Navigator

NAVO MSRC Fall 1999

News and information from...
The Naval Oceanographic Office Major Shared Resource Center

The Director's Corner

Steve Adamec,
NAVO MSRC Director

"All of these initiatives support the most important resource within DoD HPCMP—our users."

1999—A Year Of Change

Welcome to the inaugural edition of the *Navigator*, the bulletin of the Naval Oceanographic Office (NAVO) Major Shared Resource Center (MSRC). We intend to publish this bulletin semi-annually and hope that you find it informative and enjoyable.

Fiscal year 1999 has seen significant change and progress for the NAVO MSRC and the Department of Defense (DoD) High Performance Computing (HPC) Modernization Program (HPCMP). All MSRCs are completing the first round of Performance Level 3 (PL3) enhancements to their computational, mass storage, visualization, and networking environments. Here at NAVO MSRC, we are emphasizing resiliency, performance, and security in all aspects of those PL3 upgrades.

Our Defense Research and Engineering Network (DREN) connectivity is being upgraded to OC-12 speed (i.e., 622 megabits/second), creating new opportunities to explore and field improved HPC environments for larger, more dispersed user communities. We have successfully completed full implementation of several DoD-mandated security initiatives, dramatically improving the information security posture for this MSRC and the entire HPCMP. Programming Environment and Training (PET) initiatives, including metacomputing and shared storage efforts among the MSRCs and Distributed Centers (DCs), will soon yield tangible improvements in the HPC capabilities we provide to DoD scientists and engineers. As a result of these 1999 initiatives, users will see a corresponding improvement in our ability to more effectively support a diverse mix of Research and Development (R&D) (both challenge and non-challenge) and time-critical operational processing. Finally, we've completed the rigorous preparations and certifications that we believe are needed to adequately prepare for the much-publicized Year 2000 date problem. All of these initiatives support the most important resource within the DoD HPCMP—our users.

Your feedback and constructive criticism are key ingredients which help us to better serve you. On behalf of the entire NAVO MSRC team, I solicit your continued feedback and support to help us maintain a premiere HPC environment for you and the HPCMP.

The Naval Oceanographic Office (NAVO) Major Shared Resource Center (MSRC): Delivering Science to the Warfighter

The NAVO MSRC provides Department of Defense (DoD) scientists and engineers with high performance computing (HPC) resources, including leading edge computational systems, large-scale data storage and archiving, scientific visualization resources and training, and expertise in specific computational technology areas (CTAs). These CTAs include Computational Fluid Dynamics (CFD), Climate/Weather/Ocean Modeling and Simulation (CWO), Environmental Quality Modeling and Simulation (EQM), Computational Electromagnetics and Acoustics (CEA), and Signal/Image Processing (SIP).

NAVO MSRC
1002 Balch Boulevard
Stennis Space Center, MS 39522
1-800-993-7677 or
help@navo.hpc.mil

Summary of Links Found Inside

ftp://ftp7300.nrlssc.navy.mil/pub/wallcraf/
http://www.tecom.army.mil/hpcw/1998/chamber/chamber.html
http://www.navo.hpc.mil/PET/navigator/thaad
http://heron.ae.gatech.edu/ccl/
http://techpubs.sgi.com/library
http://www.navo.hpc.mil/usersupport/news/ODYSSEY.Queues
http://www.navo.hpc.mil/pet/sip/
http://www.navo.hpc.mil/pet/Metacomputing99

NAVO MSRC Navigator
www.navo.hpc.mil/navigator
NAVO MSRC Navigator is a bi-annual technical publication to inform users of the news, events, people, accomplishments, and activities of the Center. For a free subscription or to make address changes, contact the editor.

EDITOR:
Toni Hampton, toni@navo.hpc.mil
DESIGNERS:
Brian Asbury, brian@navo.hpc.mil
Patti Geistfeld, pgeist@navo.hpc.mil
Lynn Yott, lynn@navo.hpc.mil

Contents

The Director's Corner

2 1999—A Year of Change

Feature Article

4 Benchmarking an Ocean Model

Application Challenges

6 Design and Evaluation of the THAAD Missile
8 Intelligent Combustion

Computer Systems and Support

10 Daily Support of the Warfighter

Scientific Visualization

11 Scientific Visualization Highlights in FY1999

Programming Environment and Training

14 NAVO MSRC PET Program
14 Training in FY2000
15 Metacomputing: The COAMPS-Legion Demonstration Project
16 HotPage: A Multi-Queue Browser

The Porthole

18 A Look Inside NAVO MSRC

User Services

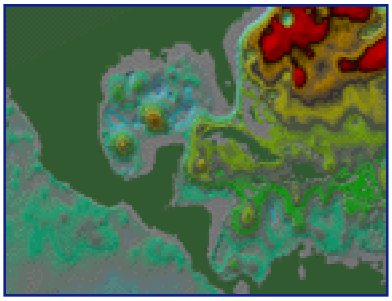
20 User Transition from the C90 to the SV-1
22 Navigator Tools and Tips

Events

23 Recent Events
23 Upcoming Events

Benchmarking an Ocean Model

Alan J. Wallcraft, Naval Research Laboratory (NRL)



PROJECT:
Global and Basin-Scale Ocean Modeling and Prediction

PRINCIPAL INVESTIGATORS:
Alan J. Wallcraft
Harley E. Hurlburt
Robert C. Rhodes
Jay F Shriver
Naval Research Laboratory
Stennis Space Center,
Mississippi

ASSIGNED SITE/SYSTEM:
NAVO MSRC T3E

CTA:
Climate/Weather/Ocean Modeling and Simulation

URL:
ftp://ftp7300.nrlssc.navy.mil/pub/wallcraf/

The NRL Layered Ocean Model (NLOM) has been under continuous development for 20 years [1], [2]. It has been used to model semi-enclosed seas, major ocean basins, and the global ocean. The current implementation of the model uses the tiled data parallel programming style [3]. It is sufficiently general that it can use any one of several parallel programming approaches, including autotasking Fortran, OpenMP Fortran, Co-Array Fortran, the Message Passing Interface (MPI) message passing library, or the shared memory (SHMEM) one-sided communication library. Therefore, NLOM is a good candidate for benchmarking both hardware and associated communication software.

The HALO Benchmark

The HALO benchmark simulates an NLOM 2-D "halo" exchange for a N by N subdomain with N = 2...1024. There are separate versions for each parallel programming technique. These can be used to compare exchange strategies for a given technique, or to intercompare techniques. HALO puts a premium on low latency, but so does NLOM as a whole, and HALO performance correlates well with overall NLOM communication performance. Halo exchanges are important operations whenever domain decomposition is used, but HALO can also be treated as a generic low-level communication benchmark. Small N performance is dominated by latency, and large N by bandwidth.

Figure 1 shows performance for the best HALO implementation of several programming techniques on a range of 16-processor machines. The best MPI implementation is typically persistent ISEND then IRECV, and MPI performance is similar on all scalable systems shown. Note that the "shared nothing" IBM SP does about as well as shared memory systems using MPI. Any one of several one-sided memory methods are always fastest (i.e., have the lowest latency) on ma-

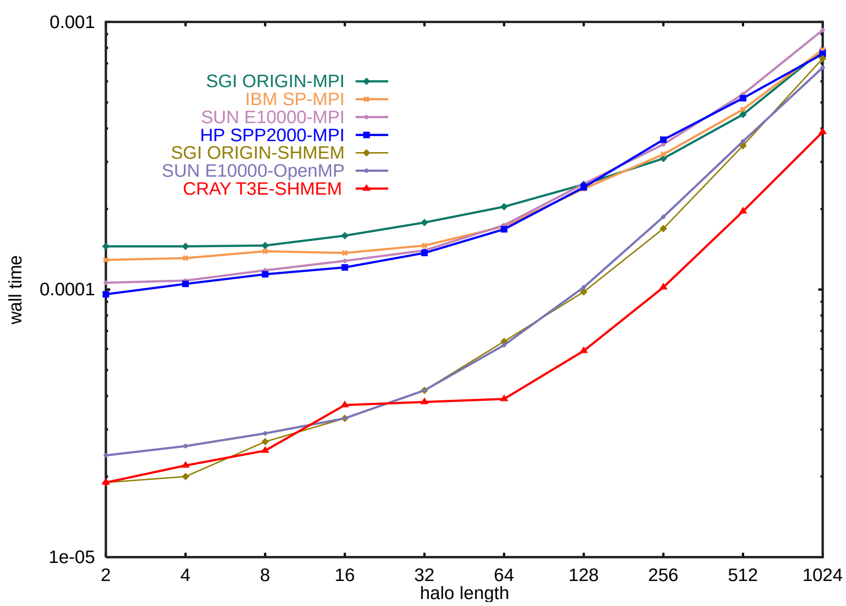


Figure 1. Best HALO Times on 16 Processors

chines with global memory. For large N, MPI and one-sided performance converge, and only the Cray T3E stands out as significantly faster than the other machines.

The NA824 Benchmark

The NA824 benchmark consists of a typical NLOM simulation of 3.05 model days on a 1/32-degree 5-layer Atlantic Subtropical Gyre region (grid size 2048 x 1344 x 5). The run includes all the typical I/O and data sampling, but it does not measure initialization time before the first time step. This is because an actual run would be for one month, ten times the length of the benchmark, so initialization time is not significant in practice. The sustained Mflop/s estimate is based on the number of floating-point operations reported by a hardware trace of a single-processor Origin 2000 run, without the multiply/add operation. Like most heavily used benchmarks, this is for a problem smaller than those now typically run. The NA824 increase in speed from 28 to 56 processors is similar to the increase from 112 to 224 processors for the 1/64-degree Atlantic model, which is four times larger, illustrating that NLOM is indeed a "scalable" code.

Table 1 summarizes the performance results for the best parallel programming approach on several machines. The Cray T3E is showing the best scalability to large numbers of nodes, but the IBM SP is faster per node and competitive on up to 64 processors. The SGI Origin 2000 is showing good scalability up to its maximum node count. OpenMP Fortran (not shown) is similar in performance to SHMEM on the Origin. The excellent scalability of the T3E and Origin is due to the low latency of one-sided (direct to memory) communications, as illustrated by figure 1. The HP/Convex SPP-2000 has a global shared memory but it lacks a one-sided application programming interface, so scalability is limited by the intrinsically high latency of MPI.

