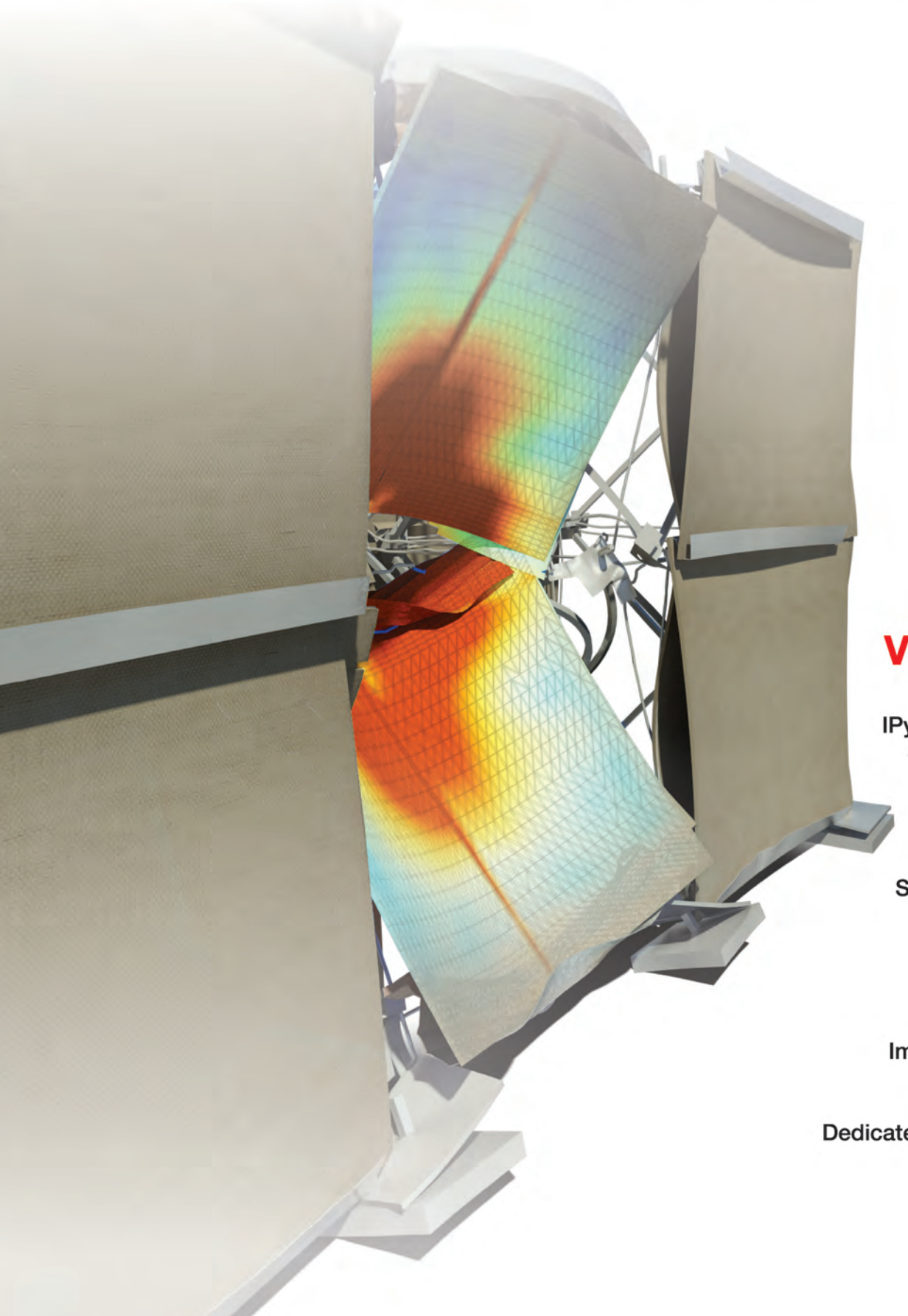
Spring 2012

DOD HPC INSIGHTS

A publication of the Department of Defense High Performance Computing Modernization Program



Data Visualization

IPython for Web-Based Scientific Computing

Using Machine Learning Software

Systems Calls Getting You Down?

Client-Server HPC Job Launching

Improving Multisystem Workflow

Dedicated Support Partitions

HPC Insights is a semiannual publication of the Department of Defense Supercomputing Resource Centers under the auspices of the High Performance Computing Modernization Program.

Publication Team

AFRL DSRC, Wright-Patterson Air Force Base, OH
Gregg Anderson
Chuck Abruzzino

ARL DSRC, Aberdeen Proving Ground, MD
Debbie Thompson
Brian Simmonds

ERDC DSRC, Vicksburg, MS
Rose J. Dykes

MHPCC DSRC, Maui, HI
Betty Duncan

Navy DSRC, Stennis Space Center, MS
Christine Cuicchi
Lynn Yott

HPCMPO, Lorton, VA
Deborah Schwartz
Denise O'Donnell
Leah Glick

MANAGING EDITOR
Rose J. Dykes, ERDC DSRC

DESIGN/LAYOUT
Betty Watson, ACE-IT

COVER DESIGN
Data Analysis and Assessment Center

Contents

About the Cover: A visualization of energy from an HPC simulation of a newly developed modular protective system. Such HPC simulations and visualizations are helping developers in the design and fabrication of the system, as well as providing input to reduce the number of costly live fire tests.

Data provided by Don Nelson, ERDC Geotechnical and Structures Laboratory, and visualization done by the HPCMP Data Analysis and Assessment Center.

First Word

By John E. West, Director

The Department of Defense (DoD) High Performance Computing Modernization Program (HPCMP) turns 20 years old this year. During that time those of you who work in the Program and use its resources have built an incredible legacy of leadership and positive impact on the ability of the DoD to accomplish its mission. It is a humbling experience to contemplate leading this Program into the future, but as the HPCMP moves into the Army, that is precisely what we are all being asked to do.

During these first several months of leadership transition, the HPCMP has been preparing to address the changing technological landscape of high performance computing (HPC) within the continuously evolving financial and operational context of the DoD. Our goal is to make sure that we provide the right services, to the right people, in the right way, ultimately achieving a modernized Defense capability through the integration of HPC into the day-to-day practices of the Department.

Stakeholders throughout the HPCMP – in the Program Office, in Centers, in the user community, and within our governance communities – have been working as a team to refine the strategic plan for the HPCMP, building upon our past successes and creating room for new opportunities. We have ensured that our mission and vision align with who we are and what we do, while allowing for a future that broadens our impact on the DoD as a whole.

Realizing our vision and maintaining a position of leadership in the application of HPC among the Research, Development, Test, and Evaluation communities as they support all of the mission areas of the DoD will require that the HPCMP target opportunities for more focused research and technology development. It will also require that we tune some of our business practices to more specifically meet the changing needs of our users.

For many of you, this will mean familiar services delivered to you in more relevant ways to maximize your productivity: HPC when you need it, how you need it. In other cases, we will be pursuing new communities of practice into which we can inject HPC, building new value for the Department.

Our strategic plan also calls for an investment in the intellectual future of our workforce. To remain a leader in the HPC field, we must educate, train, and retain the best computational professionals in the world. HPC is critical to the future success of the DoD: in research, where HPC enables DoD to explore new theories and evaluate them well beyond what is practical using experiment alone; in acquisition, through the use of validated applications in design and testing to reduce the time and cost of acquiring weapon systems; and in operations, where real-time calculations produce just-in-time information for decision makers on the battlefield.

You will hear more about the strategic planning process, and get insight into its component parts, during the June User Group Conference and in the weeks and months following that event. The plan is a living document, and we need your great ideas and (constructive) criticisms to continually make it, and this Program, more effective.

This is an exciting time for the HPCMP and an exciting time to be part of the international HPC community. I think you'll find it will bring a wealth of opportunities for our current and future users to make HPC a tool of first resort.

DOD HPC
DEPARTMENT OF DEFENSE
HIGH PERFORMANCE COMPUTING
MODERNIZATION PROGRAM

Data Visualization

By Randall Hand, U.S. Army Engineer Research and Development Center (ERDC), Data Analysis and Assessment Center (DAAC), Vicksburg, Mississippi

As with every new year, new users are entering the DoD High Performance Computing Modernization Program (HPCMP) and learning to use the systems. For many, it is their first exposure to large-scale computing and the many challenges it brings. The Data Analysis and Assessment Center (DAAC) is the Program resource for helping users manage their resulting simulation datasets and helping them find the information hidden within.

For many users, both new and old, the benefits of data visualization remain unknown. Classical numerical analysis of the same 2D slices of their data that they've used for years remains all they've ever used. However, users who have worked with DAAC are finding new visualization methods for their data that provide the opportunity for greater insight into their simulation and breathe new life into their reports and presentations for upper-level stakeholders.

Dr. Doug Dommermuth expresses the importance of visualization in his work:

> "We use flow visualization to understand the physics of flow around naval combatants. We use flow visualization every day to debug code, analyze and process data, and visualize billion-plus grid-cell simulations of complex free-surface flows. Flow visualization is a critical part of our research program, and without flow visualization, our research program would grind to a halt."

Not only has the visualization become an integral part of his research in analyzing his data, it has also presented new opportunities for communicating his results in public venues such as conferences and stakeholder

Design

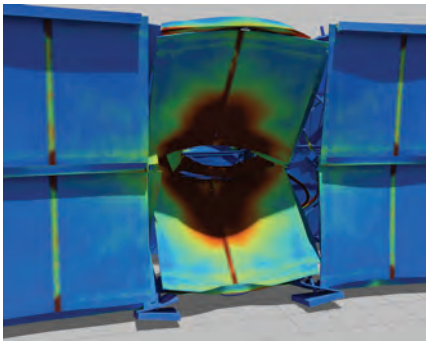
briefings. Visualizations of his research created by DAAC have won awards in events like the Gallery of Fluid Motion and the Department of Energy SciDAC Visualization Night events, giving him newfound exposure and publicity amongst several agencies and stakeholders.

Even for those without the need for public exposure, data visualization offers a fast and interactive way to see large amounts of data in a quick and natural way. Several users shy away from interactive visualization due to the complexities involved in our HPC architectures, but DAAC has done the heavy lifting to make client-server applications as point-and-click as possible. New HPC launching scripts for tools like VisIt, ParaView, and Ensight make visualizing data on the HPC platforms as simple as opening data on your local computer. This allows you to visualize your data as soon as your jobs are finished without transferring or copying your data to other systems.

High-Speed Experimental Footage

Dr. Jesse Sherburn, ERDC-GSL, discusses the importance of this in his work:

> "In a high performance computing environment, the need for fast, high-quality visualization is critical to solving large computational problems. If I am unable to look at my data from a large simulation accurately, I will not be able to learn much from the simulation let alone try to improve it. The faster and more accurate I can look at and analyze my data, the quicker I can make an informed decision based on the results."



HPC Simulation Data

DAAC is a free resource open to all users of the HPCMP, dedicated to helping you find tools and methods to visualize your data. With assistance ranging from training to full poster and video productions, we can help you come up with visualizations to help yourself, your fellow researchers, and everyone else involved in your work.

For more information including tutorials, example results, and screencasts of analysis applications in action, along with details on contacting the DAAC, visit our website at http://daac.hpc.mil.

Client-Server HPC Job Launching

By Rick Angelini, Army Research Laboratory (ARL), Data Analysis and Assessment Center (DAAC), Aberdeen Proving Ground, Maryland

Background

There are several client-server visualization packages that are actively used across the High Performance Computing Modernization Program (HPCMP) for data analysis and postprocessing of computational results. Today's modern low-cost Linux, Mac, and Windows desktop workstations with a standard commodity graphics card provide virtually any desktop system with sufficient power to drive these high-end visual analysis tools. The availability of these low-cost workstations combined with the availability of production-quality commercial (EnSight) and open-source (ParaView, VisIt) client-server visualization packages allow for unprecedented access to HPC-sized datasets from the desktop. These applications are client-server in nature; that is, there is a portion of the code (client-side) that runs on the local desktop workstation and is responsible for handling the GUI-based interface and the rendering and manipulation of the graphical components. On the HPCMP resources, the server-side of the application is responsible for the computationally intensive portions of the data postprocessing such as reading in the simulation results, subsetting or manipulating the data, etc. The client and server communicate with each other using standard network protocols; however, establishing a clear communication path between the client workstation and the allocated HPCMP resources has been challenging.

