September-October 1996
Vol. 2, No. 20

Langley-Ames SP2 Metacenter Offers Better Job Turnaround and Balanced Workload

by Mary Hultquist and James P. Jones

After three months of collecting background information and requirements, followed by four months of implementation, the new features of the SP2 metacenter are now available to users. The metacenter is currently made up of two systems, a 48-node IBM SP2 at NASA Langley Research Center (LaRC) and a 160-node SP2 at the NAS Facility, NASA Ames Research Center. It is now possible to utilize both systems from a single entry point, which will decrease the turnaround time for a user's job and balance the workload across the two systems.

Both testbeds operate under a Cooperative Research Agreement funded by the NASA High Performance Computing and Communications (HPCC) Program. (See the January-February issue of *NAS News* for background information.)

New Job Scheduler Runs at Both Sites

One of the requirements to make the metacenter operable was to rewrite the job scheduler, which interfaces with PBS (the Portable Batch System) to accommodate multiple systems. This was completed earlier in the summer by a joint team from NAS and LaRC, consisting of James Jones (NAS parallel systems group), and Ed Hook (LaRC, formerly of NAS), with assistance from Chris Batten (also LaRC). The scheduler, originally written in Tcl, was rewritten in C and combined into a single scheduler that runs independently at both sites.

A major change in the scheduler was the addition of "logic for peer scheduling." When a job is submitted to the local system, the scheduler determines whether the job can be run locally based on requirements such as number of nodes; if not, the job will be routed to the next larger machine. Since there are currently only two systems in the metacenter, a larger job would be sent to the NAS system. However, the scheduler is designed to include additional systems as they become available.

When one system has nodes available for user jobs but no jobs queued that can fit the availability, the

PBS scheduler will query the other PBS servers (its "peers") in the metacenter for jobs that can be moved and run on its system. Some jobs can be marked so that the server will not allow their removal from the local system.

This is done by specifying the "nodes attribute" for the site where the job will run. For example, instead of specifying:

`nodes=32`

the user would specify:

`nodes=32:larc`

to request that the job stays at LaRC or

`nodes=32:nas`

to keep it at NAS.

New Graphical Interface to PBS

To make job submission and tracking easier for users -- especially with the complexity of the metacenter -- Albeaus Bayucan, NAS PBS group, has written a graphical user interface (GUI) that can monitor any PBS queue on systems where the user has an account. This GUI, called xpbs (located in /usr/local/bin) was released with PBS 1.1.8 in early August.

The main screen of xpbs allows users to choose which systems to query through the "Preferences" button. After clicking "Update Data Manually" the server(s) will be displayed in the HOSTS section. Selecting any or all servers will list the queues in the QUEUES panel. Selecting the queue will then display job information based on the criteria chosen. In this way, users can easily find out the status of all their jobs on any of the metacenter systems.

Another useful option to xpbs allows users to submit a job for either batch or interactive use. Choosing the "Submit" button brings up a dialog window, which presents many of the standard PBS options for job submission. For interactive jobs, an "Interactive" button will create a window with the specified resources (for example, nodes or wall-clock time).

For more information on xpbs, see the online man pages or the WWW.

Staging Data For Effective Use

The "file staging" option is needed in cases where a user's job may be routed by the scheduler to another metacenter system. By selecting this option, another dialog window will appear with areas for specifying the hostname and files to stage-in and stage-out. Although it is possible to mark a job not to be moved elsewhere, users are encouraged to designate stage-in/stage-out of files to use the metacenter effectively.

HPCC Project Allocations

Although there are no current allocations for the LaRC SP2, named poseidon, node accounting is still being tracked. At this time, jobs that run on poseidon are not counted against the NAS allocation. However, poseidon will be a part of the allocation process for the new parallel operational year beginning October 1. Then, any jobs run on either system will be charged against the user's HPCC SP2 allocation. To find out about the availability and application process of SP2 computer time, contact Amy Lacer at the HPCC office, **lacer@nas.nasa.gov**, (415) 604-4498.

- Additional information on the metacenter project

The Virtual Skeleton: Modeling Human Movement with Supercomputers

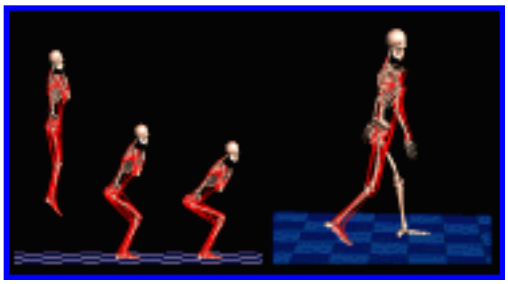
by Jarrett Cohen

Supported by a Guest Computational Investigator grant from the NASA High Performance Computing and Communications (HPCC) Program's Earth and Space Sciences Project and additional NASA awards, researchers at the University of Texas at Austin are using the NAS IBM SP2 and other supercomputers to model the human musculoskeletal system during movement on Earth and in space.

As with subatomic particles and cosmology, supercomputers are supplying insight into workings of the human musculoskeletal system that are otherwise impenetrable.

"Numerical simulation provides us with a way of estimating the forces developed by the muscles in the body. We could not do this in vivo," said Marcus Pandy, associate professor of kinesiology and mechanical engineering at the University of Texas at Austin. "If you knew the forces developed by the muscles, you would really understand how muscles coordinate limb movements."

Pandy and mechanical engineering graduate students Clay Anderson and Brian Garner combine optimal control theory and mathematical modeling to determine musculoskeletal forces during different activities. Optimal control involves finding the best way to achieve a task. For example, going as high as possible is the goal in jumping, while expending minimal energy is typical for walking.



Mathematical equations "represent the way bones move in relation to each other and the relationships between the forces in the muscles and movements in the bones," Pandy said. These dynamical equations of motion, which are vast in number, can be derived using Symbolic Dynamics' (Sunnyvale, CA) software package SD/Fast. The software for the rest of the modeling had to be developed in-house. Using these methods, Pandy's research team has been constructing three-dimensional models for studying vertical jumping, walking, rising from a chair, kicking, knee rehabilitation exercises, and, in collaboration with NASA Ames' Malcolm Cohen, various arm movements.

The Modeling Process

Modeling a physical activity begins with deciding on the number of body segments. Pandy said they have used as few as four (foot, shank, thigh, and upper body) and as many as eight. The joints come next, and how complex their movements are depends on the activity. In rising from a chair, the ankles, knees, hips, and feet are hinge joints -- they only move in one plane, much like a door. Joints in the walking model, the most intricate, together have 23 degrees of freedom. For instance, the hips move in three planes.

With a skeleton in place, the researcher then chooses the muscles to activate it. Limitations in computer power make it "infeasible to represent all of the muscles and simulate movement," Pandy stressed, so the team only models the major muscles that pull on bones. The walking model includes the largest number at 56.

The last step before running a simulation is selecting muscle parameters, most of which are in related literature or obtainable through experiments. For the latter, "we take a pool [of] people and measure their strength and such things as mass and moment of inertia," Anderson said. "We scale the parameters to be an average of the subject pool." One key parameter -- the lengths of the tendons -- cannot be measured, so they estimate them using the model, Pandy said.

A graphical interface developed by Garner furnishes a highly interactive solution process. "You can change the muscle activation levels...often and quickly...if something is going badly," such as when movement is grossly uncoordinated or unnatural, he explained. The researcher also runs the simulation code and produces visualizations from this interface. "The input is muscle activation levels," Garner explained. "The output is kinematic motions and joint angles at each instant in time," which are visualized in software based on the Silicon Graphics Inc. GL library.