Table 1. Performance of NLOM (NA824)

Machine	Method	Nodes	Time	Mflop/s	Speedup
Cray T3E-900	SHMEM	14	44.1 mins	1,064	(450 MHz)
Cray T3E-900	SHMEM	28	21.0 mins	2,236	2.10x 14 nodes
Cray T3E-900	SHMEM	56	10.2 mins	4,591	2.06x 28 nodes
Cray T3E-900	SHMEM	112	5.7 mins	8,184	1.79x 56 nodes
Cray T3E-900	SHMEM	224	3.4 mins	13,601	1.68x 112 nodes
SGI Origin 2000	SHMEM	14	57.6 mins	814	(195 MHz)
SGI Origin 2000	SHMEM	28	28.9 mins	1,625	2.00x 14 nodes
SGI Origin 2000	SHMEM	56	15.6 mins	3,015	1.86x 28 nodes
SGI Origin 2000	SHMEM	112	7.8 mins	6,030	2.00x 56 nodes
HP SPP-2000	MPI	14	56.3 mins	833	(180 MHz)
HP SPP-2000	MPI	28	25.1 mins	1,868	2.24x 14 nodes
HP SPP-2000	MPI	56	15.1 mins	3,107	1.66x 28 nodes
IBM SP	MPI	14	39.2 mins	1,197	(135 NHz)
IBM SP	MPI	28	20.0 mins	2,345	1.96x 14 nodes
IBM SP	MPI	56	11.2 mins	4,169	1.79x 28 nodes
IBM SP	MPI	112	7.7 mins	6,060	1.45x 56 nodes
IBM SP	MPI	224	5.1 mins	9,208	1.51x 112 nodes

Acknowledgements

This is a contribution to the 6.2 Global Ocean Prediction System Modeling Task, sponsored by the Office of Naval Research (ONR) under Program Element 62435N, and to the Common HPC Software Support Initiative (CHSSI) project Scalable Ocean Models with Domain Decomposition and Parallel Model Components, sponsored by the DoD High Performance Computing Modernization Office (HPCMO). The benchmark simulations were performed under CHSSI on an SGI Origin 2000 and Cray T3E at NAVO MSRC; an HP SPP-2000 at NRL, Washington D.C.; and an IBM SP at the U.S. Army Engineer Research and Development Center (ERDC), Vicksburg, Mississippi.

References

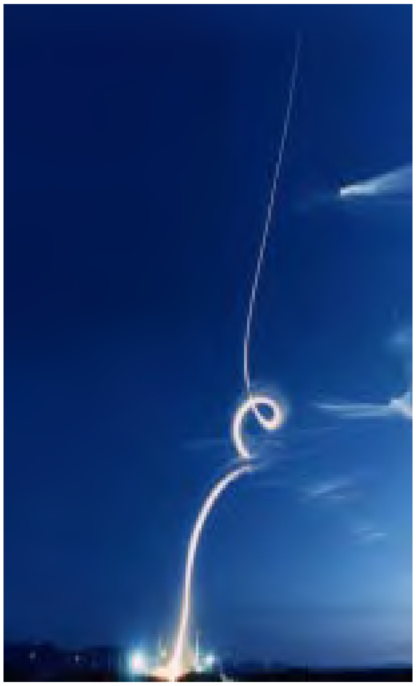
[1] H.E. Hurlburt and J.D. Thompson, "A Numerical Study of Loop Current Intrusions and Eddy Shedding," Journal of Physical Oceanography, 10 (1980), 1611-1651.

[2] A.J. Wallcraft, "The NRL Layered Ocean Model Users Guide," NOARL Report 35, Naval Research Laboratory, Stennis Space Center, MS, 1991, 21.

[3] A.J. Wallcraft and D.R. Moore, "The NRL Layered Ocean Model," Parallel Computing, 23 (1997), 2227-2242.

Design and Evaluation of the THAAD Missile

Donald McClure, PEO Missile Defense



PROJECT:
Analysis of Jet Interaction for the THAAD Interceptor

PRINCIPAL INVESTIGATORS:
Donald McClure,
PEO Missile Defense,
Huntsville, AL

Rex Chamberlain,
Tetra Research Corporation,
Huntsville, AL

Anthony Dang,
ADRC Inc., Huntsville, AL

ASSIGNED SITE/SYSTEM:
NAVO MSRC T90,
SMDC Origin 2000

CTA:
Computational Fluid Dynamics

URL:
http://www.tecom.army.mil/hpcw/1998/chamber/chamber.html

Recent computational success for the U.S. Army's Theater High Altitude Area Defense (THAAD) missile project has been made possible through the extensive use of high performance computing (HPC) assets. These HPC resources provided rapid solutions of complex equations, which generated engineering and aerodynamic data that would otherwise not have been available for the development and checkout of the missile.

This data was utilized for improved design understanding, autopilot tuning and sensitivity studies, ground and flight test data analysis, and improved flight simulations. Analysis of each aspect of reaction jet control contributed to an improved understanding of the missile aerodynamics and performance objectives. A greater confidence in the present design and future requirements has also been achieved.

During the last two years, this Challenge project examined extensive details of jet interaction (JI) phenomena associated with aerodynamic control of the missile. The JI effects of interest to investigators are caused by the aerodynamic interactions between the control jet exhaust and the oncoming flow of air. These effects generally work against the desired action of the control jet and must therefore be included in the autopilot design to achieve robust and reliable flight performance.

The analysis of various aspects of JI control for THAAD helped to increase understanding of JI from CFD simulations, which led to improved predictions of missile performance. An understanding of each of these JI control aspects fed directly into autopilot designing and tuning, flight data reconstruction (e.g., thruster performance derivation), and flight test anomaly investigations.

Background

Performance requirements for rapid and robust responsiveness, both in and out of the sensible atmosphere, compelled the use of reaction control jets as a divert and attitude control system (DACS). This system consists of ten liquid bi-propellent jets, with four located at the center of gravity for divert capability and the remaining six placed at the aft end for pitch, yaw, and roll control. Figure 1 illustrates jet placement on the kill vehicle (KV) and the computational grid used in the studies. Research focused on scaling cold jet wind tunnel data to flight, JI afterburning effects, yaw control, and transient jet phenomena.

Different models were required for the jet exhaust, depending on the level of complexity required in the simulation. For example, comparisons with wind tunnel data required only a cold air jet, while the study of afterburning effects required a finite rate chemistry model for the bi-propellent mixture. Each of these modeling aspects and their implications for missile performance and computational resource requirements were analyzed in this research.

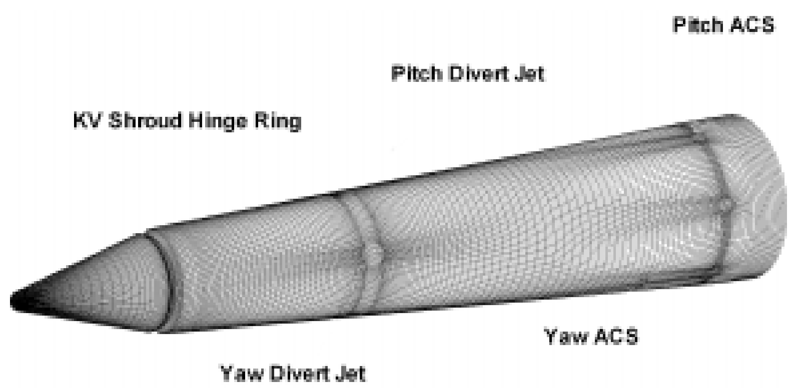


Figure 1. Jet Placement on THAAD

Technical Approach and Resource Requirements

For an acquisition program on a very aggressive schedule, it was not enough to solve the equations on an arbitrarily fine grid, perform some visualizations, and declare the problem solved, even in principle. The autopilot needed extensive force and moment input data to correct for JI control effects in appropriate parts of the battlespace. This in turn required a large number of computational solutions to define aerodynamic trends that could not be measured in ground or flight tests. The technical approach to understanding the fluid dynamics associated with JI was through the numerical solution of the compressible Navier-Stokes equations. It was clear from the beginning that Challenge resources were needed to understand each aspect of the JI control problem in a timely fashion.

The computational techniques required to successfully address JI phenomena must be robust since the numerical aspects of such demanding flow fields often created stability and convergence problems. An additional requirement for a fully coupled finite rate chemistry to study the effects of exhaust afterburning imposed significant computational resource requirements that must be addressed with multiple processors to achieve timely solutions.

The production software used for the most stressing aerodynamic problems was a parallelized version of General Aerodynamic Simulation Program (GASP). Hardware resources included the NAVO MSRC 22-processor T90 and the Space & Missile Defense Commands (SMDC) 128-processor Origin 2000.

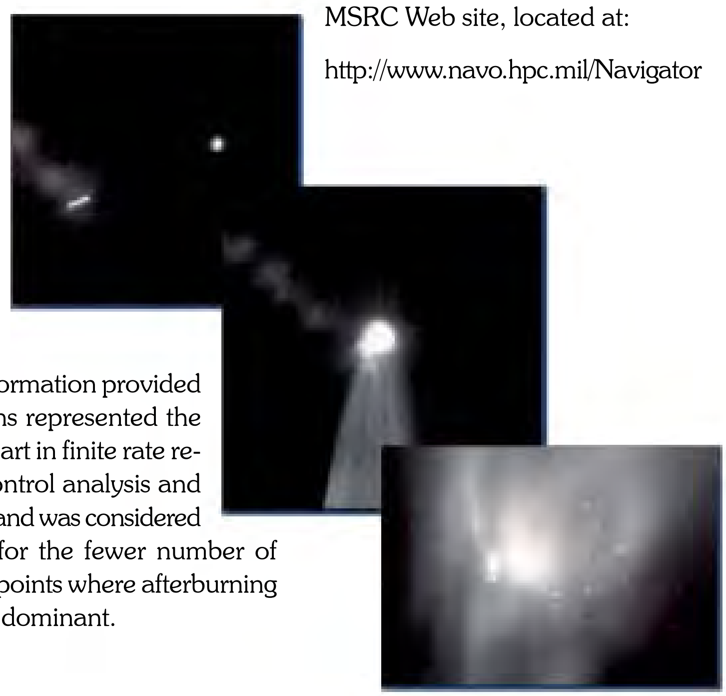
A typical run-time to achieve steady-state, non-reacting multiple (single) jet solution was 120 (60) Central Processing Unit (CPU) hours on a single T90 processor. The turnaround time at NAVO MSRC was 1 to 2 days, using eight processors in the given queue structure. Finite rate chemistry runs required an order of magnitude more computational effort, but those solutions, besides being more numerically sensitive, could still be obtained in 2 to 3 weeks. The new information provided by these runs represented the state-of-the-art in finite rate reaction jet control analysis and technology, and was considered acceptable for the fewer number of battlespace points where afterburning effects were dominant.