These client-server applications have been used for many years by HCPMP customers; however, utilization by the general customer population was limited due to the complexities required to establish a clear communication path between the client and server. An interactive session required manually launching the client and server processes independently, establishing the appropriate SSH-tunnels to allow the processes to communicate, and then having it all come together into a working, interactive session. These client-server applications had to be coerced to work within the HPC environment, overcoming obstacles such as the job-queuing systems and any number of necessary security policies that restricted communication paths between the allocated back-end computational nodes and the client workstation.

Implementation

In recent years, both ParaView and VisIt began to provide methods to automate the process of connecting to an HPC resource from a desktop client workstation, launching a job through the native queuing system, and establishing a communication path from the allocated computational nodes back to the workstation. Initial HPC job-launching support provided by these packages was inconsistent, and the ARL DSRC visualization staff worked with the developers to improve the functionality of these processes. We were eventually able to provide totally transparent HPC job launching from the desktop workstation, thereby allowing the researchers to focus entirely on their visual analysis rather than the convoluted and frustrating process of setting up a client/server session.

The ARL DSRC visualization staff also worked with the vendor Computational Engineering, Inc (CEI) to add this functionality to EnSight and served as the primary evaluator during the development cycle to test the implementation provided by CEI. In addition to enabling HPC job launching in Ensight, we worked with the vendor to incorporate server-side license checkout. EnSight is a commercial package that requires that a license seat be checked out from a pool of available licenses. HPCMP customers attempting to run EnSight from unknown subnets were not able to check out a license from the client-side process without first creating a tunnel to an HPC resource. We recognized the difficulty in accomplishing this task, and this procedure is no longer required. A server-side license checkout will occur when a user adds "-slim_on_server" to the EnSight startup command. This functionality is automatically provided in the preconfigured host profiles starting with EnSight 10.0.1a.

The ability to hide all of the complexities of client-server interaction from the HPC customers has allowed for a significant increase in the use of these packages by virtually any researcher from their desktop workstation. Many years of effort and natural application evolution has culminated in the availability of multiple production-quality software tools that provide researchers access to massive amounts of computational resources from their desktop. HPC customers are able to use the packages as a natural part of their data analysis efforts without the extraordinary burden of understanding the underlying complexities required to allow it all to function properly. The impact of these visualization tools for both classified and unclassified researchers is greater now than it has ever been.

Availability

In order to use HPC job launching from a client workstation, the application software needs to be installed along with the appropriate configuration files and scripts necessary to support HPC job launching. Within ARL, this process is managed in the Linux and Mac environment through various automated software procedures that update the software on a daily basis, therefore allowing the appropriate software and associated configuration files to be pushed to the client workstation without an additional intervention by the customer. The more difficult challenge is how to provide these applications to all HPCMP customers in such a way, as the appropriate configuration files and scripts are included in the distribution. After discussion with various HPCMP and User Productivity Enhancement, Technology Transfer, and Training (PETTT) collaborators, it was determined that the optimal solution was to bundle the complete

application and the appropriate configuration files into a single package that can be downloaded from a common location and installed on the local workstation. These preconfigured application bundles benefit the HPCMP customer by avoiding the situation where the customer first downloads the application from a vendor or application website, then does a separate download for the HPCMP-specific files, and then inserts the configuration files and scripts into the distribution.

ParaView, VisIt, and EnSight preconfigured bundles can be downloaded from any of the HPCMP Utility Servers from the directory ***/app/projects/client-server*** along with appropriate instructions for installation of the software. (NOTE: The availability of these packages from the HPCMP Utility Servers does not relieve the customer of any local organizational rules, regulations, and approvals related to the installation of workstation software.) The versions of the software available from this distribution directory match the versions of the software currently available on the Utility Servers. Each package provides job-launching functionality to all of the Utility Servers and to a number of HPC systems where the applications have specifically been requested.

Summary

Traditional client-server applications are a viable tool if they can be configured to work within the constraints inherent to coprocessing on an HPC system. ParaView, EnSight, and VisIt now provide a mechanism to allow virtually any researcher to use these tools as a natural component in their analysis process. The challenge remains to find a concise method for distributing these tools to the customer desktop in a way that is equally natural for the customer. DAAC will continue to evaluate the implementation of client-server software and develop distribution methods.

Additional documentation on HPC job launching can be found on the DAAC wiki pages located at https://visualization.hpc.mil/wiki/Main_Page, including video tutorials on the use of HPC job launching with ParaView, Ensight, and VisIt.

Navy DSRC Provides Team Approach to Improving Multisystem **Forecast Modeling Workflow**

By Christine Cuicchi, Navy DSRC Computational Science and Applications Lead, and Sean Ziegeler, PETTT Advanced Computational Environment (ACE) On-Site, Naval Research Laboratory, Stennis Space Center, Mississippi

It's easy for anyone who writes code and scripts to make a quick hard-coded fix here and there to get things running smoothly one day and forget about it the next; we've all done it. But as those quick fixes become part of legacy code, scripts, and multisystem workflows, they reduce the flexibility required to move between the varying operating systems and system architectures found in the fast-paced world of high performance computing (HPC).

The limited flexibility and resiliency of the Navy's operational oceanographic modeling community's multisystem workflow, which in part uses Navy DoD Supercomputing Resource Center (Navy DSRC) HPC systems to create forecast products delivered to the Navy fleet on a 24x7 basis, present an ideal project for process improvement. The community's workflow can weather scheduled short-term preventative maintenance periods with little impact to forecast product delivery, but is more extensively disrupted by unexpected outages of HPC systems. The relatively short overlap between the Navy DSRC's retirement of existing HPC systems and the installation of new HPC systems in the coming year emphasizes the need to make this workflow as platform-independent and adaptable as possible, and within a much shorter time period than afforded during previous system transitions.

Approach

The Navy DSRC has taken a lead role in assisting this community in modernizing their entire workflow from oceanographic model development, to computational model runs on HPC systems, to the Naval Oceanographic Office's (NAVOCEANO) operational forecast product delivery. The end goal of the modernization effort is a workflow designed towards and fully capable of moving expediently between heterogeneous HPC platforms, making operational modeling forecast product delivery as resilient as possible and providing an immediate, positive impact to the Navy and the Department of Defense.

To accomplish these goals, the Navy DSRC staff launched a modernization effort by leading a workshop in October 2011 for subject matter experts from the DSRC, NAVOCEANO, Naval Research Lab at Stennis Space Center (NRL-Stennis), and the HPCMP's User Productivity Enhancement, Technology Transfer, and Training (PETTT) program. The workshop began with an information exchange regarding current and future oceanographic model requirements and development methods. DSRC and PETTT on-site staff highlighted new and incoming HPC capabilities and project assistance available to the NRL-Stennis and NAVOCEANO teams, and the group conducted a rigorous examination of the requirements of each HPC- or non-HPC-based step of the operational workflow.

Two major improvement areas identified were workflow portability and best use of HPC resources. A wide range of solutions were proposed for these areas; some solutions leverage existing efforts, and others require innovation and workflow restructuring. Various

team members took ownership of the implementation or feasibility determination of various solution sets.

Workflow Portability

Platform dependencies built into the oceanographic models and supporting scripts are the largest issues hampering the portability of the current workflow. Proposed solutions include the following:

- An interface layer that will assign a specific runtime environment to a model via a relational database prior to the model run's start on the target HPC platform.
- Platform-independent precompiler scripts.
- Beta operational queues on HPC systems available for testing script changes during 'idle' portions of the operational HPC workflow.
- Development of an Operational Model Test Suite (OMTS) that will allow for pre-deployment testing of the OMTS against HPC system and runtime environment upgrades.

Using Dedicated HPC Resources and Managing Data

A portion of each Navy DSRC HPC system is dedicated to operational modeling jobs to prevent resource contention with allocated users. The operational models access specific resources, run on a set schedule, and generate a large number of files, all of which provide opportunities for improvements and increased efficiency. Proposed solutions include the following:

- Intelligent HPC system access scripts that target specific resources while allowing for failover.
- Surge capability within the dedicated computational queues that could return some nodes to the pool of non-dedicated batch nodes during 'idle' periods in the operational modeling run schedule.
- Improved data management methods such as automated inventory purge, intelligent file scrubbers, understanding of file system limitations, and use of the Center Wide File System (CWFS)

Team's Path Forward

Since the workshop, the DSRC has begun investigating solutions internal to the Center and provided input to the newly begun development of NAVOCEANO's Ocean Model Developer's Guide (OMDG), which will provide standards and guidance to the NRL-Stennis team and others involved in the development of oceanographic models and supporting job and file management scripts. The Center continues to collaborate with the NAVOCEANO team towards the implementation of several of the above solutions.

A PETTT 6-month Pre-Planned Effort (PPE) begun in February and led by Sean Ziegeler, the PETTT Advanced Computational Environment (ACE) On-Site at NRL-Stennis, is set to investigate and accomplish many of the proposed solutions by collaborating with the DSRC, NAVOCEANO, and NRL-Stennis teams with innovative activities. The project will also leverage current activities within NRL and NAVOCEANO such as the OMDG and NAVOCEANO's newly developed Environmental Variable Manager (EVM). It will also explore new capabilities such as version control, Center Wide File System, and automated dependency checks. The final result of the PETTT PPE will be a system in which the entire modeling run stream can (1) be more easily ported to new platforms and (2) failover or switch between two systems when one system fails or must be taken down for maintenance. NAVOCEANO and NRL are also moving ahead on the solutions for which they are the lead team members. The bulk of the modernization effort is expected to conclude by the beginning of FY13.

The Navy DSRC welcomes the opportunity to provide a team approach to problem solving for all of our users. For more information, please request Outreach assistance from the Navy DSRC via the Consolidated Customer Assistance Center at help@ccac.hpc.mil.

Applying for a Dedicated Support Partition (DSP) at the MHPCC DSRC

By Marie Greene, Air Force Research Laboratory Deputy Director, Maui High Performance Computing Center DoD Supercomputing Resource Center, Maui, Hawaii

A new HPC delivery model called Dedicated Support Partition (DSP) has been prototyped at MHPCC in FY12. Instead of the typical environment of user jobs being submitted to a batch queuing environment, each project approved has a dedicated number of computer cores assigned to it. This model differs from the Advance Reservation System in that these partitions typically are large (128-2048 cores) and have long durations (2 months to a year). This allows projects complete control of usage during the duration of the DSP. They have proved to be quite beneficial for large-scale software development and regression testing. The general concept of the DSP is endorsed by two of the Program's main advisory groups: HPC Advisory Panel (HPCAP) and User Advocacy Group (UAG).

There is an ongoing call for FY12 usage through 30 June 2012. It is expected that there will be another call for FY13 usage by publication of this magazine. All project leaders with an active RDT&E computational project are eligible to submit a DSP proposal. All application software development efforts, large-scale weapons system test support, and other activities requiring substantial dedicated time on HPCMP resources that cannot be serviced through normal batch processing, interactive processing on the new Utility Servers, nor the Program's advance reservation service will be considered. To apply for participation, proposers should prepare a short proposal (2 to 3 pages suggested) containing the following:

- Project leader's name and contact information.
- HPC computational project number.
- Which HPCMP system(s) is being

requested and portion of system needed (number of cores) (minimum of 64 cores, maximum of 2048 cores).

