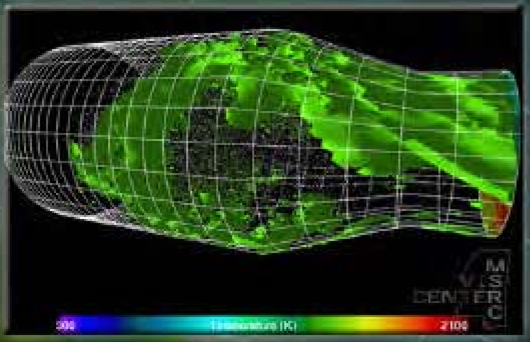
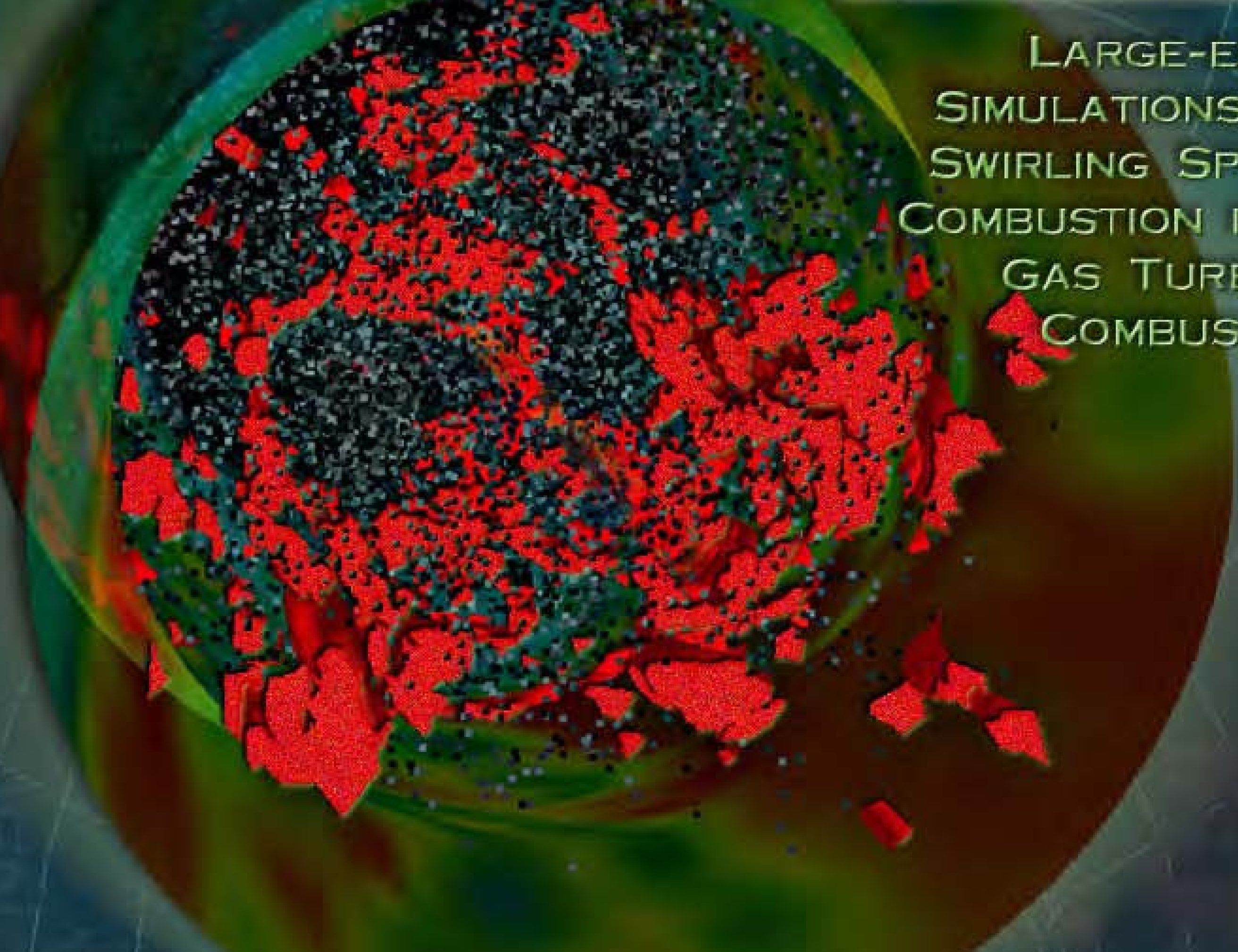
Navigator

NAVO MSRC Fall 2000



Large-eddy Simulations of Swirling Spray Combustion in a Gas Turbine Combustor



News and information from...

The Naval Oceanographic Office Major Shared Resource Center

The Director's Corner

Steve Adamec, NAVO MSRC Director

Changes at NAVO MSRC Designed to Benefit Users

The past several months have been notable ones here at the NAVO MSRC. We've completed a sweeping series of center enhancements, designated as Performance Level 3 (PL3), across all major technology areas within the MSRC. The present PL3 HPC systems provide a **thousand-fold increase in aggregate peak computing performance (i.e., 3.2 teraflops)** when compared to the aggregate peak capability (3.2 gigaflops) of the Primary Oceanographic Prediction System Supercomputer Center when it was established here at NAVOCEANO in 1990. This enormous computational capability, coupled with a sustained 10-year NAVOCEANO focus on supporting the largest and most demanding DoD computational applications, has enabled unparalleled advances in several of the key DoD science and technology areas served by the High Performance Computing Modernization Program (HPCMP).

With all of this diverse computational capability that's been fielded across more than 20 shared resource centers (SRCs) by the HPCMP, it has become critically important for us to redouble our efforts in assessing and implementing common user environments, practices, and tools within and across the SRCs. Your individual and collective user feedback at forums such as Program Review 2000, HPC Users Group 2000, and HPCAP/SRCAP meetings, makes it clear that you consider this to be one of your highest priorities for the SRCs. In response, the SRCs have undertaken or intensified strategic cross-cutting collaborative efforts in enabling technical areas such as mass storage and archival, metacomputing, HPCMP-wide shared information environments, and security. Here at the NAVO MSRC, we've supplemented those efforts with a Programming Environment and Training (PET) program that's more tightly focused than ever on user environment, tools, and productivity. We've also formally added an Inter-MSRC Facilitator (IMF) component to our user support organization. The IMF's primary function is two-fold: (1) to quickly engage and resolve user requests/issues which span multiple SRCs; and (2) to work with the other SRCs to identify and prioritize possible improvements to cross-SRC environments and practices. We hope to report substantial progress on these issues to you during the upcoming DoD HPC Users Group Conference at Biloxi, Mississippi in June 2001.

Finally, we'd like to recognize and say farewell to Mr. Serge Polevitzky, who served as Logicon's Program Manager at the NAVO MSRC for over four years. Serge's enthusiasm, technical prowess, and dynamic leadership were major contributors to the success of this center, as they will be for his new assignment on the West Coast in support of a Logicon initiative there.

About the Cover:

Pictured are images from an OpenGL-based application created by NAVO MSRC visualization experts to simulate the environment inside a gas turbine combustor. The program is helping scientists from Georgia Institute of Technology create intelligent gas turbine engines for the next generation of Army tanks and helicopters (see story on page 10).

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
Code N7
1002 Balch Boulevard
Stennis Space Center, MS 39522
1-800-993-7677 or
help@navo.hpc.mil

NAVO MSRC Navigator
www.navo.hpc.mil/navigator

NAVO MSRC Navigator is a bi-annual technical publication designed to inform users of the news, events, people, accomplishments, and activities of the Center. For a free subscription or to make address changes, contact NAVO MSRC at the address above.

DESIGNERS:
Patti Geistfeld, pgeist@navo.hpc.mil
Kerry Townson, ktownson@navo.hpc.mil
Lynn Yott, lynn@navo.hpc.mil

Any opinions, conclusions, or recommendations in this publication are those of the author(s) and do not necessarily reflect those of the Navy or NAVO MSRC. All brand names and product names are trademarks or registered trademarks of their respective holders. These names are for information purposes only and do not imply endorsement by the Navy or NAVO MSRC.