Computational Implementation

The overall task is to "find how all of the muscles should be activated so that the model can produce an optimum, coordinated movement," Pandy said. "You need a mathematical algorithm to search the solution space in order to find the best possible solution. This is why we need supercomputers; the computation time for conducting the search is prohibitive."

Anderson explained further: "In optimal control problems, the first step is calculating derivatives of performance with respect to the controls." The second step is running a parameter optimization routine to produce an improved set of controls."

Since the equations must be integrated thousands of times, the derivatives are too time-consuming for serial computers, Anderson said. The problem is better suited to parallel systems, on which the integrations can be distributed across multiple processors.

SP2 Gives `Almost Ideal' Scaling

Beginning in 1990 as a co-investigator with Ames' Robert Whalen, Pandy was the first non-aeroscience researcher to have access to a NAS parallel supercomputer. A Guest Computational Investigator grant from the NASA High Performance Computing and Communications Program's Earth and Space Sciences Project, and other NASA awards, have funded parallel code development. Pandy's team has evaluated the Thinking Machines CM-5, the Intel iPSC/860, and the IBM SP2. They found the CM-5 impractical because it requires manually aligning many arrays of data (for example, the length and shortening velocity of a muscle) in the machine's memory. The iPSC/860 is not effective because its processors have too small a cache.

The IBM SP2 is a different matter altogether, Anderson said. The fast clock speed and large cache of its processors have enabled performance 15 to 20 times that of the CM-5 or iPSC/860, conservatively 2.5 gigaflops on 128 processors. "Our problem is ideally suited to MIMD [Multiple Instruction Multiple Data] parallel machines, to compute those derivatives," he added. "We get almost ideal scaling. Even with 150 processors on the SP2, we're seeing something like 80 to 90 percent ideal [scaling]."

Applying the Knowledge

The first three-dimensional model of its kind, a jumping run takes approximately 100 hours on eight SP2 processors. Simulating walking will require four times the computation. Anderson said that the primary reasons for this difference are that walking needs separate controls for the right and left sides of the body and that a step takes twice as long to perform in real time as a vertical jump.

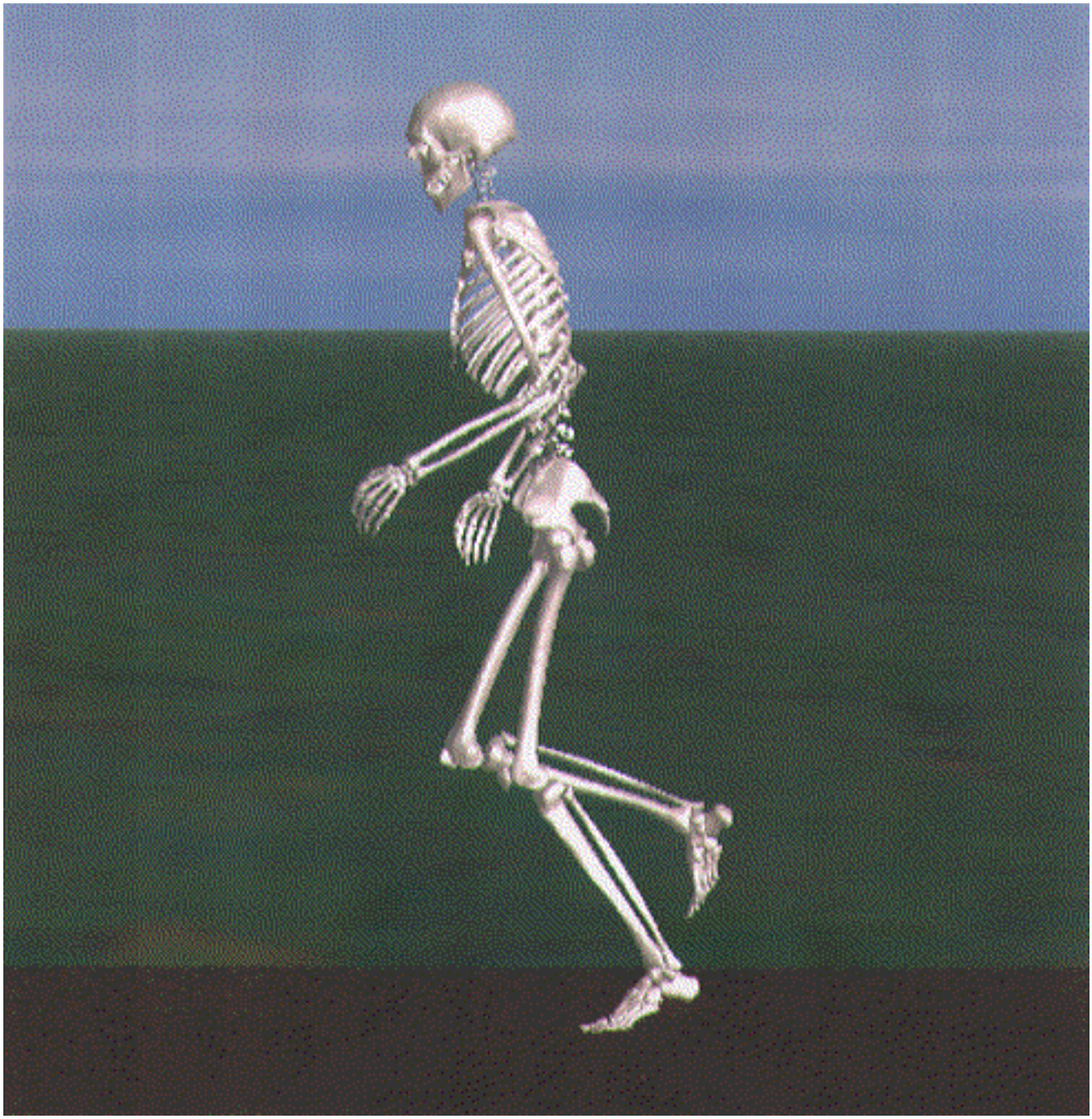
Whatever the activity, the computational models are used in conjunction with other research methods to gain as complete an understanding as possible. Before simulation, Garner said that they often videotape human subjects and then input the joint angles into simpler graphical models. The researchers also conduct cadaver studies for comparison. "This is a way of validating the model," Pandy said. "It is much more accurate, as we can do things that we can't do to live people -- such as inserting pins directly into the bones to more accurately measure movement in 3D."

"If you have a validated model, you can study a variety of things without doing experiments," Pandy emphasized. For example, by tweaking the muscle strength and gravity parameters his team has simulated jumping in space, where strength is depleted. "This capability is potentially very attractive, even more so from a rehabilitation point of view," Pandy said. "One could envision simulating surgeries, where the tendons are cut and relocated to compensate for musculoskeletal abnormalities."

For more information on this virtual skeleton work, send email to **pandy@mail.utexas.edu**.

Jarrett Cohen, a senior science writer for the NASA HPCC Earth and Space Sciences Project, is based at NASA Goddard Space Flight Center.