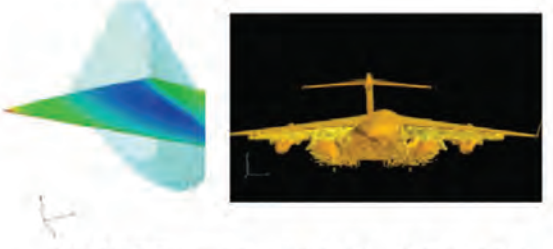
- Duration of project (projects should be no less than 1 month and may continue through the end of the planned fiscal year). Note that the dedicated period for the project need not be continuous; e.g., dedicated resources only for certain days during each week for the duration of the project may be requested.
- A brief description of the project and justification for a dedicated partition on an HPCMP resource.
- Names of users associated with this project.
- Project's impact on DoD.
- Specific considerations:
  - Software license requirements.
  - Special storage requirements.

Key Program Office staff and external experts (as needed) will select projects from these requests and give each an opportunity to have a large fraction of an HPC system dedicated to it for a specified period of time. Proposals for FY 2012 DSPs may be submitted through 30 June 2012. Proposals will be evaluated within 2 weeks of submission, and new awards will be implemented as soon as available resources are identified.

The HPCMPO requires proposals in Microsoft Word 2003 or later format. Please email the completed proposal and direct any questions to the DoD HPC Modernization Program Office at require@hpcmo.hpc.mil.

Below are some current DSP projects. The first five use cases fall into the Computational Research and Engineering Acquisition Tools and Environment (CREATE) – Air Vehicles computational applications sponsored by the DoD HPCMP. JSF Ship Integration is a project of the Naval Air Warfare Center.

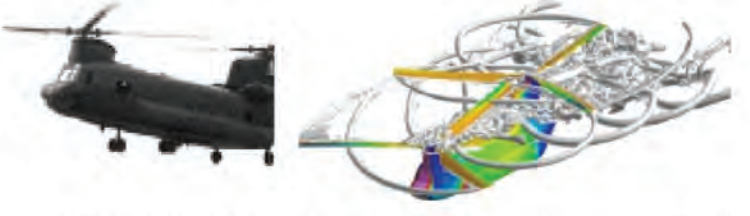
For more information, contact Peter A. Calvin at telephone 808-891-7760 or email to peter.calvin.ctr@mhpcc.hpc.mil.



KESTREL – High-fidelity, full vehicle, multi-physics analysis tool for fixed-wing aircraft

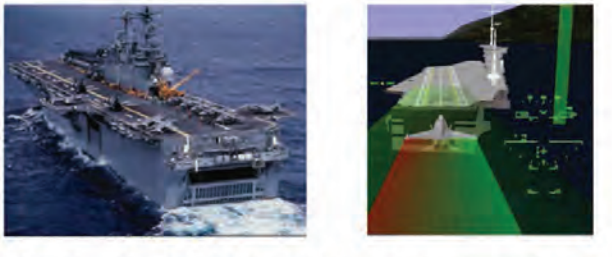
HELIOS – High-fidelity, full vehicle, multi-physics analysis tool for rotary-wing aircraft

FIREBOLT – Module for propulsion systems in fixed and rotary-wing air vehicles



SHADOW OPS – Computational tools to support acquisition programs that provide experience and establish connections and value

RADIO FREQUENCY (RF) – Computational tools that enhance antenna performance and integration with platforms on ships, aircraft, etc.



JSF SHIP INTEGRATION – Analysis of output to improve ship design, aircraft control system development, and inputs for flight stimulators

Dynamic Linear Solver Selection for Transient Simulations Using **Machine Learning Software**

By Paul R. Eller, Jing-Ru C. Cheng, and Robert S. Maier, Engineer Research and Development Center, Information Technology Laboratory, Vicksburg, Mississippi

Introduction

Many numerical models use transient simulations to accurately simulate how natural or man-made systems change over time and how different events or designs may affect these systems. For many transient simulations, the largest amount of simulation time is spent solving a linear system. Many preconditioners and solvers have been developed to quickly solve different types of linear systems. As the linear systems produced by the transient simulations change, the best preconditioned solver to solve each linear system also changes. Using the best preconditioned solver at each point in the simulation will allow us to get the fastest possible running times.

Machine learning algorithms provide the ability to generate predictive models, allowing us to create classifiers capable of taking a set of linear system attributes as input and outputting a preconditioned linear solver as the output. We test both single-label classifiers that associate a single fast linear solver with each linear system and multi-label classifiers that associate multiple fast linear solvers with each linear system.

We can generate databases by computing attributes for each linear system, physical attributes for the transient simulation, computational attributes, and running times for a set of preconditioned solvers on each linear system. Machine learning algorithms can then use these databases to generate classifiers capable of dynamically selecting a preconditioned solver for each linear system given a set of attributes. This allows us to use different preconditioned solvers throughout the simulation and provides the potential to produce speedups in comparison with using a single preconditioned solver for an entire simulation.

Background

The ADH modeling system provides users with the capability to simulate saturated and unsaturated groundwater flow, overland flow, three-dimensional Navier-Stokes flow, and two- or three-dimensional shallow-water problems. This work focuses on using the 3-D Navier-Stokes numerical flow solver to simulate free-surface flow in complex 3-D structures for the evaluation of navigation locks. ADH uses the Galerkin least-squares finite element method for solving the Reynolds-averaged incompressible turbulent 3-D Navier-Stokes equations. Turbulence is modeled with an adverse pressure gradient eddy viscosity technique. ADH uses the Newton algorithm to solve the nonlinear problem.

PETSc provides users with access to a suite of data structures and routines for parallel scientific applications, including a wide variety of fast, scalable linear solvers and preconditioners. ADH has been interfaced with PETSc, providing ADH users with access to these fast linear solvers and preconditioners. We use the AnaMod library to help compute numerical metadata. AnaMod is a part of the Self-Adapting Large-scale Solver Architecture (SALSA) software project, which aims to assist applications in finding suitable linear and nonlinear solvers based on analysis of the application-generated data. We generate Web Ontology Language (OWL) databases using the OWL API. OWL provides a language for developing ontology documents for use by applications that need greater information processing capabilities. We use the OWL API to access and modify the OWL ontologies from within the numerical model.

We use the WEKA and MULAN data mining software packages to access a wide variety of single-label and multi-label machine learning algorithms. These classifiers are generated by passing the machine learning algorithm a collection of instances, with each instance containing a set of attribute values and one or more label values. Once the classifier has been generated, we can pass the classifier a set of attributes as input and the classifier will return a

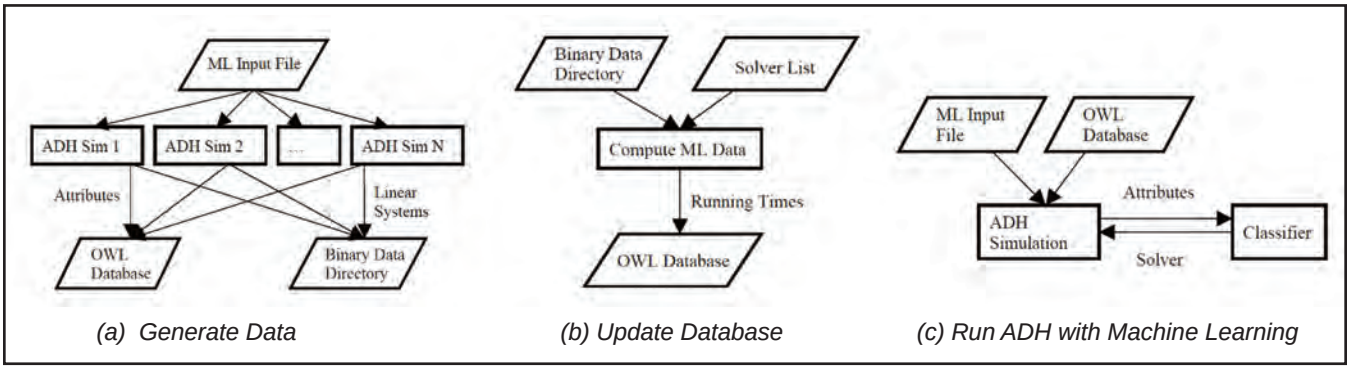


Figure 1. Process to use machine learning with ADH. First, generate a database and linear system data; then test solvers with linear systems and update database with running times; and finally, use ADH with machine learning to dynamically select the solvers for each linear system

label as the class value for the output. In this case the instances are linear systems produced by ADH, the attributes are the properties of the linear system, and the labels are preconditioned linear solvers.

Machine Learning Interface

In order to use machine learning, we must first create a database containing training data for the machine learning algorithm. Figure 1(a) shows how to generate a database and save the linear systems. We run simulations with a safe choice for the preconditioned solver and save attributes to a database. We create a database containing attributes for each linear system produced during the simulation. Using AnaMod, we compute attributes for each linear system for the categories simple, variance, normality, structure, and spectrum. Attributes related to the physical system and computational methods are added to the database. When we generate the database, we save the full linear systems produced by the simulation to a binary data directory.

Next, we must determine the best preconditioned solvers for each linear system. Figure 1(b) shows how to update the database with the running times for the preconditioned solvers. We can use a separate program to solve each saved linear system with many different preconditioned solvers, adding the running times for each solver for each linear system to the database. The user passes the program a list of PETSc preconditioned solvers that may perform well at some point during the simulation.

Once the full database has been generated, new simulations can be run using machine learning to select a solver for each linear system. Figure 1(c) shows how to use machine learning with ADH. A function to generate each machine learning classifier must be written and compiled in the machine learning section of the code. This code must use WEKA or MULAN functions to create, build, and return a classifier. The user can also define the attributes they want to compute in an input file. Attributes specific to the simulation code can also be used by the classifier. The user must add code to compute these values and pass the name of the attribute and its value to the machine learning interface. This functionality allows the user to control which attributes are computed during the simulation and choose the best classifier for their problem.

At the beginning of an ADH simulation with machine learning, the machine learning interface will access the database and create a dataset. This dataset is used by the machine learning algorithm to create a classifier. The machine learning input file determines which attributes will be used by the classifier and computed during the simulation, as well as which solvers the classifier can select during the simulation.

Test Setup

We test the effectiveness of the machine learning algorithms with the Watts Bar Lock model. ADH models are frequently used to simulate locks filling with water, resulting in difficult-to-compute transient simulations, as there are rapid changes in flow velocity and pressure at the beginning of the transient simulation. The Watts Bar Lock model has a long culvert with a slope at the beginning that leads to a tainter valve with a valve well. There are bulkheads before and after the valve well. The finite element mesh for this model uses 1,635,510 elements to simulate a 23.0 × 0.888 × 3.778 ft area. We change the values of the eddy viscosity parameter and inflow speed to create 12 variations on this model. We test eddy viscosities of $2ft^2/s$, $5ft^2/s$, $10ft^2/s$, and $50ft^2/s$ and inflow pressures of 725Pa, 740Pa, and 755Pa.

We tested the PETSc solvers, preconditioners, and subpreconditioners against ADH produced linear systems. We test BiCGStab(l) (BCGSL) with 2, 4, 6, 8, and 10 search directions and the -ksp_bcgsl_cxpoly option. We test the flexible generalized minimal residual method (FGMRES) with 10, 25, 50, 100, and 200 search directions. We test the additive Schwarz method, block-Jacobi, and Jacobi preconditioners. We test the incomplete LU subpreconditioner with 0 and 1 factor levels and the LU subpreconditioner. This results in generating a database with 70 preconditioned solvers.

We design the test setup to generate full classifiers that have knowledge about all variations of a model as well as test classifiers that have knowledge of all but one variation of a model. We test the full classifiers against all variations of the model to determine the performance of the classifiers when they have prior knowledge of the transient simulation being run. We test each test classifier against the variation of the model of which the test classifier does not have prior knowledge to determine the performance of the classifiers when they do not have prior knowledge of the specific transient simulation being run. Tests are performed on *Garnet*, a Cray XT6, using 16 nodes (256 cores). We perform tests using a variety of WEKA single-label classifiers and MULAN multi-label classifiers.

Results and Analysis

Figure 2 compares the fastest BCGSL and FGMRES solvers to the fastest solvers using machine learning for dynamic linear solver selection. We see that the fastest BCGSL solver outperforms the fastest FGMRES solver for the more difficult simulations, while the fastest FGMRES solver outperforms the fastest BCGSL solver for the less difficult simulations.