Approved for Public Release
Distribution Unlimited

Contents

The Director's Corner

Feature Articles

Scientific Visualization

Programming Environment and Training

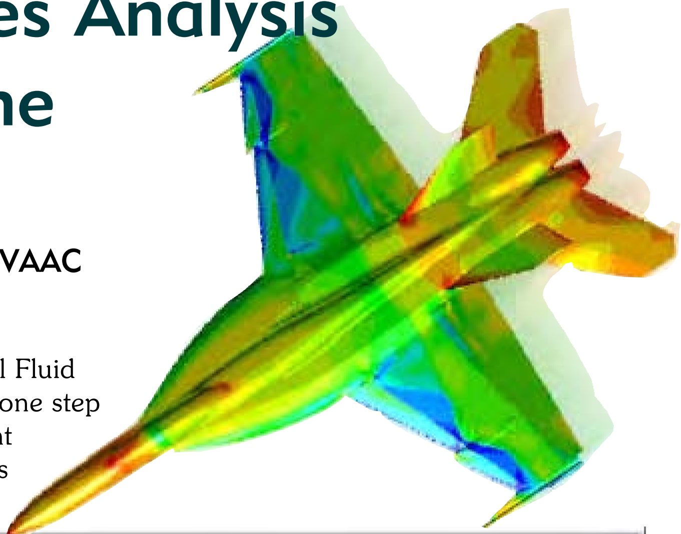
The Porthole

Navigator Tools and Tips

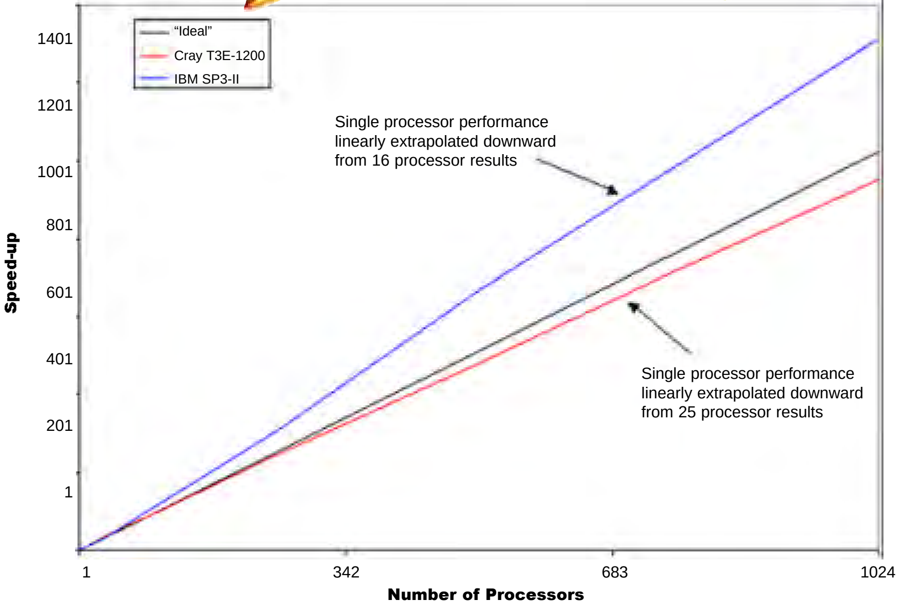
Upcoming Events

Bringing Navier-Stokes Analysis One Step Closer to the Design Process

Kenneth E. Wurtzler and Robert F. Tomaro, AFRL/VAAC



The reality of incorporating Navier-Stokes Computational Fluid Dynamics (CFD) analysis earlier in the design process is one step closer with the inclusion of parallel computing. The recent expansion of the NAVO Cray T3E to 1088 processors has made this step possible for the Applied Computational Research Group of the Air Force Research Laboratory (AFRL/VAAC). The in-house code Cobalt60 has combined the flexibility of unstructured CFD with the power of parallel computing to enable less than one-day turnaround for full aircraft analysis. After ten years of development, the ability of Cobalt60 to routinely provide meaningful data to the engineer has been magnified by the availability of up to thousands of processors.



Single processor performance linearly extrapolated downward from 16 processor results

Single processor performance linearly extrapolated downward from 25 processor results

Figure 1. **Speed-up for a large problem (3.17 million cells) for a benchmark run.**

A previous attempt to do some benchmark cases on 1024 processors on the NAVO Cray T3E in the fall of 1999 revealed a few shortcomings in some portions of the code. The routines dealing with domain-splitting and the calculation of wall-distance for the turbulence model actually showed signs of reverse scalability after several hundred processors were utilized. Once this problem was realized, corrections were implemented that kept the domain-splitting, pre-processing function limited to running concurrently on groups of user-defined size (approximately 30 to 100 processors). The wall-distance calculation was revamped. However, the chance to benchmark on 1024 processors at the NAVO Cray T3E was lost. Another opportunity to benchmark on the upgraded T3E at the Army High Performance Computing Research Center (AHPCRC) in Minnesota arose in June 2000. The benchmark for the IBM SP3 occurred recently at NAVO MSRC. A standard test case consisting of a 3.17-million-cell F-16C grid was used with the results shown in figure 1.

Using this phenomenal scalability up to 1024 processors, 12 Navier-Stokes runs were completed in 12 hours to complete a tail placement design matrix on a generic twin-tail fighter. The current AFRL/VAAC DoD Challenge Project, "Unsteady Aerodynamics of Aircraft Maneuvering at High Angles of Attack" has benefited greatly from this capability.

A research project that is closely related to this Challenge Project is focused on shock-boundary layer interaction at moderate angles of attack on an F/A-18E/F. Static analyses were obtained by first running at 6° angle of attack. The grid was then rotated 2°, and the previous solution was used as the initial start-up point for successive runs. This decreased overall convergence time for the entire suite of runs.

The initial portion of the research, run on the NAVO Cray T3E and at the Maui High Performance Computer Center IBM SP3, was aimed at a comparison of turbulence models and their ability to predict the separation over the wing. The two turbulence models investigated were the one-equation Spalart-Allmaras model and the two-equation Menter's Shear Stress Transport (SST) model. The Spalart-Allmaras model predicted the separation to occur later than the wind tunnel data suggested. Menter's SST model did a better job of predicting the separation on the wing when compared to wind tunnel data.

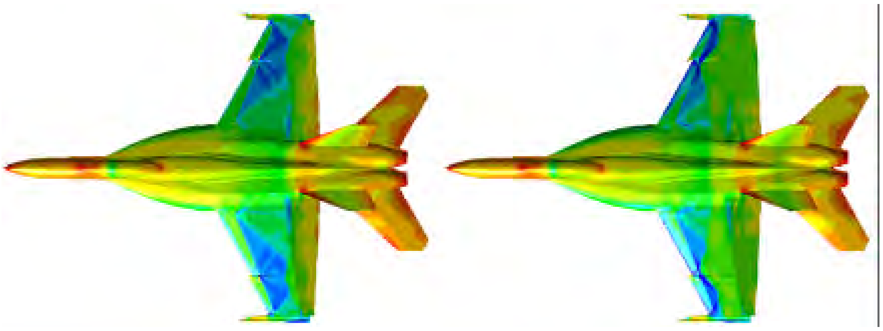


Figure 2. **Location of shocks on F/A-18 E/F.**

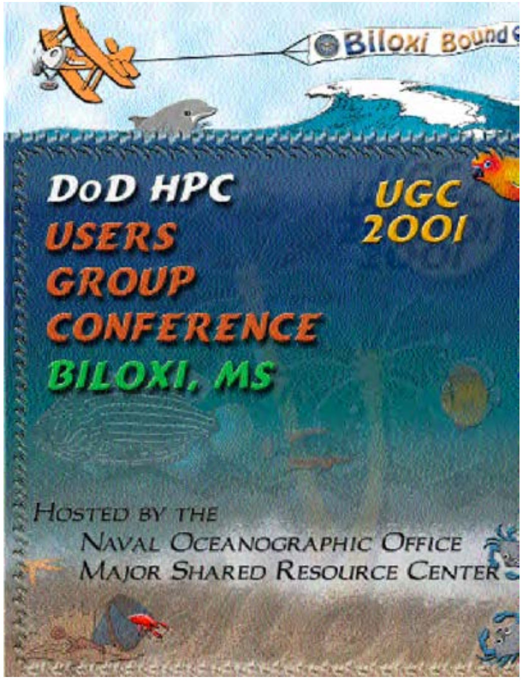
The wind tunnel geometry did not have horizontal or vertical tails for some of the runs. A new grid that accounted for the tails was created, and several CFD solutions were obtained to investigate the impact of the tails on the flow. In figure 2, the location of the shocks on top of the wing is shown for two moderate angles of attack. The presence of the tails impacts the inboard trailing edge flow over the wing.

The combination of a highly scalable CFD code and expert in-house unstructured grid-generation capabilities has given AFRL/VAAC the ability to quickly respond to projects that require highly accurate aerodynamic analysis. By accessing large numbers of processors (~512), the turnaround time on the NAVO Cray T3E allows results to be obtained in days, not weeks. Changes to the grid or turbulence model can be made if a review of the data requires them. The engineer does not have to wait weeks to determine if something is wrong with a solution. This speed is what is needed in order to bring Navier-Stokes analysis one step closer to the design environment.

COMING IN JUNE 2001...

UGC 2001
BILOXI, MISSISSIPPI

Hosted by NAVO MSRC, Stennis Space Center, MS



Newest Supercomputer Makes

NAVO MSRC recently completed installation of an IBM RS/6000 SP supercomputer, code named “Habu.” One of the largest systems ever built by IBM, Habu cruises at over 2 trillion operations per second, making it one of the fastest and most capable HPC systems in the world today. TOP LEFT: Unloading one of the two moving vans needed to transport the components. TOP RIGHT: Technicians begin installing the 24 cabinets that house the computer. BOTTOM: A panoramic photo of the completed system.

Its Debut at NAVO MSRC

The IBM RS/6000 SP, installed June 16, 2000, is the latest supercomputer installed at the NAVO MSRC. It is capable of processing two trillion calculations per second, making it the fourth largest supercomputer in the world. The two-teraflop system harnesses the computing power of 1,336 microprocessors, 1,336 terabytes of memory, and 17 terabytes of IBM disk space. With the addition of the new RS/6000 SP system, the aggregate computational capability at the NAVO MSRC exceeds 3 trillion operations per second.

Looking down the aisle at many of the 24 cabinets which house Habu's 1.336 processors and massive storage capabilities.