Results

Insight into JI control has been greatly enhanced by comparing different physical solutions (e.g., geometric scaling and altitude scaling), by examining trends across Mach number and altitude ranges, and by evaluating the resulting behavior of flight simulations using the predicted JI inputs. As of this writing, more than 150 single and multiple jet analyses have been accomplished using Challenge resources. An increased understanding of JI control behavior, ground and flight testing, and autopilot simulations extends beyond the present missile program to provide a starting point for future work. This includes the systematic development of a detailed, research-oriented experimental JI database, which is needed for CFD code and model validation, and for benchmarking solutions across HPC platforms. This complementary activity will leverage the computational efforts, which have already enhanced the understanding of critical design.

You can find this article, Jet Interaction Phenomena for the THAAD Missile, presented in full on the NAVO MSRC Web site, located at:

http://www.navo.hpc.mil/Navigator



Intercept Success

Intelligent Combustion

Suresh Menon, Army Research Office

PROJECT:
Parallel Simulations of Reacting Turbulent Two-Phase Flows

PRINCIPAL INVESTIGATORS:
Suresh Menon
Won-Wook Kim
Sreekanth Pannala
William Henry
Army Research Office, Research Triangle Park, NC

ASSIGNED SITE/SYSTEM:
NAVO MSRC T3E,
SMDC Origin 2000

CTA:
Computational Fluid Dynamics

URL:
http://heron.ae.gatech.edu/ccl/

Under a multiyear Army Research Office-funded Multidisciplinary University Research Initiative (MURI), a combined experimental-numerical study is underway at the Georgia Institute of Technology, to develop an "intelligent" gas turbine combustor for application to the Army's next-generation helicopters and battle tanks.

A key element of this research is the development of an actively controlled fuel injector that can be tuned to perform optimally over a wide operating margin. Since the flow field near fuel injectors is very complex, detailed experimental characterization of the underlying processes is difficult. Therefore, new numerical techniques are being developed to investigate the turbulence-combustion process in liquid-gas flows near such injectors.

Desirable features for the next generation of gas turbine engines are combustion efficiency, reduced emissions, and stable combustion in the lean limit. Improvements in the liquid fuel atomization and fuel-air mixing downstream of the fuel injector have been identified as two major design goals to reduce emission and to increase combustion efficiency. Achieving these goals is very difficult due to many conflicting constraints.

Background

Since fuel atomization and fuel-air mixing are highly unsteady processes, conventional steady-state methods cannot be used to reveal the finer details. The unsteady mixing process can be studied quite accurately using direct numerical simulation (DNS). Application of DNS is limited to low and moderate Reynolds numbers, typically on the order of 1,000 due to resolution requirements.

Researchers are investigating an alternative approach using large-eddy simulation (LES) being developed for high Reynolds number flows (on the order of 10,000, and more). In conventional LES, scales larger than the grid resolution are computed using a time and space accurate scheme, while the unresolved small scales are simulated using a subgrid model. For momentum and energy transport closure, a new localized dynamic single-equation model for the subgrid kinetic energy has been developed that allows accurate simulations using relatively coarse grid resolution, when compared to classical algebraic subgrid eddy viscosity models. An eddy viscosity model is reasonable for momentum closure since the small scales primarily provide dissipation for the energy transferred from large scales. However, a similar eddy diffusivity subgrid closure is not applicable for scalar mixing and combustion, since species react only after they are molecularly mixed, which occurs at small, unresolved scales.

Technical Approach and Resource Requirements

Recently, a new methodology has been developed at the Georgia Institute of Technology to simulate turbulent single and two-phase mixing and combustion processes in small scales. A unique feature of this method is the manner in which small-scale turbulent mixing and combustion is simulated within the subgrid, rather than modeling the effects of the subgrid processes on the LES-resolved processes.

The new approach can be envisioned as a simulation within a simu-

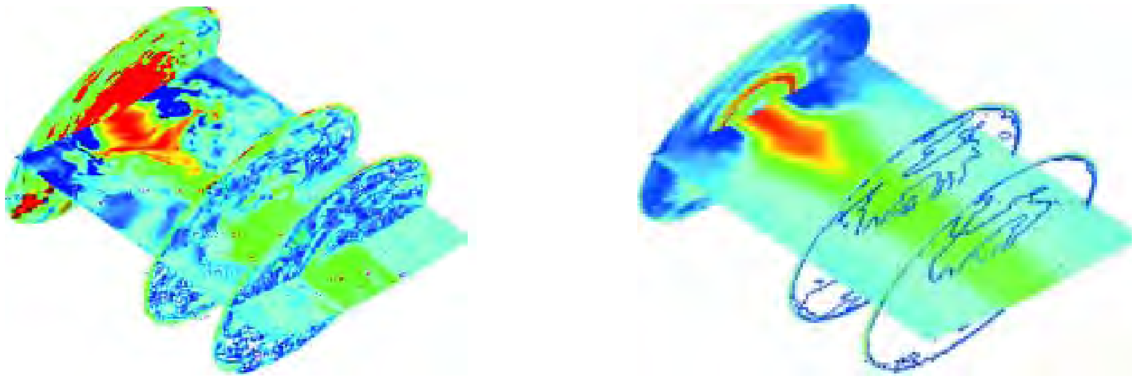


Figure 1. Comparison of an instantaneous and time-averaged result from the LES of swirling flow in a combustor. Circular planes show the swirling vorticity field, and the spanwise plane shows the fuel species density. As shown by the image on the right, many fine details of the mixing process are lost when the flow field is time-averaged. Analysis of mixing effects requires using the unsteady flow fields.

lation. Within each LES cell, a stochastic simulation of local small-scale processes is carried out in conjuction with the standard LES simulation on the larger scales. This feature allows for proper and accurate characterization of both large-scale convection and small-scale mixing on the droplet transport, vaporization, and scalar mixing processes, providing a more accurate prediction of chemical reaction effects.

This new LES method is uniquely suited to massively parallel systems. A typical run-time can be completed within 10 hours on 120 Cray T3E processors. The turnaround time at NAVO MSRC was 2 weeks for a full simulation requiring 14 runs in the given queue structure.

It is estimated that the new MIPS 12000 processors will take approximately 2 to 3 hours, using the same number of processors. At this rate, a full simulation would take two days, assuming fifty percent availability of the system. With this faster computational capability, 3-D LES of realistic flows in full-scale combustors will become practical enough to use for limited design studies. However, this would be feasible only for relatively coarse grid LES (i.e., less than two million grid points). In that case, advanced subgrid models such as those developed at the Georgia Institute of Technology must be used to ensure the accuracy of the predictions.

Results

The results for gas phase and premixed and non-premixed combustion have shown excellent agreement with experimental data. Such agreement was not possible using the conventional LES model. For conventional two-phased LES, the model was implemented within the framework of a Eulerian-Lagrangian Stochastic Separated Flow (SSF) model, which is well-suited for spray modeling since it allows for quantitative predictions. In that approach, droplets are tracked in a Lagrangian manner within the Eulerian gas-flow field.

Resource constraints are a key limitation of the conventional LES model approach. A limited range of droplet sizes is tracked. Droplets below a pre-specified cutoff size are assumed to vaporize instantaneously and become fully mixed in the gas phase. This approach has been demonstrated to be flawed, unless a very small droplet cutoff is used—in which case, the computational cost is excessive.

The new methodology developed at the Georgia Institute of Technology takes into account the effect of all droplets below the cutoff, thereby allowing the use of a larger cutoff size.

There is a significant computational advantage of the new approach, since increasing the cutoff size by a factor of four decreases the computational cost by a factor of approximately four, in spite of the increase in computational effort due to the new subgrid model. The full 3-D LESs of two-phase spatially evolving sprays performed to date have been the first of their kind reported. In addition, the numerical model developed is the only one of its kind that addresses fundamental issues in turbulent combustion such as flame stability, extinction/ignition, and pollutant formation. These issues are of considerable interest to many industries and DoD agencies. Furthermore, the LES methodology is generic in nature and can be used to study a wide range of problems that are currently unresolved, including mixing/chemistry in aircraft jet plumes, combustion in internal combustion engines, effects of buoyancy-induced turbulence generation in reacting flows, physics of flame extinction/ignition and combustion instability, flame spread and flame structure, etc.

Daily Support of the Warfighter

Dave Cole, NAVO MSRC Computer Systems and Support

The NAVO MSRC operates as a DoD shared HPC center serving over 2000 users nationwide. Support is also provided to users under the Commander, Naval Meteorology and Oceanography Command 15 percent Operational Entitlement. This relationship provides the NAVO MSRC with a unique opportunity to employ high performance computing to address the warfighter needs by providing computational resources to accelerate development of numerical atmospheric and ocean models for transition to operational use.

