting subsets from simulation datasets has been demonstrated and will undergo further development through 2001.

Road Map for Scientific Data Management and Discovery

The associated technology road map visually depicts the five-year status (calendar year 2000 to 2005) of desired capabilities/activities within a functional area. Each capability is color coded to show what level of R&D effort it requires or anticipates.

R&D Effort Indicator:

Accomplished = completed
Planned = ASCI will accomplish even with slight budget fluctuations
Hurdle = ASCI will need some help
Barrier = ASCI will need significant help from the high performance computing (HPC) community

NOTE: Both hurdles and barriers represent research opportunities for the HPC community.

Road Map for SCIENTIFIC DATA MANAGEMENT & DISCOVERY CAPABILITIES

Functional Area	CY 2000	CY 2001	CY 2002	CY 2003	CY 2004	CY 2005
Data Access and Preparation	Simple *ad hoc* queries on small simulation datasets		Optimized data access for improved storage interaction	Complicated *ad hoc* queries on large datasets Automated speculative data access DOE complex-wide integrated data access across wide range of data sources		ASCI-pertinent legacy information integrated into data access infrastructure
Meta-data Infrastructure and Applications	Automatic simulation data archiving and retrieval for high-demand codes				Automatic simulation data archiving and retrieval for all ASCI codes	
Data Discovery	Scalable pattern recognition for medium-sized mesh-oriented data	Scalable geometrical feature extraction for mesh-oriented data Smart comparison of small simulation datasets Guided feature detection on medium simulation datasets	Scalable tool kit for extraction of common features	Smart simulation/experimental data comparison tools on large datasets	Scalable pattern recognition for massive mesh-oriented data Feature extraction and analysis simultaneous with simulation Interactive-example-based discovery in simulation datasets	Postprocessing and discovery operations integrated with uncertainty quantification analysis
Data Models and Formats	Robust high-level data model for ASCI simulation data Application-oriented parallel IO capabilities on ASCI platforms	Advanced user-directed content-based subsetting of simulation results	Data sharability across high-demand ASCI codes		Application-oriented data manipulation operators	Ubiquitous data infrastructure across all ASCI codes

R&D Effort Indicator 🔵 ACCOMPLISHED 🟢 PLANNED 🟡 HURDLE 🔴 BARRIER

ACCOMPLISHED—Completed
PLANNED—ASCI will accomplish even with slight budget fluctuations
HURDLE—ASCI will need some help from the HPC community
BARRIER—ASCI will need significant help from the HPC community

TIMELINE

This timeline elaborates on the activities that appear on the preceding road map. These activities represent desired capabilities and are not a statement of ASCI milestones. To achieve these capabilities would require a multiyear effort; however, they appear in the timeline only once in the first year that the activity would be useful for consideration.

Calendar Year **Description and Status**

2000 **Simple ad hoc queries on small simulation datasets** – Support simple ad hoc queries on the original mesh data. The queries include range and point queries on field and geometric variables, as well as simple topological queries. Query results should be visualized. **Planned**

Scalable pattern recognition for medium-sized mesh-oriented data – Fully parallel and scalable pattern recognition algorithms that are deployed to ASCI platforms and are designed and tuned for the ASCI problem: densely sampled, spatially organized, mesh-based simulation data. **Accomplished** in CY2000

Robust high-level data model for ASCI simulation data – A flexible and extensible mechanism for describing the numerous complex data representations used by ASCI simulation codes. The role of the ASCI data model is much the same as that of the widely used relational data model for database management systems. **Accomplished** in CY2000

Application-oriented parallel I/O capabilities on ASCI platforms – Development of a software system to be integrated with ASCI physics-based simulations for capturing and accessing datasets with high performance on ASCI parallel architectures. The software interfaces are suitable for simulation systems and use a context/language that is familiar to computational physicists. **Planned**.

Automatic simulation data archiving and retrieval for high-demand codes – Most frequently used ASCI simulation codes will be deployed with tools to automatically archive and retrieve simulation results and datasets. **Planned**.

2001 **Smart comparison of small simulation datasets** – Scalable algorithms deployed to ASCI platforms that provide high-level, intelligent comparison of simulation datasets computed on equivalent meshes. This means that they support a flexible definition of what constitutes a "difference" and provide rich, meaningful reports of the areas found to be different. This activity is described as a **Hurdle** because of the difficulty in quantifying differences and the variety and size of datasets.

Guided feature detection on medium simulation datasets – Pattern recognition and data analysis tools that "learn by example." This means tools that are shown areas of a mesh by a user and can then find similar areas in the same or other data sets. These tools are intended to be screening tools for a user faced with datasets too large to examine exhaustively by hand. Thus, they need to be very sensitive, though a substantial false alarm rate is acceptable. **Hurdle**

Scalable geometrical feature extraction for mesh-oriented data – A source of information important for successful feature detection and pattern recognition is local geometry, which is implicit, not explicit, in simulation data sets. Thus, scalable algorithms are required for computing and recording local geometrical characteristics at all points in a simulation dataset. **Hurdle**

Advanced user-directed, content-based subsetting of simulation results – Support for a data manipulation language that uses simulation-code concepts for selecting and describing areas of interest in the dataset regardless of computer-science data structures used and physical data layout. **Planned**

2002 **Optimized data access for improved storage interaction** – The integration and capturing of user access patterns. This information will allow improved data caching, reorganization of the data for more efficient writing, and accurate, up-to-date information on what datasets are likely to be accessed soon, so that the high-performance storage system (HPSS) can optimize the serving of all of its requests. **Planned** (in conjunction with our ASCI alliance partners).

Data sharability across high-demand ASCI codes – Support for interoperable data across the most often used ASCI codes and commonly used tools such as visualization applications within the three laboratories. Computational scientists employ an enormous variety of representations when modeling physics processes on computers. Problems arise when different representations are required to exchange data with one another or with other software packages. This activity is described as a **Hurdle** because sharing data among simulation models is difficult and persistent.

Scalable tool kit for extraction of common features – Though pattern recognition and low-level data analysis can be problem independent, the selection of high-level features (e.g., vortices, crumple zones) to aid in the detection of specific behaviors in a simulation dataset is problem dependent, and is best tackled by a domain expert. Those experts will require a software library of scalable methods for calculating a broad range of common low-level features (e.g., local geometry such as inflection points, curvature) implemented with algorithms tuned to ASCI data. **Hurdle**

2003 **DOE complex-wide integrated data access across a wide range of data sources (simulations, archives, products)** – The meta-data architecture will be opened to connect the entire DP complex, and the meta-data integration effort will be opened to include information pertinent to the entire "from design to decommission" lifecycle. Because of the technical, security and organizational difficulties, this activity is defined as a **Barrier**.

Automated speculative data access – The "optimized data access" effort will be enhanced to take into account subtle workflow and process relationships between data sets and types of data, permitting the recovery and presentation of ASCI data almost before the user asks for it. **Hurdle**

Smart simulation/experimental data comparison tools on large datasets – Data comparison tools are required that can handle two very difficult problems. One is the sensible objective comparison of simulation datasets where the underlying mesh can differ across the data sets. This permits validation as a mesh is refined, and also permits adaptive mesh refinement (AMR) data to be compared.

The other problem is the sensible comparison of simulation data to supposedly matching experimental data, data that is most likely described and characterized very differently. **Hurdle**

Complicated ad hoc queries on large datasets – Support complex ad hoc queries from highly compressed models of the original mesh data. Scalability is an issue, as well as producing models rich enough to support a wide range of queries. Queries should include general topological queries and generalized filters over field variables where the operands can be functions of field variables rather than simply field variables and constants. Result return should be progressive, performance should be driven by both time and error requirements. **Planned**

2004 **Application-oriented data manipulation operators** – Development of a tool kit of methods for operating on simulation datasets captured using the data models and format (DMF) software system developed at the three laboratories. Desired data transformations and manipulations are user directed and can be applied to the data prior to making it persistent. **Planned**

Feature extraction and analysis simultaneous with simulation – The pattern recognition, feature detection, and feature characterization algorithms already deployed must be adapted to permit their effective use simultaneously with the simulation that is providing the data. One goal is to provide high-level information that permits the steering of the simulation by humans or by other computational processes. **Hurdle**

Automatic simulation data archiving and retrieval for all ASCI codes – All ASCI simulation codes will be deployed with tools to automatically archive and retrieve simulation results and datasets. **Planned**

Scalable pattern recognition for massive mesh-oriented data – The pattern recognition algorithms already deployed for medium data sets must be tuned and tested to ensure that they continue to scale and provide responsive answers even with massive datasets. Therefore, it may be necessary to implement progressive methods that can always provide an answer, no matter how little time is provided, but which provide more accurate answers the longer they have to process. Because of the technical difficulty, this activity is defined as a **Barrier**.

Interactive, example-based discovery in simulation datasets – User identifies an object/event of interest in the dataset manually, and based on this single example, the system finds other examples of similar objects on the fly. The resultant quality should degrade gracefully with differences in scale, and be rotation- and translation-invariant. The resultant return should be progressive; the performance should be driven by both time and error requirements. Because of the technical difficulty, this activity is defined as a **Barrier**.

2005 **Ubiquitous data infrastructure across all ASCI codes** – Application of common data model and data management infrastructure with all ASCI codes on all tri-lab ASCI resources. Such an infrastructure would enable data interchange and interoperability across ASCI codes and tools. **Hurdle**

2005 cont. **Postprocessing and discovery operations integrated with uncertainty quantification analysis –** Explicit handling of uncertainty is crucial in knowing exactly how, and how much, to trust an analysis. In addition to the uncertainties inherent in simulation codes themselves, feature extraction and other discovery post-processing introduce additional sources of error, especially when speed of processing is traded against detail. An explicit framework, and its supporting mathematics and software, is required to capture and integrate all of these sources of uncertainty. Because of the technical difficulty, this activity is defined as a **Barrier**.

ASCI-pertinent legacy information integrated into data access infrastructure – Seamless access to historical information and multiple distributed data sources useful to ASCI scientists including experimental results, documents, drawings and simulation codes and their results. Information is properly protected and access is subject to security constraints and need to know (NTK). This activity is classified as **Barrier** because of complexity. Many data sources are currently unavailable electronically and are controlled by numerous agencies.

SUMMARY, CONCLUSIONS, AND RECOMMENDATIONS

Future plans in this technology area include increased research and collaborations with other ASCI projects and with researchers in data exploitation/data discovery from universities, governmental agencies, and industry. The latter covers issues such as representing, detecting, and extracting features, reducing data representations, scalable pattern recognition, and querying, comparing, and mining data at the terascale level.

In addition, the evaluation of architectures for high-performance data manipulation, including the integration of hardware data servers and software, is a primary focus for CY01. Activities will be conducted to integrate and deploy prototype data services systems that exploit the architecture effort developed throughout FY00 and FY01. Areas to address in these efforts include storage capacity, good storage access rates, and high throughput transfer rates. Implementation of this end-to-end system requires integration of technologies developed with the Alliance partners: Visual Interactive Environment for Weapons Simulation (VIEWS) development centers and Data and Visualization Corridors (DVC) collaborators. These initial investigations will likely uncover many additional challenges and research directions.

This technology area maintains a strong research focus and depends heavily on collaboration. To stay on track with future ASCI requirements will require a combination of an accelerated research program involving close collaborations with external researchers in data discovery and management and the development and deployment of the new tools and techniques for the ASCI scientists and their application codes.

2.3 DATA STORAGE AND FILE SYSTEMS

As sustained parallel processing performance continues to improve on ASCI platforms, saving and retrieving huge volumes of data are becoming even more demanding. To maintain balance between computational speeds and I/O rates, many storage devices per compute node and with parallel access to all of them will be required. ASCI needs to motivate research and development into improving existing, and building new, storage and archival systems technologies

Within five years, weapons scientists will require data storage and file systems orders of magnitude improved in terms of larger capacity, higher bandwidth, lower latency, and faster transaction rates than those available today. To maintain a balance with future ASCI platform computational speed and memory capacity, file systems in 2004 to 2005 will be expected to deliver sustained throughput of 100+ gigabytes/s (see Figure 2.3-1). Archival systems are expected to deliver 10+ gigabytes/s throughput with 50+ petabyte capacities. Furthermore, these systems will need additional security and easier accessibility. Without such improvements, scientists may no longer be able to effectively save or retrieve data generated by physics applications.

Current market forces are not sufficient to meet ASCI requirements. Our objective is to meet these difficult goals by accelerating and influencing new advances in academic research efforts as well as in commercial components. ASCI may not be positioned to dictate requirements for component suppliers focused on broad commercial markets. However, the need for ASCI to motivate, develop, and integrate new parallel I/O techniques and storage system components is critical. Other scientific and engineering communities including bio-informatics, high-energy physics, real-time data capture, and signal intelligence should also benefit from accelerating and influencing data storage and file system technology development.

TECHNICAL ISSUES AND CHALLENGES

Current ASCI application programs are extremely demanding in their use of high levels of parallelism (thousands of processors), rapid generation of data (trillions of floating point operations per second

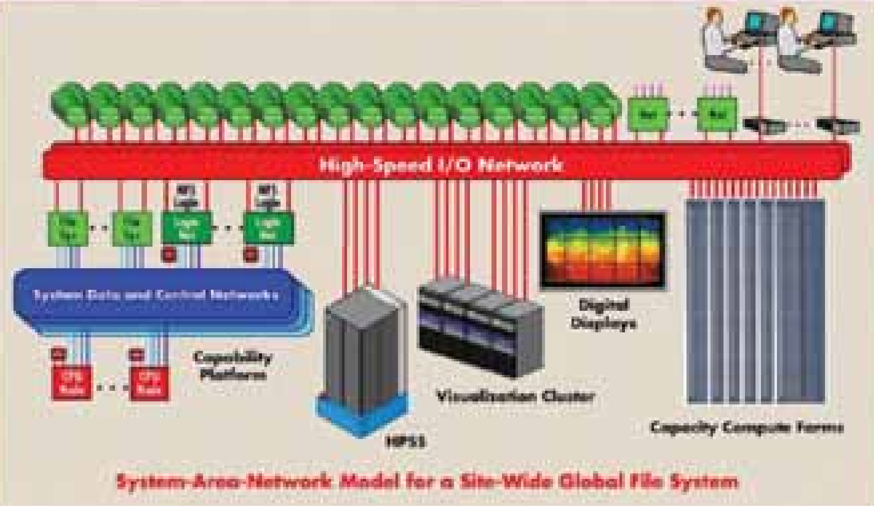


Figure 2.3-1. ASCI platform, data storage, and file system architecture for 2004.