> "High performance computing technology of this magnitude gives us unparalleled capabilities in the daily ocean- and global-scale modeling we perform to support worldwide DoD operations."
>
> — Landry Bernard,
> NAVOCEANO Technical Director

The improvements directly benefit DoD scientists and researchers by providing the capability to run the very largest DoD Challenge applications. The RS/6000 SP will be used to assemble the most detailed models of ocean waves, currents, and temperature ever constructed. The computer models will enable scientists to predict the behavior of the world's oceans with incredible precision, increasing the safety of naval vessels and commercial shipping, and augmenting search and rescue capabilities. In addition, it will enhance the forecasting of weather patterns that are heavily influenced by ocean phenomena, such as "El Nino" and "La Nina." Scientists will also use the IBM in a wide range of DoD research projects, from designing stronger aircraft and missile designs to simulating battlefield environments. One DoD Challenge project in Climate Weather Ocean Modeling and Simulation on the IBM is the 1/32 Degree Global Ocean Modeling and Prediction project. The overall objectives of this Navy project are to simulate, understand, nowcast, and forecast global ocean circulation and to increase the capability to model it. The DoD Challenge Project in Computational Electromagnetics and Acoustics uses this system for the Radar Signature Database for Low Observable Engine Duct Designs. This Air Force project will increase mission effectiveness and survivability on current and future combat aircraft, such as the F-117, B-2, F-22, and Joint Strike Force, which all have low observability as a requirement.

Landry Bernard, NAVOCEANO Technical Director, commented that, "High performance computing technology of this magnitude gives us unparalleled capabilities in the daily ocean- and global-scale modeling we perform to support worldwide DoD operations. The benefits to DoD research and development will be enormous, enabling substantive advances in the science areas which are critical to the nation's defense."

NAVO IBM RS/6000 SP: Setting New Standards

Timothy J. Campbell, Ph.D., NAVO MSRC Programming Environment and Training

Molecular-dynamics (MD) simulations continue to play a critical role in our understanding of various phenomena in physics, chemistry, biology, and materials sciences. In the MD approach, one obtains the phase-space trajectories of the system (positions and velocities of all atoms at all times). This allows one to study how atomistic processes determine macroscopic materials properties. In classical, empirical MD simulation, the total force on an atom is computed from the interatomic potential which is expressed as an analytical function of the coordinates of all atoms. The results discussed in this article are for classical, empirical MD (in contrast to the more computer intensive quantum mechanical MD approach). The present state-of-the-art in classical, empirical MD simulations involves 10 to 100 million atoms. For a recent discussion of state-of-the-art MD simulations in DoD research see the "Large-Scale Atom Simulation" article in the Spring 2000 issue of the NAVO MSRC Navigator **[http://www.navo.hpc.mil/cgi-bin/Navigator/navigator.cgi].**

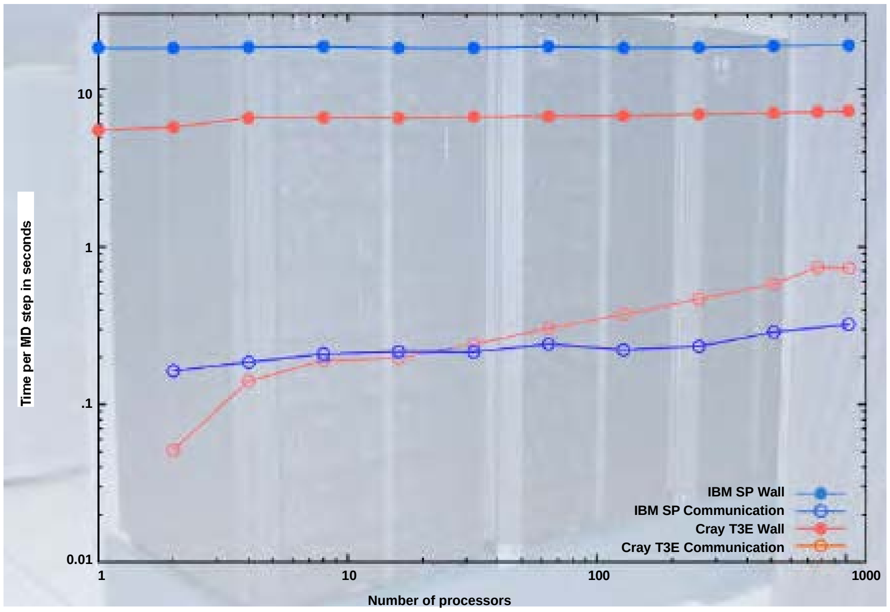


Figure 1. **Wall clock (filled circles) and communication (open circles) times for MD on the IBM SP (red) and Cray T3E (blue). The workload is scaled linearly with the number of processors: 648,000*P*-atom silica systems on *P* processors (*P* = 1,...,1024)**

To implement MD on parallel computers, a divide-and-conquer strategy based on spatial decomposition is commonly used. The total volume of the system is divided into P subsystems of equal volume, and the data associated with atoms of a subsystem are assigned to a processor in an array of *P* processors. To calculate the force on an atom in a subsystem, the data associated with atoms in the boundaries of neighboring subsystems must be communicated using a message-passing protocol. With spatial decomposition, the computation scales as *N*/*P*, while communication scales in proportion to $(N/P)^{2/3}$. The communication overhead thus becomes less significant when *N*/*P* is greater than 10^4, i.e., for coarse-grained applications.

Performance tests of MD have recently been completed on the new IBM SP computer at NAVO MSRC using up to 1280 processors and compared with results on the NAVO Cray T3E. The IBM SP at NAVO MSRC consists of 320 4-way 375-MHz POWER3 compute nodes, each with 4 GB of memory. The T3E at NAVO MSRC consists of 1088 450-MHz Digital Alpha processors with 258 GB of memory. The MD program is written in Fortran 77 with MPI (Message Passing Interface) for message passing.

Figure 1 shows the execution time of MD for silica (SiO_2) material as a function of the number of processors, *P*, for both platforms. The system size is scaled linearly with the number of processors, so that the number of atoms, *N* = 648,000*P*. The speed of the program is defined as the number of MD steps executed per second times the number of atoms; the "memory-bound" speed-up is defined as the speed divided by the single-processor speed. A parallel efficiency on *P* processors is defined as the speed-up divided by *P*. The MD implementation scales well on both platforms. The parallel efficiency on 1024 processors of the IBM SP is about 75%, and the corresponding time per MD step is 7.3 seconds. Similar performance tests on 1024 processors of the NAVO Cray T3E yielded a 97% parallel efficiency with a time per MD step of 18.9 seconds. We see that although the internal communication on the NAVO Cray T3E is faster for large numbers of processors, the wall time is decreased by more than 60% on IBM SP for the coarse-grained MD application.

The increased performance and larger memory of the NAVO IBM SP

continued on next page

NAVO MSRC Networking

Randy Becnel, Logicon, Inc.

The high performance computational and file server platforms that comprise the NAVOCEANO MSRC require high bandwidth, low latency, and high availability networks to provide connectivity between platforms and to users throughout the DREN and general Internet communities.

The internal MSRC network is a combination of HiPPI (800 Mbps), ATM OC-3/OC-12 (155 and 622 Mbps), FDDI (100 Mbps), and 10/100 BaseT Ethernet. Connectivity to the DREN network is via an ATM OC-12 (622 Mbps) Wide Area Network (WAN) link. The core of the MSRC network backbone is a pair of Cisco 12012 Gigabit Switch Routers (GSR). One GSR is positioned at the NAVO MSRC connection point to the DREN network and the second is installed within the NAVO MSRC Programming Environment and Training (PET) facility.

The two GSRs are linked via an ATM OC-12 interface. This high-end ATM switching/router has a switching backplane scalable to 60 Gbps and supports OC-3 (155 Mbps) through OC-48 (2.4 Gbps) and 1000 BaseT Ethernet (Gigabit Ethernet) interfaces, positioning the NAVO MSRC to meet current and future high performance networking requirements. Connectivity to Local Area Network (LAN) components is provided by two Cisco 7513 routers and 8540 ATM switching router.