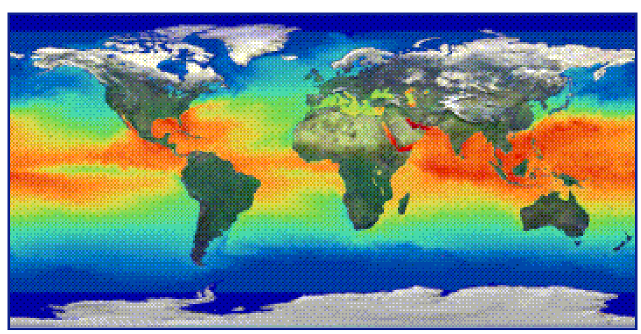
Within the DoD HPCMP, separate resources are funded and made available to support user real-time and near-real-time requirements at DoD HPC DCs. The few exceptions to this policy have all been granted in response to high-level DoD requests for MSRC direct support of time-critical events. However, near-real-time support is generally provided on NAVO MSRC systems for models that are in the process of being transitioned from the research and development community. Typically, this occurs during the Operational Evaluation (OPEVAL) phase, where the model is integrated into the operational runstream and is no longer in the development phase. This policy also minimizes interference with time-critical NAVO MSRC operational models and DoD HPC Challenge projects.

The NRL Ocean Dynamics and Prediction Branch, NRL 7320, performs basic and applied research in computer modeling of ocean hydro/thermodynamics modeling of ice dynamics, computational numerical techniques, data assimilation, and the analysis of satellite oceanographic data, as related to the development of modeling and data assimilation capabilities. The branch then translates the results of basic and applied research into accurate, scientifically valid, environmental models and analyses.

The Naval Oceanographic Office (NAVOCEANO) Warfighting Support Center (WSC), with assistance from NRL, adapts these models into their operational runstreams for use in providing nowcast and forecast products to satisfy Fleet requirements.

During April and May 1999, NAVO MSRC resources were utilized to support NRL 7320 and the NAVOCEANO WSC Operational Test (OPTEST) of the Modular Ocean Data Assimilation System Version 2.1 (MODAS2.1). This system is a modular, relocatable temperature and salinity nowcast system developed by NRL 7320 that uses optimal estimation theory to combine all available satellite and in-situ data into a three-dimensional best-estimate nowcasts. The system output is used to provide thermal structure input for Naval warfare and acoustic applications. During the OPTEST, altimetry and multichannel sea surface temperature data were received and processed in near-real time on a NAVO MSRC Origin 2000 to create 1/8-degree resolution global fields of sea surface height and sea surface temperature data. These two-dimensional data sets were then provided as inputs to MODAS2.1 to produce and deliver three-dimensional temperature and salinity fields for designated geographic regions of interest to the Navy.

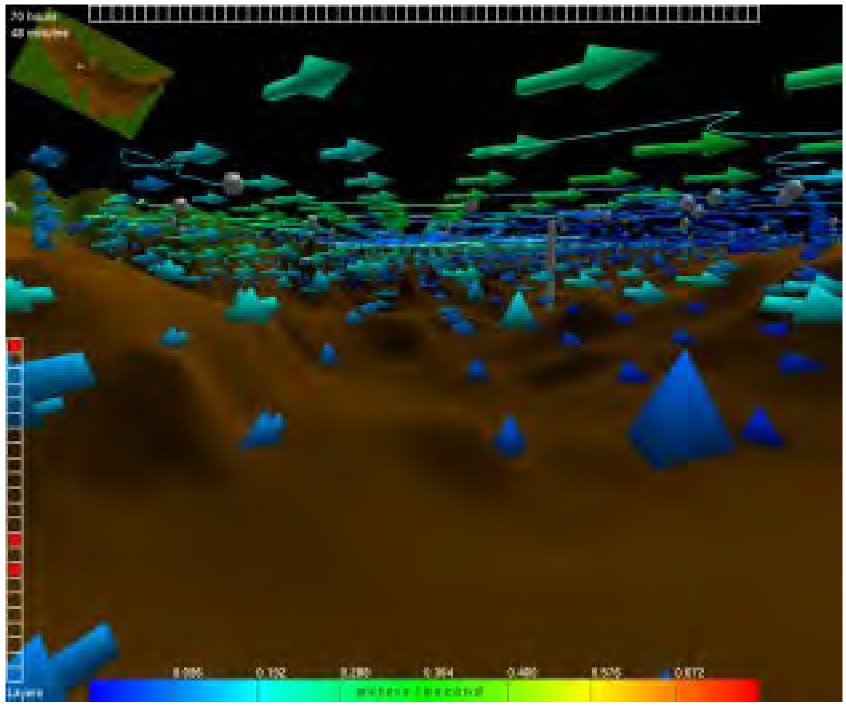
The use of NAVO MSRC systems in the OPTEST of MODAS2.1 established a milestone in the use of scalable systems to provide computational support for numerical ocean models transitioned from NRL to the NAVOCEANO WSC. This effort clearly demonstrates NAVO MSRC's commitment and capability to provide near-real-time world-class high performance computing products to address the warfighter needs.

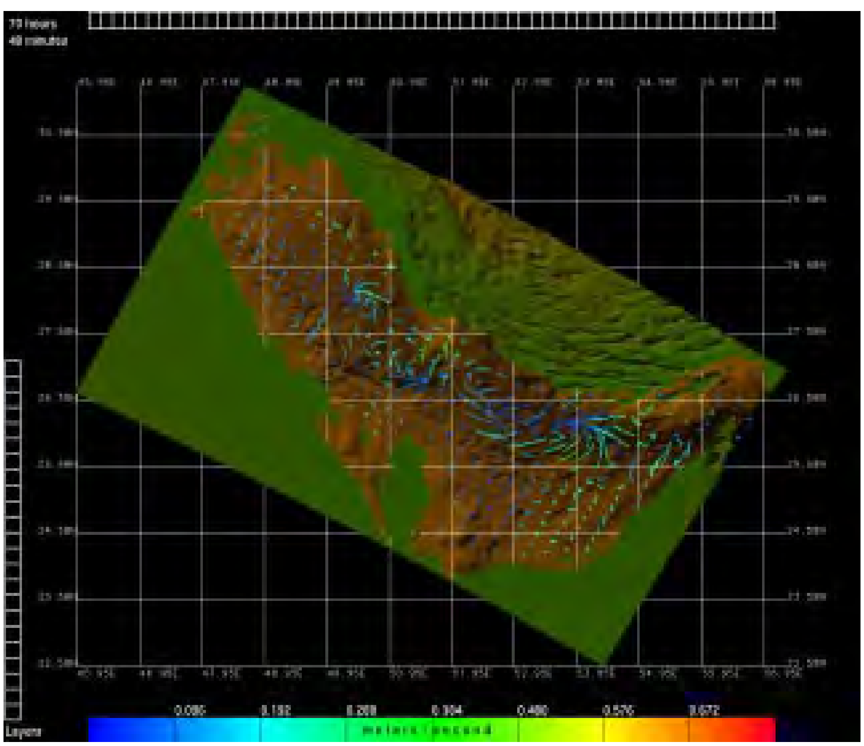


This is an example of a two-dimensional multichannel sea surface temperature and sea surface height product developed for the daily operational needs of the warfighter.

Scientific Visualization Highlights in FY1999

An integral part of the NAVO MSRC scientific visualization support program is the outreach activities we conduct. Since the majority of our users are remotely located, we must engage the users in their environment. These outreach activities can be comprised of telephone conversations, e-mail, or site visits. The Scientific Visualization Staff works closely with the User Services Staff to accommodate any data visualization requirements that originate through them. The Visualization Staff has been very successful providing support to Challenge users from various services involved in several CTAs. Some of our success stories include the support provided to researchers at Louisiana State University and their computational chemistry project designed to provide improved high-temperature materials to the Air Force. We provided innovative support to a Navy CFD project, which studied ship hydrodynamics. The Visualization Staff also provided a 3-D interactive computer environment for an ocean modeler at the University of Miami. Most recently, we are supporting the Defense Threat Reduction Agency (DTRA) on the West Coast to visualize blast/dispersal phenomena. The task of helping remote users visualize large data sets is one of the most challenging and rewarding aspects of working with the Visualization Staff here at the Center.