[FLOPS]), and the bulk of their data sets (terabyte memories, 100-terabyte file systems, and petabyte archives). Processor speed over the last decade has doubled about every 18 months, following a rule known as Moore's law. Actual computational power grows faster than this rate, however, fueled not only by faster processors but by the increase in processor count that can be applied to the application. Similarly, ASCI I/O requirements are growing by an order of magnitude every three years — again faster than Moore's Law. Sustained processing performance will continue to improve on ASCI platforms, and to maintain balance between computational speeds and I/O rates will require many storage devices per compute node, and parallel access to all of them as shown in Figure 2.3-1.

Users have historically saved the same kinds of data, in the same kinds of ways, regardless of available processing speed. Physics restart files, mesh subsets, and visualization files will almost certainly need to be saved, and these tend to grow with the size of memory and mesh resolution. Writing, reading, and saving these files can be categorized as either productive or defensive I/O. Productive I/O produces data used for subsequent analyses. Defensive I/O operations are used to generate restart files as a way to ensure that long-running calculations can be resumed from a checkpoint or snapshot rather than from the beginning of a run. Restart files can also be used to adjust job parameters or to repair meshes, in addition to safeguarding against intermittent hardware failure. Present plans for 100-teraOPS ASCI platforms define daunting rates for both productive and defensive I/O (50 to 100+ gigabytes/s).

It may not be possible to save all of the files produced by productive and defensive I/O given the available data storage and file system technologies. Users will face several choices: generate less data, discard or overwrite some of the generated data, or move generated data to archival storage. Unfortunately, discarding data partially defeats the purpose of advanced physics simulations, which is to analyze results and create new scientific understanding. Users will likely be more willing to overwrite restart files resulting from defensive I/O operations, alleviating some of the load on archival storage.

I/O, storage, and file systems traditionally have received less attention than CPU speed, memory size, and peak FLOPS in the world of scientific computing. Compounding the problem are the magnetic storage devices themselves, upon which file systems and archives are implemented. These devices are highly dependent on mechanical movement, whose speed has remained nearly constant (solid-state disk being a notable, but expensive, exception). While areal density of magnetic recording media has increased such that disk capacities have recently been doubling every nine months, single disk bandwidth is increasing only about 40% per year [1,2]. Furthermore, advancements in disk seek times and rotational latencies are usually incremental and provide only small improvement.

In recent years, single-spindle disk capacity has grown from less than 1 gigabyte up to 100 gigabytes (vendors now promise 200 and 400 gigabyte disks in the next few years). Disk I/O rates have moved through a narrower range of about 4 to 40 megabytes/s per spindle because of the slower rate of change in linear bit density and mechanical latency. Tape evolution has followed a somewhat similar path. In the past decade, half-inch longitudinal tape capacity increased from about 0.5 gigabyte to 60 gigabytes per cartridge, while tape bandwidth only increased from 3 megabytes/s to about 15 megabytes/s. During this same period, tape vendors improved "time-to-first-byte" characteristics by employing serpentine patterns (reading or writing back and forth on different tracks, as in a continuous S-pattern) to minimize access to a given record. More recently, they have also offered dual-reel, mid-point load technology to further reduce latency.

Tape latencies are still extremely poor when compared to disk. The price of disk is now so low that disk competes directly with tape in several market segments. Some users of large data have started to question the wisdom of using tape (see Figure 2.3-2), given the

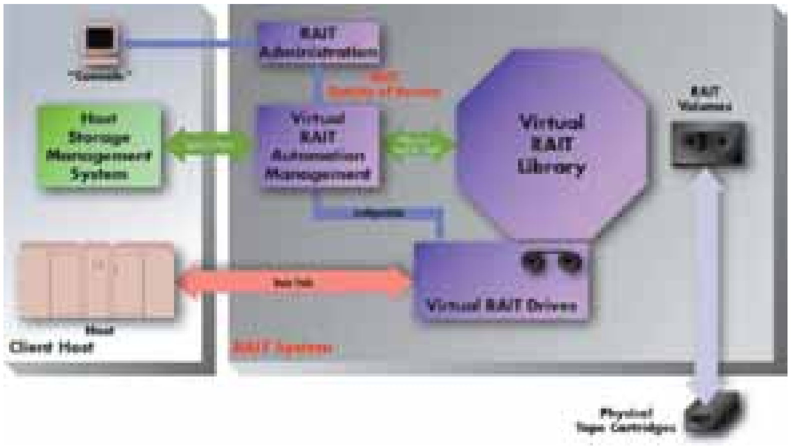


Figure 2.3–2. Current Redundant Array of Independent Tape (RAIT) device architecture.

increase in disk areal densities and the more-than-adequate reliability provided by mirroring and Redundant Array of Independent Disk (RAID). The price of tape cartridges has not fallen as quickly as disk, and historical cost/benefit ratios that made tape preferable for lower levels of a traditional storage hierarchy are under debate. Magnetic tape also has several difficult engineering problems to overcome not present in disk, such as media stretch and track alignment at high tape speed. An ASCI 100-teraOPS platform will have as much as 2.5 petabytes of global disk, larger than the current capacity of ASCI's tape-based archival storage systems. The ever-increasing size of ASCI platform disk coupled with changing cost/benefit arguments for tape will require increased attention from ASCI storage system architects.

ASCI storage system developers and integrators planned for relatively few, but very large, "records" — restart files, mesh subsets, and visualization files. Code execution support environments also assumed capacity and bandwidth would be much more important than latency (the time it takes to move the first byte of a terabyte size file is negligible compared to the time it takes to move all the data). For example, at 100 megabytes/s, the time to transfer a terabyte of data is almost three hours. Contrary to expectation, some code team users decided not to produce large files during their runs, but rather to produce tens of thousands of smaller files to help simplify I/O issues and code development. Design physicists who will later use these codes may be able to produce fewer, larger files. File "bundling" utilities now under development may be able to reduce the number of physical files managed by a file system or an archive, increasing overall throughput. However, there will still be interest in organizing, managing, and accessing millions or billions of files or data objects, particularly for data discovery, a nontrivial issue.

A further complication is predicting and planning appropriate interfaces and interconnects to move data between ASCI resources. Fibre Channel is moving from 1 gigabit/s to 2 gigabits/s and eventually 4 gigabits/s speeds. Ethernet has progressed from 1 to 10 gigabits/s, with 40 gigabits/s demonstrated in the laboratory and 100 gigabits/s rates now in definition. InfiniBand^{TM}, a new architectural specification for processor and device interconnection [3], promises to make clusters look more like a tightly coupled symmetric multiprocessor (SMP) than existing interconnect technology and will eventually be inexpensive enough for wide-scale deployment. InfiniBand^{TM} calls for a series of 1x, 4x, and 12x implementations through multiple physical lanes (each lane is currently specified at 1.25 gigabits/s). It would require 70 12x connections to achieve 100 gigabytes/s in one direction. While Ethernet and InfiniBand^{TM} promise higher rates than Fiber Channel, these technologies are not yet being used as storage device interfaces.

Alternative ultra-high-density data storage technologies are also being researched at several companies, universities, and industrial consortia. Volume holographic storage has been investigated for more than a decade and attempts to change the traditional storage paradigm by writing vast amounts of data "inside" a storage medium, instead of "on top of" a storage surface [4]. A potential problem with holographic storage is that it may allow high capacity or high bandwidth, but probably not both. MEMS-based (micro-electro-mechanical system) mass storage, which positions small probe tips over storage media, can potentially create a new generation of nonvolatile, inherently parallel, rewritable mass storage [5]. Holographic and MEMS-based storage, while not yet ready for the broader commercial market, are examples of potentially disruptive technologies that may have large impact on the design of ASCI data storage and file systems in the next decade.

ASCI is not positioned to strongly influence storage component builders targeting the broad commercial markets. Rather, the need to recognize, help develop, and integrate new parallel I/O and storage systems is the main challenge for ASCI. Parallelization and distribution of storage resources complicate things at every level. To scale throughput rates to 100+gigabytes/s, software bottlenecks arising from contention for metadata resources and locks will need to be eliminated. In addition, memory, bus, and switch fabric bandwidths are extremely important in parallel I/O and may present hardware bottlenecks if the interconnects are not fast enough to handle thousands of parallel disks being simultaneously read or written.

Another problem is global accessibility of files; it is easy to connect a set of local disks with a separate file system to each computational node, but applications really need arbitrary access from any node to any file. In fact, ideally they should have concurrent access from all nodes to any single file, with a proportionally large aggregate throughput. This accessibility is difficult to implement, especially with high transfer rates. Moreover, an even higher degree of global accessibility is desired; different, possibly heterogeneous platforms at a single site, or even from geographically distant sites should have access to a shared set of files, with high performance. Distributed "global" file system implementations that manage locking through networks connecting cooperating machines need to look like a single file system on a tightly coupled SMP for good performance.

Yet another problem is that the large number of mechanical devices in storage systems attached to ASCI computers can mean problematic overall reliability of storage; aggressive use of fail-over software can be employed to gain back reliability, but this is complex and difficult to develop. Data reliability and data accessibility are fundamental concerns for ASCI scientists. Loss of physics data is unacceptable. The ability of archived data to be accessible also presents problems related to continual technology turnover. Archived data must usually be copied from older storage devices to newer devices, as new devices become available. Rewriting of data will always be limited by the slower, older devices, no matter how fast the new devices are.

Newer storage system architectures, such as Network-Attached Storage (NAS), Storage-Area-Networks (SANs), Internet SCSI (iSCSI), Internet Fibre Channel Protocol (iFCP), and Object-based Storage Devices (OSD) are rapidly emerging and gaining wide acceptance [6]. Some of these approaches may help provide uniform accessibility, heterogeneous data sharing, and efficient layout and space allocation, but they do not themselves solve the ASCI-specific problems of high performance. ASCI also has critical security and NTK issues that present problems when attempting to integrate new storage system architectures. Secure, authenticated access has not yet been completely addressed in NAS, SAN, and OSD. Another difficulty that arises in using new architectures and parallel I/O systems is the need for standard programming interfaces. Old serial I/O interfaces are inadequate to use parallelism effectively, and current parallel file systems do not extend to multiple vendor platforms. Therefore, inherently parallel, standardized interfaces such as MPI/IO must be developed and provided to sit atop vendor proprietary file systems and storage devices.

The ASCI-supported High Performance Storage System (HPSS) is faced with meeting many difficult requirements for multipetabyte archival storage systems. Hurdles include integration of legacy systems, on-the-fly software upgrade and conversion, meta-data interchange, inter-site data space linkage, technology migration, and implementation of new storage architectures. Still more issues are security, scalability, bulk data movement, data replication, multiple serial and parallel interface support, multivendor operating system and device support, resource allocation, load balancing, and parallel and low-latency protocol definition. These are all significant challenges and must be overcome in a cost-effective way.

In the timeline that follows, throughput and storage capacity numbers represent what we believe to be realistic high-end ASCI requirements driven by an analysis of user scenarios and technology assumptions. Estimating data storage and I/O needs is at best an inexact science, influenced heavily by changing needs of ASCI scientists and capabilities of the ASCI platforms. An excellent discussion on estimating I/O requirements in ASCI-sized computational environments, as well detailed information on I/O in high-performance scientific computing, can be found in Reference 7. One data rate assumption used by ASCI is that users dump some, or all, of memory on a periodic basis (for defensive I/O as described above). These dumps are typically taken once an hour for restart files and should account for no more than 10% overhead (i.e., 90% for computing, 10% for restart dumps). If dumps are not overwritten, but instead sent to an archive before the next restart dump, this would require an archival (tape) transfer rate about an order of magnitude less than platform disk rates. An older estimation technique that has been used in the past to size mass storage capacity is 750 times the platform's memory size per year moved to the archive.

In summary, the complex storage and I/O needs of ASCI applications exceed the mass market's requirements; meeting ASCI needs takes strategic planning and ongoing discussion with the ASCI user community, collaboration with academic researchers, and cooperation with the data storage and file system industry.

Road Map to Data Storage and File Systems

The associated technology road map visually depicts the five-year status (calendar year 2000 to 2005) of desired capabilities/activities within a functional area. Each capability is color coded to show what level of R&D effort it requires or anticipates.

R&D Effort Indicator:

Accomplished = completed
Planned = ASCI will accomplish even with slight budget fluctuations
Hurdle = ASCI will need some help
Barrier = ASCI will need significant help from the high-performance computing (HPC) community

NOTE: Both hurdles and barriers represent research opportunities for the HPC community.

Road Map for
DATA STORAGE & FILE SYSTEMS CAPABILITIES

Functional Area	CY 2000	CY 2001	CY 2002	CY 2003	CY 2004	CY 2005
Security	Access Control Lists in HPSS					Fine-grained need-to-know access control
Accessibility	Layered tri-laboratory applications I/O architecture RFI issued for SGS-FS	Prototype tri-laboratory federated namespace RFP issued for SGS-FS	Production tri-laboratory federated namespace Demonstration of SGS-FS on heterogeneous clusters	Distributed Resource Management (DRM) support for storage systems Demonstration of SGS-FS on heterogeneous clusters	Scientific Data Management support for storage systems	
Aggregate Speed	40 MB/s Tape Prototype optical tape and high-end RAIT systems	5 GB/s Disk COTS RAIT (tape)	1 GB/s Tape COTS optical tape	20 GB/s Disk Next-generation HPSS systems (post-Release 5)	4 GB/s Tape HPSS secure access for Object-based Storage Devices	100+ GB/s Disk Evaluation of alternative mass storage systems (e.g., MEMS holographic)
Storage Capacity	500 TB Archive	1 PB Archive	5 PB Archive	10 PB Archive	25 PB Archive	50+ PB Archive Evaluation of alternative mass storage systems (e.g., MEMS holographic)

R&D Effort Indicator ● ACCOMPLISHED ● PLANNED ● HURDLE ● BARRIER

ACCOMPLISHED—Completed
PLANNED—ASCI will accomplish even with slight budget fluctuations
HURDLE—ASCI will need some help from the HPC community
BARRIER—ASCI will need significant help from the HPC community

TIMELINE

This timeline elaborates on the activities that appear on the preceding road map.