Single-label classifiers are generated using WEKA. We see that the fastest single-label classifiers for WEKA and WEKA Full perform well for some model variations, but do not perform as well for other model variations, producing slower running times than the fastest BCGSL and FGMRES solvers. We also see that the test classifiers outperform the full classifiers in some situations. The additional information that the full classifier are given should allow them to outperform the test classifiers. This suggests that the single-label solvers do not have enough data to accurately predict the best linear solvers in some situations. Additional tests demonstrated that individual WEKA and WEKA Full classifiers would perform well for some model variations, but would produce significantly slower running times than

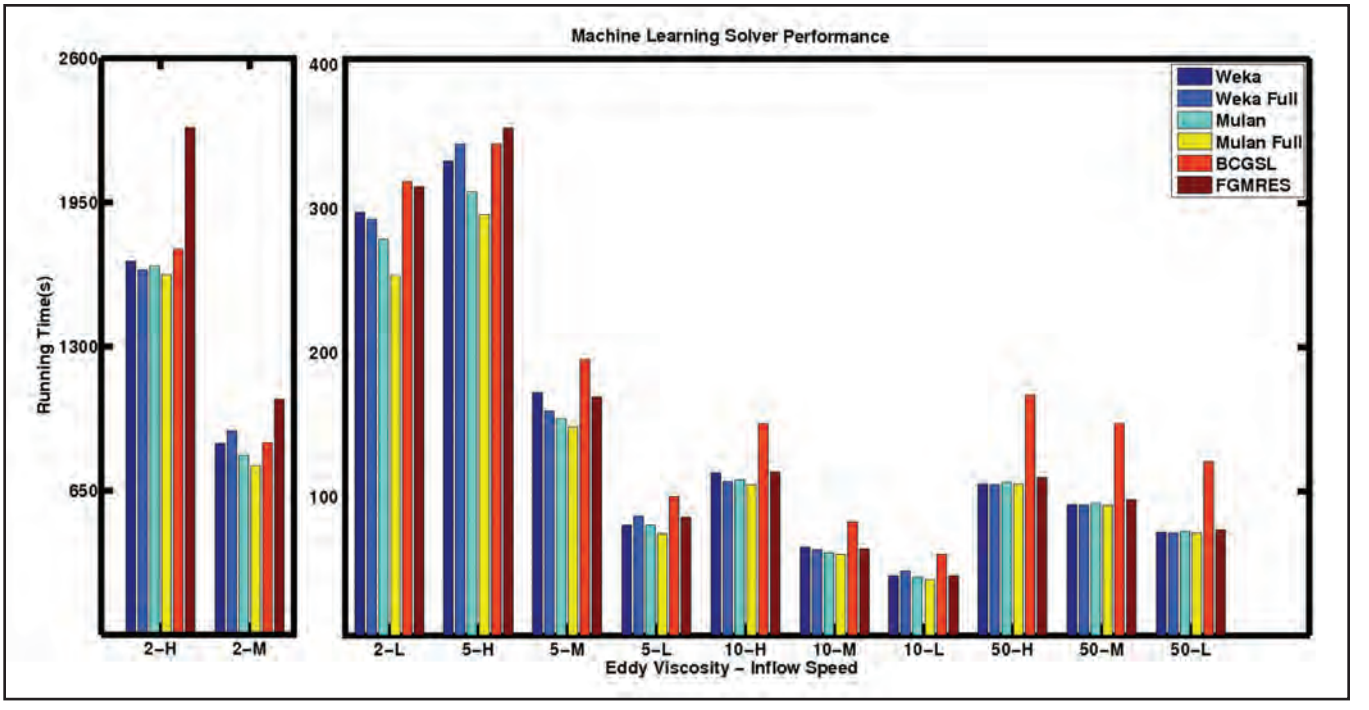


Figure 2. Comparison of fastest PETSc BCGSL and FGMRES solvers for each model variation with the fastest WEKA single-label and MULAN multi-label classifiers for each model variation for ADH Watts Bar Lock simulations with varying eddy viscosities (2ft²/s, 5ft²/s, 10ft²/s, and 50ft²/s) and inflow speeds ((L)ow, (M)edium, or (H)igh hydrostatic pressure at inflow boundary)

the best BCGSL and FGMRES solvers for other model variations. At times, the WEKA and WEKA Full classifiers would fail to solve some linear systems, preventing the ADH simulation from running to completion.

We test the multi-label classifiers using MULAN. We see that the fastest multi-label classifiers outperform the fastest BCGSL and FGMRES solvers for every model variation and that the fastest MULAN Full classifiers outperform the fastest MULAN classifiers in every case. The MULAN classifiers produce significant speedups for the more difficult simulations and produce running times slightly faster than the best BCGSL and FGMERS solvers for the simpler simulations. This suggests that the additional information provided by listing multiple fast preconditioned linear solvers for each linear system provides enough information to accurately predict fast linear solvers for each linear system encountered by an ADH simulation.

Conclusions

This work demonstrates that dynamic linear solver selection using multi-label classifiers allows us to outperform the fastest BCGSL and FGMRES solvers for transient simulations. The single-label classifiers are not able to consistently produce fast running times due to only being able to associate one solver with each linear system. The multi-label classifiers are able to obtain fast running times due to their ability to associate multiple solvers with each linear system, providing the machine learning algorithms with more examples of the linear systems each solver can quickly and accurately solve.

Acknowledgment

We would like to thank Allen Hammack at the U.S. Army Engineer Research and Development Center Coastal and Hydraulics Laboratory for providing us with ADH models. This study was supported by the U.S. Army Engineer Research and Development Center Civil Works Basic Research Program and an allocation of computer time from the DoD High Performance Computing Modernization Program.

IPython for Web-Based Scientific and Parallel Computing

By Dr. José Unpingco, Space and Naval Warfare Systems Center Pacific (SSC-Pacific), San Diego, California

Introduction

The traditional way to share scientific and computational work is with an oral presentation or a static paper. However, web-based technology makes it easier and easier to share both code and data, instead of just results. This potential for detailed reproducibility enhances scientific relevance by providing a clear look into the underlying machinery used to develop the advertised results.

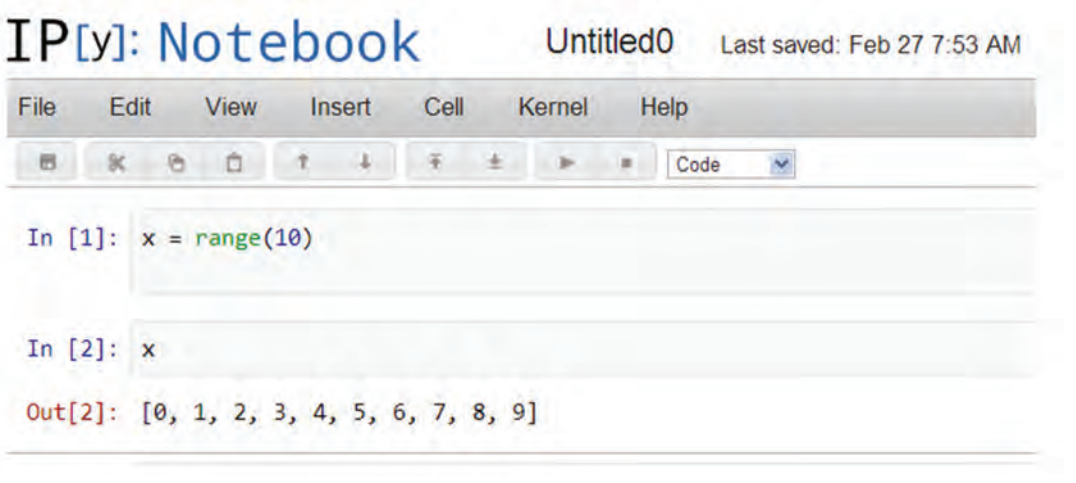
IPython is an open-source, Python-based toolkit for providing interactive interfaces to scientific and parallel computations. As an easy-to-use interactive development environment, IPython has been part of the PETTT "Scientific Python" training course for years where it provided a rich scientific computing experience, complete with graphics. Now, this experience has been extended to modern web browsers. In this article, we will focus on the special capabilities made available by this modern web interface called the IPython Notebook.

IPython Notebook in the Browser

You can start the web interface to IPython using the following command:

```
% ipython notebook
```

The following shows the IPython Notebook running in the Google Chrome web browser:

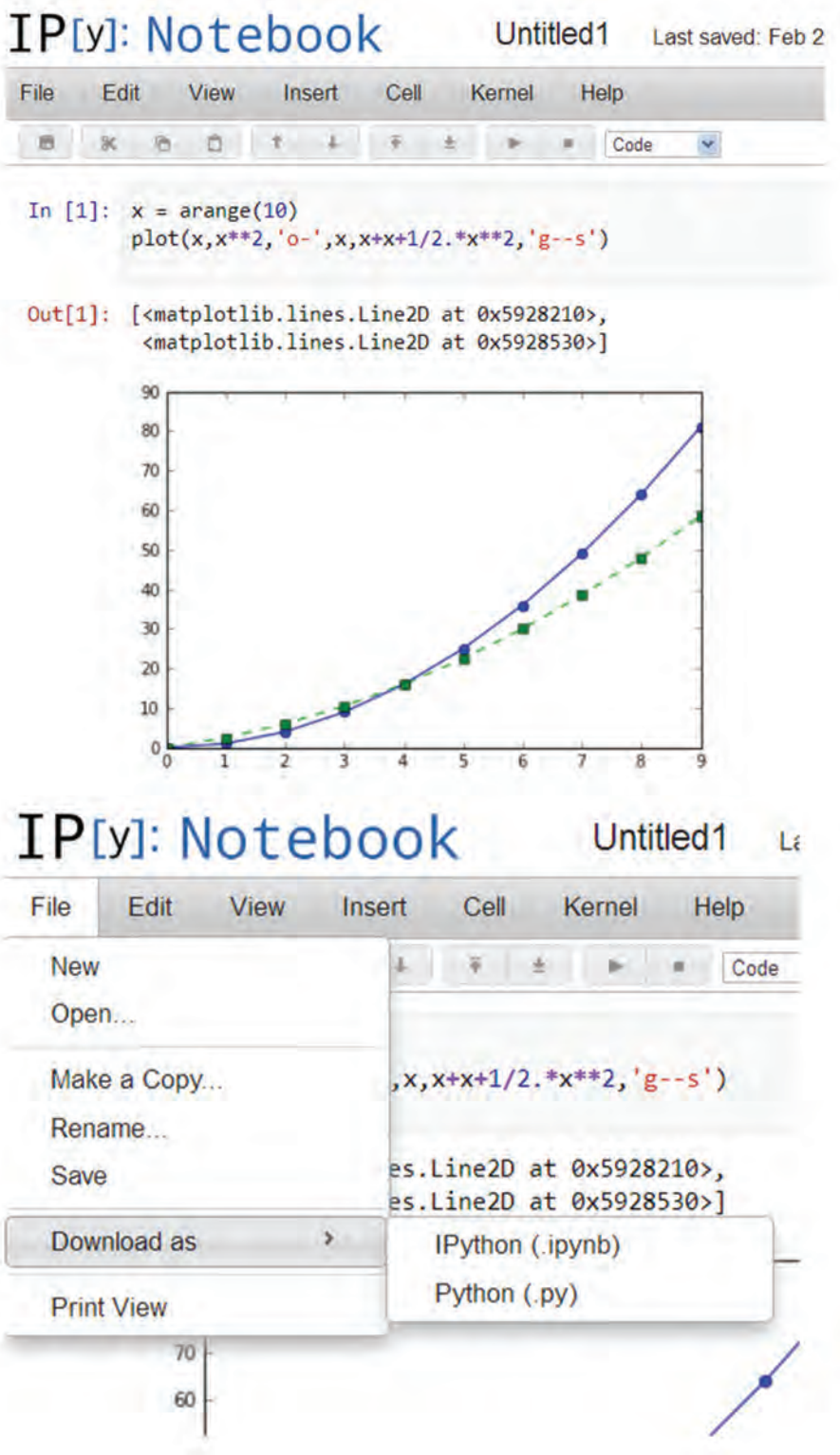


Now, you can type in python commands in each cell and then hit SHIFT+ENTER to execute those commands just as you would in a regular console. Internally, the IPython notebook is running a fast ZeroMQ message passing framework on top of a Tornado web server. This means that the IPython process (i.e., kernel) that is actually executing these commands is separate from the browser.