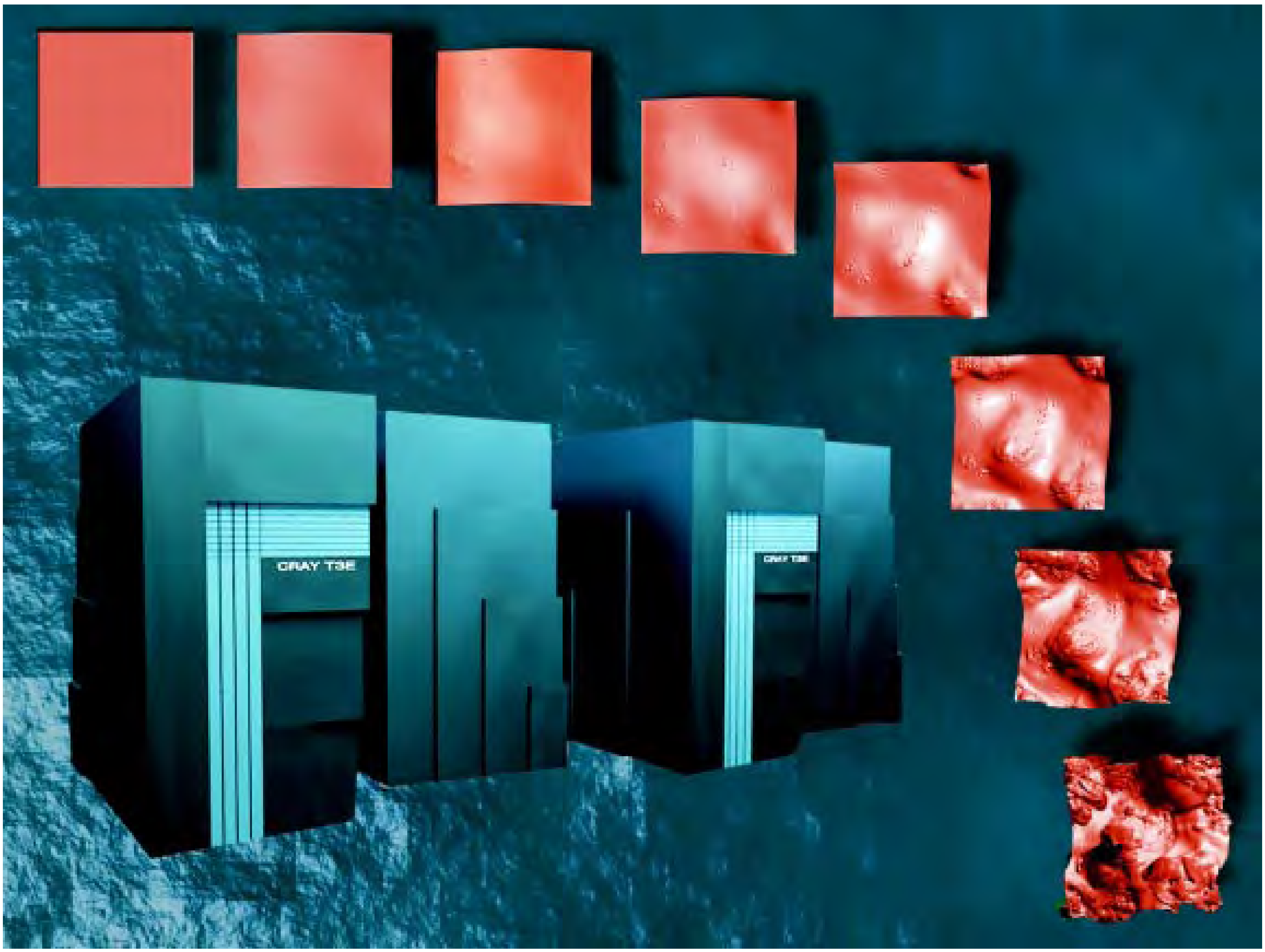


These images illustrate a Shallow Water Analysis Forecast System (SWAFS) model output of sea currents inside the Persian Gulf. The view position is marked on the map in the upper left corner. Arrows show the direction and speed of ocean currents at several layers of depth, and the white particles reconstruct the path of these currents in a two and a half day simulation. The image on the bottom is a charting view, which illustrates relevant current action and trends. These images were developed using SeaFlow, a numerical ocean current modeling application developed by Andy Haas at NAVO MSRC during 1999. SeaFlow is used by the Modeling and Techniques Division at NAVOCEANO to visualize SWAFS data used in nowcast and forecast products to satisfy Fleet requirements.

Scientific Visualization Highlights in FY1999

Parallel Processing of Time Series Scalar Data Using Marching Cubes

This rendering illustrates a sheet of water breaking up over a 2-millisecond time period. Instead of gathering and processing each data step sequentially, the Message Passing Toolkit uses the Marching Cube algorithm to distribute data over many processors, thus reducing computational time. This implementation was written in C++ for the SGI and Cray computers by Scientific Visualization Staff member, Ludwig Goon, and student intern, Sean Ziegeler, during 1999. The computations for this rendering were performed on the NAVO MSRC Cray T3E.

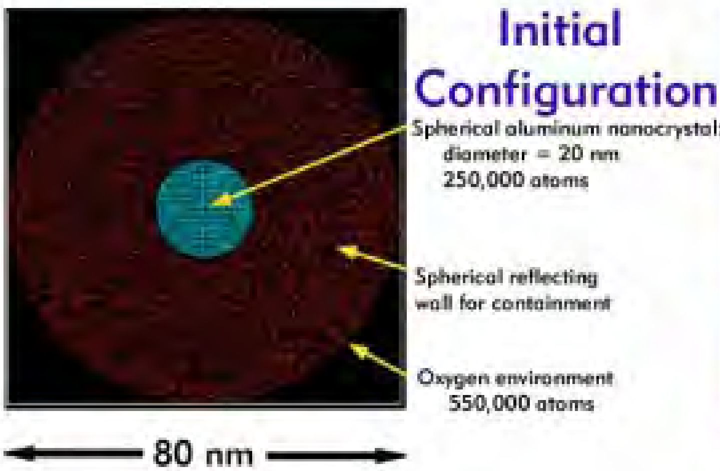


DOD Challenge

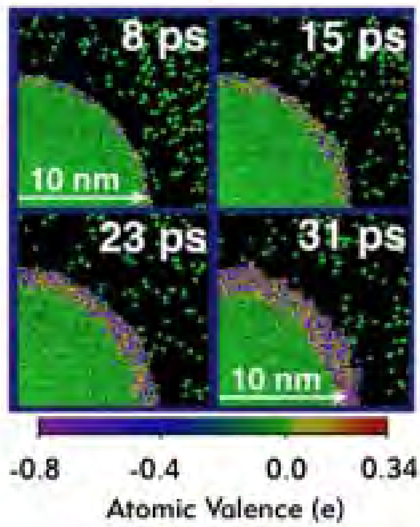
Computational Assisted Development of High Temperature Structural Materials

CTA: Computational Chemistry and Materials Science

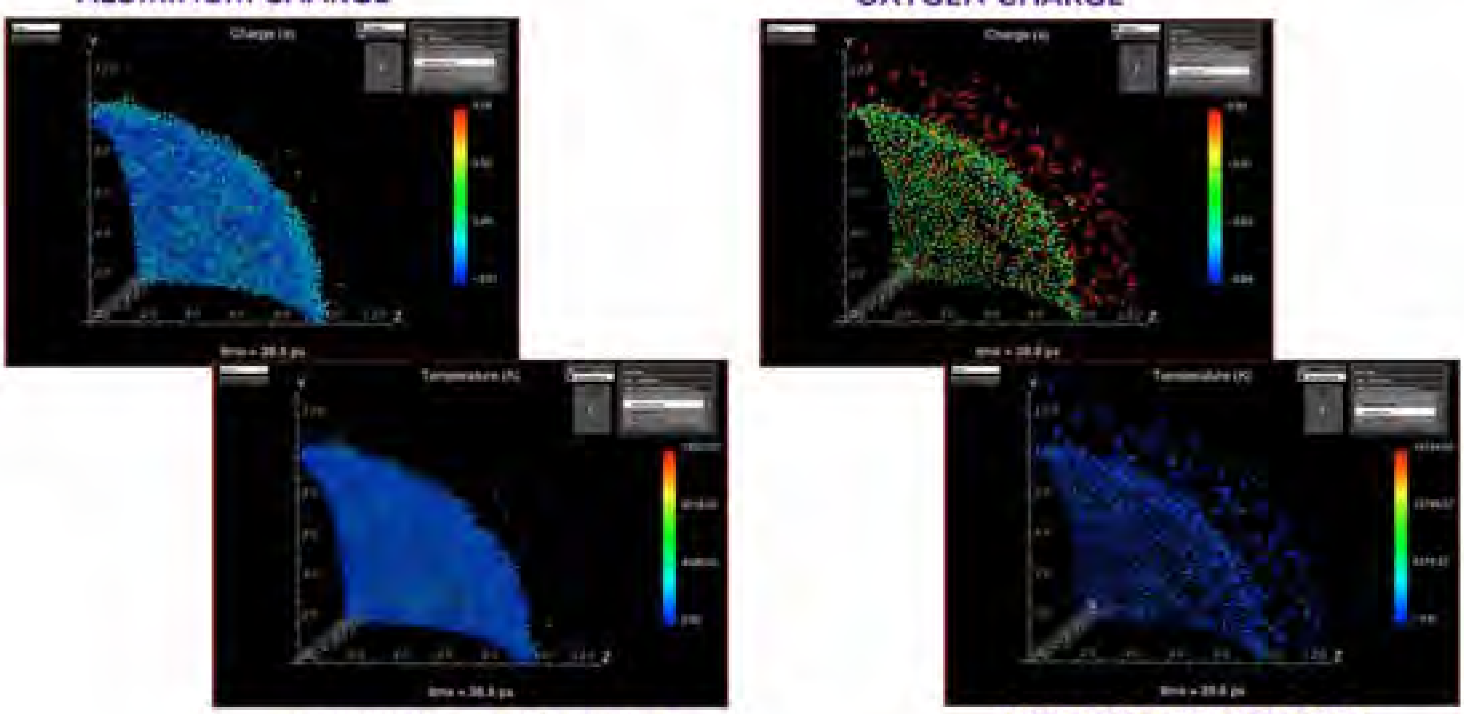
OBJECTIVE: Using state-of-the art computational methods on the Cray T3E-900, the scientists on this project seek to identify and understand basic technical factors controlling and limiting the performance of high temperature structural materials and to discover windows of opportunity for improving existing materials, particularly at higher service temperatures.



Dynamics of Oxidation



- No temperature control applied
- Oxide thickness increases linearly with time
- Nanocluster heats up and eventually fragments explosively



ALUMINUM CHARGE

OXYGEN CHARGE

ALUMINUM TEMPERATURE

OXYGEN TEMPERATURE

NAVO MSRC PET Program

Eleanor Schroeder, NAVO MSRC PET Government Lead

October 1, 1999 marked a significant milestone—the beginning of Year 4 for the NAVO MSRC PET Program. After much discussion, it was decided to align our contract year with the fiscal year. Of course, this created challenges for us in closing out our third year. Working as a team with our academic partners, we met the challenge with grace and creativity!

Our third year was filled with many changes. Walter Shackelford, formerly of the Environmental Protection Agency, joined the NAVO MSRC team as the Chief Technologist and Integrator Lead in January 1999. Also in January, John Pormann joined the team as our new Technology Evaluation Lead. We had to refocus many of our projects due to the advent of Kerberos, delaying the operational deployment of some projects.

Eleanor Schroeder, NAVOCEANO MSRC PET Government Lead

Year 4 strategic planning and programming was begun shortly before Walter Shackelford arrived, but no major decisions were made until he was in place. The leads for those CTAs that are associated with the NAVO MSRC were invited to participate in the review of the academic proposals that we received. Their comments definitely assisted us in defining the scope of our Year 4 plans. Many, many thanks go to them for their time reviewing these proposals and for their valuable insight. For more information: http://www.navo.msrc/pet/pet.html

Training in FY2000

In order to provide structure and detail to the PET Training Program, a mixture of workshops and classroom training is being proposed for FY2000. In addition, San Diego Supercomputing Center, one of our academic partners, will provide training with the Parallel Computing Workshop. These workshops will be conducted in each quarter of FY2000, as will classroom training. Workshops will focus on discussions and hands-on examples. Classroom training will focus on software installed at MSRCs, that has immediate utility to users and will involve some portion of the local user base. Other classroom training opportunities may arise during the year, such as TANGO courses and special need courses requested by users. In addition, the series of video courses that was begun in FY1999 is slated for completion in FY2000.