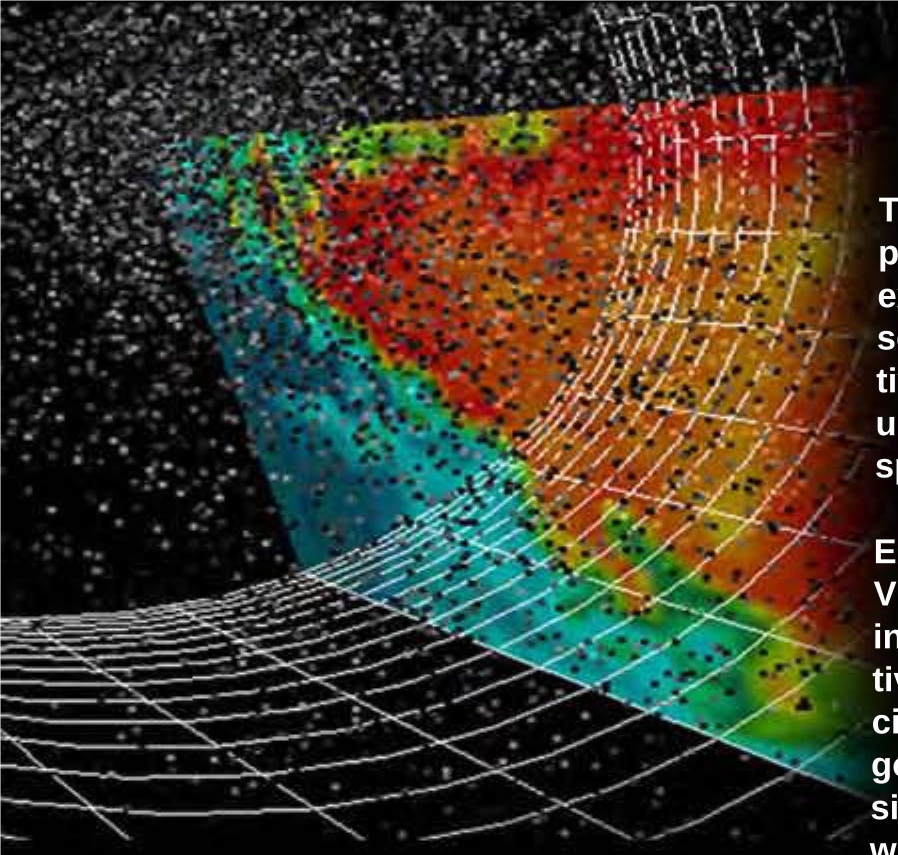
High-speed data transfer between computational servers and the mass storage servers is accomplished primarily via the 800 Mbps HiPPI network. Varying combinations of ATM OC-3/OC-12, FDDI, and Fast Ethernet provide user access to the computational and mass storage resources of the NAVO MSRC. The support and visualization workstation network consists of multiple Cisco 5500/5000 network switches providing switched 10/100 BaseT connectivity for support analyst workstations. The network switches are linked to the NAVO MSRC backbone via multiple full-duplex 100 BaseT trunk links providing high-speed access and fault tolerance. Legacy FDDI connectivity for Cray platforms is provided via Cisco 1400 concentrators. FDDI will be phased out over time in favor of ATM, Fast Ethernet, and Gigabit Ethernet connectivity for host access.

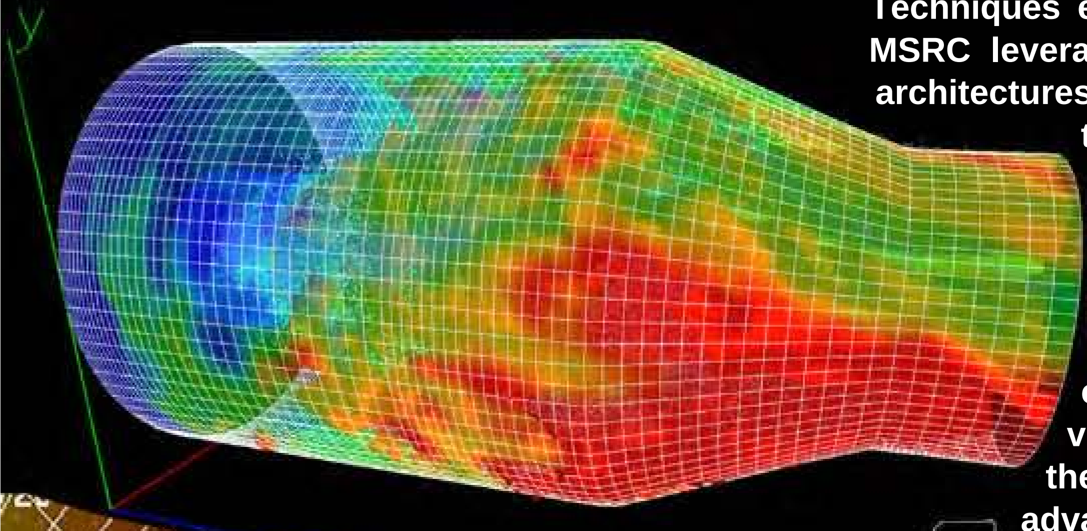
Future enhancements to the NAVO MSRC networking infrastructure include expanded use of Gigabit Ethernet (GigE and 10 GigE) technologies for both backbone and host interface connectivity. We also plan to explore emerging Gigabit System Network (GSN or SuperHiPPI) technology as a means of high-speed host-to-host data transfer and Storage Area Network (SAN) implementations. GSN promises transfer rates of 600+ Mbps. Additionally, ATM technology will continue to be a major part of the NAVO MSRC network. In addition to speeds beyond OC-48, emerging ATM protocols, such as Packet Over SONET (POS), are planned to be a part of the infrastructure. Developments in broadband optical technologies, such as wavelength division multiplexing (WDM), are also being monitored by the NAVO MSRC network engineers for possible applications to the center.

NAVO IBM RS/6000 SP (continued from previous page)

allows us to simulate atomic systems much larger than has ever been done. In fact, MD simulations of silica have been performed on all 320 compute nodes (1280 processors) of the NAVO IBM SP that involve up to 8 billion atoms with a corresponding physical size of about 500 nanometers. Because each MD step for the 8 billion atom system takes several minutes, simulations of that size are limited to studing structural relaxations and stress distributions. However, simulations of advanced ceramic materials involving 2 to 4 billion atoms to study longer time important processes, such as fracture, are now a reality.

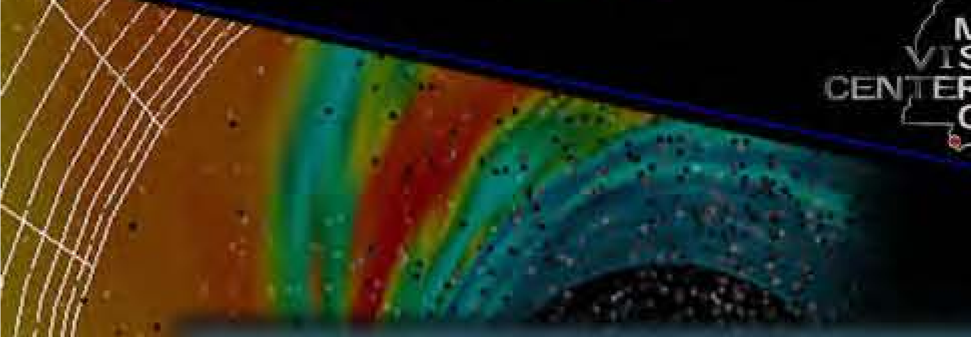
Recent advances in scalable multiresolution algorithms coupled with access to massively parallel computers, like the new IBM SP at NAVO MSRC, have enabled practical MD simulations to move beyond 1 billion atoms, where the corresponding physical size of the systems are on the order of hundreds of nanometers. The significance of this is immediately apparent when we consider that the design of advanced materials and reliable devices in extreme environments such as high temperatures incorporates nanometer-scale features. These recent performance tests represent the new state-of-the-art in molecular dynamics simulations and how NAVO MSRC is setting new standards in supporting DoD research.



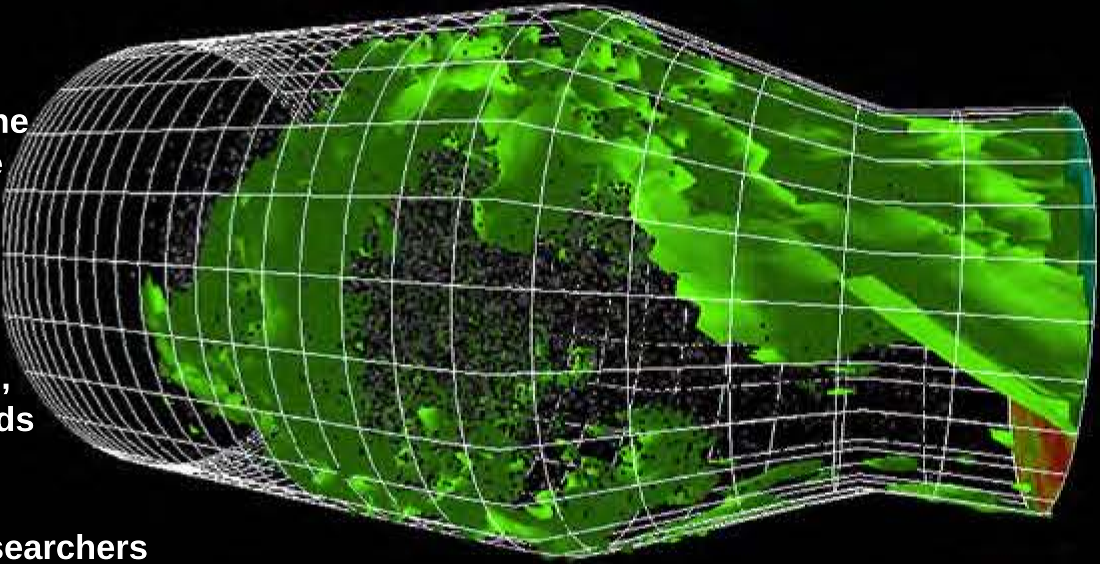


The ability to interactively explore computational domains is one of the most exciting and effective methods used in scientific visualization. These interactive environments are built around the user's data structure and are tuned specifically for interactive frame rates.

Experts within the NAVO MSRC Visualization Center are busy developing OpenGL-based immersive interactive environments which are both efficient and portable. Resolution and polygon counts are issues that must be considered in order to maintain interactivity within a software application. Techniques employed within the NAVO MSRC leverage the latest in hardware architectures and software techniques to provide both optimum resolution of the data and full control over both the temporal and spatial domains. This strategy supports full visualization capability using commodity visualization technology at the user's site, while taking advantage of high-speed networks such as DREN to use specialized and very expensive visualization server equipment within the MSRC.