Calendar Year **Description and Status**

2000 **Throughput and capacity** – 500-terabyte archives deployed, multiple gigabytes/s parallel disk I/O demonstrated, 40 megabytes/s production striped HPSS tape performance, 80-megabytes/s parallel tape demonstrated with prototype Redundant Array of Independent Tape (RAIT) hardware and software. (A 500-terabyte tape archive could be read or written in 2 to 3 months at 80 megabytes/s.) **Accomplished**

Layered tri-lab applications I/O architecture – Integrated, multilayer end-to-end architecture to support ASCI scalable I/O, data management, and visualization approaches (API layer for applications, middle layer for data models, lower layer for parallel I/O interfaces, and connections to underlying file systems). **Accomplished**

Prototype optical tape and high-end RAIT systems – PathForward-funded efforts in high-capacity, high-speed optical tape, and high-capacity, high-speed RAIT (see Figure 2.3-2) were pursued, to accelerate development of early prototype systems for ASCI evaluation in testbed environments. **Accomplished**

RFI issued for scalable, global, secure file systems (SGS-FS) – Initial RFI issued to investigate industry interest and progress in some of the more critical technologies in file systems for ASCI: globally distributed file systems, scalable access, scalable management, security, and WAN accessibility. **Accomplished**

Access Control Lists (ACLs) in HPSS – Authorization and NTK protection are made available through the use of the distributed computing environment (DCE) ACLs in new releases of HPSS. Multivendor HPSS support in 2000. **Accomplished**

2001 **Throughput and capacity** – 1 petabyte archives, 5 gigabytes/s parallel disk I/O, 160 megabytes/s commercial off-the-shelf (COTS) RAIT devices or prototype optical tape devices or both. (A 1-petabyte tape archive could be read or written in 2 to 3 months at 160 megabytes/s.) **Planned**

COTS RAIT (tape) – COTS RAIT tape systems deployed for ASCI use (assuming a successful PathForward effort). Potential for introducing architectural changes due to reevaluation of traditional tape price/performance ratios. **Planned**

RFP for SGS-FS – A request for proposal (RFP) will be issued to pursue industry interest in some of the more critical technologies in file systems for ASCI: globally distributed file systems, scalable access, scalable management, security, and WAN accessibility. **Accomplished**

Prototype tri-lab federated namespace – Begin work on a prototype, federated namespace for data storage and file systems in support of more efficient ASCI user accessibility. **Planned**

2002 cont.

Throughput and capacity – 5-petabyte archives deployed, 10-gigabytes/s parallel disk I/O demonstrated, 1 gigabyte/s parallel tape I/O demonstrated. (A 5-petabyte tape archive could be read or written in 2 to 3 months at 1 gigabyte/s.) **Planned**

Production tri-lab federated namespace – Complete work on a production, federated namespace for data storage and file systems in support of more efficient ASCI user accessibility. **Planned**

Demonstration of an SGS-FS system on homogeneous clusters – Early demonstration possible on homogeneous platforms mounting the same file system shown; similar to network file system (NFS), each platform would see the same file space. **Hurdle**

COTS optical tape – COTS optical tape systems can be deployed for ASCI use (assuming a successful PathForward effort). This technology holds potential for introducing major archival tape marketing shifts due to reevaluation of traditional tape price/performance ratios. **Hurdle**

2003

Throughput and capacity – 10-petabyte archives deployed, 20 gigabytes/s parallel disk I/O demonstrated, 2 gigabytes/s parallel tape I/O demonstrated. (A 10-petabyte tape archive could be read or written in 2 to 3 months at 2 gigabytes/s.) **Planned**

Distributed Resource Management (DRM) support for storage systems – The data storage and I/O hardware and software services environment will require DRM support. ASCI DRM capabilities must be expanded to provide storage and I/O resource allocation and management capabilities. **Hurdle**

Next-generation HPSS systems (post-Release 5) – A "next-generation" HPSS system must be developed to support new Data Resource Management capabilities, 100-teraOPS distance applications, speculative pre-fetching, and performance enhancements for SANs and OSDs. Hurdle

Demonstration of an SGS-FS system on heterogeneous clusters – Early demonstration possible on multiple heterogeneous platforms mounting the same file system shown. Similar to NFS, each platform would see the same file space. High-end performance goals for FY03–FY04 will be up to tens of gigabytes/s for parallel access to a single file. **Barrier**

2004

Throughput and capacity – 25-petabyte archives deployed, 40-gigabytes/s parallel disk I/O demonstrated, 4-gigabytes/s parallel tape I/O demonstrated. (A 25-petabyte tape archive could be read or written in 2 to 3 months at 4 gigabytes/s.) **Hurdle**

HPSS secure access for Object-based Storage Devices – A "next-generation" HPSS system must be augmented to support secure access for network components such as OSDs. **Hurdle**

Scientific Data Management (SDM) support for storage systems – Data storage and I/O hardware and software services environment will require scientific data management support.

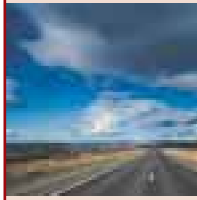
ASCI SDM capabilities must be expanded to provide storage and I/O integration with data discovery and data query tools. **Hurdle**

2005 **Throughput and capacity** – 50+ petabytes archives deployed, 100 gigabytes/s parallel disk I/O demonstrated, 10+ gigabytes/s parallel tape I/O demonstrated. (A 50+ petabyte tape archive could be read or written in 2 to 3 months at 10+ gigabytes/s). **Barrier**

Fine-grained NTK access control – Additional access control mechanisms will be needed for data storage and file systems to meet ASCI fine-grained NTK requirements. **Barrier**

Evaluation of MEMS-based storage systems for ASCI – Miniature micro-electro-mechanical systems that position probe tips over the storage media can potentially create a new generation of nonvolatile rewritable mass storage devices. **Barrier**

Evaluation of holographic storage systems for ASCI – Volume holographic storage that writes data "inside" a storage medium, instead of "on top of" a storage surface also has potential to create a new generation of mass storage devices. **Barrier**



CURRENT STATE OF ASCI I/O

ASCI has worked closely with vendors and collaborators to develop and improve performance and reliability of parallel file systems, such as IBM's General Parallel File System (GPFS), archives such as HPSS, and higher-level I/O libraries (such as MPI-I/O), with excellent results [8,9,10]. Improved implementations of MPI-I/O [11] use the underlying file systems more efficiently and can improve performance for many nonoptimal access patterns. An extensive suite of I/O test programs has also been created to check correctness and performance. Over the past five years, performance and capabilities of HPSS have been significantly enhanced to meet growing needs of ASCI [12]. For example, sustained throughput rates of 2 gigabytes/s on GPFSs and 1 gigabyte/s to the disk cache of an HPSS archival storage system have been achieved. HPSS archives have been built that can hold 500 terabytes to 1 petabyte or more of data. ASCI applications that store 15 terabytes of data in the HPSS archive for each run are now supported. HPSS systems also support a single name space of tens of millions of files.

HPSS authentication services are currently provided by Kerberos. Authorization and NTK protection are made available through DCE Access Control Lists. Because of HPSS's performance, security, and usability features, it is currently being used as a distributed, tri-lab data repository. Through a fast WAN implemented as a part of the ASCI DisCom^2 Program, laboratory users can move data directly between the HPSS systems at each site. This facilitates data sharing and allows users to store data where it can be used most effectively. For example, a user may prepare data for a calculation at one site and move the data to another for the actual computation to take advantage of a particular architecture. The user might then move the data to a third site for specialized visualization, and then return the results to the original site for post-processing.

Through multiyear ASCI PathForward contracts (PathForward is an ASCI program element supporting industrial acceleration of needed high-performance computing technologies), ASCI also hopes to accelerate availability of new tape storage devices. The Storage Technology Corporation (StorageTek) is accelerating the delivery of reliable, parallel tape I/O analogous to disk RAID. This new technology is called Redundant Array of Independent Tape (RAIT) and enables the writing of parity information on a parallel set of tapes. RAIT allows the reconstruction of all the data in the parallel set, if the data on one of the tapes becomes corrupted. When appropriately configured, RAIT guarantees the reconstruction of all data even in case of corruption of the data on multiple tape cartridges. The first prototype RAIT system was tested at Los Alamos in the winter of 2000. It is anticipated that production-quality RAIT systems will be generally available in late 2001.

Another technology promising increased capacity and bandwidth is optical tape. Laser-based optical tapes encode data by "burning" bits onto a tape. New high-capacity, high-speed optical tape drives are under development by LOTS Technology, Inc. Optical bits are not necessarily smaller than magnetic bits (the wavelength of light limits optical bits to about one micrometer). However, advanced optical tape technology using an array of parallel laser beams allows placement of optical data tracks at micrometer distances, something not achievable with bulkier magnetic tape heads. High bandwidth is the result of parallel read/write channels, implemented through an array of parallel laser beams and fast tape speeds. A single tape cartridge will be able to store well over a terabyte of data. Initially, 25 to 40 megabytes/s transfer rates are planned, scaling to 100 to 160 megabytes/s over the next few years. In an optical drive, read/write heads never make contact with the media, so reliability and longevity characteristics should be excellent.

SUMMARY, CONCLUSIONS, RECOMMENDATIONS

In summary, good progress has been made accelerating the performance of data storage and file systems, reaching file system sustained throughput rates of 2 gigabytes/s, and deploying petabyte HPSS archives. Strategic storage-related PathForward contracts have helped accelerate availability of high-performance tape based devices. Still, ASCI desires to work more closely with industry and academia, on additional technologies, including scalable, global file systems, and secure access to object-based storage. ASCI needs to continue to convey to the external community the details of our requirements, assumptions, user scenarios, and methods of operation, and help to test and improve data storage and file systems so that they might better meet ASCI needs.

ASCI is interested in accelerating scalable file system development over the next one to three years, possibly using PathForward-like contracts. An ASCI file system may be characterized as secure, extremely scalable, and able to support multiple supercomputer sites. *ASCI would prefer industrial-strength, end-to-end solutions but is prepared to act as a system integrator in the event no academic prototypes or industrial products are able to address all of our needs in the necessary timeframes.* Some of the more critical technologies in file systems for ASCI are globally distributed file systems, scalable access, scalable administrative management, security, and WAN accessibility. In addition to these, all of the usual requirements of a file system remain in place. For example, POSIX compliance, standard locking mechanisms, persistence, integrity, and stability would be assumed. ASCI will continue to gauge the level of industrial interest in file system areas through ongoing RFI and RFP PathForward processes.

The hardest challenges are yet to come. We still must scale ASCI storage device throughput and capacity by up to three orders of magnitude over what is currently obtainable. We will need fine-grained need-to-know access control and better integration with object-based devices, distributed resource management, and scientific data management tools. In the longer term, we will also need to start evaluating experimental technologies that may become more common at the end of the decade, and their possible place in ASCI storage hierarchies, including volume holographic storage and MEMS-based storage technologies. ASCI will continue to monitor and leverage relevant industry consortia efforts in these areas, including the National Storage Industry Consortium (NSIC), the Storage Networking Industry Association (SNIA), and the Internet Engineering Task Force (IETF) IP Storage (IPS) Working Group [13,14,15].

REFERENCES

1. *Eighth NASA Goddard Conference on Mass Storage Systems and Technologies, in Cooperation with the Seventeenth IEEE Symposium on Mass Storage Systems,* College Park, Maryland, March 27–30, 2000. http://esdis-it.gsfc.nasa.gov/MSST/conf2000/index.html

2. Toigo, J. W., "Avoiding a Data Crunch," *Scientific American,* May 2000. http://www.sciam.com/2000/0500issue/0500toig.html

3. InfiniBand$^{\text{TM}}$ Trade Association, formal InfiniBandTM IBTA 1.0 Specifications are available at http://www.infinibandta.org/

4. Orlov, S. S., "Volume Holographic Storage," in *Communications of the ACM*, Vol. 43, No. 11, November 2000.

5. Carley, L. R., Ganger, G. R., and Nagel, D. F., "MEMS-Based Integrated-Circuit Mass-Storage Systems," in *Communications of the ACM*, Vol. 43, No. 11, November 2000.

6. Gibson, G., and Van Meter, R., "Network Attached Storage," in *Communications of the ACM*, Vol. 43, No. 11, November 2000.

7. May, J. M., *Parallel I/O for High Performance Computing*, Morgan Kaufmann Publishers, 2001.

8. Sturtevant, J., et al., PDS/PIO: *Lightweight Libraries for Collective Parallel I/O*, SAND2000-0326, Sandia National Laboratories, Albuquerque, NM, February 2000.

9. Jones, T., Koniges, A., and Yates, K., "Performance of the IBM General Parallel File System," in *Proceedings: International Parallel and Distributed Processing Symposium*, IEEE Computer Society Press, May 2000.

10. Chen, P., and Ward, L., "Cplant I/O," *Fifth NASA/DOE Joint PC Cluster Computing Conference*, October 1999. http://www.cs.sandia.gov/cplant/presentations/10-8-99/jpc4-5/index.htm

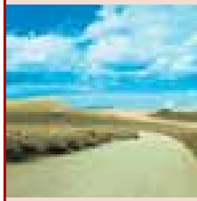
11. Prost, J-P., et al., "Towards a High-Performance Implementation of MPI-I/O on Top of GPFS," in *Proceedings of Europar 2000 Conference.*

12. HPSS Website, http://www.sdsc.edu/hpss

13. NSIC Website, http://www.nsic.org

14. SNIA Website, http://www.snia.org

15. IETF IP Storage Website, http://www.ietf.org/html.charters/ips-charter.html



3. DISTRIBUTED AND DISTANCE COMPUTING

ASCI computers, scientists, and their collaborators are located in the three NNSA defense laboratories and on university campuses. Just as the "Net" and the "Grid" are important parts of mainstream computing today, they are important issues for ASCI project development. Support for terascale hardware systems, huge datasets, and utmost security requires these technology areas to expand to cover the unique problems associated with ASCI simulations. Here we discuss grid services software and issues as well as technology and network communication levels necessary to provide the backbone of support for ASCI computational programs.