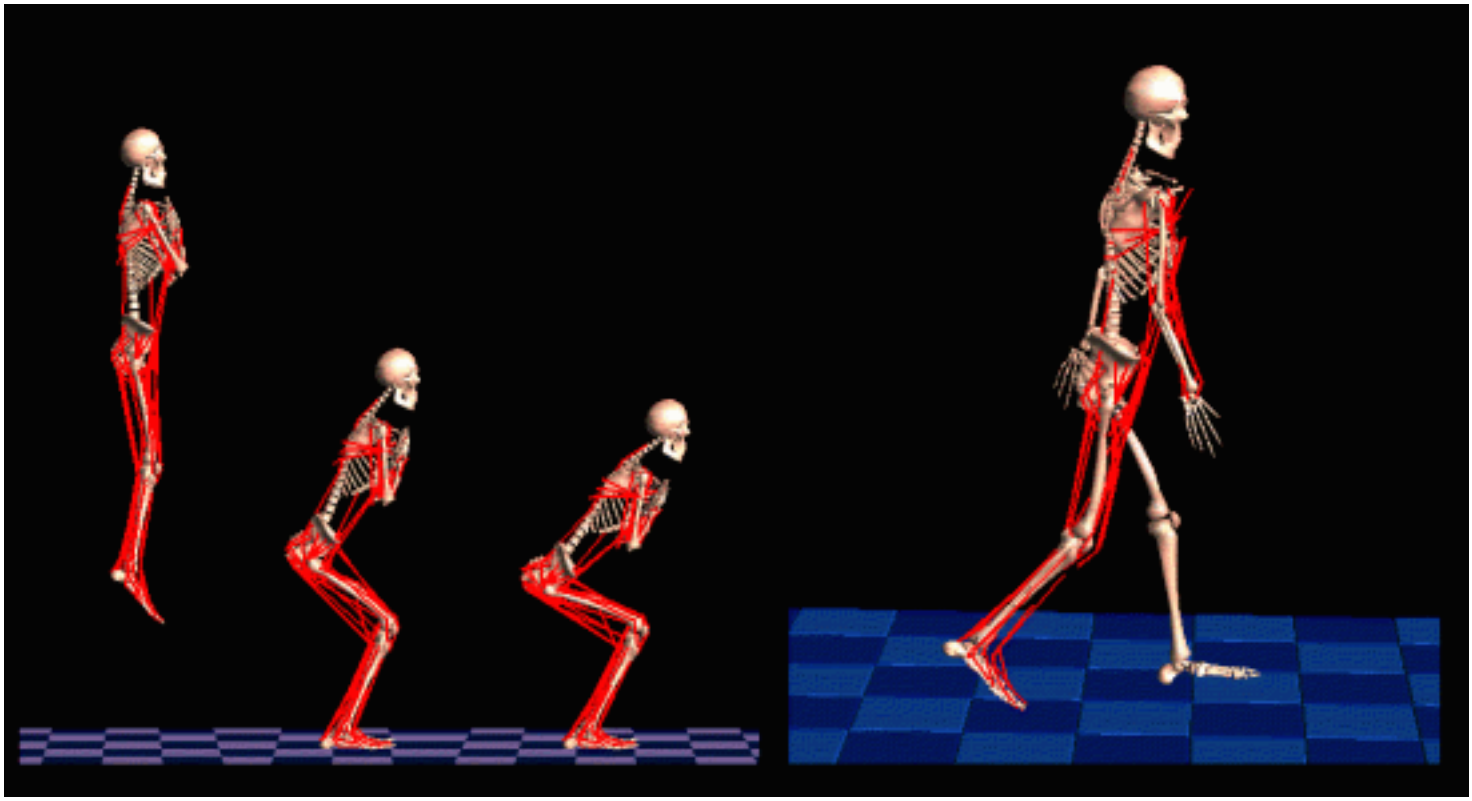
Next Article | Contents | Main Menu | NAS Home



Frames from an animation displaying a running motion recorded experimentally and visualized using bone data derived from a two-dimensional image set of the National Library of Medicine's Visible Human Project. Created for NBC's coverage of the 1996 Olympic Games, the animation depicts 49 stride cycles of running in real time.

Graphics courtesy of Marcus Pandy, NASA Goddard Space Flight Center.

Back to the article



These graphical models of jumping and walking incorporate joint angles from videotaped human subjects. Each muscle is represented by a three-element entity in series with tendons. A three-dimensional jumping supercomputer model, the first of its kind, consumes 800 CPU hours on an IBM SP2. A walking model, under development, will need approximately four times the computation.

Graphics courtesy of Marcus Pandy, NASA Goddard Space Flight Center.

Back to the article

First Research Projects Chosen for NAS CRAY J90

by Chuck Niggley

Nine projects have been selected for work to be performed on the CRAY J90 testbed system, newton. These projects, encompassing 14 applications, were selected to represent the first phase of newton users -- in keeping with the original plan to put a small number of projects (fewer than 10) on the system during the first year. Users began accessing the system in July.

A total of 17 proposals -- all noteworthy -- were received in response to a Request for Proposal released last May to current NAS CRAY C90 users and other selected customers. It is anticipated that other projects will be added to the system in early 1997, following some scheduled hardware upgrades and enhancements to the support software systems, including MPI (Message Passing Interface). Although some of the researchers who submitted proposals already had well-tuned parallel applications using MPI, it was decided that the system's performance was not adequate to support those users at present.

Much Preparation for New Users

Several important events have taken place since the newton system was announced (see the March-April issue of *NAS News* for background and configuration). The four systems were installed in late February and passed acceptance tests. New and updated software packages were installed, and various systems configurations were tested for I/O performance and ease of use for users. During this time, a version of the PBS scheduler was written for newton and tested. Cray Research Inc.'s beta version of the MPI software was also installed and tested. Various NPB (NAS Parallel Benchmarks) were ported and timing runs accomplished. In preparation for the new users, documentation was published on the World Wide Web (WWW).

Below are summaries of the nine projects selected, listing NASA locations or company names, and Principal Investigators (PIs).

Boeing Commercial Airplane Group

Jeffrey Lewis, PI. The NPARC Navier-Stokes code is a general purpose CFD (computational fluid dynamic) tool which is applicable to a wide variety of aerospace design and analysis problems involving fluid flow. It is actively supported by the NPARC Alliance, a partnership between NASA Lewis Research Center and the Arnold Engineering Development Center.

TLNS3D solves three-dimensional, time-dependent, thin-layer Navier-Stokes equations with a finite-volume formulation on structured grids. Expertise will be gained in applying and developing load-balancing tools to convert grid block topology that is defined by the geometric configuration to a grid block topology that will make efficient use of a desired number of processors.

Lockheed Martin Corp.

Erich Bender, PI. FALCON is a structured-grid, multiblock, finite-volume Navier-Stokes code for general-purpose analysis of aerodynamic and propulsion flow fields.

SPLITFLOW is an upwind, finite-volume, unstructured-grid CFD code with automatic grid generation and adaptation for the Euler analysis of flow over complex geometries. The purpose of this project will be to test the performance of these vectorized codes in a large-scale parallel environment and to improve the parallel efficiency and scalability of these codes on systems of this class.

McDonnell Douglas Aerospace

Samson Cheung, PI. CFL3D solves the three-dimensional, time-dependent, thin-layer Navier-Stokes equations with a finite-volume formulation on structured grids. Work on this code will be to implement a solution-restart capability. The overset grid and patched grid methodologies need to be tested. It also needs a three-level multigrid capability; at present, it has two levels. The benchmark results must match those of the serial counterpart.

Northrop Grumman Corp.

Kari Appa, PI. ENSAERO is a NASA-developed Euler and Navier-Stokes CFD computer program that is capable of handling multiblock grid models and flexible wing-body combinations. Aerodynamic analysis incorporating multiple maneuvers in a single computational run has never before been accomplished and, when implemented, will reduce the time required for an aircraft CFD analysis by hundreds of hours. The code must be optimized for multinode efficiency.

NASA Ames Research Center

Karen Gundy-Burlet, PI. At present, aircraft engine compressors are among the least efficient elements of an aircraft engine. This is due to the complex, unsteady, three-dimensional nature of the flow, coupled with the adverse pressure gradient inherent in compressors. A three-dimensional, unsteady, thin-layer Euler/Navier-Stokes zonal code (STAGE-3) has been developed to analyze these flows. This code will be implemented using the High Performance Fortran (HPF) compiler.