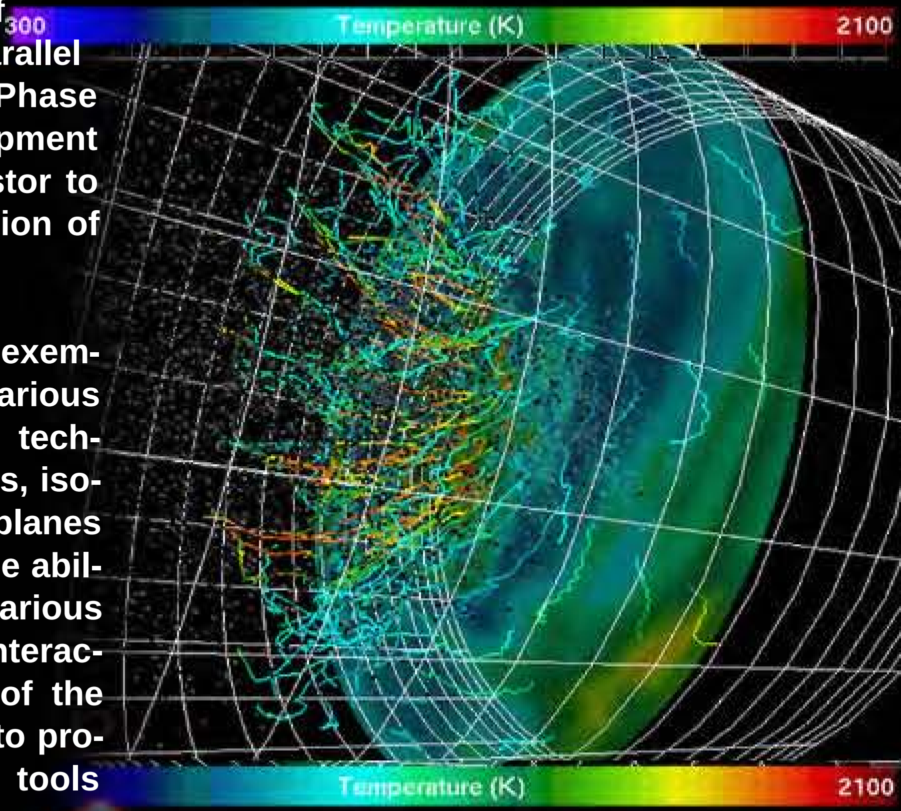


LARGE-EDDY SIMULATIONS OF IN A GAS TURB



While bandwidth is still the primary issue in remote rendering of this type, the strategy is attractive because it avoids costly file transfers of increasingly large datasets, which often are hundreds of gigabytes in size.

Dr. Suresh Menon and researchers at the Georgia Institute of Technology, in a project entitled "Parallel Simulations of Reacting Two-Phase Flows," are actively pursuing development of an intelligent gas turbine combustor to be used by the Army's next generation of helicopters and tanks.



The imagery shown on these pages exemplifies this type of work, showing various aspects of the project. Standard techniques including streaklines, particles, isosurfaces, and colormapped cutting planes are applied to represent the data. The ability to toggle (turn on and off) these various features is critical to providing an interactive environment. A primary goal of the NAVO MSRC Visualization Center is to provide remote researchers like Suresh tools to help decipher the complex dynamics of this extremely critical work.

SWIRLING SPRAY COMBUSTION
BINE COMBUSTOR

Highlights from "A Case Study of an Object-Oriented Parallelized Isosurfacing Algorithm"

Ludwig Goon, Logicon, Inc., and Sean Ziegeler, NAVO MSRC

The NAVO MSRC High Performance Computing (HPC) environment provides an opportunity for users to explore ways of constructing software to run on various types of hardware. Most of the "Big Iron" is multiprocessor oriented, with high memory capacities. However, where visualization is concerned, oftentimes an interactive solution is required. In some cases preprocessing data is necessary to ensure interactive exploration of large data volumes.

In the case of a ship hydrodynamics simulation, many time steps are given (130 at the time of the project), each being 64 Mb of three-dimensional scalar data. Concentrating on one scalar value, or threshold, in volume as the simulation progresses in time is another given. The solution is to use an isosurfacing method to produce the desired effect.

Isosurfaces are advantageous because they are generally constructed using polygons. Graphics hardware platforms use polygons and textures as performance benchmarks. Presciently, non-geometric volume rendering, such as splatting, is not ideal for this application due to lack of hardware support, placing the bulk of interactive transformations and manipulations on the CPU and software (figure 1).

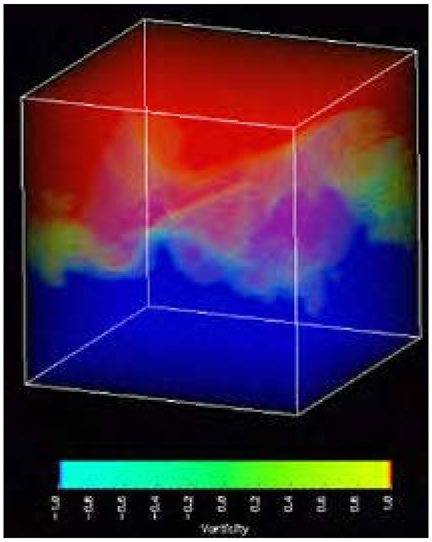


Figure 1. **Non-geometric volume-rendered time step of vorticity using Open DX.**

The Marching Cubes algorithm is a perfect selection for extracting isosurfaces and directly converting them to polygons. Originally developed for medical imaging applications, this technique is the de facto algorithm for many geometric isosurfacing applications. The principle involves taking volume data, dividing it into smaller adjoining "cubic" samples with scalar values at the vertices, and generating polygons that represent the isosurface in the sample. If any of the scalar values at the vertices are above and below the threshold, the iso-polygon(s) vertices are formed via linear interpolation along all applicable cube edges. The procedure is repeated by "marching" to the next cube (figure 2).

Object-oriented programming involves analyzing the problem and abstracting and modeling the elements that are solvable via computer. For instance, a cube is an object that contains edges, sides, and vertices; abstracting the necessary information, a "marching cube"(MCube) is constructed with vertices adding scalar and xyz point data.

The C Plus Plus (C++) programming language provides object constructs, called classes, where the marching cube is defined along with any necessary data allocation, member functions, and any other incorporated classes. Once the Mcubes are created, parallel methods of generating the isosurfaces are explored. This process is dependent on machine hardware and the availability of parallel programming environments.

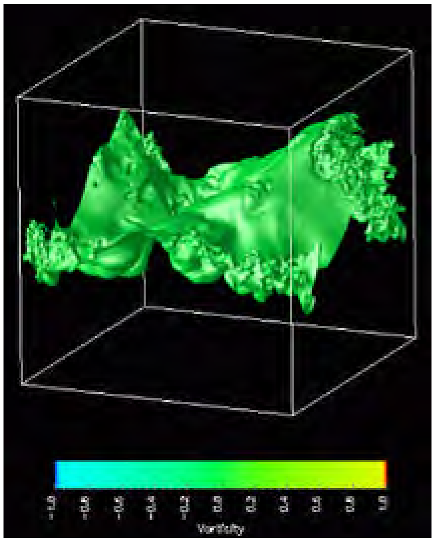


Figure 2. **Polygonal isosurface of vorticity at threshold value 0.0.**

To parallel process or not to?

The real challenge began when processing the hydrodynamics data became tedious due to transferring the data between the Cray T3E and a Silicon Graphics Onyx 2. Both systems have the Message Passing Interface (MPI) toolkits and a C++ compiler. The source code didn't refer to any machine-specific libraries or calls, so a port to the SGI proved successful with an exception. Running MCubes on distributed memory systems could result in data and memory exceptions (or allocation errors), because each processor has its own physical memory. Running MCubes on shared memory systems is better when allocating data because all processors have access to the entire physical memory.

In order to process each time step, the volume is split into layers according to data size and number of processors. The data volume is 257x256x256 values per time step. Given the operating environment of Cray computers, the system is divided into queues. In the most extreme case, depending on the machine, no more than 60 processors were used.

MPI is made to work on many types of multiprocessor computers, which are either heterogeneous or homogeneous. MPI uses either internal processor networks or network hardware to communicate with worker processors from a master processor. Mcubes uses the master processor to determine the amount of data to distribute, the number of worker processors to create, and allocation of the Mcubes to each processor.

More recent advances have expanded the project to include sockets, threads, and shared memory. Hardware platforms now include the SGI Origin 2000 and the Sun E 10000. The ability to incorporate MCubes in interactive applications and post-rendering applications is also included. Another interesting fact is that Mcubes is not tied to the traditional Euclidean 3D coordinate system; it is adaptable for rectilinear and nonuniform grids as well, since coordinate data are contained within each allocated cube.

Our main goal was to find out what parallel environment works best with what system. The MPI version, which ran across all platforms, proved to be stable on many systems, offering good performance. MPI does well on distributed systems such as the Cray T3E; however, sockets on the T3E did not offer any better performance running on one processor (figure 3).

More detailed results on system performance using the various parallel environments are in the forthcoming paper entitled "A Case Study of an Object-Oriented Parallelized Isosurfacing Algorithm."