Bob Melnik is responsible for workshop coordination and arrangements. Brian Tabor and Andrew Schatzle are responsible for classroom training and video courses.

Workshops

We are planning various types of workshops and classroom training for FY2000. Each session will last approximately 2.5 days each.

A Visualization Workshop (1st Quarter FY2000), coordinated with the NAVO MSRC Visualization Laboratory, will include participation from Robert Moorhead of Mississippi State University, an academic partner. This workshop will focus on helping scientists get more out of their data.

A CWO/CFD Workshop (2nd Quarter FY2000) will be coordinated with George Heburn, CWO CTA Lead and conducted at NAVO MSRC.

A SIP/CEA Workshop (3rd Quarter FY2000) will be coordinated with Richard Linderman, SIP CTA Lead, and Bob Peterkin, CEA CTA Lead. We hope to make NAVO MSRC the home for this annual workshop.

Creating Scalable Code for Structured and Unstructured Grids (4th Quarter FY2000) will be conducted with NASA Langley. The location for this workshop will be determined at a later date.

Classroom Training

Classroom training will be held primarily at NAVO MSRC and the dates will depend on student/instructor availability. Classroom training will be limited to one day, unless there is sufficient justification and student availability for longer times. Plans in FY2000 include:

- MPI Tips and Tricks (1st Quarter FY2000)
- MPI and OpenMPI for Scalable Architectures (2nd Quarter FY2000)
- Tracing and Debugging Tools (3rd Quarter FY2000)

For more information visit the PET training web site:

http://www.navo.hpc.msrc/cgi-bin/pet/Training/training.cgi

Metacomputing: The COAMPS - Legion Demonstration Project

Eleanor Schroeder, NAVO MSRC, PET Government Lead

The Navy is currently using the Coupled Ocean/Atmospheric Mesoscale Prediction System (COAMPS) as their regional atmospheric forecast model. This model has many attractive features, such as local refinement, full three-dimensional equations, and a robust physics package. Furthermore, this model was developed to integrate with various ocean models through a common interface. Similarly, the Naval Coastal Ocean Model (NCOM) is slated to become the Navy's regional ocean forecasting tool. This model was also developed to integrate directly into COAMPS.

A scalable version of the NCOM model is now available, providing much quicker turnaround times for ocean simulations. However, a scalable version for the atmospheric portion of COAMPS is still some time off. Currently, the two models can be run either as a single code on a single processor machine, which best suits the atmospheric portion of the program, or the models can be coupled through input files and a series of restarts. Neither of these solutions represents the ideal situation, which is to have both programs running in lock-step in their native environment, with information being shared as needed.

Matthew Bettencourt of the Center of Higher Learning, one of our academic partners, has proposed that an executable library be developed to handle requests for boundary information from both the NCOM and COAMPS programs. This library would satisfy the following technical goals:

1. Produce identical results as either of the current methods.

2. Allow numerical models to run on different hardware simultaneously.

3. Utilize the Legion software environment.

4. Enable portability across many hardware platforms provided at NAVO MSRC.

This library will be executed by both the COAMPS and NCOM codes, as required. Furthermore, this library will have the ability to share data between all executables referencing it. This will be the mechanism that facilitates the sharing of surface boundary information.

As stated in the technical goals, the use of Legion is a critical part of the program. Legion is an object-based metasystems software application that links together hosts and objects through high-speed networks. Users working on their home machines have the illusion of working on a single computer, with access to all kinds of data and physical resources, such as digital libraries, physical simulations, cameras, linear accelerators, and video streams. Groups of users can construct shared virtual work spaces, to collaborate research and exchange information. Legion supports this abstraction with transparent scheduling, data management, fault tolerance, site autonomy, and a wide range of security options.

Legion provides an extension to the FORTRAN language through its Basic FORTRAN Support (BFS). A key feature provided by BFS is the simple interface for Remote Procedure Calls (RPC). This allows the developer to create a library program. This library executes its code on a remote machine returning the results through a predefined interface. BFS enhances the functionality further by allowing the library to store permanent data between successive calls. In the current project, this library will be used to buffer, interpolate, and return surface boundary information for both the ocean and atmospheric codes.

This implementation will require the modification of both NCOM and COAMPS code. The surface boundary condition routines will need to be made "Legion aware." This will require the addition of hooks to reference the newly created boundary condition library. A second new routine will need to be added to both prediction codes to send their surface information to the boundary condition library. This will be in lieu of the previous file I/O interface.

The major effort will focus on development of the boundary condition library. The role of this library will be to store and distribute boundary condition information to multiple applications, upon request. Initially, the code will be developed specifically for the COAMPS and NCOM models. However, it will be coded to allow for future extension to other codes. Five basic routines will represent the shell of the boundary condition interface library: initialize, store, precompute, retrieve, and finalize. There is no reference in the specification to NCOM or COAMPS and therefore, the library should be portable to other applications.

Work began on the development of this library in mid-1999. A simple non-Legion version utilizing sockets is running on the Origin 2000 system at the NAVO MSRC. The idea behind this implementation is being incorporated into the Legion version, which is also underway. There are two programs that can share information through a simple library. This will be expanded to the more general interface defined above.

Hotpage: A Multi-Queue Browser

Ken Ferschwelier, NACSE/Oregon State University
Mary Thomas, San Diego Supercomputer Center

The DoD's system of MSRCs and DCs provides users of high performance computing resources a variety of machines on which they may choose to execute their programs. Access to high-speed networks reduces the importance of geographic location, further expanding the feasible set of target machines. However, few tools exist to help users choose the most efficient option for execution of a particular job.

Turnaround time is dependent on a variety of run-time conditions, such as availability of resources (e.g., memory, particular hardware characteristics), machine load, and current job queue length. Complicating matters further, users must learn a different command set for displaying information from different platforms and queuing systems. The Queue Browser is a web-based tool which provides a unified view of resources and queue characteristics for a selected set of computer systems. Each user can personalize the information so that only appropriate systems and queues are shown for each application.

This tool is intended to reduce the time and effort needed to acquire current information on queue status, as well as the need for users to learn a new set of commands for retrieving machine characteristics for each platform. This feature is particularly important where the user must consider a large number of systems and architectures. The features and user interface design of the browser were developed after extensive discussions with users, who clearly preferred simplicity over an excessive features list—they wanted the simplest tool which would provide "one-stop shopping" for systems and queue information.

An example Queue Browser display is shown in Figure 1. Systems in the display may be of various architectures, located at various sites, and use various queuing systems. Since users are known to have preferences for particular systems, the machines in the display are arranged in a user-specified order, so that the user sees his or her "favorite" systems first. The display provides links to pages describing the system characteristics, if the host site provides such pages. Current system status and any system messages are also displayed; this information, again, is provided by the machine administrators.

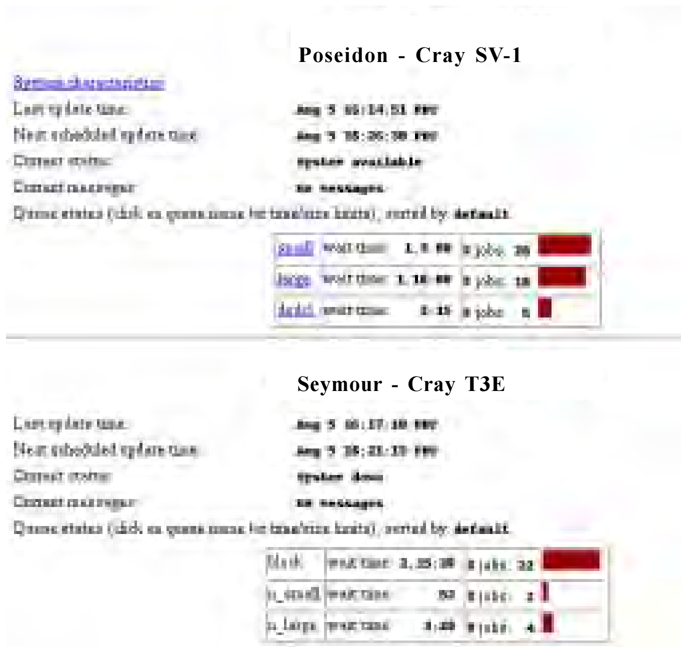


Figure 1. Example Queue Display

The most obvious feature of the display is the queues table. This is also personalizable by the user, so that he or she sees only the queues of interest. Each queue name in the table is a link to a description of the policy for use of that queue (if available). The maximum wait time for the queue is shown, if applicable; this is simply the sum of the time limits of all jobs in the queue and is not shown for queues which don't have time limits. The number of jobs already in the queue is shown graphically; users emphasized number of jobs as the most important piece of information for choosing where to run their jobs. The number of jobs is highlighted by showing it as a bar of length proportional to the number of jobs. The user may also choose to have queues sorted by number of jobs or time. The displayed page shows the time at which the queue information was last updated and the time at which the next update is expected; the server pushes a new page when queue information is updated so the user won't be misled by an out-of-date display of the queue status. The page gives an at-a-glance view of the machine status and queue states of the machines in which the user is interested, simplifying the choice of where to submit a particular job for execution.