NASA Ames Research Center

Dennis Jespersen, PI. The OVERFLOW computer code is a widely used Navier-Stokes solver. The code handles complex geometries by allowing multiple zones with arbitrary overlapping, interpolating from one zone to another to provide appropriate boundary conditions. Splitting zones across nodes implies that the implicit solvers in OVERFLOW would have to be augmented by linear solvers that can span nodes.

NASA Goddard Space Flight Center

Jose Zero, PI. The proposed experiments are part of an international effort (organized under the World Climate Research Programme's Climate Variability and Predictability project) to assess predictability on seasonal time scales. Improved understanding and prediction of seasonal-to-interannual variability is a national priority of the U.S. Global Change Research Program and has recently been designated as a high priority for NASA's Mission to Planet Earth project.

The Data Assimilation Office (DAO) runs a full General Circulation Model (GCM) of the terrestrial atmosphere as part of its Data Assimilation System. Large parts of the GCM have been ported to a message-passing paradigm and are currently being tested in several High Performance Computing and Communications Program facilities.

The fluid dynamics module DYCORE is functional but its scaling characteristics are prone to degradation under latencies above 100 microseconds. DAO proposes to work in conjunction with the NAS staff to reduce existing high latencies between nodes and at the same time work toward a more robust DYCORE code that will depend less on network latencies.

NASA Langley Research Center (LaRC)

Christopher Riley, PI. LAURA (Langley Aerothermodynamic Upwind Relaxation Algorithm) has been the workhorse analysis tool for LaRC's Aerothermodynamics Branch for the past several years. At present, an elementary MPI version of the code exists and has been run on the IBM SP2 at LaRC.

The DPLUR (Data Parallel Lower Upper Relaxation) code was developed under a grant at the University of Minnesota. It was created specifically for parallel architectures and was developed on a Thinking Machines CM-5. The intent is for this code to eventually have the same functionality as the LAURA code.

FELISA (Finite Element Langley Imperial Swansea Ames) is a closely coupled, unstructured grid generator/flow solver. Although the flow solver codes function in a parallel environment, they have not been optimized for performance.

NASA Langley Research Center

Veer Vatsa, PI. The TNLS3D code will combine microtasking with message passing to attain a high level of scalability, where coarse-grained parallelization (across blocks) will be achieved across the nodes with

message passing, and finer grained parallelization (within blocks) will be achieved through microtasking.

Users Get Individualized Help

The NAS scientific consulting group is implementing a new support model for assisting newton users in porting and tuning their codes. User groups assigned to newton have a specific consultant designated to be their interface to all other NAS staff (such as the systems group), as necessary.

Planned enhancements to the newton cluster over the next few months include adding Cray Research Inc.'s GigaRing hardware to support faster communications among the four systems (Q1FY97). Several performance-enhanced MPI releases are expected during this period. In addition, the HSP and parallel systems staff are working on performance improvements to the system and developing utilities to help users; for example, the release of P2D2, the parallel debugger developed by the NAS applications and tools group.

- More information on newton, including CRAY J90 documentation.

Next Article | Contents | Main Menu | NAS Home

NAS Users Note: New Accounting Commands

by Elisabeth Wechsler

Effective October 1, the NAS Facility will complete its transition to a new accounting system for user allocations on the CRAY C90 (vonneumann), CRAY J90 (newton), and ACSF CRAY C90 (eagle).

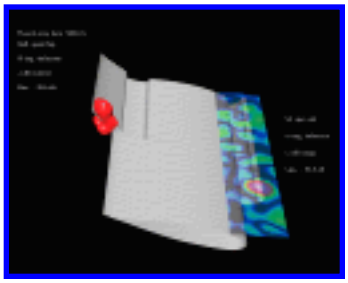
The new software is also scheduled to be operational on HPCC-funded systems at the NAS Facility -- the IBM SP2 (babbage) and the Silicon Graphics Inc. POWER CHALLENGE workstation cluster (davinci). However, the old pbsuse command will be retained through October for these testbed systems, according to James Jones, parallel systems engineer.

"We wanted to be able to store accounting data for all systems in one database with the same utility," said Chris Kleiber, systems control manager. Another reason was to switch from proprietary-based to architecture-independent software -- a program that was developed in-house by the NAS accounting group.

Beta testing was conducted last spring for six weeks. The 50 users -- mostly from the NAS Resource Monitors group at Ames, Langley, and Lewis Research Centers and NAS Users Group -- contributed feedback and suggestions for modifications that were incorporated into the software's first production version.

Users will need to use different command sequences, but other changes to the accounting system will be transparent, according to Kleiber. The former nasja command, which gave individual and group usage breakdowns, has been replaced by acct_query. The jaq command, which provides a summary line of total time allocated for an account vs. total time used, is now acct_ytd. For information about how to use the new software, users should consult the man pages for acct_query and acct_ytd.

Since April 2, both the old and new sets of commands have been offered to Cray users. However, on October 1, the nasja and jaq commands will no longer be available to users, Kleiber said.



NAS Team Helps Boeing Link Acoustical Data to IBM SP2 for Dynamic Wind Tunnel Tests

by Elisabeth Wechsler

Acoustic data from Boeing-NASA wind tunnel tests conducted last spring at Ames was churned out in one week by the NAS Facility's IBM SP2, thanks to hard work and network problem solving by a team from both groups.

"The quantity of data and processing required [was] quite large. The ability to handle [this type of test] in a timely manner is a major factor in making it practical," commented Boeing engineer Guy Neubert. "The NAS Facility and the people involved are a major part of making the concept viable."

The tests were conducted as part of the NASA Advanced Subsonic Initiative (AST), a project devoted to improving the technical base of aerospace industry. Specifically, the Boeing Commercial Airplane group is researching airframe noise that generated primarily by wings, flaps, and landing gear.

"We've learned a lot about the causes of airframe noise in the last few years," explained Bob Dougherty, the Boeing researcher who developed the technique used in the wind tunnel tests. Imaging technology can help pinpoint the sources of noise in two ways: a 2-meter diameter telescopic "dish" that reflects noise and a phased array of microphones, arranged to measure the sound source level relative to its position in the array.

Milliseconds of Difference

"The milliseconds of difference are significant," Dougherty said, adding that noise travels at a rate of one foot per millisecond. By knowing the time that a sound arrives at each microphone, one can compute the source location. Because neither the 12 ft. nor 7x10 ft. wind tunnels (used for the June test and the April-May tests, respectively) had acoustic lining on the walls, it is "impossible to make accurate measurements without spatially selective methods and the software to implement them," he said.

Neubert developed a C program using MPI (Message Passing Interface) to calculate time differences measured in the acoustic tests. This program was run on the NAS Facility's SP2, which was networked to a Silicon Graphics Inc. (SGI) POWER CHALLENGE workstation set up by a NAS team at the wind

tunnel. The computational time was six times faster than what Boeing had used in previous tests, according to Neubert.

"It was especially useful to run a lot of identical processes for each microphone. For example, with 100 microphones, there could easily be 5,000 time differences to compute, each for a different acoustical measurement such as angle of attack. Each calculation required 2.5 hours, and subsequent conditions often depended on the results of previously measured conditions," Neubert said.

After Neubert calculated the time differences, Dougherty determined the noise source location using beamforming techniques, which he described as "a comparison of each possible source location with data from Neubert's calculations to see if there was a high degree of agreement between the expected time delays by the microphones and the observed delays." The beamforming was also done on the SP2, using a Fortran code written by Dougherty.

Results Visualized in 3D

The results of Neubert's and Dougherty's calculations were visualized in the 3D grid of the aircraft model using FAST on the SGI workstation in the wind tunnel. Color contour plots of sound level vs. frequency were graphed for different parts of the model. In addition, grid files were generated that mimic the model's geometry, which were then overlaid and read into FAST.