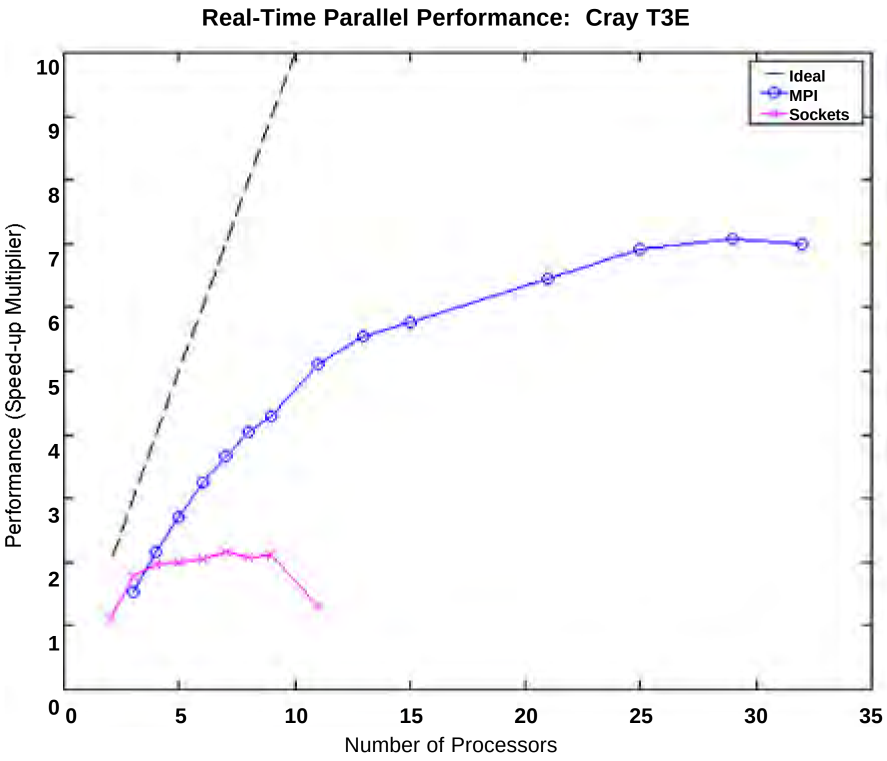


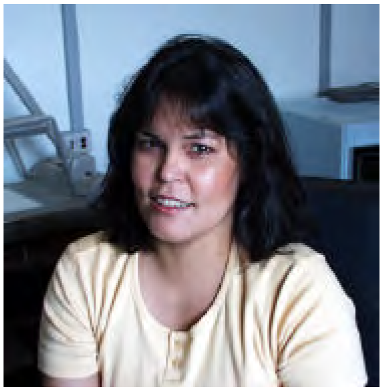
Figure 3. **Performance graph of Message Passing Interface and Sockets on Cray T3E.**

Acknowledgements:

The authors would like to thank Dave Cole of NAVO MSRC for all of the FORTRAN to C conversions from the Cray T3E to the SGI systems; Douglas Dommermuth of SAIC for providing Gigs of data to play with; Sheila Carbonette and the NAVO MSRC User Services staff for helping to understand Cray-isms and un-migrating data; and Pete Gruzinskas of NAVO MSRC for his uncanny support and enthusiasm for going to the threshold of visualization.

NAVO MSRC PET Update

Eleanor Schroeder, NAVO MSRC Program Environment and Training Program (PET) Government Lead



As we enter our fifth year in the Programming Environment and Training (PET) Program, we begin to look back at the accomplishments of our team of government personnel, integrators, and academia.

The NAVO MSRC PET followed a different model than the other three MSRC programs. As a result, our academic concentrations have primarily been focused on the general programming environment. While assisting our valued Computational Technology Area customers is of importance to us, we felt that we could obtain the most for our dollars by leveraging from those utilities and tools that were developed by our esteemed academic partners under other auspices. Hence we were able to develop tools such as Web-based Queue Stats which evolved from the National Partnership for Advanced Computational Infrastructure/San Diego Supercomputer Center (NPACI/SDSC) Hot Page project and the Resource Allocation Database and associated tools that spawned from work done under Northwest Alliance for Computational Science and Engineering (NACSE)/Oregon State auspices. We were able to fund the hardening of the University of Virginia's Legion project, which has become the prototype metacomputing model for the DoD High Performance Computing Modernization Program.

We look forward to our efforts in year 5 and believe that we will have some very exciting deliverables this year. We will also be continuing our Tiger Team efforts, expanding to include more academic partners as well as two excellent on-site senior analysts.

We know that the PET Program is undergoing some major revisions for year 6 and beyond. We embrace and welcome the changes that will be made. We look forward to the future of this program and continuing our work with current partners and beginning new and exciting work with additional partners.

So to borrow from a couple of well-used phrases, we've come a long way, baby, but the best is yet to come!

Signal and Image Processing Forum

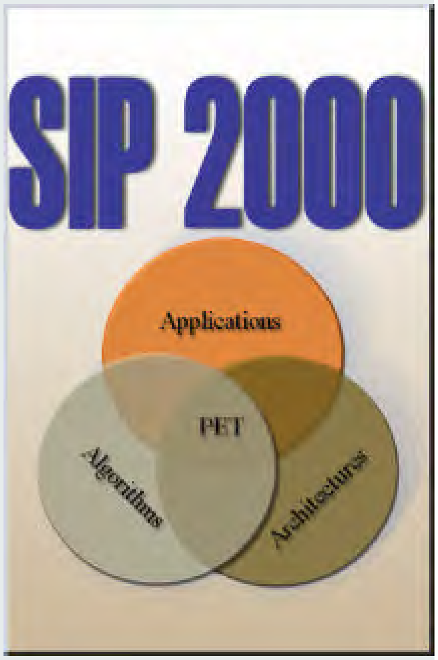
Dr. Bob Melnik, CTA Coordinator

The NAVO and Army Research Laboratory (ARL) MSRC PET programs recently co-sponsored the third annual Forum in Signal and Image Processing (SIP2000). The forum was held June 13-14 in Fairborn, Ohio, near Wright Patterson Air Force Base. The Aeronautical Systems Center (ASC) MSRC PET Program served as the local host of the meeting.

The SIP forums bring together a group of select SIP researchers with diverse expertise in order to identify critical areas of need for DoD SIP research. This year's forum provided the SIP community with another opportunity to identify critical SIP problem areas that could be the focus of DoD high performance computing (HPC) research and resources.

Fifty-two researchers and managers from the DoD SIP and MSRC communities attended this year's forum. Thirty-two papers on a variety of subjects were presented, including overviews of SIP technology trends, Common HPC Software Support Initiative (CHSSI) project status, ARL, ASC, and NAVO MSRC activities in SIP, as well as an overview of the HPC Modernization Program. Other papers were presented in sessions titled: Programming and System Technologies, SIP Processing Technologies, Enabling Technologies, SIP Applications, and Future Directions. Participants took advantage of the forum to have lively discussions in several open sessions that were arranged for this purpose. Get more details about the SIP2000 forum at:

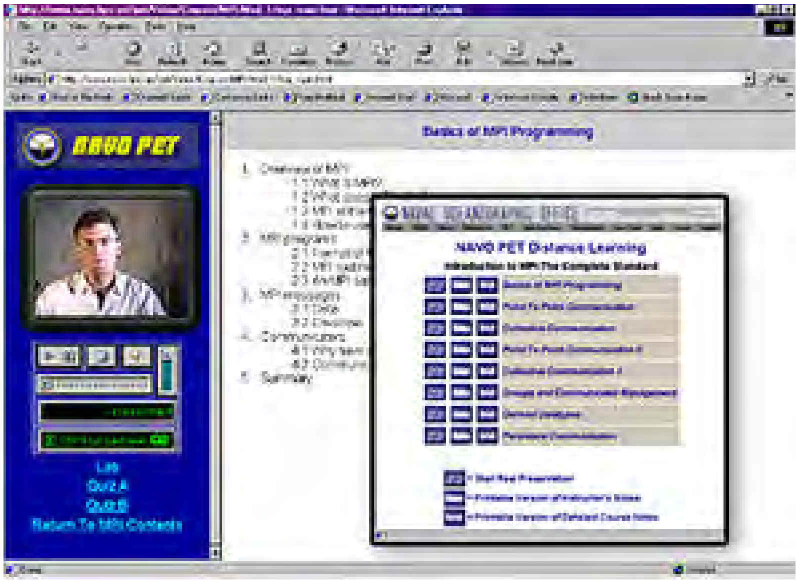
http://www.navo.hpc.mil/pet/sip2000/



Online, On Demand, On Your Desktop Training

Dr. Bob Melnik, CTA Coordinator, and Brian Tabor, Training Coordinator

The NAVO MSRC Program Environment and Training (PET) program is pleased to announce the availability of a distance learning program—a series of online courses in parallel program development. These online courses are directed at DoD MSRC users who either would like to make a transition from single CPU (serial) processing to multiprocessor (parallel) processing or would like to optimize pre-existing parallel code.



This distance learning program covers the multiprocessor programming styles associated with two current programming paradigms, MPI and OpenMP. MPI (Message Passing Interface) is a practical and flexible standard for developing portable and efficient message passing programs on distributed memory architectures. OpenMP is a portable and scalable thread-based interface that provides programmers with a simple and flexible parallel development tool for shared memory architectures.