The browser can also embed rendered graphics if the notebook is started with the "pylab" flag as in

```
% ipython notebook --pylab=inline
```

The Notebook can now embed lush graphics using `matplotlib` (Python scientific graphics package). For sharing and collaboration, the notebook can be downloaded as shown here



as either a pure Python script or in the IPython Notebook format. This means that if your colleagues are using the IPython Notebook with same setup, they can not only render the page, but also change the embedded computations and corresponding results and figures. Furthermore, the page can be saved in HTML format using the browser and distributed as a read-only document for those without the IPython Notebook. Either way, this is a quick and powerful way to share complex scientific computations and graphics. In fact, given the right local network setup, you can share the live IPython Notebook with your colleagues by passing them the corresponding URL. This means that others using a modern web browser can run your notebook

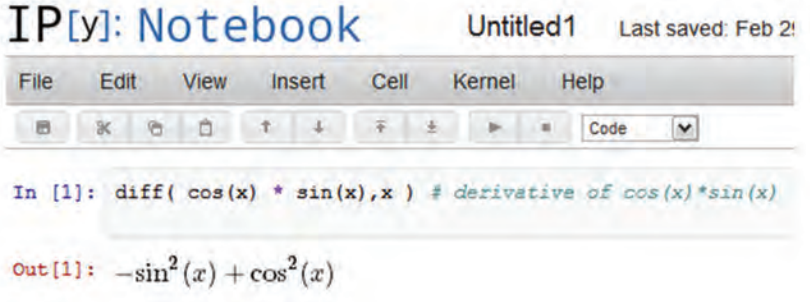
on your machine, without having to set up the individual python modules on their platforms. This means that even if your work involves many difficult-to-install or customized modules, you can still share your work with others who do not have the same installation.

IPython Notebook for Symbolic Math

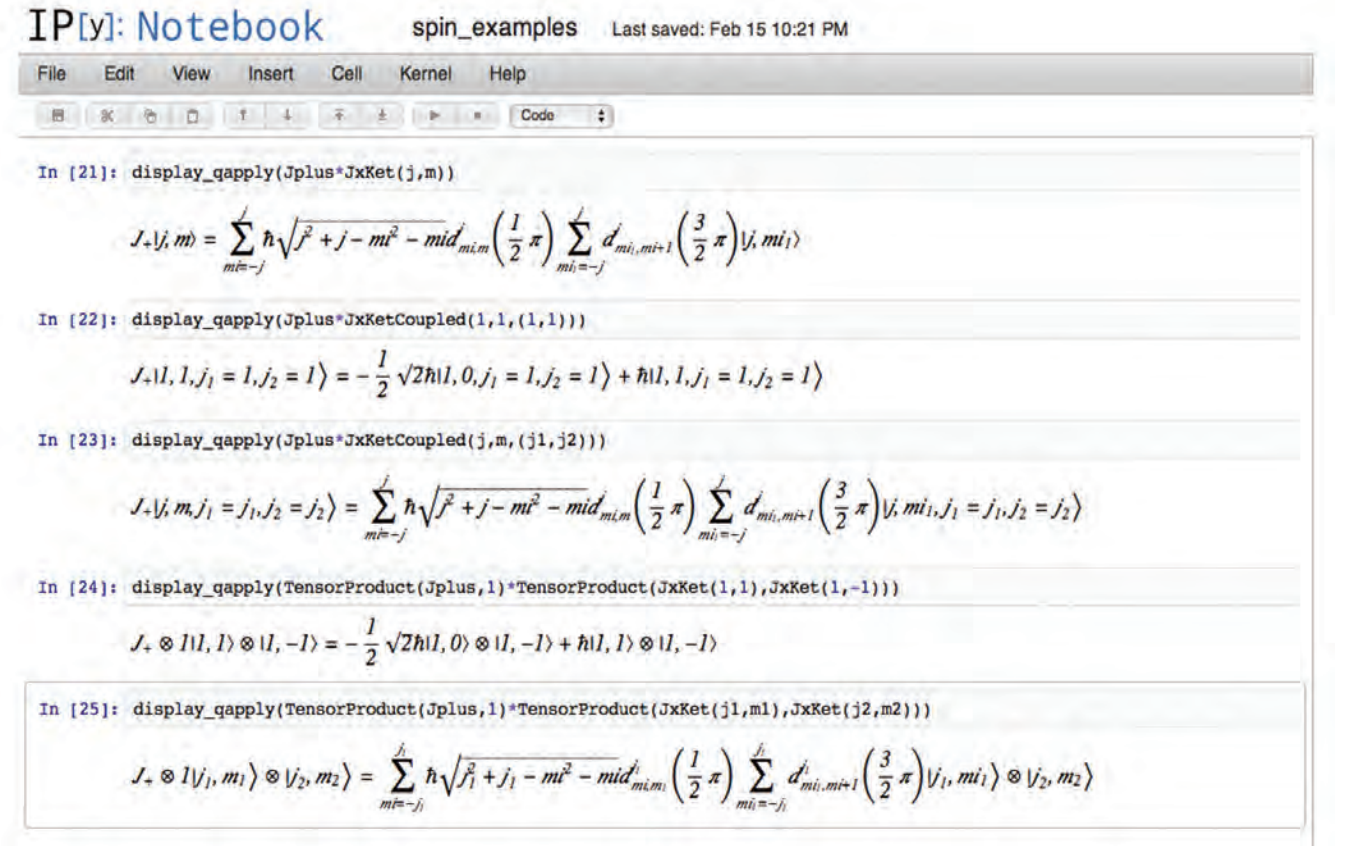
One outstanding benefit of the IPython Notebook for those who need symbolic math (e.g., Mathematica®, Maple®) is that the SymPy module for Python can render equations in the notebook using autogenerated LaTeX and MathJax. MathJax is a toolkit for web browser math, developed by the American Mathematical Society (AMS) and the Society for Industrial and Applied Mathematics (SIAM), which can display complex mathematical formulas in the browser. This is enabled by starting the notebook using

```
% ipython notebook --profile=sympy
```

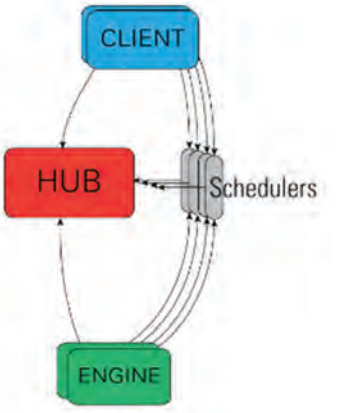
The following example shows what this looks like in the notebook



where we compute the first derivative of a simple trigonometric expression. The SymPy module contains many more features beyond basic calculus, including matrices, differential equations, logic expressions, and algebraic equations. The combination of the IPython Notebook and SymPy can create complex and beautifully rendered calculations as shown in the following figure from Sean Vig's Google Summer of Code project:



Parallel Computing Patterns in IPython



IPython enables interactive parallel computing that supports many different styles of parallelism such as SPMD, MPMP, MPI, and task farming (and combinations thereof). Furthermore, IPython supports many cluster configurations including PBS, Windows HPC Server, and even Amazon's EC2 using StarCluster. As always, ZeroMQ provides the message passing framework in all of these configurations. The figure to the right shows the architecture overview.

The IPython engine is a separate Python process that receives instructions from the network connections. This is where the actual computing takes place. The IPython controller consists of the "hub" and "schedulers" that manage and distribute requests to the engines. The controller provides a "direct" or "load-balanced" interface to the engines. In the direct case, the user controls which computations are sent to which engines, whereas the load-balanced case assigns these automatically. The IPython client provides the primary interface for the entire framework where users type in python commands.

Using the IPython Client

As an example of setting up multiple processes on a local machine, you can start with

```
% ipcluster start -n 4
```

which starts four Python processes. From inside the IPython notebook, you can import and initialize the client as in the following:

IP[y]: Notebook Untitlec

File Edit View Insert Cell Kernel

```
In [1]: from IPython.parallel import Client

In [3]: c = Client()
        c.ids

Out[3]: [0, 1, 2, 3]
```

This shows that the client has successfully connected to the individual four Python processes we started above. Now, to interact with these processes, we choose and set up the "direct" view as follows:

```
In [4]: dview = c[:]
```

and now you can distribute calculations to each of the four processes as shown:

```
In [5]: parallel_result = dview.map_sync(lambda x: x**10, range(32))

In [6]: parallel_result

Out[6]: [0,
         1,
         1024,
         59049,
         1048576,
         9765625
```

It is important to emphasize that the interactive experience is the same regardless of where the python engine processes are running because the controller mediates these issues. In particular, this means that you can connect to a long-running job on a far away cluster using the client from a laptop, for example.

Summary

The IPython notebook represents a significant advance in Python-based scientific computing by making the interactive IPython experience available within a modern web browser. This design makes it much easier to share complicated scientific results and research using common web standards and to provide rich graphics and mathematical fonts. IPython's interactive parallel computing capabilities support a wide variety of cluster technologies and styles of parallelism. Furthermore, there is already a PETTT 12-month effort underway to enhance IPython's parallel capabilities for the Proteus multiphysics toolkit at ERDC and to further extend the IPython Notebook.

For further information or assistance with Python tools, please contact the author or help@pettt-ace.com.

Acknowledgments

The author would like to thank and acknowledge Dr. Fernando Perez (UC Berkeley), Dr. Brian Granger (Cal Poly San Luis Obispo), and Dr. Chris Kees (Coastal and Hydraulics Laboratory, U.S. Army Engineer Research and Development Center) for their support.

System Calls Getting You Down?

By Brent Andersen, Systems Administrator, Air Force Research Laboratory DoD Supercomputing Resource Center, Wright-Patterson Air Force Base, Ohio

It's tax time and you have just submitted all the necessary information to your tax person. He's busy this time of year, but will complete everyone's taxes first-come, first-served.

Imagine calling your tax person every 5 minutes for a progress report. How is your goal affected? Now imagine everyone calling him every 5 minutes.

The operating system is a bit like your tax person. When parallel applications flood it with frequent unnecessary system calls (such as requesting CPU time), the system has little time left for other, more important tasks. Delaying vital system calls too long can even cause a system crash. Asking for CPU time frequently is like incessantly interrupting the tax accountant to find out how long he has been working on your taxes. It would be much better to ask to be notified upon completion.

Plan ahead for the amount of data you want to read or write. A teaspoon is the perfect tool for putting sugar into your coffee; but for borrowing a cup of sugar from the neighbor, it would take 48 trips. Yet again, a forklift truck would be more suitable for moving sugar in a grocery warehouse. System defaults work fine for small files, but huge files may benefit from the use of larger buffers. The size of the buffer will determine how much data are written per system call. The optimal buffer size could save the equivalent of a thousand trips to the grocery store.

How's the weather? When one individual checks outside to report the weather to the entire office, everyone saves time. System calls are costly, and even a supercomputer can be overtaxed. It is sometimes better for one process to make a system call (such as checking for the existence of a file) and then broadcast the information to its fellow processes. Everything counts in large amounts.

Running out of memory is a little like running out of rocket fuel. You may crash and burn and even take others with you. So it is important to request enough memory at the outset. Some possible strategies for increasing available memory are as follows:

(1) Adjust the job script to run fewer processes than cores.
(2) Adjust application input parameters.
(3) Repartition the problem dataset.

Don't forget to allow room for system buffer cache if your code is I/O intensive.

Hint: If you can't modify your application, sometimes environment variables can be set to adjust the behavior of your program.

The above are only a few examples of problems that can affect application performance. If your code has high system CPU time or it seems to perform poorly, consider contacting CCAC for assistance.