The NAS team helped transfer the raw data files to the NAS Facility for processing on the SP2, the 160-node testbed funded by the High Performance Computing and Communications Program. The challenge was to use as fast a network as possible because of the acoustic data files' size -- 310 megabytes (MB) per condition. A FDDI network was used to connect the wind tunnels with the NAS Facility; however, a workaround was necessary to get the files to the SP2 because it couldn't accept the larger packets directly. In addition, Boeing wanted to simultaneously archive the data.

"We didn't have the full FDDI bandwidth -- the capability was not available from the SGI workstations to the SP2 -- so we sent the data to chuck [the Convex unit hosting NAStore] and then ran the data across a local HiPPI network to the SP2," said Leigh Ann Tanner, manager of the NAS parallel systems group.

As part of the DARWIN project (see the May-June issue of *NAS News*) the NAS wide area networks group has installed a high-speed FDDI connection between all of the Ames wind tunnels and NAS Facility resources. When necessary, access is extended to remote resources such as the testing company's home base, explained networks team member Jude George.

Making Tests `Dynamic'

For this set of tests, the main point of using a computer was to adjust the wind tunnel tests dynamically based on that day's results, said Steve Heistand, of the NAS scientific consulting group. He pointed out

that there wasn't enough time to convert Neubert's serial code to parallel code and debug it in order to make use of the SP2's parallel processing capabilities. Boeing planned to analyze the parallel aspect of these results later.

The goal for the NAS networks group was to "maximize throughput via existing networks," George said. Even though the FDDI network supports high-speed access, not all computer hosts can utilize this feature to its full potential, making extra work necessary.

He explained that problems arise when two hosts from different subnets are communicating. Subnets are divisions of a larger Internet Protocol network used to intentionally isolate traffic, enabling hosts in close logical proximity to communicate without sending their data to the entire local area network. George worked with Archemedes deGuzman and Lou Zechtzer, of the parallel systems group, to explore vendor options in various hosts -- especially the SGI workstation -- to facilitate the use of larger packets between subnets.

The data flow from the wind tunnel was serpentine: First collected by the Hewlett-Packard HP735 workstation (furnished by Boeing) from the wind tunnel acoustic sensors, the data flowed to the SGI workstation running Boeing's aero codes and FAST, then to chuck to kick in the HiPPI network, then to the SP2 for processing, and finally back to chuck for storage.

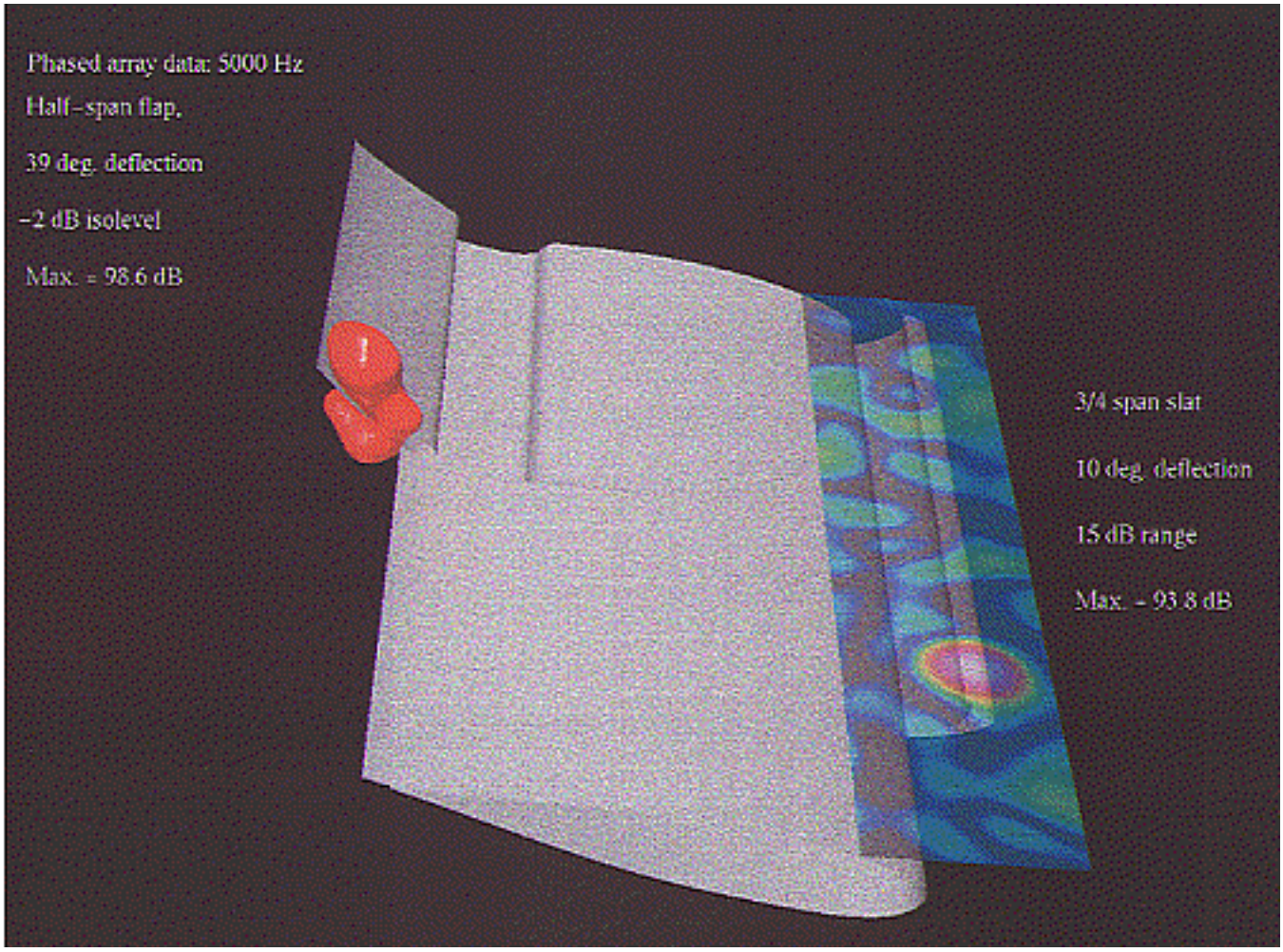
Optimizing the SGI workstation involved rebuilding the kernel with different parameters and implementing an option to reset the packet size on a subnet-by-subnet basis. The outcome was only partially achieved because the SGI workstation was able to send large packets but only to certain hosts -- the reason it didn't send directly to the SP2. The actual throughput to chuck was 3.5 MB/second, George said, adding that the rate could be improved "by determining the requirements of the test beforehand and setting aside time to redesign parts of the network to accommodate those requirements."

Valuable Lessons Learned

"One thing we learned from this test is, when we're trying to maximize performance we should probably install a new subnet to connect those hosts involved in the test," George said, estimating that this would take about two weeks. "It can be done as a hack job, but if the need for this occurs again, we'd like to do it right -- with some planning." He added that the parallel systems group was very effective in providing and configuring the SGI POWER CHALLENGE on such short notice.

The Boeing researchers were especially pleased with the SP2's performance. "It's the right architecture for Neubert's calculations," Dougherty said, adding that "this Boeing team plans to use the SP2 for all future tests -- we're addicted."

Next Article | Contents | Main Menu | NAS Home



This figure shows airframe noise generated by portions of a simplified model of an aircraft high lift system. In this case, the lateral edges of the half-span flap (at left) and the 3/4 slat (at right) are the principal noise sources. Two phased arrays of microphones flush-mounted in a wall and in an artificial floor of the 7x10 ft. wind tunnel at Ames Research Center were used for the measurements. The signals were processed on the IBM SP2 at the NAS Facility, and the isosurface plot was produced with FAST (Flow Analysis Software Toolkit). The image of the flap-edge noise source is an isolevel surface: the set points that are 2 decibels (dB) lower than the peak source strength. The two arrays were used together to define this surface. The contour plot of the slat noise (15 dB range) was made with the wall array alone.