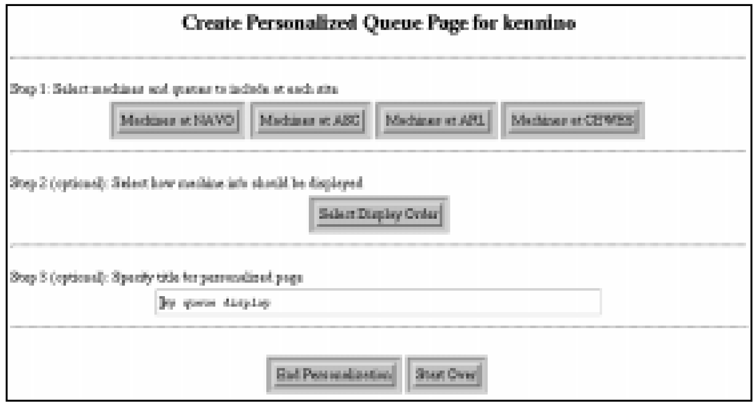


Figure 2. Example Personalization Page

Queue Page Personalization

As mentioned, the user may personalize some aspects of the queue browser display: which systems and queues are listed and the order in which they are displayed. Use of the browser would be cumbersome if the user had to specify, for instance, the target systems each time invoked, so per-user personalization information is stored in a small database on the machine which hosts the Browser (Figure 2).

Personalization is done via the web interface, and is easily changed as the user's needs change. The page relies on whatever security mechanism is used by the web server—htaccess, kerberos, etc.—to provide user identification for access of personalization information.

Queue Browser Maintenance

A utility that provides information about machines across a number of sites has the potential to become a maintenance headache. Who updates the utility when configurations change at any given site? The implementation of the queue page addresses that issue by delegating the responsibility of providing site-specific information to administrators at each site.

The Queue Browser is implemented as a Common Gateway Interface (CGI) program which queries each site of interest and produces the queue display page dynamically. Information specific to each site is accessed from that site via the Hypertext Transfer Protocol (HTTP); hence sites can supply the necessary information via new pages on existing web servers, rather than needing to install any new servers or other software, with concomitant maintenance and security considerations. A current snapshot of queue information is provided to the Queue Browser dynamically by CGI programs run at the site of interest. In the current implementation, queue information is produced by cron scripts run at regular intervals on the various unclassified HPC systems at the NAVO MSRC site. All scripts are compliant with the NAVO MSRC security model, and are fully Kerberos compliant. The results are filtered to exclude sensitive data and written to local files, and a data collection script gathers all the files and moves them to the web server system where they are served in response to the Queue Browser's query.

USER SERVICES NOTE: Hotpage is currently undergoing Beta testing at NAVO MSRC through the end of 1999. It will be implemented across the program during the first quarter of 2000.

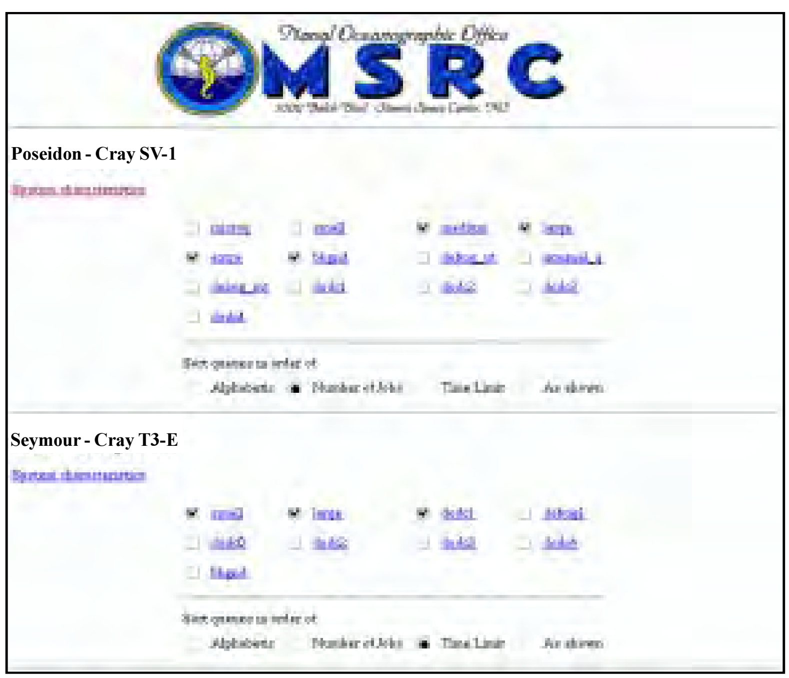


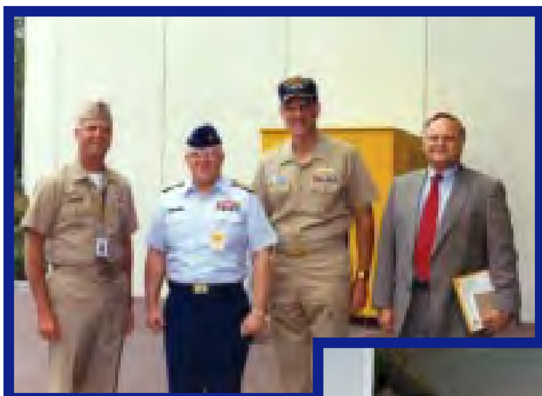
Figure 3. Example Site-Specific Personalization

The Porthole

A Look Inside NAVO MSRC

NAVO MSRC hosted many visitors during 1999. Pictured below are a few of the many welcome visitors that have toured our facility.

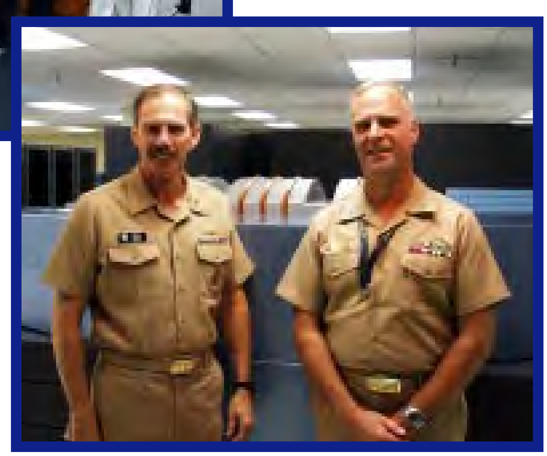
Senator Trent Lott, Mississippi; Terry Blanchard, NAVO MSRC Deputy Director; RADM Kenneth Barbor, Commander, Naval Meteorology and Oceanography Command; and Kent Kresa, Chairman of the Board, President and Chief Executive Officer Northrop Grumman Corporation (right)



RADM Kenneth Barbor, Commander, Naval Meteorology and Oceanography Command; RADM Paul Pluta, Commander, Eighth Coast Guard District; CAPT Larry Warrenfeltz, Commanding Officer, Naval Oceanographic Office; and Landry Bernard, Technical Director, Naval Oceanographic Office(left)

Diane Sawyer, Charlie Gibson and the Good Morning America crew (left) take a visual tour.

Congressman Gene Taylor, RADM Kenneth Barbor, Commander, Naval Meteorology and Oceanography Command and Steve Adamec, NAVO MSRC Center Director (above)



RADM Richard West, Oceanographer of the Navy and RADM Kenneth Barbor, Commander, Naval Meteorology and Oceanography Command (right)

On July 8, 1999 NAVOCEANO sponsored an open house for employees and their families. This day allowed employees to spend the day with their families, while showing off the functions and facilities of NAVOCEANO. This summer's open house organized tours of the departments, games, entertainment, and giveaways provided by various agencies and contractors within NAVOCEANO. The weather was hot (it was Mississippi in July!), but a good time was had by all.

Clay Harper taking a break from the snowball machine.

User outreach coordinator Eileen Crabtree showed off her hot air balloon basket and offered a balloon ride for two.

Tina Young, Beth Laboe, and Mike Miller strategized over water bottle distribution.

Bernard Chmiel explained the mass storage archive system to visitors.

Families and friends enjoyed the afternoon.

User Transition from the C90 to the SV-1

Eileen Crabtree, User Services Lead

An SGI Cray SV-1 supercomputer system was introduced to the NAVO MSRC user community July 2, 1999, as a replacement for the Cray C916. The SV-1 system is SGI's first-generation scalable vector supercomputer, which combines complementary metal oxide semiconductor- (CMOS-), based vector processing, dynamic random access memory (DRAM) technology, and custom-streaming cache into a new and unique supercomputer architecture.

The SV-1 is configured as a two-node system. The first node is named Poseidon and the second node is named Trident. The system is configured with 32 one-gigaflop CPUs and 32 gigabytes of central memory divided equally across the nodes. The SV-1 system configuration doubles the number of CPUs and quadruples the amount of central memory previously available on the C916 system.

Accessing the SV-1

To access the SV-1 system node named Poseidon, one may use "krlogin -x", "ktelnet -x", "krsh -x", or "ssh" to poseidon.navo.hpc.mil from a kerberized workstation. Utilities *kftp* and *krcp* are available for file transfers between the SV-1 and other MSRC systems. To facilitate batch mode data transfers, SV-1 users with the proper .rhosts file on msas1 may use the non-kerberized *rcp* and *remsh* commands from the SV-1 to msas1.

Batch Jobs on the SV-1

Interactive login access is limited to Poseidon, the SV-1 system node one. Trident, node two, is available only for user batch queue access via SGI's Network Load Balancer (NLB) software, which routes batch jobs to the appropriate node based on the current system load. Because the two nodes of the SV-1 system have different /tmp file systems, users are allowed access to utilities *krcp*, *kftp*, and *krsh* to move data to and from Trident's /tmp directory, as needed among other MSRC systems. File movement is not required for user home directory access because directory /u/home is Network File System-mounted (NFS-mounted) from Poseidon to Trident.

One may monitor batch jobs running on Trident by using the *-h* option of the *qstat* command to specify a target host. For example, one may use *qstat -h trident -a* to view a list of all batch jobs queued and running on Trident.

Figure 1. Cray C916

The Transition

Users of the new system should experience few software transition issues. The SV-1 uses the same UNICOS 10 operating system previously available on the C916. The same compilers, utilities, tools, and libraries that were available on the C916 also reside on the SV-1. All user codes transitioned from the C916 to the SV-1 must be recompiled due to differences in the architecture of the two systems.