Steven L. Humphreys
(lead writer and tri-lab curve owner)
Sandia National Laboratories
Albuquerque, New Mexico
(505) 844-7223

K. Jerry Melendez
(tri-lab curve owner)
Los Alamos National Laboratory
Los Alamos, New Mexico
kjm@lanl.gov
(505) 667-7785

Moe Jette
(tri-lab curve owner)
Lawrence Livermore National Laboratory
Livermore, California
jette1@llnl.gov
(925) 423-4856

Pete Dean
(lead author and tri-lab curve owner)
Sandia National Laboratories
Livermore, California
pwdean@sandia.gov

Steven Tenbrink
(tri-lab curve owner)
Los Alamos National Laboratory
Los Alamos, New Mexico
sct@lanl.gov
(505) 667-4935

Dave Wiltzius
(tri-lab curve owner)
Lawrence Livermore National Laboratory
Livermore, California
wiltzius1@llnl.gov
(925) 422-1551

3.1 GRID SERVICES

With the increasing complexity and diversity of ASCI's high-performance computing, weapons scientists require quick and easy access to these compute resources that are physically distributed. The development of a network of computational grid services with a well-defined set of interfaces to simplify remote access, make usability efficient, provide needed security, and coordinate scheduling of disparate resources is not unique to ASCI. However, ASCI requirements continue to push the limits of security and scalability, while benefiting from continued and new collaborations with academia and industry in the building of a generic computational Grid infrastructure

ASCI's dependence on the most advanced architectures in addition to the size and scale of its data and visualization services mandates ease of access and usage of remote platforms. Even today ASCI scientists and engineers are required to perform increasing numbers and kinds of simulations in a rapidly changing and complex distributed computing environment. In the future, they will depend on even more diverse and geographically distributed compute resources located throughout the Nuclear Weapons Complex (NWC).[3] The mandate of Grid services is to obtain effective tools and common interfaces for distributed compute resources. This mandate includes monitoring tools for viewing the status of work in progress and the available computing environment, coordination of resource use, and DOE Complex-wide scheduling for the effective use of resources.

Without the accelerated development of Grid services, weapons scientists and engineers will be forced to spend much of their time trying to learn the details of how to use a newly added resource, access an application at a remote site, or wait for a local resource to be free. They will not have the time to complete all the necessary simulations and perform all the analysis required for the Stockpile Stewardship Program. This basic problem of how to provide access to a set of diverse and geographically distributed set of compute resources is not a problem unique to ASCI. There are many government agencies, consortiums, universities, and private industries that have distributed high-performance computing environments of similar complexity. Many of these organizations are also trying to increase the accessibility and usability of their compute resources through the application of Grid technologies [1,2]. These organizations will greatly benefit from ASCI's accelerated development of Grid services.

TECHNICAL ISSUES AND CHALLENGES

The concept of a computational grid is analogous to that of an electric power grid that provides easy access to electric power through a well-defined set of interfaces and services. Similarly, the Grid services outlined above will provide the interfaces and services required to easily and effectively access the Nuclear Weapons Complex's compute resources. ASCI's Grid services are expected to interact with users, applications, frameworks, batch queuing systems, and operating systems. This effort involves major challenges of management and performance of the ASCI resources, security services, coordinated scheduling of multiple resources, and application environment support.

Because of the many levels of interaction required, the Grid services will be developed as a set of layered services, with each layer interacting with the layer above and below it. This layered approach is illustrated in Figure 3.1-1. The layers, progressing from the lowest to the highest layer will be the local resource managers, Grid interface, core Grid services, and Grid accessibility services layers, respectively. The lowest layer, the local resources managers (these are the local batch queuing systems in the case of the ASCI teraFLOPS supercomputers), will interact directly with the resources. The top layer, the Grid accessibility services layer, will interact with the user through the use of desktop tools and environments and with applications and other user-level services that are developed to be Grid aware. These and

[3] Compute resources include compute platforms and clusters, storage devices, visualization services, data services, networks, and software resources. The NWC computing environment contains all of these resources and is distributed among the DOE's nuclear weapons laboratories and production sites.

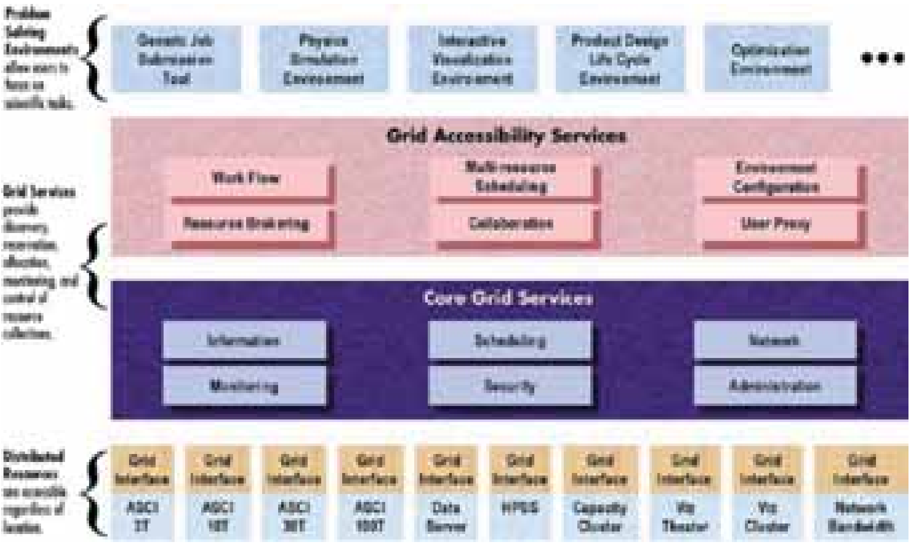

Problem Solving Environments allow users to focus on scientific tasks.
Generic Job Submission Tool
Physics Simulation Environment
Interactive Visualization Environment
Product Design Life Cycle Environment
Optimization Environment
•••
Grid Services provide discovery, reservation, allocation, monitoring and control of resource collections.
Grid Accessibility Services
Work Flow
Resource Brokering
Multi-resource Scheduling
Collaboration
Environment Configuration
User Proxy
Core Grid Services
Information
Monitoring
Scheduling
Security
Network
Administration
Distributed Resources are accessible regardless of location.
Grid Interface
ASCI 3T
Grid Interface
ASCI 10T
Grid Interface
ASCI 30T
Grid Interface
ASCI 100T
Grid Interface
Data Server
Grid Interface
HPSS
Grid Interface
Capacity Cluster
Grid Interface
Viz Theater
Grid Interface
Viz Cluster
Grid Interface
Network Bandwidth

Figure 3.1-1. Grid services architecture overview.

the rest of the layers are described in more detail below. The Grid Services technology road map also outlines the services that need to be provided by each of these layers.

Local Resource Management Layer

The ASCI supercomputing workload encompasses a mix of a few very large jobs running a few hours, a time-sharing model, many moderately sized jobs running hundreds of hours, and a space-sharing model. Accommodating this mix requires pre-empting the long-running jobs to make processors available to the large jobs. To address the problem, Lawrence Livermore and IBM have jointly developed a gang scheduler [3]. While time-sharing is in widespread use for nonparallel computers, gang scheduling extends time-sharing to parallel computers. The gang scheduler ensures that all processes belonging to the same job, though running on different nodes, are context-switched as a unit, or "gang," thus effectively ensuring the co-allocation of a very large fraction of the system to a single, large job.

Achieving the ASCI platform performance goal will require continued research and development of local resource management capabilities. The growing size and complexity of individual ASCI computers continues to stress the scalability and fault tolerance of local resource managers. The accounting systems need modifications to track individual jobs consuming tens of processor years, with hundreds of processor years in the near future. To prevent bottlenecks as the computation and data sets scale up, access to storage, network bandwidth, visualization resources, and capacity computing clusters must be coordinated with scheduling of the ASCI computers. For example, an archival storage system must prioritize requests associated with work being performed by the supercomputers over other requests. Local resource managers must become Grid aware, providing mechanisms for coordinating with other resources and services even across administrative domains. For example, the local scheduler must be able to queue a job request until notified by a data service or another computing platform that a run condition has been satisfied. Coordination mechanisms include advance reservation, prioritized access, level-of-service delivery, and dependencies on external events.

Grid Interface Layer

The Grid interface layer will make the individual resources accessible via common access methods over the distributed network. The Grid resource interface adapts the resource-specific commands to a common Grid access method, and performs several other functions to integrate the resource into the Grid. Since Grid users do not log-in to the resource directly, its services must be made available to remote users and to distributed processes acting on a user's behalf. The interface must provide a mechanism for a Grid user or process to acquire access, use its services, and monitor progress. It must register or publish the resource with the Grid to make the resource known to Grid users. It must provide current information about the resource, such as attributes, capabilities, and status. As resources become Grid aware, the Grid interface functional area will be subsumed in the resource layer.

Core Grid Services Layer

The core Grid services are provided by the Globus tool kit co-developed by Argonne National Laboratory and the University of Southern California [4]. These core services are the focus of the Global Grid Forum efforts and of most Grid technology developers; they have been demonstrated in academic and research environments and are moving rapidly into production environments at many high-performance computing centers. Core services include information, monitoring, remote job submission, remote data access, accounting, and security. Security services authenticate access to resources for basic distributed computing functionality, but many security challenges remain. As Grid technologies continue to move beyond cutting-edge users to the scientific community at large, additional development will be required to enhance features, improve fault tolerance, and optimize performance for these services.

Grid Accessibility Services Layer

This layer will provide a layer of middleware above the core Grid services to support the programming and use of Grid resources and services by higher level applications. With today's research and production grids, building a sophisticated distributed computing application, such as an immersive visualization environment, requires expert knowledge in Grid technology. Grid computing must be easier before it will be widely adopted by users and developers. Ultimately, most users will access resources via point-and-click environments in relative ignorance of the underlying technology. To achieve this, the enabling capabilities of the core services must be built into more complex functionality or software services. ASCI is targeting two main categories of services in the Grid accessibility layer: services that coordinate the use of multiple resource, and common support services that tailor Grid interactions for particular application environments.

Security

In addition to the functionality that each layer must provide, each layer must satisfy the DOE's stringent security requirements. Kerberos Version 5 is the primary authentication method in the nuclear weapons complex. A major focus of the past year has been to partner with the Globus Project to provide a Kerberos-secured Globus infrastructure. ASCI developers adapted the Generalized Security Services Application Programming Interface (GSSAPI) security abstraction layer used by the open Grid and by other technologies into a more object-oriented framework called the Generalized Security Framework (GSF). The GSF is available to ASCI applications and environment developers writing C++ or Java code. Below are major technical issues for Grid security services and frameworks:

- Native authentication services are typically required on a wide variety of desktop computing platforms.
- User credentials may expire before a queued or long running job ends.
- Security credentials are not easily acquired by a distributed component.
- Mechanisms and protocols for handling delegated credentials with restricted privileges are lacking.
- Secure and COTS compatible solutions for Web servers that can acquire and use end-user security credentials are lacking. Our current secure Web server is based on old technology that does not support critical features like servlets.
- Interoperability with evolving commercially available secure systems (e.g., HPSS, DCE, Distributed File System (DFS), GPFS, NTFS, Win2000, Web application servers, document management systems, configuration management systems, databases) must be supported.

- Future compatibility with an evolving Cyber security infrastructure must be anticipated, which may be based on PKI client certificates, Windows 2000 Kerberos servers, CryptoCard authentication requirements, commercial GSSAPI object libraries, and Windows 2000 SSPI object libraries.

Many high-performance computing centers are developing and promoting Grid technologies. Argonne National Laboratory and the University of Southern California are jointly developing the Globus tool kit [4]. Other leading Grid technologies include Legion from the University of Virginia [5], Condor from the University of Wisconsin [6], and NetSolve from the University of Tennessee [7]. NASA, with the Information Power Grid (IPG) project [8], is a leader in the effort to develop these technologies into a robust, production-quality, high-performance computational Grid and data management infrastructure. NASA will enable collaborative, multidisciplinary science and engineering using geographically distributed supercomputing resources. To accomplish this, the IPG project is taking the approach of building domain-specific problem-solving environments using diverse middleware technologies that are layered on top of a common set of Grid services. The DOE2000 Collaboration program [9], the National Science Foundation's Partnership for Advanced Computational Infrastructure [10], and the DoD High Performance Computing Modernization program [11] are all developing Grid technologies for distributed computing and resource management. A German consortium has developed UNICORE for a national network [12]. CERN has adopted Grid technologies for providing their experiment data to scientists worldwide through a distributed hierarchy of sites and repositories [13]. Japan and Europe have regional Grid programs. These efforts have matured enough that the Global Grid Forum [14], an international consortium of academic, government, and commercial institutions, is engaged in defining best practices and interoperability standards for the core technologies that enable computational grids.

ASCI, as well as the entire high-performance computing community, requires coordinated scheduling of multiple resources across hardware platforms, local resource management software, and administrative domains. Three types of resource coordination are critical to the ASCI Grid: dependency-based workflow scheduling, level-of-service access, and concurrent resource scheduling. Achieving the full vision for coordinated resource use may be difficult without vendor-supplied, Grid-aware resources or local scheduling software; however, significant capability can be provided with Grid resource interface software and core Grid services. ASCI will work with the high-performance computing community in the Global Grid Forum to develop resource coordination mechanisms and interoperability standards.

Dependency-based workflow scheduling supports loosely coupled resource use. This is a high priority for ASCI, fundamental to the vision of problem-solving environments where users focus on the specific domain rather than the computing environment. Workflow services can automate some of the time-intensive manual operations characteristic of today's computing environment: overlap data transfer and queue wait time, submit restart requests, view intermediate results, archive data, and post-process output data. Workflow services can also implement contingency actions and other fault tolerant behavior. Flows may have serial, parallel, and conditional tasks. The resource use can be considered separate actions, but connected through scheduling dependencies. A generalized Grid event service is essential to workflow scheduling. Resource interfaces that receive and generate Grid events are also needed.

Level-of-service access provides a guaranteed rate of delivery, allowing higher-level applications to treat multiple resources as if they were concurrently scheduled. Since resource use is not overtly scheduled or coordinated, however, achieving reliable, high-performance level of service across multiple resources depends on the actual workload and resource demand. Level of service access is particularly important in network and storage resources for interactive visualization of distributed datasets. These resources need interfaces or software services that can provide a known rate of delivery, such as priority access, level of service, or resource reservation.

Concurrent resource scheduling is required for applications that transfer large volumes of data with tight coordination between resources. For example, a high-fidelity simulation may generate more data than can be stored on the local storage system, requiring the data to be siphoned off to an archive while the simulation is running. In another example, a single job may need to reserve a computational resource, a visualization resource, and network bandwidth simultaneously. Concurrent resource scheduling requires an infrastructure for advance resource reservation, and potentially complex Grid scheduling algorithms. Workload management is very complex because of fault tolerance, allocation policy, imprecise prediction of execution time, and unpredictable changes in the workload. Unlike the rest of the high-performance computing community, ASCI has little interest in coupling multiple computers to solve a single problem. It is expected that most of ASCI's needs will be met with loosely coupled and level-of-service resource coordination mechanisms.