Graphic courtesy of Boeing Commercial Airplane group.

to the article

Using MPI-IO for Fast Parallel I/O: A User's Perspective

by Rob F. Van der Wijngaart

Users of the Intel iPSC860 and Paragon systems formerly located at NAS will remember the special commands that were necessary to write large grid and solution arrays, spread over many processors, to a single file that could be read by a postprocessing tool such as FAST or PLOT3D. Unfortunately, when these systems were removed from the NAS Facility, the I/O modules containing these special commands became obsolete and a whole new coding effort was required for using the successor system, the IBM SP2, which also has a limited lifetime at the NAS Facility. (*The Cooperative Research Agreement ends in July 1997.*)

Just as MPI (Message Passing Interface) provided a comprehensive and successful standardization of existing message-passing systems on a variety of architectures, MPI-IO is now offering the means for extending the useful life of parallel I/O in application programs through a uniform user interface. This in itself is a significant reason for parallel systems users to adopt MPI-IO. In addition -- again following the example of MPI -- MPI-IO offers enhancements of existing parallel I/O libraries that can yield substantial program performance improvements.

When to Use MPI-IO

Applications that can take advantage of these improvements must be written using MPI, with which MPI-IO is tightly integrated. The greatest speed-ups by far are obtained when the original code uses multiple processors to write a single file in a non-trivial, interleaved manner. In CFD applications, this typically happens when several processors share computations on a single computational grid and are required to produce a single output file.

In more coarse-grained applications, such as the latest version of the flow solver OVERFLOW (see the related front-page article in the July-August issue of *NAS News*), each processor writes its own output file and little performance improvement is obtained through using MPI-IO -- though portability is still a plus.

Preparing to Use MPI-IO

Once you have identified I/O as the bottleneck of a parallel MPI application, there are several steps to take in preparing to use MPI-IO.

First, check with your system administrator to see if MPI-IO is available on your system and if you have access to its parallel file system. NAS users with accounts on the IBM SP2 can use the current pilot version of the library. The parallel file system, called piofs1, can be accessed from any node of the SP2, but not from the front end. User directories are created within piofs1, for example:

```
/piofs1/wijngaar
```

Next, change your makefile so that the MPI-IO libraries are linked with your program, and that the public-domain MPI compiler MPICH is used to compile the program. A sample compilation and link statement for the Fortran 77 program RANS-MP.f (using MPI-IO on the SP2 at NAS) follows:

```
mrf77.mpichRANS-MP.f -L/usr/local/lib/ pmpio -lmpio -I/usr/local/ include/pmpio
```

In addition, insert the include file mpiof.h in any Fortran subroutine that uses reserved words from the MPI-IO library, such as MPI_OFFSET_ZERO.

Finally, determine the structure of the file that you want to write, in terms of the processors that need to create it. Then, create the MPI derived datatypes that correspond to the layout. This is the most (and only) difficult step in using MPI-IO, although -- ironically -- derived datatypes are not new constructs within MPI-IO, but are part of the definition of MPI. Two derived datatypes are required for each processor: one to describe the layout within the program of the data to be written (called buftype), and one for the layout of the same data as it should appear in the file (called filetype).

Since each processor writes only part of the total dataset, each filetype will consist mainly of "holes" -- locations within the file where the current processor cannot write data but that are reserved for data from other processors.

Those locations within the filetype that are accessible by the current processor make up the pieces of the jigsaw puzzle that this processor contributes to the filetype. Usually, a single filetype will consist of many small pieces of the puzzle, even for very simple cases.

This is exactly the reason that parallel I/O is typically slow: all these small pieces must be written by different processors in an interleaved way. But because the definition of derived datatypes gives MPI-IO a global picture of how data on the different processors ends up in the file, it becomes possible to reshuffle the data among the processors before writing to disk, so that only large chunks are written by each processor. This process, invoked by the collective I/O command MPIO_WRITE_ALL, is completely transparent to the user.

Solving Datatype Problems

Creating derived datatypes to define filetypes and buftypes poses two problems. First, they rely on a single elementary datatype for each piece of information to be written, such as reals or integers. Mixtures are possible, but are harder to manage and are usually unnatural. However, most data files written for postprocessing purposes contain some header information besides the distributed-array data (such as grid dimensions in PLOT3D files), which is typically of a different type that does not fit the elementary datatype.

The solution is as follows: open the data file using a very simple filetype (for example MPI_BYTE); let one processor write the header; close the file, and then let all processors reopen it using the complex filetypes that describe the distributed array. To skip over a header, simply reopen the file with a certain initial "displacement."

The second problem is that it is difficult to determine the correctness of filetypes due to their swiss-cheese nature caused by the holes. For that purpose, several filetype constructors have been defined in MPI-IO in addition to the general derived-datatype constructors in MPI. The most useful is MPIO_TYPE_SUBARRAY, which can be employed to describe how any subblock of a grid owned by a processor can be embedded in the total grid that winds up on disk (see Sam Fineberg's article in the May-June issue of *NAS News*).

Still, manipulation of derived datatypes is a somewhat arcane business that may look unnatural to the uninitiated. Fortunately, relatively little code is needed to define very powerful datatypes.

Significant Increases in I/O Performance

The result of using MPI-IO is a binary data file. These types of files are characterized by a lack of structure, such as record information contained in Fortran unformatted files. The layout of the first MPI-IO data file you write will probably be wrong, and it may be hard to figure out where you erred. Perhaps the best debugging strategy is to write a very small file (small number of grid points) using only a few processors. Selectively let only one (different) processor write during each run and see where the data ends up in your file, for example by using the UNIX binary-file inspection utility "od" (Octal Dump).

Once the filetype is debugged, the read and write rates should experience increases of one to two orders of magnitude over "traditional" non-collective I/O for even moderate-sized problems and numbers of processors (see Parkson Wong's article in the July-August issue of *NAS News*).

For example, experiments with RANS-MP, a portable parallel Navier-Stokes solver, on a grid of about 200,000 points showed an I/O speed-up of a factor of 40 on 16 processors on the NAS Facility IBM SP2.

Caution: Be Prepared to Adapt

Parallel I/O has recently been adopted as a chapter in the next release of the MPI standard. Whereas the functionality of MPI-IO will likely be fully absorbed within MPI, the syntax will probably be slightly different, so early users should be prepared to do some editing of their MPI-IO calls.

For more information:

- Contact ACSF/NAS User Services
- Technical specifications for MPI-IO

Next Article | Contents | Main Menu | NAS Home

New Tutorial Format Takes Advantage of Web Features

by George B. Myers

The NAS scientific consultants have developed a World Wide Web-based online tutorial format for class presentations and reference. The group has also begun to use this format to provide the primary information for on-site classes.

The first class that used this method was held in mid-April to prepare users for converting their codes from NQS (Network Queuing System) to PBS (Portable Batch System) on the CRAY C90 systems at the NAS and Aeronautics Consolidated Supercomputing Facilities. The many positive comments received lead the consultants to establish this format as the standard for all their NAS classes.

In class presentations, the instructor can also make use of live demonstrations of commands running in other windows. However, the tutorials provide the core of the information presented and allow users to link to additional information -- such as a page with the output from an example -- at their own discretion.

Works Best With Netscape 2

The tutorials are designed to work with Netscape version 2 or higher, in order to take advantage of "frames." The frames feature gives the instructor the ability to divide the window into independent "panes," which can be used to view different documents simultaneously. The accompanying graphic shows a sample tutorial format.