A New Feature: the SV-1 Data Cache

Each processor on the SV-1 contains a 256-Kilobyte (32-Kilo word) data cache. The cache is shared by scalar and vector data and instructions. This cache is quite different from those to which users may have grown accustomed. This cache is neither direct-mapped nor associative, nor does it consist of lines of multiple words. Each word in the cache comprises a cache line, and each line can access and store any word in memory. The entries in cache are overwritten on a least recently used basis, so any data value remains in cache until it becomes the least recently used entry, at which time it becomes the candidate to be overwritten with other data.

The SV-1's data cache can improve the scalar performance of application codes. Scalar read references prefetch eight words from memory by allocating eight single word lines for each scalar reference. This provides spatial locality for scalar codes.

Overall application performance depends moderately on cache usage. Modification of application codes for cache usage, however, remains challenging. In some cases, one readily observes how a code might be optimized for cache usage; in most cases optimizations are not clear. The cache is automatically enabled, however, on the SV-1, and the compiler automatically optimizes application codes for cache usage. Thus users need not necessarily concern themselves with this feature of the SV-1. Currently, much work is being performed on the compiler to fully implement cache optimizations.

Price/Performance

Users of the SV-1 system may notice performance differences compared to the C916. On average, single CPU applications executed on the SV-1 exhibit approximately two-thirds the performance of the C916 system. This rough conversion factor was determined by examining numerous vector applications. Some users may observe a performance increase using the SV-1; others will observe a decrease. However, one may expect the performance ratio of the two machines to approach unity as the SGI compiler developers further optimize the compilers for the SV-1, and as the user community learns to optimize vector codes for data cache usage.

SV-1 users should be aware that supercomputers are no longer measured by pure performance. Cost issues are driving the industry and its customers to compare price/performance measurements. Because of this, the supercomputing industry is developing commodity component systems. For example, the SV-1 system uses CMOS-integrated circuit technology rather than the higher performing and higher cost emitter coupled logic (ECL) technology. Further, the SV-1 uses DRAM memory rather than higher performing and higher cost static random access memory (SRAM). As a result, the price/performance ratio of the SV-1 is five times better than the Cray J90, and eight times better than the T90.

Figure 2. Cray SV-1

At the same time, the SV-1 is truly a supercomputer that provides unprecedented vector processing capabilities. The SV-1 scales to one teraflop of peak CPU performance and more than one terabyte of central memory. When compared to the C916, the SV-1 scales to 64 times the peak CPU performance and 128 times the available memory.

Performance Optimizations

Users wishing to optimize application codes for the SV-1 should concentrate on the usual vectorization, parallelization, and I/O performance optimizations. The easiest method is to simply ensure that the application code has no bottlenecks on the computer's hardware. Performance monitoring tools are available on the SV-1 to assist in this effort. These tools will direct the user to performance bottlenecks and will indicate the type of optimization one might perform to remove a bottleneck. Vectorization can result in up to a 10x increase in code performance. Parallelization can result in a performance increase of up to Nx, where "N" is the number of available CPUs on the system. Finally, a code which is I/O bound can experience up to a 1000x performance improvement with I/O optimization! Of course individual users' mileage may vary for each application.

Many users have become familiar with the C916's performance analysis tools. Each of these tools is available on the SV-1 and include hpm, ja, flowview, jumpview, perfview, procview, profview, ATExpert, TotalView, and Xbrowse. The tools are easy to use and fully documented.

Learn More About Optimization

Users who want to learn more about application optimization should visit the SGI Technical Library located at:

http://techpubs.sgi.com/library

Of particular interest at this web site are the following UNICOS books, available for online viewing or printing:

- Application Programmer's I/O Guide
- CF90 Commands and Directives Reference Manual
- Fortran Language Reference Manual, Volumes 1 - 3
- Guide to Parallel Vector Applications
- Optimizing Application Code on UNICOS Systems. Web site: http://techpubs.sgi.com/library/tpl/cgi-bin/browse.cgi?db=bks&coll=uncs&pth=ALL#O

Navigator Tools and Tips

Forwardable Tickets

Simone Crider, User Services

Kerberos credentials are forwardable from one MSRC host to another. A forwardable ticket will allow a user access to another MSRC host without a password prompt. Forwardable tickets are obtained using the following command:

% kinit -f
Password for user_login@NAVO.HPC.MIL:
{Enter Kerberos Password}
Passcode:
{Enter 6-digit passcode generated from SecurID Card

Use the following command to verify Kerberos credentials:

% klist -f
Ticket cache: **/tmp/krb5cc_xdm_0**
Default principal: **user_login@NAVO.HPC.MIL**
Valid starting Expires Service principal
08/03/99 14:28:47 08/04/99 00:28:47
krbtgt/NAVO.HPC.MIL@NAVO.HPC.MIL
Flags: FIHA

To forward credentials using Kerberos commands, use the "-F" option. However, the Kerberos Secure Shell command will automatically forward credentials to another host. The following commands will transfer Kerberos credentials to another host:

% krlogin -x -F hostname
% krsh -F hostname
% ssh hostname

File Transfer

Ray Sheppard, User Services

There are three basic types of file transfers at NAVO MSRC: external to a non-Kerberized host, external to a Kerberized host, and internal transfers. Data transfers between a non-Kerberized host and a NAVO MSRC server may only be done from one of our servers to your machine. For transfers between Kerberized machines, a user has three options: *kftp, krcp,* and *scp*.

To use all of these, a valid Kerberos ticket is required. Kftp (which works like ftp) is faster than the others and is less CPU intensive. For batch jobs, *krcp* is syntactically the easiest. The following example syntax for *krcp* moves file "xfer_file" from Neptune to my_kerberos_machine:

neptune% krcp /tmp/ray/xfer_file my_kerberos_machine.world.net:/home/ray/

For a significant speed-up in transfers between servers at NAVO MSRC, the High Performance Parallel Interface (HPPI) interface may be used by replacing the machine name by the machine_name-hip0:

unicos% kftp neptune-hip0

Finally, Kerberized scp uses the same basic syntax as krcp, but is much slower because the data bits are encrypted/decrypted during the kerberized secure shell transfer.

All of the Kerberized transfer methods work between NAVO MSRC systems, but there is an additional possibility. We have a waiver for non-Kerberized rcp to function to our mass storage device (msas1) to facilitate storage. The rcp command works like its krcp counterpart, but without the need for a Kerberos ticket. More information about this command may be found on the web at: http://www.navo.hpc.mil/usersupport/news/remote_commands.

Show Usage

Debbie Franklin, User Services

We have created a command to check account allocation usage on the systems at NAVO MSRC called "/usr/local/bin/show_usage". When this command is executed, the date the accounting file was last updated is given, as well as the number of hours allocated, number of CPU hours used, the resulting balance, and percentage used. Each project is shown for a user in multiple projects. Note, accounting data is shown for the entire project, not per user.

odyssey% /usr/local/bin/ show_usage

Date	Time	System	Account
08/02/99	08:45:13	odyssey	APROJID

Allocated	26233.00
Used	19369.55
Balance	6863.45
Percent	73.84 used

Recent Events

SIP'99 Forum

The NAVO MSRC and Army Research Laboratory (ARL) MSRC PET programs recently co-sponsored a forum on "The Role of High Performance Computing in Signal and Image Processing." It was held at the Grand Hotel and Casino in Gulfport, Mississippi, on May 25, 1999. This was the second SIP forum; the first one was held in Aberdeen, Maryland, on February 3-5, 1998. The two forums brought together a select group from the DoD SIP community with diverse expertise in order to identify critical areas of need for DoD SIP research.

The SIP'99 forum was attended by 46 researchers and managers from across the DoD SIP community. Twenty-six papers on a diversity of subjects from current DoD SIP research activities to overviews of the ARL MSRC and NAVO MSRC PET programs were presented. Additionally, the forums provided a venue to describe the ongoing activities and services of CHSSI and PET programs, and solicited guidance from the SIP community as the programs attempt to support the DoD SIP community. Get more details about the SIP'99 Forum at:

http://www.navo.hpc.mil/pet/sip/

Metacomputing Workshop

NAVO MSRC PET hosted the first cross-MSRC workshop on Metacomputing at the University of Virginia School of Computer Science August 11 to 13, 1999. The ARL MSRC PET team led the program, which consisted of presentations from four metacomputing system developers and the four MSRCs. Discussions between the MSRCs, the DCs, and metacomputing experts enable the creation of a metasystem roadmap for HPCMO.

The purpose of metacomputing is to increase both the effectiveness and efficiency of high performance computing to support the DoD mission, according to John Baird, HPCMO Metacomputing Lead.

Metacomputing systems that are being supported or evaluated by the MSRCs include Legion, from the University of Virginia; Globus, from the Department of Energy; and Gateway, from Syracuse University. The progress of National Aeronautics and Space Administration's (NASA's) Information Power Grid (IPG) is also being closely watched by many.

More details about the workshop and metacomputing approaches can be found at http://www.navo.hpc.mil/pet/Metacomputing99.

Upcoming Events

October 19-21, 1999
Grid Forum
Chicago, IL
www.gridforum.org/

November 13-19, 1999
Supercomputing '99
Portland, OR
www.sc00.org/

November 29-December 1, 1999
Parallel and Real Time Systems (PART)
Melbourne, Australia
www.cs.rmit.edu.au/PART'99/

December 2, 1999
International Workshop on Cluster Computing
Melbourne, Australia
www.dgs.monash.edu.au/~rajkumar/tfcc/IWCC99/

February 29-March 2, 2000
NAVO MSRC PET Review
Stennis Space Center, MS
www.navo.hpc.mil/pet/

May 1-5, 2000
International Parallel and Distributed Processing Symposium
Cancun, Mexico
www.ippsxx.org/ipdps2000.htm

June 5-9, 2000
2000 HPC Users Conference
Albuquerque, New Mexico
www.hpcmo.hpc.mil

July 23-28, 2000
SIGGRAPH 2000
New Orleans, Louisiana
www.siggraph.org/s2000/

November 4-10, 2000
Supercomputing 2000
Dallas, Texas
www.sc00.org/

NAVO MSRC
1002 Balch Blvd.
Stennis Space Center, MS 39522