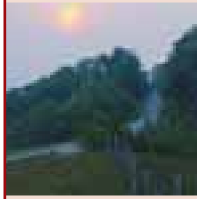
In addition to the challenges of resource coordination and scheduling, configuring Grid interactions for classes of application environments, such as an interactive visualization of distributed data or a distributed computation, is another challenge. Problem-solving environments need to tailor Grid services and views for the specific products, work processes, resource usage, events of interest, and constraints pertinent to the problem domain and to security requirements. These customized environments will increasingly become Web-based, but other delivery mechanisms must be supported. To support these environments, persistent software services will be installed throughout the Grid that, like other resources, will need discovery, information, monitoring, and secure access services. The uses of a Grid may be varied and specific to an application, but some examples of commonly needed support capabilities that will greatly increase Grid accessibility are listed here.

- A Grid access language that allows applications and frameworks to request higher-level middle-ware services such as workflow management, resource brokering, and Grid event notification. Flows and dependencies must be specified. Conditional workflows can implement contingency action plans, such as specifying an alternate action based on file size or timeout. Application environments will both publish and subscribe Grid events. Typical events today include job status, resource status, error conditions, and output available. Additional events could support access to subsets of data, application-level signaling, or distributed control for collaborative environments. Fault detection can trigger recovery operations, such as restarting a job, terminating a workflow, or extracting pertinent information from system logs.

- A workflow management system that performs dependency-based Grid-computing activities on behalf of a user or higher-level application.

- Resource brokering services that can locate and restrict activities to the set of resources that meet criteria specified by the application environment. This includes hiding resources that the user is not authorized for or chooses to ignore, specifying attributes and constraints to determine which resources are suitable for a given task, and specifying a preference strategy like current load or anticipated completion time for selecting the "best" of the available resources.

- Limited delegation of credentials, allowing the problem-solving environment to configure needed levels of security, protection, and privacy depending on the domain, instance of resource access, and workflow of a complex product.



Grid Services Road Map

The associated visual depicts the five-year status (calendar year 2000 to 2005) of desired capabilities/activities within a functional area. Each capability is color coded to show what level of R&D effort it requires or anticipates.

R&D Effort Indicator:
Accomplished = completed
Planned = ASCI will accomplish even with slight budget fluctuations
Hurdle = ASCI will need some help
Barrier = ASCI will need significant help from the high-performance computing (HPC) community

NOTE: Both hurdles and barriers represent research opportunities for the HPC community.

Road Map for
GRID SERVICES CAPABILITIES

Functional Area	CY 2000	CY 2001	CY 2002	CY 2003	CY 2004	CY 2005
Grid Accessibility	Kerberos-secured grid access services	Grid administration Resource coordination via dependencies	Kerberos-secured web access	Grid-aware programming components Personalization of grid environment		Grid scheduling
Core Grid Services		Grid monitoring service	Grid instrumentation	Limited delegation of credentials		
Grid Resource Interface		Grid interface for 10 teraOPS ASCI platform Grid interface for HPSS	Grid interface for 30 teraOPS ASCI platform		Grid interface for 100 teraOPS ASCI platform	
Local Resource Manager		Gang scheduler for 10 teraOPS ASCI platform			On-demand scheduling	

R&D Effort Indicator ● ACCOMPLISHED ● PLANNED ● HURDLE ● BARRIER

ACCOMPLISHED—Completed
PLANNED—ASCI will accomplish even with slight budget fluctuations
HURDLE—ASCI will need some help from the HPC community
BARRIER—ASCI will need significant help from the HPC community

TIMELINE

This timeline elaborates on the activities that appear on the preceding road map.

Calendar Year **Description and Status**

2000 **Kerberos-secured grid access services** – The GSSAPI security abstraction layer used by the open Grid and by other technologies was adapted into a more object-oriented framework, called the Generalized Security Framework (GSF), for Java and C++ applications. **Accomplished**

2001 **Gang scheduler for 10-teraOPS ASCI platform** – Time sharing was extended to the 10-teraOPS ASCI platform through the implementation of a gang scheduler that synchronizes context switching across nodes for all processes belonging to the same job. **Accomplished**

Grid interface for 10-teraOPS ASCI platform – Grid capabilities will be extended to the latest ASCI computer, a 10-teraOPS computer located at Lawrence Livermore and used extensively by ASCI's customers throughout the ASCI complex. **Accomplished**

Grid interface for High-Performance Storage System (HPSS) – A common interface for the HPSS systems at each of the three laboratories will be made accessible via the ASCI Grid. **Planned**

Grid administration – Tools and services are needed to provide system metrics and accounting information, performance measures and fault detection, regression testing, and support for administering the Grid. **Planned**

Grid monitoring service – A general event-based Grid monitoring service is needed, whereby a sensor generates an event upon a defined condition, and a listener registers to be notified of events of interest. **Planned**

Resource coordination via dependencies – The coordinated use of multiple resources via dependencies will be needed to facilitate Grid programming. Local resource managers must support dependencies on external events. **Hurdle**

2002 **Kerberos-secured Web access** – Kerberos-secure Web access is needed so users can authenticate and use the Grid from any HTTPS-compliant Web browser. **Hurdle**

Grid instrumentation – The Grid needs be instrumented for performance metrics and fault detection that can be incorporated into applications and data services. While monitoring of individual sensors is straightforward, it is more complex to make a meaningful interpretation of the set of sensors in context of the Grid. **Hurdle**

2002 cont. **Grid interface for 30-teraOPS ASCI platform** – The ASCI Grid will be extended to the latest ASCI computer, a 30-teraOPS computer located at Los Alamos and used extensively by ASCI's customers. **Planned**

2003 **Limited delegation of credentials** – Limited delegation provides the ability for a user or process to restrict privileges in a delegated credential, and to specify rules for whom, when, and where the credential may be forwarded. **Barrier**

Grid-aware programming components – Grid resources and services will be made available to applications and data services as grid-aware components that have mechanisms to support coordination with other resources, which may be managed by different schedulers or different administrative domains. **Hurdle**

Personalization of Grid environment – Middleware services and application environments will accomplish complex tasks on the user's behalf. User perceptions of fairness and flexibility for obtaining resources when needed will be improved through scheduling and brokering services that support deadline scheduling and user allocation swapping. **Hurdle**

2004 **On-demand scheduling** – Resource allocation and management services that support application-level scheduling, by providing a mechanism to schedule resources to part of the problem on demand, are needed both in Grid services and in local resource managers. **Barrier**

Grid interface for 100 teraOPS ASCI platform – ASCI's latest platform, a 100-teraOPS supercomputer, will be connected to the ASCI Grid. **Planned**

2005 **Grid scheduling** – Adaptive, fault-tolerant scheduling for Grid-aware applications is required. Grid-aware applications will replace the single complex script characteristic of today's scheduling requests with a composition of smaller work units that can be separately scheduled, and must be designed for variable latency, fault tolerance, and hardware failures. **Barrier**

CURRENT STATE OF ASCI GRID SERVICES

A new level of integration was achieved in 2001 with the introduction of the classified ASCI Grid in the nuclear weapons complex. This computational Grid permits authorized users at any of the ASCI sites to easily use the classified compute resources physically located at another ASCI site. Remote access to the 10-teraOPS ASCI computer is a major focus of the initial ASCI Grid services. This is the first platform to be shared on near-equal terms by all of the nuclear weapons laboratories.

The ASCI Grid is founded on Kerberos-based security services and the Globus tool kit. A major focus has been to partner with the Globus Project to provide a Kerberos-secured Globus infrastructure for authenticating and authorizing access to the ASCI resources; actual access control and data protection are provided by the underlying security mechanisms of the resource. ASCI developers have adapted Kerberos authentication services for compatibility with open Grid and Globus technology, including the implementation of user-to-user authentication. The

ASCI Grid information service uses the Netscape LDAP Directory Service. Access is authorized via a plug-in from PADL Software Pty. Ltd. that provides a Simple Authentication and Security Layer (SASL) mechanism. Authorization is provided using LDAP Access Control Instructions. ASCI developers adapted the GSSAPI security abstraction layer into a more object-oriented framework called the Generalized Security Framework (GSF). The GSF is available to ASCI applications and environment developers writing C++ or Java code. ASCI will share these adaptations with the greater Globus community, and will promote their adoption in IETF standards documents and by the developers of open Kerberos source at Massachusetts Institute of Technology (MIT). A diagram of the current ASCI Grid is shown in Figure 3.1-2.

Workflow management and brokering services have been developed to support Grid computing. Workflow management services provide coordinated job submission and file movement between sites, restart loop control, and the ability to define start-finish dependencies between jobs. A resource broker is used to select the "best" resource for any work based upon requisite features and a preference strategy such as anticipated completion time or lowest load. Installed executables are registered as Grid software resources, allowing a user to simply request "run code X."

The ASCI Grid will undergo a trial phase, where selected users will work closely with Grid developers to establish the functionality and reliability of the needed services. An incremental development, testing, and deployment strategy will evolve the initial Grid into the integrated computing environment for the Complex. The initial production Grid services have been submitted for DOE accreditation.

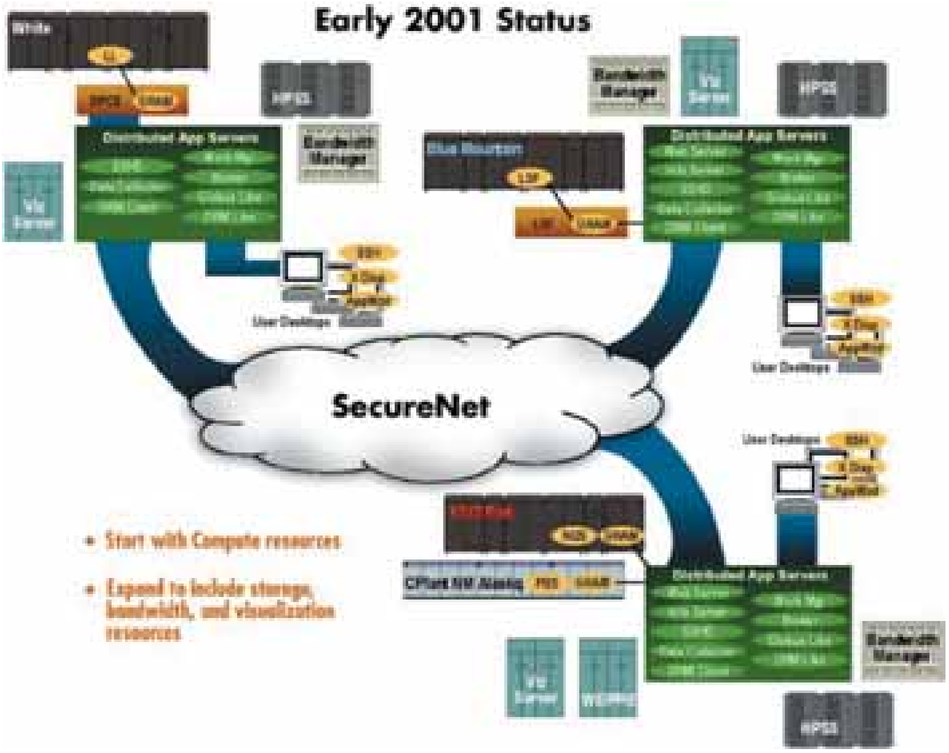


Figure 3.1-2. Grid computing scenario in the future Defense Programs Complex.

SUMMARY, CONCLUSIONS, AND RECOMMENDATIONS

Grid services complement the ASCI vision of terascale computing. The ASCI platforms, focused on the performance of an individual execution, increase the capability of individual computing resources; the ASCI Grid, focused on the aggregate workload from a multitude of simulations to analysis product, increases the speed at which the real work of weapons scientists and engineers can be accomplished. The principal measure of success for Grid technologies is increased user productivity through the increased accessibility of resources and the user's consequent ability to focus on the science at hand.

Security services for a complex-wide integrated computing environment are critical to the ASCI mission. Next steps include a credential agent and Web-based services for secure desktop clients. A generalized credential agent is envisioned as a container-based service that manages and refreshes delegated user credentials and provides these credentials to authenticated and authorized services. The credential agent would support providing a refreshed user credential to a queued job, which would help solve the expiring credentials problem. It also should provide some degree of "limited delegated credential" functionality, and possibly eventually serve to support better interoperability between standard SSL-based services and Kerberos-based services. Kerberos-secured Web services for secure clients are desirable so that users can authenticate and use the Grid from any HTTPS compliant WebTop. In the short term, this will involve developing and deploying a Web server that uses Distributed Computing Environment (DCE)/Kerberos (or CryptoCard) passwords in basic HTTP authentication prompts. Desktop authentication based on native Windows 2000 SSPI libraries is also needed. In the long term, incorporating the credential agent into a Web server is desired. ASCI will continue to work with groups such as the Global Grid Forum [15], the IETF, and with MIT to ensure that the security solutions used by the ASCI Grid are based on open standards, and thus support use of standards-based open Grid technology.

The coordinated scheduling of multiple resources presents several challenges. Dependency and work flow-based resource coordination services can be put in place on top of existing resources, though effective performance will need Grid-aware resources that support dependencies on external events. A workflow engine coupled with dynamic performance monitoring and a generalized event mechanism are needed for fault-tolerant Grid computing. On-demand scheduling is needed to obtain appropriate resources for different parts of the problem to support, for example, the coupling of different models or dynamically resizing the problem upon some condition.

Another difficult problem is concurrent access to multiple resources. Concurrent resource scheduling requires an infrastructure for advanced resource reservation, and complex scheduling algorithms that balance fairness and application performance against resource utilization and workload management. Level of service access for network bandwidth and storage resources is essential for efficient access to large datasets. Adaptive, fault-tolerant scheduling for Grid-aware applications that reallocates resources upon hardware failure or unacceptable performance levels is required in local resource managers and Grid schedulers. Brokering services that provide discovery and negotiation are needed to obtain a suitable resource set for a given application or domain environment.

Grid technologies provide middleware components and services for a complex-wide integrated computing environment. Using Grid middleware, integrated simulation and data services, interactive visualization environments, and collaborative tools can be built. Ultimately, Grid services technology will enable the paradigm shift to network-centric computing necessary to realize end-to-end problem solving environments where users can focus on science.