The distance learning program will cover, as well, a hybrid MPI/OpenMP style of programming designed for Nonuniform Memory Access- (NUMA-) based architectures. NUMA-based architectures represent a collection of tightly coupled symmetric multiprocessors (SMPs)—a system of multiple processors, each of which can access common shared memory nodes. Every SMP node is connected to every other SMP node through high-bandwidth network interconnects.

Currently, the NAVO PET distance learning program in parallel programming includes the following courses:

- **Overview of Parallel Computing Hardware**
- **Overview of Parallel Computing Software**
- **Fortran 90**
- **Introduction to MPI for Finite Difference Models**
- **Introduction to OpenMP for Finite Difference Models**
- **Introduction to the Complete MPI Library**

These courses can be taken at your desktop at any time by going to the NAVO PET home page at **http://www.navo.hpc.mil/pet/**. To access the online courses you can use any desktop computer that has a web browser and an installed copy of the "Real Player G2" streaming media viewer. A free copy can be downloaded from a link to the Real Networks web site provided on the NAVO PET video library web page at **http://www.navo.hpc.mil/pet/Video/.** The lecture notes for the courses, in PDF format, can also be downloaded from that page.

Additional online courses on parallel programming are in development and are scheduled for completion by the end of the year:

- **Introduction to OpenMP—the Complete API**
- **Single CPU Optimization/Cache Management**
- **Introduction to Parallel Linear Algebra Solvers for Sparse Systems**
- **Introduction to IBM SP**

Traditional style classes are planned for the following:

- **Introduction to IBM SP (same as the above online course)**
- **Parallel Program Debugging and Performance Analysis and Optimization Tools**
- **Numerical Algorithms for Scalable Programming of Partial Differential Equations**

We also regularly provide traditional classes on parallel programming and other topics in high performance computing (HPC) at the NAVO PET training facility at Stennis Space Center. We can offer these classes at your site if there is a sufficient number of users taking the class.

NAVO PET is very interested in satisfying your training needs. If you desire training on any HPC subject, either online or live, please contact Brian Tabor at **taborb@navo.hpc.mil.** We would appreciate receiving any other feedback you might offer.

For more information, visit the NAVO PET web site: **http://www.navo.hpc.mil/pet/**.

The Porthole

A Look Inside NAVO MSRC

We welcome our visitors...

Right:
Mississippi Governor Ronnie Musgrove visits the MSRC Visualization Center with Dr. Don L. Durham, Technical/Deputy Director, COMNAVMETOCCOM, and RADM Kenneth Barbor, Commander, COMNAVMETOCCOM.

Below: Dave Cole, Computer Systems and Support, leads a group of science teachers on a tour of the MSRC facility.

Left:
Dr. Don L. Durham, MSRC Director Steve Adamec, and MSRC Deputy Director Terry Blanchard greet RADM Jay M. Cohen, Chief of Naval Research.

Right:
Terry Blanchard; Lieutenant General James King, Director, National Imagery and Mapping Agency; RADM Kenneth Barbor; and Steve Adamec.

Left:
Dr. Don L. Durham, Terry Blanchard, RADM Jon Greenert, and Mr. Gary Cohen of the Office of Budget, Dept. of the Navy, meet during a FY2000 CNO Midyear Review.

Right:
Tom Cuff, Deputy Technical Director, CNO and Terry Blanchard in the computer center.

Navigator Tools and Tips

Tips for Batch Jobs on Wolfe (Sun E10000)

The batch queuing system on Wolfe is handled by Platform Computing's Load Share Facility (LSF). A sample batch script follows. It should be changed to reflect your own needs on the system. However, certain options are necessary for LSF to run:

```
#BSUB -P AAA000
#BSUB -J Test1
#BSUB -e batch_csh.e%J
#BSUB -o batch_csh.o%J
#BSUB -n 8
cd /scr/$LOGIN
cp $HOME/mywork.dir/* .
# Compile F90 MPI program, myprog.f
f90 -fast -xarch=v9a -dalign myprog.f -o myprog.exe \
-I/opt/SUNWhpc/include -L/opt/SUNWhpc/lib \
-R/opt/SUNWhpc/lib -lmpi
# Run MPI program, myprog.exe
pam -n 8 ./myprog.exe
# Run a regular program
 ./cleanup
# END SCRIPT
```

Explanation of the options is as follows:

`#BSUB -P projectname`

This tells LSF which project should be charged for the runtime. You can find your project name by issuing the command **`groups $LOGIN`** on the system. Either this option should be explicitly set **`-P projectname`** or **`#BSUB -P projectname`**. If not, LSF environment variable in your login session needs to be set as follows:

```
csh% setenv LSB_DEFAULTPROJECT NA1234

ksh$ LSB_DEFAULTPROJECT=NA1234
ksh$ export LSB_DEFAULTPROJECT
```

`#BSUB -J jobname`

This option will name your job for the queue.

`#BSUB -e batch_csh.e%J`
`#BSUB -o batch_csh.o%J`

To get stderr/stdout files with jobid-related names, add this to your LSF batch script. **`%J`** must be added to your file name.

If you instead use the default or a file name without the **`%J`**, each LSF run will keep appending data to the same filename, rather than overwriting it. This can result in unwieldy and hard to read output files.

`#BSUB -n #procs`

This informs LSF how many processors you wish to run your job on. It overrides the **`pam`** directive for MPI jobs.

After you have created your script, submit it to the queue with the following syntax:

```
bsub < batch.csh
```

Use the following command to submit to the interactive batch queue, which is where MPI interactive jobs run:

```
bsub -I < batch.csh
```

LSF Commands of Interest

bsub	submit a job for batched execution (qsub)
bkill	send a signal to one or more unfinished batch jobs (qdel)
bpeek	display the stdout and stderr output of an unfinished batch job
bjobs	get information about batch jobs
bacct	report accounting statistics on completed batch jobs (qacct)
bhist	display the history of batch jobs
bhosts	get information about batch server hosts
bqueues	get information about batch queues

For more information on these commands please use the man pages or contact User Support.

Navigator Tools and Tips

IBM Math Libraries

The NAVO MSRC has recently procured a new IBM machine named **habu.navo.hpc.mil.** IBM has been a leader in high performance computing for many years. However, IBM systems are new to our site, and there may be many users who are unfamiliar with some of IBM's "built-in" math libraries and how those libraries relate with the ones our users may already be familiar with.

After logging into Habu for the first time, it might be disconcerting to see (for example, with the env command) that the expected LD_LIBRARY_PATH variable is missing! This is because the most common library paths are invoked at compile time. Of these, three groups are the most useful: ESSL, PESSL, and MASS.

The MASS group is actually three libraries. This family of thread-safe libraries may be used to speed up intrinsics like cos, sqrt, tan, etc. They may be called from either FORTRAN or C (but C only supports calls by reference).

There are three libraries of interest in this group. These include libmass.a (-lmass), which supports scalar calls, libmassv.a (-lmassv), which supports vector calls for any of the IBM SP family of processors, and libmassvp3.a (-lmassvp3), which supports vector calls tuned specifically for the POWER3 architecture. A test of the scalar cosine (cos) call sped up this call by almost a factor of two (1.91) by doing nothing but linking the library (-lmass) at compile time! A list of functions and other information may be found in the readme file. This file is located at **/usr/local/lib/MASS/MASS.readme** on Habu.

The second library group is ESSL (Engineering and Scientific Subroutine Libraries). This group contains subsets of the BLAS and LAPACK libraries as well as many others. There are two thread-safe libraries of interest in this group. If you plan to calculate a function on a single processor, the libessl.a (-lessl) should be used. If you wish to take advantage of multiple threads to calculate the function, the libesslsmp.a (-lesslsmp) library should be used, and the environment variable, XLSMPOPTS, should be set to declare the number of threads to be created for the calculation.