Technical Notes:

The most commonly used debug technique is adding "print" statements. While this can be expedient and effective, it tends to degrade application performance. Some tips for mitigating this effect are as follows:

(1) Remove or comment out debug statements after a problem has been solved.
(2) Make debug statements conditional on a debug flag.
(3) Write debug output to a file opened for that purpose. This will result in fewer system calls than I/O to the standard output and standard error files. A drawback is that output buffers may not be flushed if the application is aborted abruptly.

One of the best ways to speed up a task is to eliminate unnecessary work. For example, if a program prints huge volumes of totally useless output, simply redirect standard output to /dev/null.

Discovering High System CPU Usage

A high ratio of system CPU to user CPU time is an indicator of possible excessive system calls. You can determine how much wall time, user CPU time, and system CPU time a program uses by prepending "/usr/bin/time -p" to the program command line. For example:

```
hawk-0> /usr/bin/time –p sleep 3
real 3.00
user 0.00
sys 0.00
```

System CPU time is usually no more than 10 percent of total CPU time. If it is greater than user CPU time, further investigation is recommended. You can get a summary of the time, calls, and errors for each system call made by a process and its children by prepending "strace –qfc" to the command line. For example:

```
hawk-0>  strace -qfc /bin/echo

% time     seconds  usecs/call     calls    errors syscall
------ ----------- ----------- --------- --------- ----------
100.00    0.000064           4        18           fstat
  0.00    0.000000           0         3           read
  0.00    0.000000           0         1           write
  0.00    0.000000           0        38        21 open
 [snip…]
```

From the Director's Desk – Frank Witzeman

Shortly before publication of this issue of *HPC Insights*, the AFRL DSRC extended once again the decommissioning date of the last-standing, shared-memory platform in the DoD HPCMP. *Hawk*, purchased and installed under the Technology Insertion 2007 (TI-07) process, is a 9216-core SGI Altix 4700 with 18 nodes each containing 512 cores (1.6 GHz dual-core Intel Itanium). Two of the nodes address large shared-memory applications by configuring 4 GB/core or a total of about 2 TB/node. The other 16 nodes contain 2 GB/core or about 1 TB/node. *Hawk's* workspace/scratch file system is SGI's CXFS with approximately 440 TB of addressable space. SGI's proprietary "NUMAlink" interconnect enables memory access, communications traffic, and I/O across the system. At the end of September, *Hawk* will be split into two reconfigured systems and delivered to new homes in the Air Force testing and Army research communities.

About a year ago, we determined that the average utilization of shared-memory applications running on *Hawk* was almost 30 percent of the entire utilization, and that over 50 projects with more than 90 users consumed the 14 million core-processor hours for these applications. Shared memory was deemed essential to sustain high levels of performance and enable large problem sets in the areas of computational chemistry, image processing, and some finite element analyses. In a number of cases, the memory requirements exceeded those available from a single *Hawk* large-memory node (2 TB). Several projects indicated that shared-memory applications were being ported to distributed-memory architectures and were experiencing significant challenges in attaining acceptable performance. Questions remain regarding the purchase and support of future shared-memory capabilities for HPCMP users with known or forecasted requirements.

Supercomputing systems have evolved toward heterogeneous architectures of distributed shared memory and accelerators. More computational cores per socket and more sockets per node, along with graphics processing units (GPUs), have enabled this evolution. Users are increasingly turning to software developers to modify existing codes or develop new ones that are hybrid in nature to take advantage of the hardware to maximize performance. This trend is not unlike the one in the early days of massively parallel computing; however, it is interesting to note that fewer system options are available to users, as the hardware vendors are focusing product development toward distributed multicore nodes with or without GPU accelerators. Processor, memory, interconnect, and data storage technology are becoming nearly standard and similar among vendors, so features such as scalability, performance dependability, and energy efficiency are emerging as differentiators.

What can be done to better support users in their quest to achieve the best performance for their applications on any HPC system of today or the future? One possibility entertains a new shared-memory system to replace all or some of Hawk's capabilities. Another includes potential modifications and more emphasis on the existing Appro Utility Servers to include larger shared-memory nodes (note that GPU-based nodes already exist). Yet another idea incorporates more substantial support from the User Productivity Enhancement, Technology Transfer, and Training (PETTT) component of the HPCMP to help users modify codes, utilize existing tools, or create new tools that exploit the hardware. The AFRL DSRC is fully engaged in exploring these possibilities and would like to hear from users regarding their specific needs. It is important for users to identify their applications that will suffer from poor performance or will fail to run on particular systems. While the HPCMP will continue to provide the latest HPC technologies to its users, the Program will become more adaptable in providing innovative services and solutions that alleviate the burden users experience by having to re-engineer their applications and processes to keep up with the hardware. We welcome users' ideas for how we can work together to enhance the relevance and impact of HPC through new ways of doing business.

New Supercomputing Center

Today's supercomputers are solving some of the most complex scientific and engineering problems in the Department of Defense. From developing new fuels for weapon systems to generating weather forecasts for theater of operations, supercomputers are greatly reducing the time to solution or design for our scientists and engineers. While supercomputers continue to increase in capability, the power, cooling, and structural load support to host these supercomputers is also increasing.

To meet these new infrastructure requirements, the Air Force Research Laboratory Defense Supercomputing Resource Center (AFRL DSRC) is constructing a new supercomputing center in the newly built Information Technology Complex (ITC) at Wright-Patterson Air Force Base, Ohio. The ITC will provide state-of–the-art computing and a collaborative modeling and

ITC facility (front view)

simulation environment required for the rapid infusion of supercomputing resources to enhance weapon system life-cycle acquisition and support capabilities. Phase 1 consolidates computing and work space for modeling analysis and design for the Aeronautical Systems Center Engineering Directorate and a large supercomputing center for the AFRL DSRC. Groundbreaking for this new facility started in Oct 2010, and it will be the first of a five-phased effort to consolidate advanced computing functions at Wright-Patterson AFB.

In an effort to reduce the facility's impact on the environment, the ITC is constructed to meet qualifications for the Leadership in Energy and Environmental Design (LEED) Silver Certification. According to the U.S. Green Building Council, "LEED certification provides independent, third-party verification that a building, home, or community was designed and built using strategies aimed at achieving high performance in key areas of human and environmental health: sustainable site development, water savings, energy efficiency, materials selection, and indoor environmental quality." The ITC will incorporate high-efficiency HVAC equipment, modular expansion capability, and environmentally friendly building materials to host world-class supercomputing systems for years to come.

The AFRL DSRC is excited to be a part of this new facility and is looking forward to hosting its Technology Insertion 2011/2012 (TI-11/12) supercomputing system in 2012 here.


Future HPC facility

Spirit Coming to AFRL DSRC

In March 2012, GSA and the HPCMP announced a contract award of nearly 24M to SGI for the AFRL DSRC TI-11/12 system. The new SGI Altix ICE X supercomputer consists of 73,440 processor-cores (4590 nodes, 16cores/node, plus 18 nodes as hot spares) with over 146 TB of memory and over 2.7 PB of disk space. The processors are the latest Intel Sandy Bridge, rated at 2.6 GHz and 115W. The compute nodes will be contained within 32 racks with the disk storage and admin nodes in 24 additional racks. The system will be water cooled by SGI's M-Cell and cold sink technology, which delivers chilled water directly to each processor.

Ranking as one of the most powerful computers in the DoD and one of the top 20 in the world, the system's peak performance will be on the order of about 1.5 PFLOP/sec, about 3.5 times that of *Raptor.* Final acceptance of this system to be named *Spirit*, in recognition of the B-2, is expected in mid-October.

From the Director's Desk – Dr. Raju Namburu

On March 1, 2012, ARL accepted a new building as the new home for the ARL DSRC. The ARL DSRC team is working energetically to prepare the facilities to meet the deadlines associated with the acceptance of both classified and unclassified Technology Insertion 2011/2012 (TI-11/12) systems for this new facility. As opposed to our earlier ARL DSRC facilities, this building is a three-story building and has sufficient space to accommodate office spaces and both classified and unclassified high performance computing (HPC) systems. We have already moved some of the ARL DSRC management staff into this building and are planning to move the majority of the ARL DSRC staff before we accept new HPC systems.

Recently, we added more capability to the ARL DSRC by upgrading MRAP, our Cray XT5's AMD Opteron processors from quad core to hex core. This upgrade resulted in an increase of 10,400 processor cores to 15,600. ARL DSRC benchmarks with CTH software showed considerable performance improvement with the new Opteron processors. As a result of this increased capacity, the HPCMP released an additional 26.2 million processor core hours to the customer community.

Emergence of petascale and exascale HPC concepts has led to new research thrusts including power efficiency. Now, power efficiency is an important area of expertise for managing large-scale computing centers. The DoD HPC Modernization Program Office initiated a number of new initiatives to build this expertise across the DSRCs. As part of this initiative, the ARL DSRC continues its efforts with green technologies, and early experimentation results here are promising.

The first effort is the Energy Aware Scheduler (EAS). The EAS concept is based on powering down idle HPC nodes by taking advantage of the scheduling policy of DoD HPC systems. Typically, an HPC system supports numerous large parallel jobs, and parallel jobs reserve nodes until they have the required nodes to start execution. While waiting for reservation, the idle nodes can be powered down. That is, backfill scheduling is the central part of the scheduling policy. Based on experimentation, a reported savings of 1982 kWh was achieved over a period of 4 weeks on the SGI HPC system. The next step is to extend the EAS to other DoD HPC systems. The second effort is to reduce losses in circulation of cooling air by innovative design of enclosures. Based on the configuration of the facility, the ARL DSRC team designed smart enclosures that reduce cooling losses. The newly designed enclosure reduced the cooling load by 90 tons, translating to savings of more than $20K per year.

As always, we are committed to fully support the DoD user community by providing computational resources and expertise to help solve their research challenges.

Enclosure Reduces Cooling Load by 90 Tons

By Brian Simmonds, Mark Purdue, and Mike Knowles, ARL DSRC

Sometime during the planning for the Technology Insertion 2006 (TI-06) delivery, the integration team was discussing how to fit the 28 racks of compute nodes that were to be the then massive *MJM* cluster into the limited space of the computing facility. This was the largest system that had ever been located at ARL. How will we ever cool it? What if we arrange the racks in a big square, with the backs facing out, and perforated tiles in the middle? After 35 test fits, the cube arrangement was born.

The next TI, TI-09, brought an even denser 10,000 plus core SGI Altix ICE system to ARL. Each of the 21 cabinets of this system has 512 cores and draws 25.37 kilowatts of power. The cube arrangement had worked in the past, so why not give it a try? After convincing SGI of the merits, the system was installed in this configuration and named *Harold*. *Harold* performed as expected, and cooling was not an issue.

Maintaining the power and cooling systems in a supercomputer center is a full-time job. The ARL DSRC facilities staff is always at work tuning and adjusting the systems so that *Harold* and the other systems can stay comfortable and productive. It was during these interactions that the team noticed massive amounts of cool air by-passing the tops of the racks. Perhaps this could be an opportunity to tune the cooling system to be more efficient.

This being a research laboratory, the scientific approach was in order. With the use of a volometer, a device to measure moving air, they recorded airflow in cubic feet per minute (CFM) and temperature and humidity readings in multiple areas inside and outside the cube. This gave the facilities engineers a baseline for testing and evaluation of the changes.

Containment enclosure installed to block airflow over tops of system cabinets

A simple test was devised, using a single sheet of paper attached to the ceiling over the top of the system on all four sides. The excess air blew the paper horizontal, indicating excessive air by-passing the racks returning to the cooling systems. This was lowering the air conditioning system efficiency and basically wasting electricity and cooling. At this time, *Harold* was utilizing 224 tons of mechanical cooling and

over 99,000 cubic feet of air being delivered under the floor

Using the paper "test," the team shut down air handlers surrounding *Harold* one at a time. After shutting down the second one, the paper on the ceiling dropped considerably. Additional temperature, humidity, and airflow readings were recorded with no significant change in temperature on the discharge side of the racks where the heated air was being returned to the computer room air conditioners or CRACs.