REFERENCES

1. Foster, I., and Kesselman, C., eds., *The Grid: Blueprint for a New Computing Infrastructure*. San Francisco: Morgan Kaufmann Publishers, 1999. http://www.mkp.com/grids/

2. Foster, I., Kesselman, C., and Tuecke, S., "The Anatomy of the Grid: Enabling Scalable Virtual Organizations," to be published in *International Journal of Supercomputer Applications*, 2001.
http://www.globus.org/research/papers.html - anatomy

3. Yoo, A., et al., "Gang-Scheduling LoadLeveler (GangLL) for IBM RS/6000 SP ASCI Computers," presented at the IBM SP Scientific Computing User Group, SCICOMP. La Jolla, California: August 15, 2000.
http://www.spscicomp.org/2000/presentations/Jette.ppt

4. Foster, I., and Kesselman, C., "The Globus Project: A Status Report," in *Proceedings of the IPPS/SPDP '98 Heterogeneous Computing Workshop*, 2000, 1998, pp. 4–18.
http://www.globus.org/research/papers.html - GlobusHCW98

5. Grimshaw, A., et al., "Legion: An Operating System for Wide-Area Computing," *IEEE Computer*, 32:5, May 1999, pp. 29–37. ftp://ftp.cs.virginia.edu/pub/techreports/CS-99-12.ps.Z

6. Basney, J., and Livny, M., "Deploying a High Throughput Computing Cluster," *High Performance Cluster Computing*, Rajkumar Buyya, ed. , Vol. 1, Ch. 5, Prentice Hall, May 1999.
http://www.cs.wisc.edu/condor/doc/hpcc-chapter.ps

7. Casanova, H., and Dongarra, J., "NetSolve: A Network Enabled Server, Examples and Users," in *Proceedings of the Heterogeneous Computing Workshop*, 1998. http://www.cs.utk.edu/netsolve/papers/examples-apps.ps

8. Johnston, W., Gannon, D., and Nitzberg, B., "Grids as Production Computing Environments: The Engineering Aspects of NASA's Information Power Grid," in *Proceedings of the 8th IEEE Symposium on High-Performance Distributed Computing*, 1999.

9. http://www.emsl.pnl.gov:2080/docs/cpse/workshop/TechPresentations/johnston1.pdf

10. DOE2000 Home Page, http://www-unix.mcs.anl.gov/DOE2000/

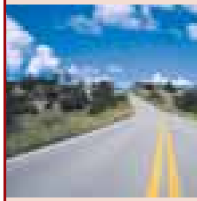
11. National Partnership for Advanced Computational Infrastructure Home Page, http://www.npaci.edu/

12. DoD High Performance Computing Modernization Program Home Page, http://www.hpcmo.hpc.mil/

13. Erwin, D., "UNICORE: Uniform Interface to Computing Resources," presented at the Desktop Access to Remote Resources Workshop, Argonne National Laboratory, October 8 and 9, 1998.
http://www.kfa-juelich.de/zam/RD/coop/unicore/unicore_presentation.ps

14. The DataGrid Project Home Page, http://www.eudatagrid.org/

15. Global Grid Forum Home Page, http://www.eu-datagrid.org/



3.2 NETWORKING

ASCI's overall architecture must enable the timely movement of terascale data files through the entire network (system, local, and wide area) at speeds of hundreds of gigabytes/s. This involves not only greater parallel bandwidth and increased speeds for network interface cards but also a reconsideration of existing end-systems such as computer architectures and I/O devices. In general, a robust end-to-end parallel data transport is essential to ASCI's future success. The vision of this technology area is to provide the end-to-end network capability required to remotely execute large classified simulation codes on teraflop computing platforms and analyze the resulting terascale datasets.

ASCI requires tight integration of the high-performance computing resources among NNSA's three defense laboratories (Los Alamos, Lawrence Livermore, and Sandia). Here, we define the term "tight integration" to mean provides users access to the ASCI platforms from remote sites in a way comparable to that obtained by local users. Such an integrated environment will enable tri-lab users to collaborate and effectively utilize the ASCI resources in an efficient manner. This goal requires a high-speed, encrypted (National Security Agency [NSA] Type1[4]), wide area network (WAN) that interconnects the three laboratories in California and New Mexico. Extrapolations from current applications and usage patterns indicate that the ASCI WAN could be required to achieve throughputs in the range of 100 gigabits/s during 2005. With respect to the bandwidth issue in the WAN, this goal appears to be quite achievable. Currently, dense wave division multiplexing (DWDM) optical technology provides increments of 10 gigabits/s, and within 12 to 18 months increments of 40 gigabits/s will be available

Initial estimates of required network capacity are based on the "hero usage mode," essentially a single application dominating an entire ASCI machine set and stretching capacity to the limit. The hero mode has a heavy peak impact on the required WAN bandwidth. Other usage modes have somewhat different requirements so that flexibility in the logical network architecture is required. The "past to present" usage mode models a user who uses the ASCI resources serially: computational and visualization platforms, the network, storage, etc. Data movement that occurs between computation and the visualization processing tends to scale with the size of the platform and the problem. The "present" usage mode shows data manipulation. Regardless of whether data manipulation is occurring on the computation platform or a computational resource local to the user, large datasets are moving from the platform to the user's site for visualization (output/display) processing. The "present" bandwidth requirement for the network is derived from this usage scenario. Requirements in the "present" usage mode were influenced by the hero mode usage, dominated by serial bulk file transfers.

A "future" usage scenario considers a concurrent usage mode in which multiple users require concurrent use of multiple machines for parallel workflow. In this "future" usage mode, the hero mode is subsumed by an aggregation of multiple file transfers, file/data accesses, visualization streams, control streams, etc., that characterize the "future" usage mode.

Under the "present" scenario, 10 gigabits/s of WAN bandwidth would be required to support the 10-teraOPS computational resource and 100 gigabits/s for the 100 teraOPS. Current estimates from some of the codes show that this bandwidth may not be sufficient to allow timely movement of the larger data sets. For example, a simulation problem was recently run on the Los Alamos platform that produced a restart file of 8.7 terabytes. That file was transferred to the ASCI Red machine at Sandia, restarted, and run again for 196 hours. The resulting 5.3 terabytes of data were then transferred back

[4] Compute resources include compute platforms and clusters, storage devices, visualization services, data services, networks, and software resources. The NWC computing environment contains all of these resources and is distributed among the DOE's nuclear weapons laboratories and production sites.

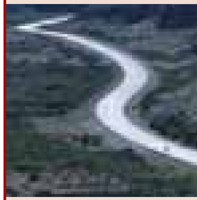
to Los Alamos for post processing, visualization, and archiving. The data transfer time was over 138 hours, almost as long as it took to run the restarted problem.

A simple way to estimate the necessary bandwidth for the WAN is to assume that a given platform has a certain number of nodes dedicated to parallel I/O. For instance, a 4096-node cluster with 128 I/O nodes, each capable of pushing 1 gigabits/s results in a parallel I/O rate of 128 gigabits/s to the WAN. The basic argument here is that WAN speeds should keep up with local area network (LAN) speeds. There is no reason (other than cost) to maintain the current LAN/WAN bandwidth difference.

TECHNICAL ISSUES AND CHALLENGES

In general, the telecommunications industry is undergoing revolutionary growth in bandwidth capacity in order to meet the exponentially growing demand for higher bandwidth Internet services. The development of DWDM optical technology has resulted in demonstrated simultaneous transmission of up to 500 wavelengths of light, each modulated at OC-192 rates (10 gigabits/s), producing an aggregate of 5 terabits/s. Thus, capacity in the WAN is not a technical barrier, but rather one of cost. Costs could go down dramatically within two to three years because DWDM represents a 1000-fold increase in supply (bandwidth). Furthermore, optical technology appears to be increasing at a rate two to three times faster than Moore's law and is expected to sustain that growth rate for at least a decade. However, the ASCI community remains optimistic but cautious about relying upon this assumption.

The end-to-end goals of the Internet service providers differ significantly from those of ASCI networking. The Internet's architecture is based upon many individual small files entering and exiting a high-capacity WAN. In contrast, ASCI's architecture must enable the movement of terascale data files through the entire network (system, local, and wide area) at speeds of hundreds of gigabytes/s. The end-to-end movement of these very large files focuses the technical challenges on achieving high source and sink data rates on the end systems (compute platforms, storage, and visualization) connected to the WAN/LAN/SAN. In the ASCI end-to-end environment, the speeds of network interface cards (NICs) on the various host machines are one governing factor. The network trunk speeds regularly employed by the telecommunication carriers generally exceed the NIC speeds by one to three orders of magnitude. Furthermore, while the network trunk speeds work as advertised, NIC speeds generally achieve only 30 to 50% of the advertised bandwidth, using conventional network protocols (e.g., TCP/IP) and hardware currently available. This issue highlights the necessity for very careful tuning of the network, including the ability to employ jumbo frames (which is currently not supported on critical networking hardware).



The additional constraint that the data be encrypted using NSA-approved Type 1 encryptors means that the throughput of individual networking paths is also constrained by the throughput capability of the fastest Type 1 encryptors. Hence, there is great interest in the ASCI community on striping or parallel transfer mechanisms (software and hardware). The requirement for end-to-end parallel paths particularly influences the logical architecture of the ASCI WAN.

Achieving throughput not only requires deployment of a well-designed and architected network, which provides sufficient aggregate and parallel bandwidth, but it also requires consideration of the end-systems (e.g., computer architecture, I/O devices, etc.). Even after successfully addressing all of those issues, there are challenges in developing the tools (e.g., File Transer Protocol, FTP) that can coordinate I/O access strategies and network programming on a variety of computer platforms to provide effective end-to-end throughput for the users.

Design. The design of the WAN architecture for ASCI has been based upon two major considerations: trading off risk mitigation against cost and parallelism to achieve end-to-end performance. Based upon these criteria, the architecture selected, shown in Figure 3.2-1, lacks complete redundancy and is therefore somewhat susceptible to catastrophic failures (fiber cut, failures of the carrier's electronics, etc.). However, this approach has

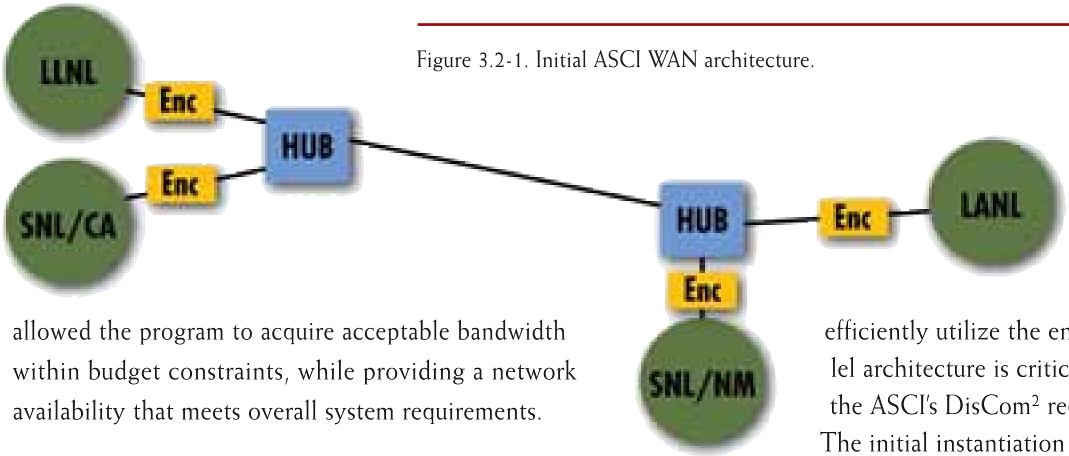


Figure 3.2-1. Initial ASCI WAN architecture.

allowed the program to acquire acceptable bandwidth within budget constraints, while providing a network availability that meets overall system requirements.

Figure 3.2-2 illustrates the current design of the ATM-based WAN architecture whose key attributes are parallelism and an ATM-based infrastructure. The WAN ATM switch allows encrypted traffic from one site to be routed past another site en route to the next with no decryption occurring until the traffic reaches the destination site. This satisfies a primary requirement of the program.

Greater site detail of interfaces is shown in Figure 3.2-3 to illustrate how each of the three sites will connect to the ASCI WAN, and to emphasize the end-to-end parallel nature of the architecture. The ability to deploy and efficiently utilize the end-to-end parallel architecture is critical to fulfilling the ASCI's DisCom^2 requirement.

The initial instantiation of the network consists of four parallel paths, providing a fairly optimal match of the available encryptor rates and NIC speeds. If the computer system hosts are able to fully utilize each stripe at the full encryptor bandwidth, then roughly 200 megabytes/s of WAN bandwidth will be available. Given the current capabilities of our hosts and network limitations, initial utilization of the WAN bandwidth is expected to range between 50 and 75% of the full potential.