Using Netscape version 2 or higher gives users two ways to navigate tutorials: viewing every page using the arrows to move through the material in the order intended by the instructor, or choosing their own path by selecting topics from the table of contents.

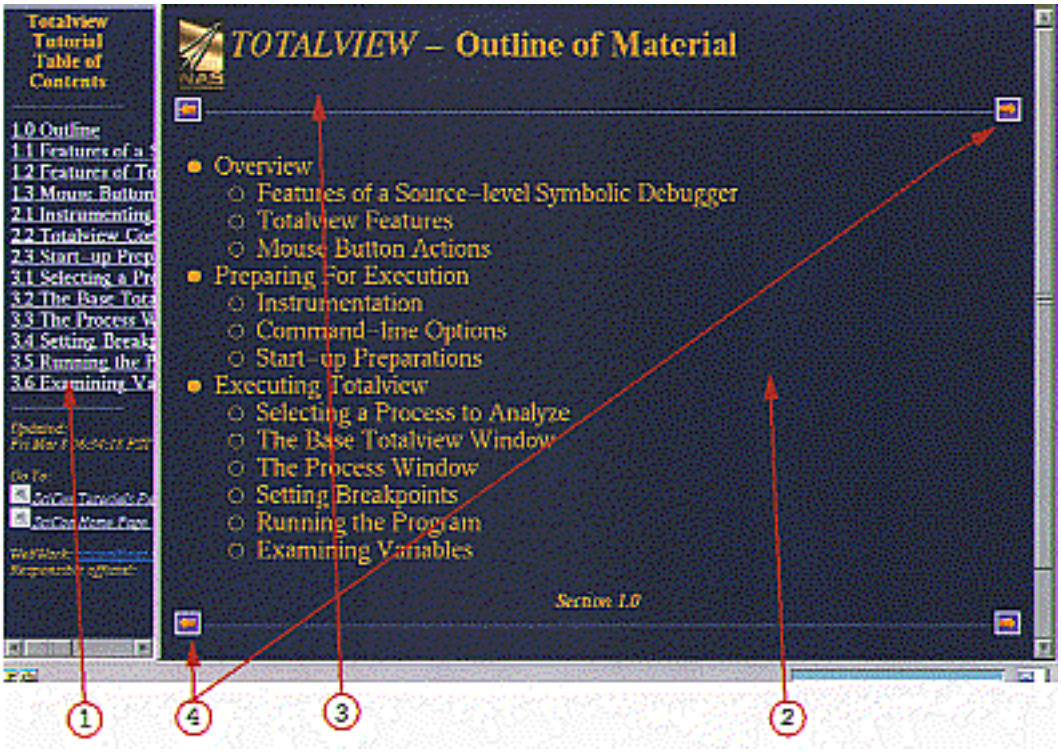
If Netscape 2 is not available, the tutorial will still work, but the table of contents will only be available from the first page. In addition, there will only be one pane, and the look may vary depending on the browser. The forward/backward arrows will work the same as in Netscape 2. The tutorials have been tested using Mosaic, and Netscape 1 and 2.

More Tutorials, Classes Coming

As more tutorials are developed, they will be announced on the Scientific Consulting home page. All new-user classes for the 1996-97 Operational Period, scheduled for October, are being modified to take advantage of the new format.

For more information, send email to **gmyers@nas.nasa.gov**.

Next Article | Contents | Main Menu | NAS Home



The viewing area, or window, of the Netscape 2 browser is divided into two "panes." The left pane (1) contains a table of contents into the tutorial, with links to those topics. Clicking on a topic here causes the right pane (2) to display the actual tutorial information, which may have links to additional information on that topic, or link to later topics in the tutorial. Other features shown in this pane are the tutorial title (in italics), the specific topic (3), and upper and lowers sets of forward/backward arrows (4).

Back to the article

Readers Speak Out: Results of *NAS News* Survey

by *Elisabeth Wechsler*

NAS News readers had a chance to sound off when they received a questionnaire in the March-April issue of the newsletter. While there were some creative suggestions for improvement and some dislikes that came through loud and clear, most of those who responded to the survey gave the newsletter a favorable review. About 95 percent rated the publication "excellent" or "good," both overall and in terms of readability.

The front-page, general interest article announcing the CRAY J90 cluster at the NAS Facility received the highest overall rating, with 70 percent judging it "very useful" or "useful." The next highest rating went to the High-speed Processor Techniques column, "Getting Familiar with PBS on the CRAY C90s": 58 percent of respondents found this technical article relevant to their work.

Color graphics and overall publication design were also given high approval ratings ("excellent" or "good") by 98 and 94 percent, respectively. Captions were rated "excellent" or "good" by 97 percent, although there was a comment that, "sometimes captions aren't descriptive enough."

`Too Much Hype'?

The mix of technical vs. general articles was rated "excellent" or "good" by 95 percent of respondents. However, a few readers requested increasing the number of technical articles, and some objected to what was termed "too much hype" or "very political marketing." Another cautioned, "Keep it non-political to maintain credibility."

The sharpest criticism was directed at the publication's 11 x 17-inch format. Although only one-fifth ranked the tabloid-style format as "fair" or "poor," more than 30 respondents took extra time to state explicitly their preference for an 8 x 11-inch format. Some of the reasons given: "11 x 17 is hard to copy...file...read on airplane...too big for desk...unwieldy...unattractive...prefer magazine size..."

Some Surprises

The editorial staff was surprised by the low percentage of readers who use the online version of *NAS News*. About two-thirds of those who submitted surveys by mail said they "never" or "rarely" look at

NAS News online.

Most respondents who do use the online version noted its convenience for archiving or reference. One person suggested "advertising the online version's location on the front page" of the print version -- a change that's been made in this issue. Another requested "email notification" when a new issue is published online.

Another somewhat unexpected finding of the survey was that only about 40 percent of respondents indicated that *NAS News* had helped them make contact with others doing related work. However, one reader noted that *NAS News* "provides easy access to email addresses of folks mentioned in articles," presumably facilitating follow-up contact.

The "News Bites" section (a variety of short topics on the back page) was rated "very useful" or "useful" by more than half the respondents. One reader requested more of the same: "shorter articles about many things."

More `Technical' Articles

Somewhat predictably, the "usefulness" rating of specific articles that appeared in the March-April issue seemed to depend on which NAS systems respondents currently use. For example, NAS CRAY C90 users, who comprised about half the respondents, rated articles about the C90 higher than those featuring other NAS systems.

Several readers stated their preference for more full-length technical articles and offered these specific topics:

- "case studies of code improvement for both RISC and vector architecture"
- "improving performance for users who aren't computer scientists"
- "more on combustion flows"
- "more on HPC [High Performance Computing] instead of just visualization and CFD"

Rating `The Competition'

NAS News was rated "better" than four other U.S. supercomputer center publications by an average of 46 percent of respondents and "worse" by 4 percent; the remaining half rated *NAS News* "about the same" as *Access*, National Center For Supercomputing Applications; *Buffer*, Lawrence-Livermore National Laboratory; *Gather-Scatter*, San Diego Supercomputer Center; and *PSC News*, Pittsburgh Supercomputing Center.) One reader commented: "Compared with other newsletters, *NAS News* is

outstanding."

Reader Profile Emerges

The "typical" *NAS News* reader, according to the survey, has a Ph.D. (59 percent), a NAS CRAY C90 account (49 percent), and works in technical development (49 percent) at a NASA center (29 percent), or in the aeronautics industry (26 percent), or at a university (23 percent). Aerodynamics and fluid dynamics were the most frequently checked fields of research (64 percent).

Almost one-third of all respondents work in the San Francisco Bay Area and about one-fourth listed the Ames Research Center ZIP code (94035). Otherwise, responses represented all parts of the continental U.S. Nine respondents noted foreign ZIP codes.