The next step was to install a containment enclosure. This is a physical barrier that in effect seals the passage from the top of the system racks to the ceiling of the room, eliminating almost all airflow over the tops of the racks. The immediate result was that the enclosure door was hard to keep closed due to the cube interior being under excessive positive pressure. The team then decided to reduce mechanical cooling and airflow further by shutting down a third CRAC unit. Taking more airflow, temperature, and humidity readings, they noticed the contained cube system was still under a desirable slight positive pressure, and underfloor temperatures actually reduced by 4 degrees Fahrenheit.

All in all the mechanical cooling was reduced by 90 tons, and airflow was reduced by 40,140 CFM, resulting in an annual savings of more than $20k based on calculations using electricity rates and maintenance costs on the three CRACs. The cost for the materials, engineering, and installation of the enclosure was returned within just 80 days based on daily savings.

As all of the Centers become more and more powerful and the capabilities and the scale of the systems increase exponentially, it is encouraging that efficiencies and savings can be found in the simplest observations. We should all be vigilant as well as creative in our quest for cost reductions and resource savings. Thanks to the ARL Facilities Engineering Team for being good stewards of the DSRC resources.

Observations after Two Months of Energy-Aware Scheduler Experience

By Michael Knowles, Bill DeSalvo, and Kathy Smith, ARL DSRC

The Energy-Aware Scheduler (EAS) project was initiated in the summer of 2011 to investigate and develop the ability to save energy by controlling power to the nodes of various high performance computing (HPC) assets, under the direction of the DoD HPC Modernization Program (HPCMP). The EAS project is based on the premise that across the Program there is a potential to reduce power consumption by millions of kilowatt hours per year by powering down idle HPC nodes. The project was a collaboration between DoD personnel, Lockheed Martin (HPCMP prime contractor), Altair (PBSPro scheduler developer—used across all HPCMP assets), and Instrumental (dedicated project coordinator).

Starting in mid-December 2011, the EAS project team has been gathering statistics to determine the efficacy of EAS on a traditional HPC machine. The majority of commercial EAS implementations occur on smaller machines with "bursty" workloads. Systems with cyclical workloads, for example, are ideal candidates for EAS. These type systems tend to experience many extended periods in which nodes are idle. Cyclic and extended idle times are not generally seen at major HPC centers. Hence, the gathering and analysis of EAS performance data is crucial to understanding whether EAS on DoD HPC machines is a viable strategy.

Because DoD HPC machines are heavily utilized (typically, 85 to 90 percent busy), it is natural to question whether EAS will be able to identify enough nodes that can be powered off for a "reasonable" length of time in order to obtain value from EAS. The configurable parameter "power down delay" (PDD) can be adjusted on a site-by-site and machine-by-machine basis to define this "reasonable" time. For the purposes of this project, nodes are powered off only if it is known that they will not be needed by a user job for at least 60 minutes (default PDD). The predictive scheduling feature of PBS is therefore a critical requirement of EAS.

Since HPC workloads include numerous large parallel jobs, backfill scheduling is a central component of the DoD scheduling policy. Parallel jobs reserve nodes until they have acquired enough nodes to start execution. When these nodes are idle and projected and won't be required for the PDD duration, EAS can decide to power them off. This is also true for advance reservations. While waiting for a reservation to start, EAS can power down idle nodes that have been allocated to that reservation. The EAS has been extensively designed to account for these HPC scheduling scenarios.

During preventive maintenance (PM) periods, system administrators need to ensure that all jobs have completed before powering off the entire system. Running jobs are allowed to complete, and new jobs are not started as PM approaches. EAS can be used to automatically power down nodes as they become idle. It is expected that this would result in substantial power savings.

One concern that is also under investigation regarding these actions is the long-term reliability of the equipment. As most new systems are diskless, and system components have become more reliable over time, repeatedly powering off nodes has become less of a concern. The EAS team is aware that the equipment reliability is absolutely a criterion in the evaluation of this capability and is working closely with vendors to track node reliability over time.

Software installation and testing began at the ARL DSRC on a 12-node SGI Altix ICE test system. After a few days of running in "simulation" mode, EAS

was reconfigured to run in what is called "live" mode. Nodes were powered off (and on) by EAS based on job demand and a suite of control parameters. Testing of this early beta version of the software revealed that EAS was now able to account for nodes reserved for backfill jobs and advance reservations. In early December 2011, EAS was installed and run in "simulation" mode on a 1344-node SGI Altix ICE production system (*Harold*). After a week of testing, a portion of the large production system switched to EAS "live" mode on December 8, and the EAS team began gathering usage statistics. EAS log files contain entries that describe when a node is powered off or powered on. A Perl language script was developed to generate a report on the number of nodes power cycled and the total number of kilowatt hours saved by EAS. The results of the testing are described in Tables 1 and 2.

After measurements taken by the hardware vendor (SGI), it was determined that the power draw for each node is 147 watts. During the month of January 2012, there were two factors that diminished the effects of EAS. First, EAS was only active during non-prime-time hours (18:00 – 08:00) due to a PBSPro 10.4 problem that was exacerbated by the EAS software. In addition, only a portion of the 1344 nodes was subject to EAS control. One particularly useful feature of EAS is the ability to designate which nodes are eligible for EAS control. This made testing on a large system feasible, as potential problems could be limited to a small partition of the larger system.

Table 1

Dates	EAS Hours Saved	EAS Hours **Saved x 147w**
01/02 – 01/08	720	105 kWh
01/09 – 01/15	682	100 kWh
01/16 – 01/22	431	63 kWh
01/23 – 01/29	1241	182 kWh
Total		393 kWh

In Table 2 (February), it is clear that major improvements and adjustments have been made to EAS. First, an upgrade to PBS v11.2 minimized inefficiencies in the EAS-PBS interface. This was followed by expanding EAS monitoring to the entire 1344 nodes as well as extending EAS usage to include prime time. Note that data for the first week of February is missing from Table 2. During this week, EAS accounting was unreliable due to preparation for the PBS upgrade, EAS accounting file changes, and other EAS improvements. It is of particular interest that the PBS upgrade took place on February 8. On February 7, it was necessary to "drain" the system of batch jobs, as this PM approached. On February 7 alone, 1457 node-hours (214 kWh) were saved.

Table 2

Dates	EAS Hours Saved	EAS Hours **Saved x 147w**
02/06 – 02/12	3971	583 kWh
02/13 – 02/19	4282	629 kWh
02/20 – 02/26	5240	770 kWh
Total		1982 kWh

The EAS project team is currently in the process of assessing the viability of the technology to other DoD HPCMP sites, and other architectures (XE6, Utility Servers, and Dell Clusters within the HPCMP) are being evaluated. The investigation is scheduled to be completed in September 2012.

From the Director's Desk – Dr. Robert S. Maier

Academia and industry are important sources of innovation for the Department of Defense. New ideas and approaches are often developed in a public setting and adopted by DoD researchers for use in applied research projects. The HPCMP supports this technology transfer process with the Open Research System (ORS), a supercomputer designated for public, unclassified research, operating on a network separate from its sensitive systems. The current ORS is a 728-node Cray XE6, located at the ERDC DSRC in Vicksburg, Mississippi.

The ORS provides a collaborative environment for public research. DoD researchers can work with scientists from outside agencies on DoD-sponsored projects, sharing group permissions to facilitate data exchange on the system. Only data classified for public release may reside on the ORS.

The security requirements for working on the ORS are similar to those for any of the DoD unclassified but sensitive systems: a U.S. citizenship or a valid visa, a visit request to ERDC Security, a DoD Government sponsor for the research project, and annual Information Assurance training. A National Agency Check Inquiry (NACI) is not required to use the ORS. This also allows new DoD researchers and contractors to work on the ORS while waiting for a clearance to work on DoD sensitive systems.

To protect the security of our systems, the ORS is maintained on a separate network with a unique domain. The security credentials necessary for access to the ORS are not valid for access to sensitive DoD systems and vice versa. This prevents accidental "spillage" of sensitive data onto the ORS from other DoD systems.

The ERDC DSRC operates the ORS for use by DoD-sponsored projects where the data are cleared for public release. In Fiscal Year 2011 (FY11), over seventy million CPU hours were used by DoD researchers and their partners. In late FY12, the ORS will be upgraded to a newer system with faster processor cores, and will continue to serve DoD as a platform for basic research.

Green HPC at the ERDC DSRC

By Paula L. Lindsey, Systems Integration Lead

The ERDC DSRC began a move towards Green HPC in 2007 with the installation of 20 Liebert® flywheel UPS systems. The flywheels eliminated the need for lead-acid batteries to provide backup power for the critical computing and network equipment. This technology uses the inertia of a spinning carbon-composite "wheel" to provide ride-through power until a generator starts. The flywheel floats in a vacuum and uses bio-degradable vegetable oil for lubrication. Flywheels can operate in temperatures up to 105 degrees, thus eliminating the need for the chilled air environment required by lead-acid batteries. Every component in the flywheel can be recycled at the end of its 20-year lifecycle.

The next Green HPC initiative at the ERDC DSRC involved installation of S&C Electric's line-interactive, medium-voltage UPS as part of the power plant infrastructure servicing the RDT&E Shelter in 2008. The S&C UPS is connected to a high-speed transfer switch and is only used when utility power is lost. Unlike an online system utilizing lead-acid batteries, the S&C UPS is built for outdoor use. This type of system is 98 to 99 percent efficient since it does not convert the power from AC to DC to AC like the online system. Batteries in a line-interactive system are only connected to the DC bus when power is lost. Since the batteries are not continuously connected to the DC bus, there is no ripple effect, thus maintenance-free batteries can be used. Batteries in online systems require monthly, quarterly, and annual maintenance. Line-interactive systems do not inject harmonics into the system and thus do not require the generators be over-sized to compensate for the harmonic currents. The line-interactive system we chose is designed to be installed outdoors. No special shelter and cooling system are required. The increased efficiency, longer lasting batteries, and outdoor installation reduce the carbon footprint.

In 2010, the ERDC DSRC installed its first 480 V computer system – a Cray XE6. This system requires only one 480 V power connection per rack. Typical 208 V systems require two, or more, power connections. Since wire size for the circuits is determined by amperage, and not voltage, one 480 V, 100 A circuit uses less copper wire than two 208 V, 100 A circuits required by other systems. Systems using 480 V connections typically do not use plugs to connect the circuit to the computer; instead the circuit is directly wired to the computer's power supply. Systems using 208 V power often use pin-and-sleeve plugs. Use of 480 V circuits results in reduced installation costs, less wire under the floor and thus less copper use, and less line loss in the circuits.

In late 2010 and early 2011, the DSRC commissioned a cooling study of the RDT&E Shelter. This study would measure air temperature, pressure, and flow rate throughout the room, both above and below the raised floor. Models of the room, as well as the equipment within the room, were input into CoolSim® software along with the room airflow depicted based on the air measurements. These airflow depictions indicated where hot spots were occurring in the room and assisted with

better alignment of perforated tiles in specific areas.

As a result of a DOE power and cooling audit in the summer of 2011, we were able to raise chilled water temperatures to the RDT&E Shelter 4° F, resulting in reduced chiller load. Airflow readings taken during the DOE audit highlighted the need to change the location of some of the perforated floor tiles to provide more efficient cooling. The study also revealed that reducing airflow from the computer room air handling units caused the computer's fans to go to high-speed mode. Even though the air handlers were using less power, the increased power draw caused by the fans in high mode caused an overall increase in the number of kilowatt hours of power being used. Returning the air handlers to their original settings reduced the power draw. The study provided an excellent example of how saving power on one type of equipment could cause an overall rise in power usage due to power increases in another area.