There are several challenges. A prime consideration in the ASCI networking development is that the inter-site bandwidth, which could range from 10 to 100 gigabits/s, must be secured for classified operation. Development of encryptors capable of meeting the ASCI networking

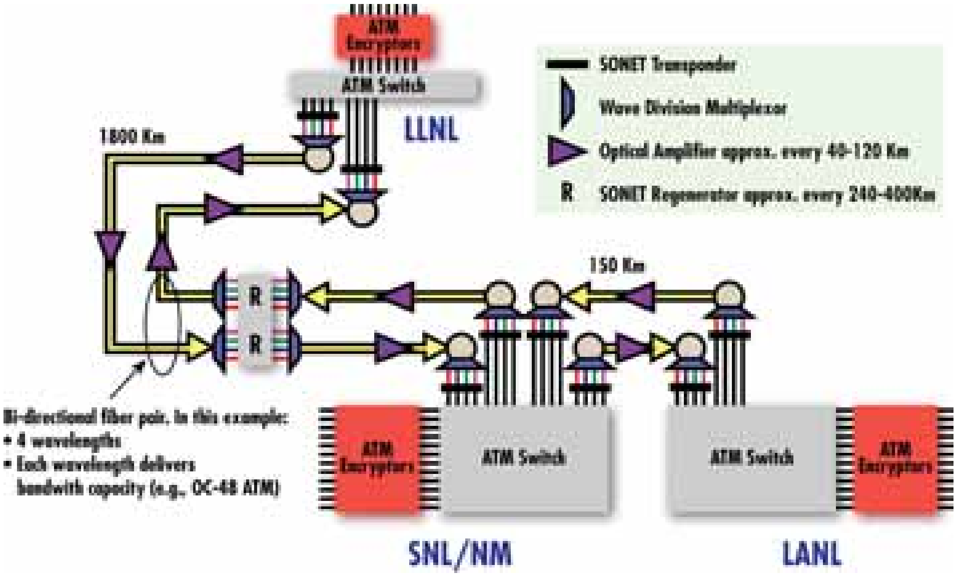

ATM Encryptors
ATM Switch
LLNL
1800 Km
SONET Transponder
Wave Division Multiplexor
Optical Amplifier approx. every 40-120 Km
R SONET Regenerator approx. every 240-400Km
150 Km
R
R
Bi-directional fiber pair. In this example:
• 4 wavelengths
• Each wavelength delivers bandwith capacity (e.g., OC-48 ATM)
ATM Encryptors
ATM Switch
ATM Switch
ATM Encryptors
SNL/NM
LANL

Figure 3.2-2. Details of the ATM-based WAN architecture.

throughput requirements is the sole responsibility of the NSA, and therefore not within the sphere of NNSA's control. To shorten the long development cycle for encryption devices, NNSA has both cultivated a technical collaboration with NSA and contributed significant funding to the development of high-speed Type 1 encryptors. In FY98, work began on NSA Type 1 ATM encryption (UltraFastLANE) capable of operating at a line speed of 10 gigabits/s (OC-192). At that time, development efforts ended with an OC-12 ATM encryptor. Funding was established to develop an OC-48 ATM encryptor by 2001 and an OC-192 ATM encryptor by 2003. This development track committed the LAN/WAN edge network components at each site to interface to an ATM WAN at speeds of OC-12 in 2000, OC-48 in 2001, and OC-192 in 2003. The challenge presented by this constraint is that the network industry does not appear to have an aggressive development and deployment schedule for ATM services. Correspondingly, it remains unclear as to which network services the telecommunication carriers will embrace and aggressively deploy and price in the next few years. The two most likely choices are ATM and Internet Protocol (IP) (over SONET or a new framing format over a wavelength). Almost certainly the ASCI WAN will continue to require an IP service in the foreseeable future.

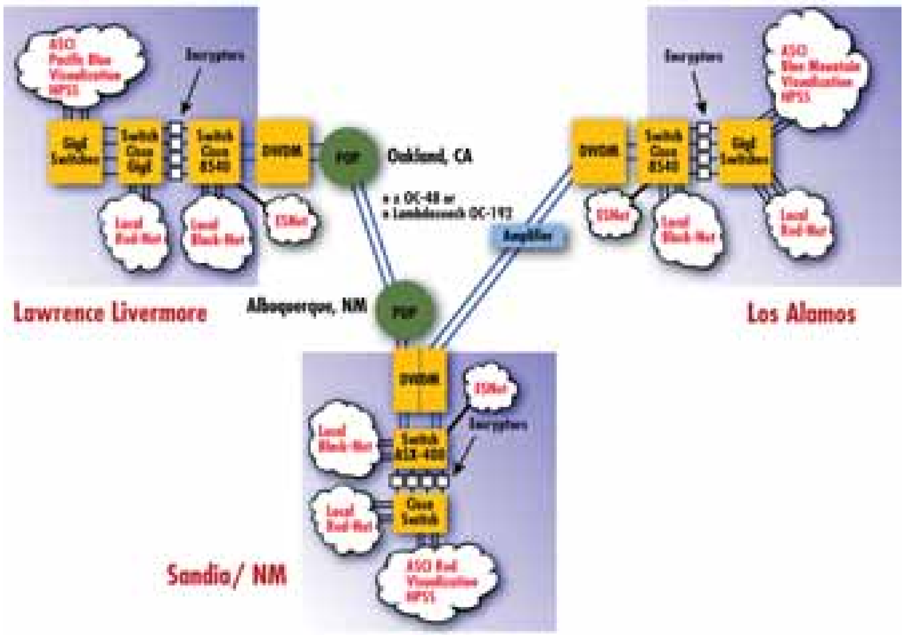


Figure 3.2-3. Site interfaces to the WAN.

A second technical challenge is associated with the I/O capability of the various hosts. One of the critical I/O bottlenecks in future computer platforms may be the ability to effectively utilize the high-speed NICs that will soon be available. Processing the standard IP stack requires significant CPU resources. For example, even though gigabit Ethernet NICs have been available for over a year, the best network performance using the standard 1500-byte Ethernet packets is roughly 500 megabits/s and utilizes 100% of the CPU. Unfortunately, the use of standard packets is currently forced upon the ASCI network because of the availability of only ATM encryptors. The existing ATM-compatible hardware does not support jumbo frames. Although processor speeds will continue to improve over the next several years, the rate of increase will not keep up with the improving NIC speeds. Many vendors are claiming that 10 gigabit/s interfaces will be available as early as FY01.

The emergence of the "psuedo-standard" large MTU has the potential of providing a path forward for network performance. Jumbo frames continue to have wide market support. For example, Cisco supports this capability on all of their switch ports (i.e., 10, 100, and 1000 Mb/s Ethernet on their 65XX series. Conversely, the large MTU was specifically voted down in the 10 gigabit Ethernet standards body. It is imperative that the industry leaders continue to champion this issue in that high-speed networks require extensive interoperability; niche market solutions in networking are not viable. Finally, the fact that jumbo frames are currently being supported on the new gigabit Ethernet equipment but not on the older ATM-compatible models is a strong reason for developing Type1 IP encryptors.

Road Map for Networking

The associated technology road map visually depicts the five-year status (calendar year 2000 to 2005) of desired capabilities/activities within a functional area. Each capability is color coded to show what level of R&D effort it requires or anticipates.

R&D Effort Indicator:

Accomplished = completed

Planned = ASCI will accomplish even with slight budget fluctuations

Hurdle = ASCI will need some help

Barrier = ASCI will need significant help from the high-performance computing (HPC) community.

NOTE: Both hurdles and barriers represent research opportunities for the HPC community.

Road Map for NETWORKING CAPABILITIES

Functional Area	CY 2000	CY 2001	CY 2002	CY 2003	CY 2004	CY 2005
Bandwidth	OC-48 (2.5 Gb/s) parallel network		OC-192 (10 Gb/s) parallel network	Three OC-192 (30 bits) parallel network	Six OC-192 (60 Gb/s) parallel network	Ten OC-192 (100 Gb/s) parallel network
Security	Encryptors operating at OC-12—created a parallel network of four stripes of OC-12	OC-48 ATM encryptor accredited Provide support for development of OC-192 IP encryptor				OC-192 encryptor accredited

R&D Effort Indicator 🔵 ACCOMPLISHED 🟢 PLANNED 🟡 HURDLE 🔴 BARRIER

ACCOMPLISHED—Completed
PLANNED—ASCI will accomplish even with slight budget fluctuations
HURDLE—ASCI will need some help from the HPC community
BARRIER—ASCI will need significant help from the HPC community

TIMELINE

This timeline elaborates on the road map on the previous page. The networking projection is characterized by bandwidth and encryptor capability. As stated previously, the original networking bandwidth requirements were derived from scenarios that predicted that one gigabytes/s of bandwidth would be required to support each teraOPS of computational resource. Given the delivery schedule of the ASCI platforms, this requirement translated into the following networking throughput/year:

- 3 gigabit/s – 1999
- 10 gigabit/s – 2000/2001
- 30 gigabit/s – 2001
- 100 gigabit/s – 2004

As the various elements (computational platforms, storage, and visualization) of the ASCI infrastructure began to actually interact, the limitations associated with the I/O capabilities of the various hosts included in the end-to-end structure raised uncertainties concerning these original requirements. As a result of these uncertainties, the approach taken has been to create a networking architecture based upon striping/parallel transfer mechanisms (software and hardware) capable of scaling to the necessary bandwidths, given that the requirement exists and the funding is available to procure the WAN portion of the total network.

Calendar Year	Description and Status
2000	The development and delivery of high-speed encryptors are important requirements. FASTLANE encryptors operating at OC-12 (622 megabit/s) were required and delivered in mid-2000 to allow the creation of a parallel network composed of four stripes of OC-12, for an aggregate bandwidth of 2.5 gigabit/s. **Accomplished**
2001	The next critical encryptor requirement is for UltraFASTLANE encryptors operating at 2.5 gigabit/s and accredited to handle secret restricted data to be available for testing in mid-2001. **Hurdle**
2002	In 2002, the existence of the OC-48 version of the UltraFastLANE will allow each of the network stripes to increase to OC-48, thereby making it possible to achieve a WAN bandwidth of OC-192 (10 gigabit/s) with four stripes. Current plans are to acquire the 10 gigabit/s WAN bandwidth at the beginning of 2002. **Planned**

Following this 10 gigabit/s planned upgrade, there is considerable uncertainty surrounding the issue of further upgrades to the network. Both budgetary concerns and questions related to the requirements placed upon the network contribute to the lack of a definitive plan. However, the ability of the architecture to accommodate an increased number of 10 gigabit/s network stripes and NSA's planned development of a 10 gigabits/s IP Type 1 encryptor in the 2005 timeframe provide the necessary critical elements for scaling up the network's throughput.

CURRENT STATE OF ASCI NETWORKING

Starting in 1998 and continuing through much of 2000, the WAN has had a bandwidth of OC-3 (155 Mb/s). The encryption is achieved using FastLANE Type1 encryptors, which have a throughput of 155Mb/s. Although well below the target of 3 gigabits/s initially set for the 2000 timeframe, this network has served the ASCI program by providing a means for introducing users to remote computing at a bandwidth that allows real work to occur.

Concurrently, a DoD/NNSA joint project was started to design and build the UltraFastLANE Type 1 ATM encryptor capable of OC-48 to OC-192 throughput. This project is going to continue through 2001, with anticipated delivery of the OC-48 version in August of 2001.

In September 1999, the DisCom^2 network architecture team released the ASCI WAN Architecture document. This design document describes the parallel networking end-to-end architecture anticipated with implementation in the last quarter of 2000 and expected use for the next several years to provide the necessary WAN capability.

In 2000, three significant events took place. First, the OC-12 (622 Mb/s) version of the FastLANE encryptor arrived; second, a project was initiated and completed to establish a multiyear contract for acquiring additional WAN bandwidth among the three laboratory locations; and third, a new encryptor project was planned. The goal was to initiate the new parallel network architecture in the last quarter of 2000 with a network composed of four stripes of OC-12, supported by a OC-48 (2.5 gigabits/s) WAN. The network that will employ the OC-12 versions of the FastLANE ATM encryptors is expected to remain at the OC-48 capability level through 2001. During 2001, the OC-48 version of the UltraFastLANE is scheduled to be available and accredited for use in the classified network.

The new encryptor effort is focused on IP encryption. Currently, industry pressure in the direction of IP over SONET has resulted in an NSA decision to curtail their development and production work on the ATM-based UltraFastLANE encryptor at the OC-48 level and direct their efforts at producing a 10 gigabits/s IP Type 1 encryptor by the year 2005. NNSA is collaborating closely with NSA to develop the IP encryption capability.

The installation of the WAN bandwidth has proceeded with the awarding of two contracts in June 2000 to supply the bandwidth. One contract provides bandwidth from Sandia to Los Alamos and the second contract provides bandwidth from Sandia to Lawrence Livermore. Both contracts require an initial bandwidth of 2.5 gigabit/s at a minimum reliability of 99% for the Los Alamos route and 98% for the Lawrence Livermore route. To provide flexibility, options were provided in each of the contracts to expand the bandwidth up to 100 gigabit/s during the contracts' three-year period. Based on the delivery schedule of new computer capability at each of the laboratories, it is anticipated that the option to increase the bandwidth to 10 gigabit/s in the late FY01 to early FY02 time period will be exercised.

SUMMARY, CONCLUSIONS, AND RECOMMENDATIONS

In general, the forecast for ASCI networking is supported by a solid network architectural design and strong commitments by NSA to provide the necessary encryptors in a timely fashion. As discussed below, consideration has been given in the architectural design to possible changes in the base technology used by the telecommunications industry. In addition, contractual arrangements are in place to obtain the necessary WAN bandwidth.

Future Design Option. The strategy for providing the necessary bandwidth has taken into consideration that the telecommunications carriers continue to deploy high-speed bandwidth services primarily as IP services. In Figure 3.2-4, an ASCI WAN architecture is shown for an IP infrastructure. Note that the IP encryptors replace the ATM encryptors in the ATM architecture, and the WAN edge ATM switch is replaced by an IP router. One advantage of an IP infrastructure is the inherent independence on the underlying IP transport media [e.g., packet over SONET (POS), POW, IP over ATM, etc.].

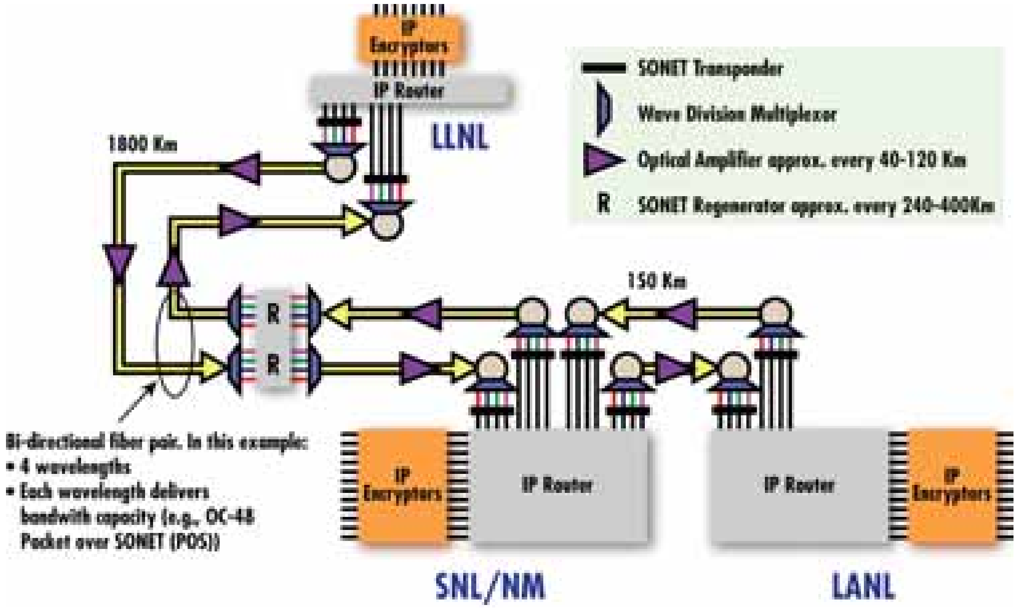

IP Encryptors
IP Router
LLNL
SONET Transponder
Wave Division Multiplexor
Optical Amplifier approx. every 40-120 Km
R SONET Regenerator approx. every 240-400Km
1800 Km
150 Km
R
R
Bi-directional fiber pair. In this example:
• 4 wavelengths
• Each wavelength delivers bandwith capacity (e.g., OC-48 Packet over SONET (POS))
IP Encryptors
IP Router
IP Router
IP Encryptors
SNL/NM
LANL

Figure 3.2-4. Possible IP-based future configuration.