A sampling of additional opinions from readers includes the following -- sometimes conflicting -- comments and suggestions:

- "Spotlight remote users."
- "Keep providing state-of-the-art information."
- "More space science articles."
- "Fewer shuttle topics, more aircraft."
- "Info on NASA-funded industry research projects would be useful."
- "It's great as it is! Keep up the good work."

Almost 16 percent of *NAS News* readers completed questionnaires for the survey, distributed in the March-April issue. (Polling experts consider an 8 percent or better response rate statistically reliable.) Responses were submitted both by mail and online form.

The Buck Stops Here

In the coming weeks, detailed survey results -- including recommendations for changes -- will be presented to NAS management, and then published on the World Wide Web. Focus groups may be formed to gather more information about particular issues or recommendations. This issue of *NAS News* incorporates two other suggestions: "focus on other NASA centers besides Ames," with an article featuring work done at Goddard Space Flight Center; and "more articles on NAS working with outside companies (non-universities)" in the "Boeing acoustical data" article. Future issues will continue to

reflect changes as a result of these ideas.

The *NAS News* staff is always interested in readers' feedback. Fill out the online survey or send comments and suggestions at any time to **nasnews@nas.nasa.gov**, attention: Editor.

Next Article | Contents | Main Menu | NAS Home

Next Article | Contents | Main Menu | NAS Home

NAS Staff Participates at SIGGRAPH

Several NAS staff members made presentations at SIGGRAPH, held August 4-9 in New Orleans, where more than 28,000 people explored state-of-the-art technology. SIGGRAPH is the largest international conference in computer graphics and interactive techniques.

Samuel P. Uselton (data analysis group) organized the panel, "Graphics PCs Will Put Workstation Graphics in the Smithsonian." Conference press releases featured Uselton's panel as a highlight; attendance was about 1700.

James D. McCabe, (wide area networking group) spoke on the panel "Issues in Networking for Entertainment, Graphics, and Data." His topic was "Issues in LIS Consolidation."

Pamela P. Walatka (customer communications group) presented a Technical Sketch, "WebToons: A Method for Organizing and Humanizing Web Documents."

The Computer Animation Festival in the Festival Screening Rooms, included a video by Michael Gerald-Yamasaki, Sandy Johan, David Kenwright, Vee Hirsch, David Lane Kau, Gail Felchle: "Visualizing Time-Dependent Particle Tracing for the V-22 Tiltrotor Aircraft."

Conference Chair John Fujii said, "SIGGRAPH has expanded online activity to make participating and contributing even more dynamic. This process will...create new connections within the vibrant community of cultures and technologies that is the SIGGRAPH phenomenon.

``Connectedness" was a key element throughout the conference, which was linked to the World Wide Web.

Next Article | Contents | Main Menu | NAS Home

September-October 1996
Vol. 2, No. 20

Executive Editor: Marisa Chancellor

Editor: Jill Dunbar

Senior Writer: Elisabeth Wechsler

Contributing Writers: Jarrett Cohen, Mary Hultquist, James P. Jones, George B. Myers, Chuck Niggley, Rob F. Van der Wijngaart

Image Coordinator: Chris Gong

Other Contributors: Clay Anderson, Joel Antipuesto, Chris Buchanan, Mi Young Cho, Bob Ciotti, James Donald, Sam Fineberg, Brian Garner, Steve Heistand, Chris Kleiber, Terry Nelson, Bill Nitzberg, R.K. Owen, Marcus Pandy, Marcia Redmond, Bill Saphir, Leigh Ann Tanner, Dani Thompson, Dave Tweten, Parkson Wong

Editorial Board: Nick Cardo, Marisa Chancellor, Jill Dunbar, Chris Gong, Mary Hultquist, David Lane, Chuck Niggley, Elisabeth Wechsler

NAS NEWS
www.nas.nasa.gov/NAS/NASnews
Volume 2, Number 20 September – October 1996

The Virtual Skeleton: Modeling Human Movement with Supercomputers

by Jarrett Cohen

Supported by a Grand Computational Investigator grant from the NASA High Performance Computing and Communications (HPCC) Program's Earth and Space Sciences Project and additional NASA awards, researchers at the University of Texas at Austin are using the NAS IBM SP2 and other supercomputers to model the human musculoskeletal system during movement on Earth and in space.

As with subatomic particles and cosmology, supercomputers are supplying insight into workings of the human musculoskeletal system that are otherwise impenetrable.

"Numerical simulation provides us with a way of estimating the forces developed by the muscles in the body. We could not do this before," said Marcus Pandy, associate professor of kinesiology and biomedical engineering at the University of Texas at Austin. "If you know the forces developed by the muscles, you would really understand how muscles coordinate limb movements."

Pandy and mechanical engineering graduate students Clay Anderson and Brian Garner combine optimal control theory and mathematical modeling to determine musculoskeletal forces during different activities. Optimal control involves finding the best way to achieve a task. For example, going as high as possible is the goal in jumping, while expending minimal energy is typical for walking.

Mathematical equations "represent the way bones move in relation to each other and the relationships between the forces in the muscles and movement in the joints," Pandy said. These dynamical equations of motion, which are vast in number, can be derived using Symbolic Dynamics' Autolev package, SD/Fast. The software for the rest of the modeling had to be developed in-house. Using these methods, Pandy's research team has been constructing three-

Continued on page 2

Frames from an animation displaying a running motion sequence experimentally and visualized using bone data derived from a two-dimensional image set of the National Library of Medicine's Visible Human Project. Created for NBC's coverage of the 1996 Olympic Games, the animation depicts a full stride cycle of running in real time. Graphic courtesy of Marcus Pandy.

Langley–Ames SP2 Metacenter Offers Better Job Turnaround and Balanced Workload

by Mary Hultquist and James P. Jones

After three months of collecting background information and requirements, followed by two months of implementation, the new features of the SP2 metacenter are now available to users. The metacenter is currently made up of two systems, a 48-node IBM SP2 at NASA Langley Research Center (LaRC) and a 160-node SP2 at the NAS facility, NASA Ames Research Center. It is now possible to utilize both systems from a single entry point, which will decrease the turnaround time for a user's job and balance the workload across the two systems.

Both facilities operate under a Cooperative Research Agreement funded by the NASA High Performance Computing and Communications (HPCC) Program. (*See the January–February issue of NAS News for background information.*)

New Job Scheduler Runs at Both Sites

One of the requirements is to make the metacenter operable was to rewrite the job scheduler, which interfaces with PBS (the Portable Batch System) to accommodate multiple systems. This was completed earlier in the summer by a joint team from NAS and LaRC, consisting of James Jones (NAS parallel systems group), and Ed Hook (LaRC, formerly at NAS), with assistance from Chris Rutten (also LaRC). The scheduler, originally written in Tcl, was rewritten in C and combined into a single scheduler that runs independently at both sites.

A major change in the scheduler was the addition of "logic for peer scheduling." When a job is submitted to the local system, the scheduler determines whether the job can be run locally based on requirements such as number of nodes. If not, the job will be routed to the next larger machine. Since there are currently only two systems in the metacenter, a larger job would be sent to the NAS system. However, the scheduler is designed to include additional systems as they become available.

When one system has nodes available for user jobs but no jobs queued that can fit the availability, the PBS schedulers will query the other PBS servers (its "peers") in the metacenter for jobs that can be moved and run on its system. Some jobs can be marked so that the server will not allow their removal from the local system.

This is done by specifying the "nodes attribute" for the site where the job will run. For example, instead of specifying

```
nodes=32
```

the user would specify

```
nodes=32:Ames
```

Continued on page 6

THIS ISSUE