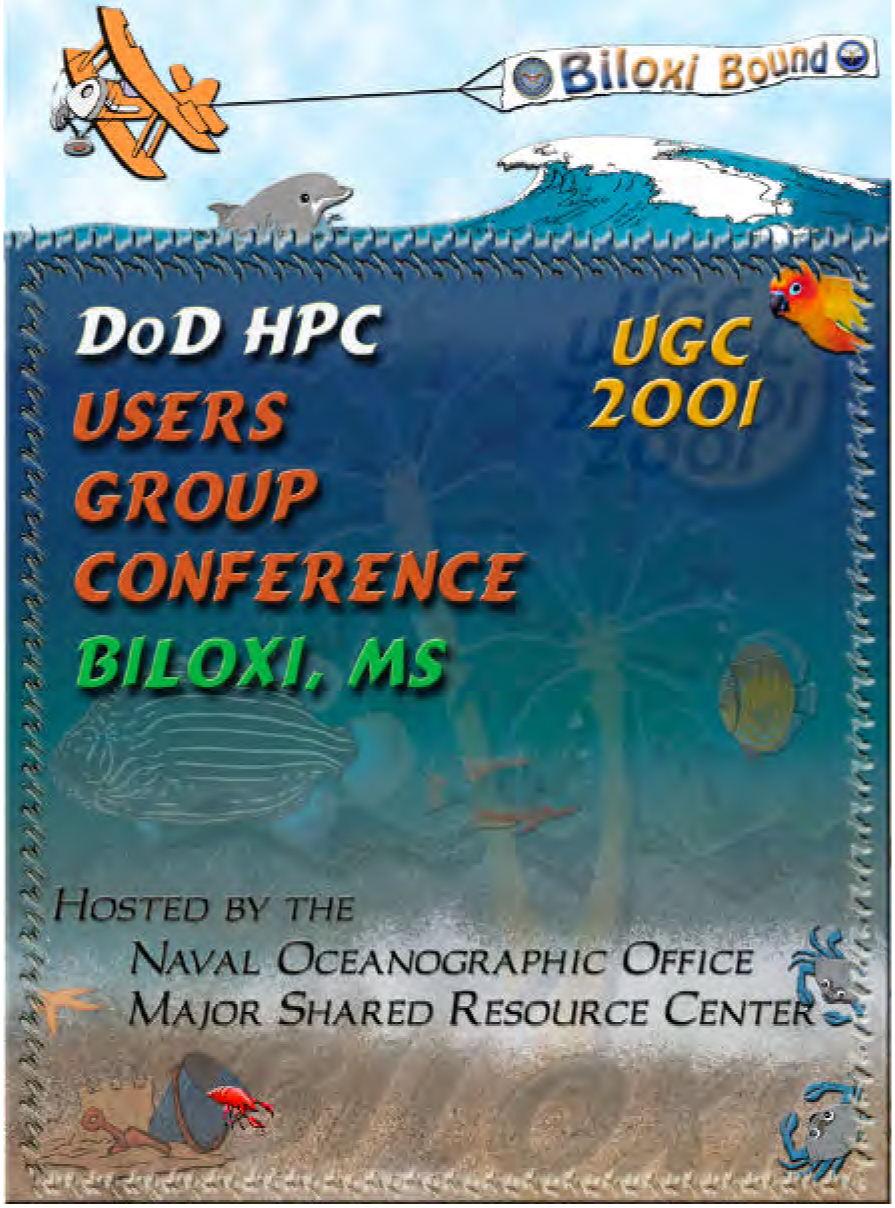
If you are using a generic BLAS (levels 1, 2, or 3), the call is straightforward, but ESSL does not support modified plane rotations. If you are used to calling a LAPACK driver routine, most of these calls do not exist. For example, a call to SGESV does not exist. However, the functionality of this call does exist.

From the site **http://www.netlib.org/** a search on SGESV will bring you to **http://www.netlib.org/lapack/single/sgesv.f**, which is the source code for theSGESV driver. This driver is actually only a simple subroutine containing an "if" statement (for error checking) and two calls to LAPACK computational subroutines, SGETRF and SGETRS. Both of these subroutines exist in the ESSL libraries and use the same inputs as the original SGESV call. Thus, SGESV can be successfully implemented in your code through ESSL. Other drivers may be invoked using the same procedure.

For those users who have codes which utilize PBLAS, BLACS, or ScaLAPACK subroutines, these subroutines may be found in the thread-safe PESSL libraries. They are structured much the same as the ESSL libraries with both a serial (-lpessl) and a multi-threaded (-pesslsmp) library for use. Once again, driver routines may not exist, but the actual computational subroutines may be accessed through the libraries. IBM offers a great deal of documentation on both ESSL and PESSL. This documentation includes a listing and description of each available library, as well as many sample programs.

They may be viewed in Adobe (.pdf) format at: **http://www.rs6000.ibm.com/resource/aix_resource/sp_books/.**

Upcoming Events



November 2000

28 Nov.-2 Dec.,**Cluster 2000,** IEEE International Conference on Cluster Computing, Chemnitz, Germany. Contact Rajkumar Buyya, rajkumar@csse.monash.edu.au. See **http://www.tu-chemnitz.de.cluster2000**

December 2000

17-20 Dec., **HiPC 2000** 7th International Conference on High Performance Computing, Bangalore, India. Contact Viktor K. Prasanna, University of Southern California, EEB 200C, Los Angeles, CA 90089-2562. See **http://www.hipc.org**

17-20 Dec., **GRID 2000,** International Workshop on Grid Computing (with HiPC 2000), Bangalore, India. Contact Rajkumar Buyya, rajkumar@csse.monash.edu.au. See **http://www.dgs.monash.edu.au/~rajkumar/Grid2000/**

January 2001

To Be Determined, **Sixth Grid Forum (GF6)** will be held in January 2001—details at **www.gridforum.org**

February 2001

7-9 Feb., **Network & Distributed System Security Symposium**, San Diego, CA. Contact Carla Rosenfeld, carla@isoc.org. See **http://www.isoc.org/ndss01**

March 2001

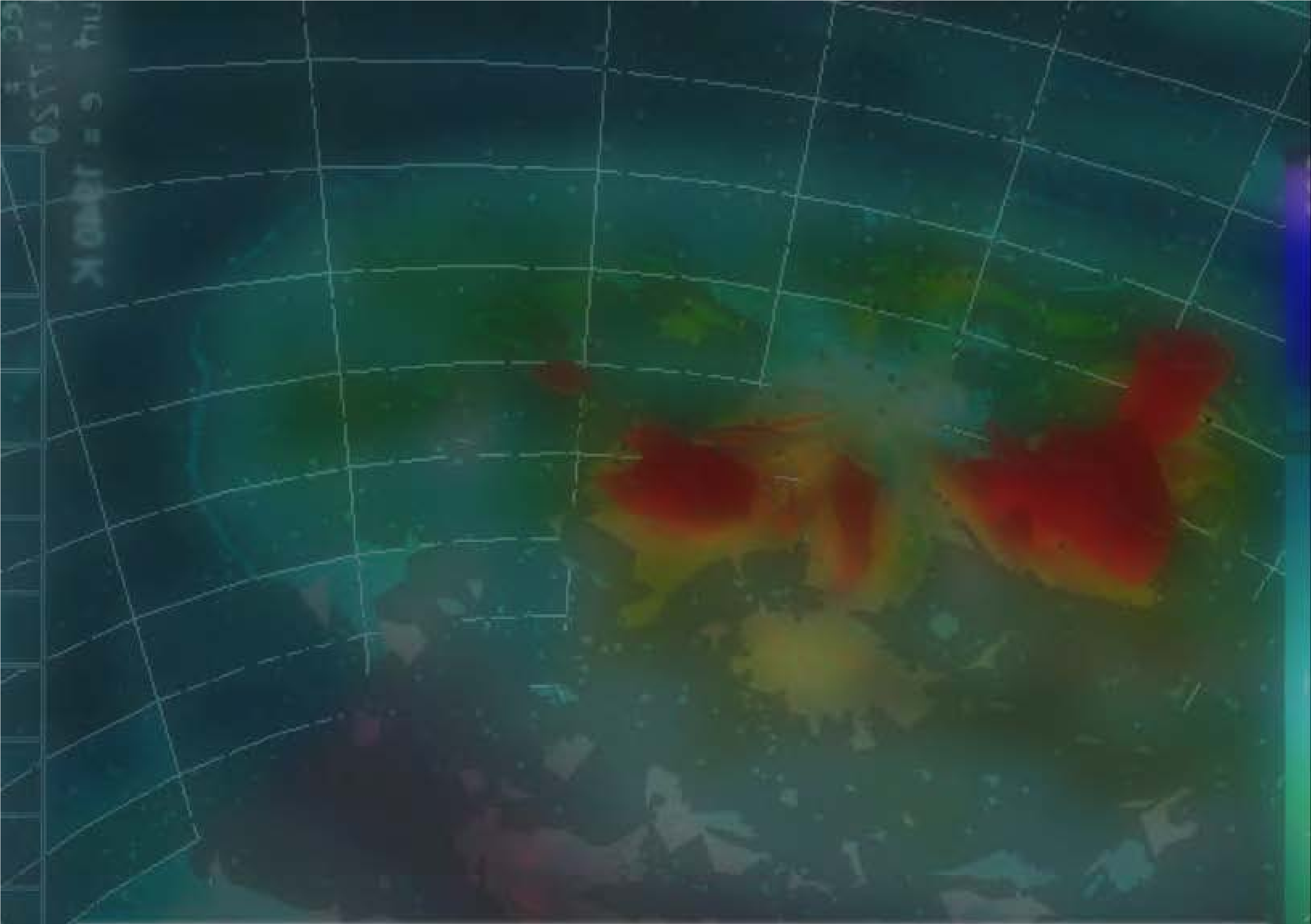
27-29 Mar., **High Performance Computing and Communications Conference (HPCCC),** Newport, Rhode Island —details at **www.hpcc-usa.org**

April 2001

16-19 Apr., **21st International Conference on Distributed Computing Systems (ICDCS 2001)**, Phoenix, AZ. Contact Forouzan Golshani, golshani@asu.edu. See **http://cactus.eas.asu.edu/ICDCS2001**

22-27 Apr., **15th International Parallel Processing Symposium & 12th Symposium on Parallel & Distributed Processing,** San Francisco, CA. Contact IEEE Computer Society, 1730 Massachusetts Ave. NW, Washington, D.C. 20036-1992

23 Apr., **HIPS 2001, 6th International Workshop on High-Level Parallel Programming Models & Supportive Environments**, San Francisco, CA. Contact Frank Mueller, Humboldt University, Berlin, Institut Für Infomatik, Unter den Linden 6, 10099 Berlin, Germany. See **http://www.informatik.hu-berlin.de/~mueller/hips01**



Naval Oceanographic Office * MAJOR SHARED RESOURCE CENTER

1002 Balch Boulevard . Stennis Space Center, Mississippi . 39522