Another advantage is that the telecommunication industry continues to be influenced by the growth of the Internet, encouraging speculation that greater IP bandwidths will be available at lower costs.

WAN Bandwidth. The strategy followed in acquiring WAN bandwidth was to contract for a three-year period for the required bandwidth among the three laboratories. As mentioned earlier, we have two separate contracts for the bandwidth: one for bandwidth between Sandia and Lawrence Livermore and one for bandwidth between Sandia and Los Alamos. The contracting strategy is a consequence of the unique circumstances of acquiring infrastructure between Sandia and Los Alamos within New Mexico. Each of the two contracts was structured with options to acquire WAN bandwidth as a service with options for scaling up both the bandwidth and the availability. The contract also included options for leasing dark or dim fiber. After receiving and analyzing the proposals, it was determined that the least cost for acquiring the bandwidth up to 10 gigabits/s was to purchase services, and for bandwidths above 10 gigabits/s, acquiring leased fiber would be the lowest life-cycle cost. The dark or dim fiber options required significant initial investments. Since there is significant uncertainty in acquiring Type 1 encryptors at the OC-192 (10 gigabits/s) and because the OC-48 (2.5 gigabits/s) encryptors will not be available until late CY2001, we elected (a) to provide the initial bandwidth through acquiring service and (b) to monitor how users structure their applications to utilize the bandwidth. These data will be used to guide the adjustment of the available bandwidth, within available budgets. The parallel network architecture provides for increasing the total bandwidth by either acquiring "bigger pipes" or by increasing the number of pipes. The contracts were structured to accommodate either approach in being able to meet the needs of ASCI users. At the end of the current three-year contract, it is anticipated that the contracts will be re-competed to take advantage of some of the cost reductions that generally occur in the telecommunications industry. We anticipate that the New Mexico infrastructure problems may improve over the next three years, affording us the opportunity to acquire dedicated dark fiber within New Mexico at competitive prices.

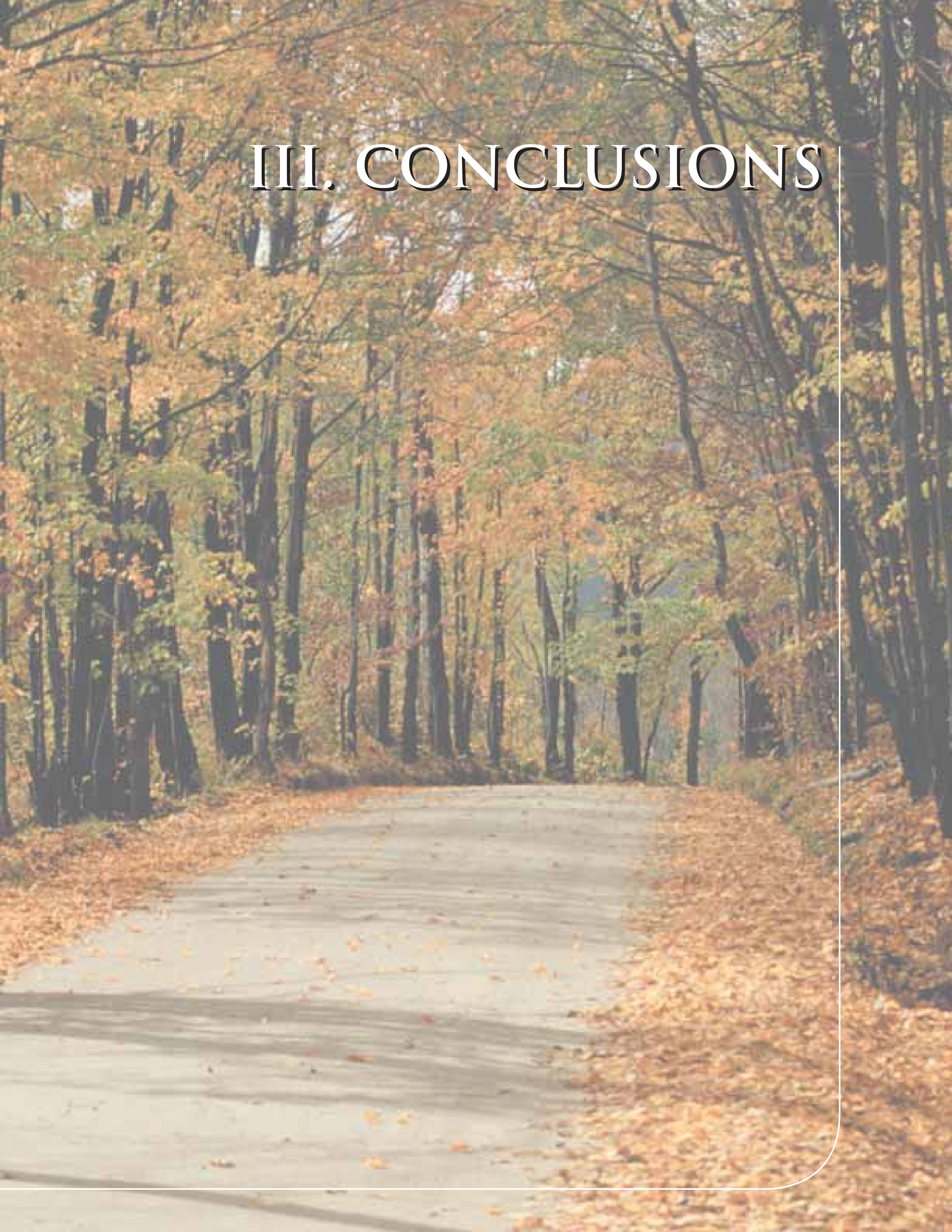
Future recommended work is to be on robust end-to-end parallel data transport. The general consensus is that a parallel SAN/LAN/WAN architecture with tools that effectively use parallel streams is required to meet ASCI's aggregate I/O performance requirements. Already the parallel FTP implementation strategy currently deployed is proving to be very successful when it is required to transfer a few large files.

A goal for future work is to focus on making parallel tools very fault tolerant. For example, presently optimal parallel FTP performance uses 8 to 16 streams distributed evenly and statically over four OC-12 paths. One or more streams may stall due to congestion, encryptor, or network problems. Since the file transfer completes only after all data over all streams are received, stalled streams severely impact end-to-end throughput. Most parallel FTP information is static. Parallel tools will be more robust if acquiring information is more automated, and using and responding to information is dynamic. For example, work could be done on parallel tools to dynamically load balance the data over the streams, thus minimizing the effects of stalled streams. There is also work to be done that could monitor the end-to-end state of the parallel network and machines, and dynamically and automatically set up and tear down streams.

A second goal for future work is to implement an appropriate part of "resource management" that would automatically determine the optimal use of network tools for a given user's application running on a machine in a particular network environment. One simple example pertaining to data movement is to optimize the use of FTP, for example, one or more streams over one or more NICs, multiple FTP sessions on one I/O node, multiple FTP sessions on multiple I/O nodes, or a combination of all of these options.

A third goal for future work is to work with industry to inject new ideas into future products. For example, inefficiencies in the TCP/IP protocol stack exist for high-speed network links that even occasionally drop packets, thereby drastically limiting performance. Research has suggested several potential solutions to this problem. As practitioners on high-speed networks, we could incorporate these in open source (e.g., Linux) operating systems, provide testing and analysis with thorough external peer review in the hope that these ideas will be embraced by industry leaders.

III. CONCLUSIONS



This *Prospectus* outlines both the extent and intent of ASCI's simulation and computer science component. By the year 2005, ASCI will deploy a 100-teraOPS computing environment that will be used as part of a process to certify the U.S. nuclear weapons stockpile. The technology road maps support this goal by outlining a well-defined five-year research and development agenda. ASCI management at the three national laboratories will use these road maps to guide future work in simulation and computer science. They have developed this strategic perspective in conjunction with research experts who participated in reviewing the *Prospectus*. However, the success of ASCI in general and the viability of these road maps, more specifically, will depend on significant advancements in computing technology over the next five years.

These eight road maps — simulation and development environment; scalable solvers; software interoperability; visualization; scientific data management and discovery; data storage and file systems; grid services; and networking – were initially developed as the result of ASCI "curves-and-barriers" workshops beginning in 1997. An early ASCI activity, these workshops were designed to identify the technology capabilities needed and to define a common direction for the three national laboratories.

Each technology road map describes an approach that addresses the "hurdles and barriers" in achieving these technology advancements. Although all are individually described, the question still remains as to which has the most immediate priority. The following list was generated at the 2000 Curves and Barriers Workshop and summarizes the top ASCI barriers, in priority order:

- Data access for visualization
- End-to-end I/O throughput between the platforms, visualization, and storage
- Solvers
- Scalable visualization platform
- Distributed file system deployment
- Tools for feature detection
- High-speed encryption
- Scalable, parallel, visualization data mining algorithms and tools
- Data delivery to offices
- Fast message passing
- Display technologies

Collectively, this list of hurdles and barriers is overwhelming and clearly makes the case that the success of ASCI is dependent on effective partnerships with industry, academia, and other government agencies. ASCI does not have the resources alone to overcome these barriers. We invite you to help us with these technical challenges.

The NNSA Stockpile Stewardship Program (SSP) has resulted in ASCI becoming the de facto leader in the high-performance computing community. However, a terascale computing environment, and the required underlying technologies, are common to many other applications. ASCI has historically collaborated with key industries through its PathForward Program and with universities through its Alliances Program and will continue these efforts. These collaborations have helped ASCI demonstrate success in accelerating the time-to-market for computing technologies required to meet SSP's objectives. This document is intended to be another mechanism for facilitating these mutually beneficial partnerships in high-performance computing.

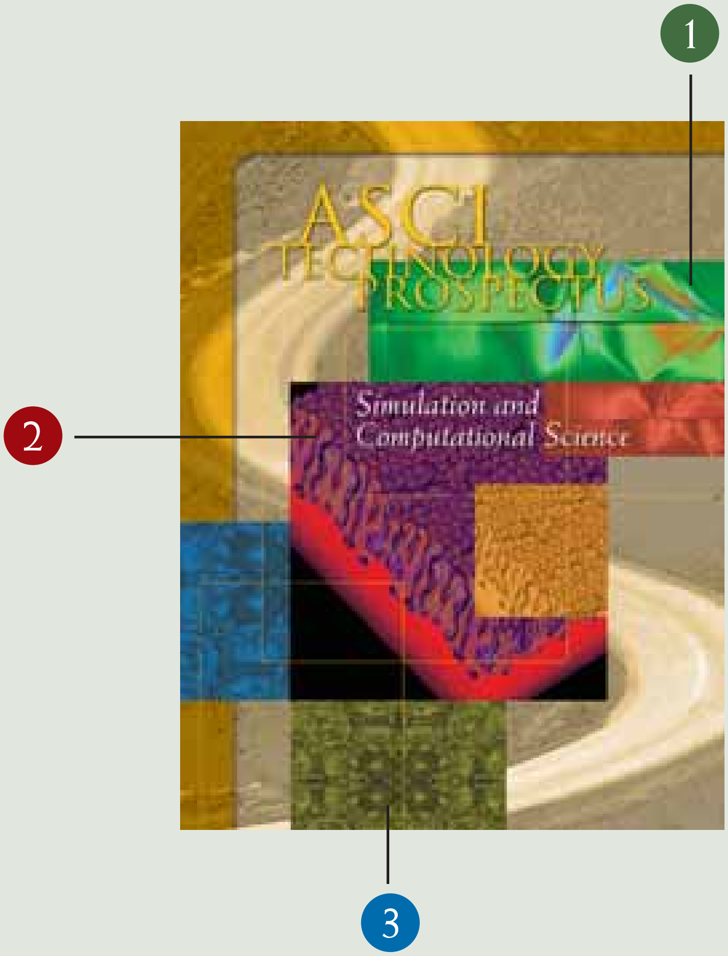
The *Prospectus* therefore reaches out to those working in the field of high-performance computing: computing vendors, universities, and government agencies. We invite your partnership and subsequent help in two important areas: (1) solving the very challenging technical hurdles and barriers described in this document and (2) ensuring that these road maps continue to reflect the projected technology evolution and timescale. Your feedback can be provided either directly to the authors identified at the end of each road map section or through the ASCI website at http://www.asci.doe.gov/

SAND2001-1765P.
Issued by Sandia National Laboratories. Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin company, for the United States Department of Energy under contract DE-AC04-94AL85000.

Notice: This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government, any agency thereof, or any of their contractors or subcontractors. The views and opinions expressed herein do not necessarily state or reflect those of the United States Government, any agency thereof, or any of their contractors.

Available to United States Department of Energy employees and contractors from the Office of Scientific and Technical Information, U.S. Department of Energy, Post Office Box 62, Oak Ridge, TN 37831.
Telephone:423-576-8401. Telefax: 423-5762865.
Internet address; reports@adonis.osti.gov

Available to the public from:
http:/www.asci.doe.gov



On the cover:

1. Recent science runs in December 2000 on the ASCI Frost platform provided a unique opportunity to study material dynamics at the atom level with unprecedented problem sizes. Dr. Farid Abraham from IBM Almaden, in collaboration with LLNL scientists and systems experts, successfully ran computations on Frost involving a billion atoms on 2000-5000 processors. Results include exciting discoveries that cracks can travel at supersonic speed. The image is from a Frost simulation run and shows in unprecedented detail the complexities and structure of the dislocation dynamics — *Lawrence Livermore National Laboratory*

2. Rage simulation of 3-D Rayleigh-Taylor Instability — *Los Alamos National Laboratory*

3. From *Sandia National Laboratories* — a rendering of a 470 million triangle isosurface from LLNL's Gordon Bell dataset.

Lawrence Livermore
National Laboratory

Los Alamos
NATIONAL LABORATORY

Sandia
National
Laboratories

CA.28125.MV.7/01