In 2011, the HPCMP funded a Green HPC Initiative. As a result of this initiative, and after researching options for how best to implement green options that would make an immediate impact, the Daikin-McQuay magnetic chiller and Baltimore Air Coil evaporator towers were selected. Magnetic chillers, such as the Daikin-McQuay pictured below, utilizes magnetic levitation to "float" the compressor shaft and eliminate the need for oil in the chiller. Magnetic bearing-type chillers are up to 30 percent more efficient than traditional oil-filled, air-cooled chillers utilizing metal bearings. The magnetic chiller works in unison with the Baltimore Air Coil closed-loop water cooling tower to remove heat from the chiller, while the chiller provides cooling to the RDT&E Shelter. During winter months, when temperatures are cooler, the cooling towers can work alone to provide chilled water to the supercomputing center. During these months, the chiller can actually be turned off to save energy costs.

The energy savings resulting from the actions described in this article means less funds are spent on overhead costs. These savings increase the ability of the ERDC DSRC to retain intellectual capital and to invest in additional technology.

From the Director's Desk – David Morton

The MHPCC DSRC has been investigating and implementing a number of new HPC delivery and energy savings paradigms since my last article. Those efforts include Dedicated Support Partitions (DSPs), a web-based "Software as a Service" (SaaS) model delivery for HPC applications. On the energy front, we are working toward reducing energy used to deliver computational resources as well as efforts toward generating a substantial portion of our overall energy needs from renewable resources. A quick update on these efforts follows:

Dedicated Support Partitions (DSPs) were designed to provide resources to projects in need of dedicated processing for a significant period of time to accomplish work that could not otherwise be done in a shared resource environment. A significant portion of this work is being done on the Dell Quad Core Xeon Cluster (*Mana*) with 9216 cores at the MHPCC DSRC. The requests for DSPs continue to mount at the MHPCC DSRC with the provision of outstanding services. Several users have requested more than 2000 CPUs to run their project. It appears to be particularly well suited for debugging and regression testing of large HPC software applications. Computational areas utilizing this special program are Computational Research and Engineering Acquisition Tools and Environment (CREATE), which includes CREATE Air Vehicles–FIREBOLT, CREATE Air Vehicles–HELIOS, CREATE Air Vehicles–KESTREL, CREATE Radio Frequency, CREATE Air Vehicles–SHADOW OPS, and Joint Strike Fighter–SHIP INTEGRATION. An article on page 5 describes how you can request a DSP at the MHPCC DSRC.

At the DoD DSRCs, in our current paradigm of inexpensive commodity large-scale systems, some of the greatest challenges we face are to make our incredible resources available to users in ways enabling their computational problems to be solved faster and easier. The current delivery model requires high bandwidth connections, ability to locally install software kits, and a detailed understanding of Linux commands, batch scheduling, etc. We are basically interfacing with our systems in the same way I did when I started working at Cray Research Inc. in 1989. If you think about that, that is an incredible long time for something in high technology to remain static. While the current operating paradigm is incredibly powerful, which I believe we should continue to deploy for a certain class of users, it does limit the ability of the HPCMP to impact certain types of users and applications in the broader DoD community.

The HPCMP Enterprise Portal employs the "Software-as-a-Service" (SaaS) model to remove barriers to access HPC resources and data. MHPCC is working with the Army Research Laboratory (ARL) DSRC, the USACE Engineer Research and Development Center (ERDC) DSRC, the CREATE Team, and HPCMP leadership to develop and deploy these new interfaces. As I write this article in February, we are planning on deploying an initial test system delivering a MATLAB drag and drop portal and a CREATE Kestrel application on Mana in about a month. We'll be discussing our successes and issues and giving an update at the Users Group Conference in June.

On the energy front, MHPCC has been working on reducing the amount of energy required to do a computer calculation. We have participated in the Technology Insertion 2011/2012 acquisition process, and while I cannot detail the technology as the award has not been released as I write this article, we are hopeful that our next-generation system will be able to deliver about 50 percent more performance than our current *Mana* system while using 50 percent less energy. This is done through next-generation efficient cooling and power technologies at the server level, as well as more efficient computing center technologies. Energy efficiencies and power usage are becoming more and more important to supercomputing, as energy is becoming a major cost driver and increasingly seen as a huge barrier to successfully deploying the next-generation exascale supercomputers.

MHPCC is also working on the generation side of the energy equation. We have deployed a 100 KW research solar array using advanced concentrated triple junction photovoltaic technology that is currently powering part of our computing center. That is just the start. We are working toward a much larger solar array that could power all of our entire facility during daylight hours. This will use more conventional photovoltaic technology. The challenges are large, but we are confident that we will be successful.

As you can tell, the MHPCC has many interesting efforts underway. We thank the HPCMP for their fantastic support of our efforts and look forward to delivering some of the innovative technology that is required for continued Program success.

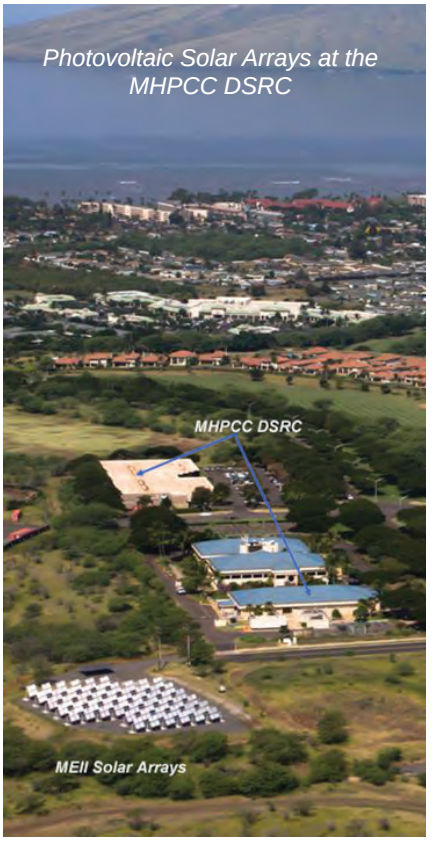
MHPCC DSRC Energy Efficiency Update

By Captain Joseph Dratz, AFRL Technical Director, MHPCC DSRC

Escalating energy consumption is straining the ability of military supercomputing centers to deliver cost-effective support to the warfighter in an environmentally sound manner. Many experts believe that energy and energy costs are a key limiting factor in scaling to next generation exascale supercomputers. Within the HPCMP, energy costs take an increasing amount of the limited budget that could be better used for service delivery and R&D efforts. Proactively, the HPCMP has been working to address this future problem with a number of energy-related initiatives. Many of these initiatives have been started at MHPCC due to the high cost of electricity here. Since October 2009, the MHPCC DSRC has partnered with several Department of Energy labs through the Federal Energy Management Program (FEMP) to conduct four comprehensive supercomputing center efficiency assessments. Over those years, MHPCC has significantly reduced energy consumption through segregation of hot and cold isles, increasing the chilled water set point, and installing more efficient chillers and CRAH units.

The MHPCC DSRC is continuing to tackle energy consumption through renewable generation and demand reduction. The Maui Solar Initiative (MSI) is a partnership with Navy Facilities Command Pacific; U.S. Army Corps of Engineers, Honolulu District; ERDC; and the Maui Electric Company to provide an approximately 15-acre solar farm interconnected with the MHPCC. MSI builds on the experience and data gained from the 100 kW Maui Energy Improvement initiative. That existing system is still providing PV power to the MHPCC DSRC and serves as a long-term test bed for emerging solar technology.

Demand reduction at MHPCC is being tackled through the Energy Efficient Computing (E2C) project. Sponsors include the HPCMP, ONR, and the Deputy Assistant Secretary of Defense Rapid Fielding Directorate who has funded the monitoring and evaluation portion of MHPCC's demand reduction efforts. E2C is a partnership between ERDC, ARL, ONR, PACOM, Lawrence Berkeley National Lab, Pacific Northwest National Lab, and MHPCC to field, test, and evaluate the latest in supercomputing center efficiency technology. Each organization brings specific expertise and equipment. ERDC is providing building instrumentation; ARL is working on node power management software; and ONR is providing power factor improvement through Magnetic Energy Recovery technology. The DOE labs serve as design consultants and outside verification of efficiency.



Photovoltaic Solar Arrays at the MHPCC DSRC

Finally, PACOM is working hard to coordinate efforts and plans to use specific E2C recommendations in theater.

Both MSI and E2C are well into the planning stage. Leasing, surveying, and NEPA activities are occurring simultaneously on the PV farm. Additionally, the physical plant upgrades to accept the E2C system procured through the TI process are being designed and planned.

From the Director's Desk – Tom Dunn

This issue of *HPC Insights* coincides with the 2012 Users Group Conference, which is being held in our back yard, New Orleans. The region has undergone several years of renewal and rebuilding, and the Navy DSRC finds itself undergoing significant changes to our long-established HPC environment. Our new HPC systems will be coming online during this fiscal year, and we've been preparing numerous parts of our infrastructure in advance of these larger, faster HPC capabilities and the ever-increasing amounts of data those capabilities will produce. As part of green energy initiatives, we are removing aging centrifugal chillers and are replacing them with high-efficiency magnetic bearing chillers. The Navy DSRC staff have also implemented enhancements in storage and network infrastructure, as detailed by Bryan Comstock and Rob Thornhill on page 26.

We welcome the opportunity to help our users enhance their use of the Navy DSRC's HPC environment in both traditional and innovative ways. We are in the midst of a significant effort to provide leadership, guidance, and innovation to the Navy's operational oceanography modeling community by partnering with the Naval Oceanographic Office (NAVOCEANO), Navy Research Lab-Stennis (NRL-Stennis), and the HPCMP's User Productivity Enhancement, Technology Transfer, and Training (PETTT) program in modernizing the entire operational modeling workflow from model development to model deployment to daily model runs that produce products. This effort will add a robustness and redundancy to NAVOCEANO's operational workflow that has long been out of reach, despite the significance of the requirement to deliver model forecast products on a 24x7, disaster-resistant basis. Christine Cuicchi details the effort from its inception to its current state on page 4.

By the end of calendar year 2012, the Navy DSRC will experience a complete turnover of HPC systems, retiring our 2008-installed Cray XT5 *EINSTEIN* and IBM Power6 *DAVINCI*, and our long-running 2006-installed IBM Power5+ *PASCAL*. At press time, it is expected that our new HPC systems will deliver an increase to our computational capacity by 300 percent or more. With the addition of our Utility Server (US), Center Wide File System (CWFS), and additional HPC Enhanced User Environment (HEUE) components, the Navy DSRC will have an exciting new array of capabilities available for our users. We are looking forward to broadening both our HPC and support services to our user community.

Incoming HPC Systems Kick Off Navy DSRC Infrastructure Upgrades

By Bryan Comstock, Navy DSRC HPC Systems Analyst, and Rob Thornhill, Navy DSRC Facilities Lead

Preparation for the imminent arrival of the Navy DSRC's new HPC systems and the expected increase of data movement and data storage has taken shape in several projects to enhance the Center's networks, mass storage infrastructure, and facilities. These enhancements ensure that the Navy DSRC user experience continues to feature resilient and highly available systems and data storage.

Storage

The Navy DSRC storage system administrators have undertaken significant enhancements and upgrades to the Center's mass storage infrastructure. One upgrade certain to boost the availability and performance of our SUN M5000 archive server, *Newton*, is an increase in the amount of Random Access Memory (RAM) available. Previously, *Newton* had 32 GB of RAM that supported every user of the Navy DSRC mass storage system. *Newton* has been upgraded to 96 GB of RAM, which will help to mitigate previously vexing issues such as physical server memory depletion due to the amount of memory in use supporting the archive tape drives.

Another storage infrastructure enhancement involved the connectivity of the archive server *Newton*, the disk cache, and the StorageTek SL8500 tape library. The Navy DSRC now has two Storage Area Network (SAN) fabrics supporting these mass storage components, which allows the Center to provide redundant archive storage connectivity. The Center also swapped a Brocade 48K switch for two Brocade 5300 switches, which will provide increased bandwidth by supporting 8 GB fibre channel.

Networks

The Navy DSRC Network Engineering team recently planned and implemented upgrades to the Center's HPC system and storage network infrastructures,

Network engineer Benjy Rigney

involving the replacement of legacy Cisco hardware with an Arista Networks solution. The team's evaluation of various commercial products found that Arista's provided the best mix of lower port-to-port latency, lower cost per port, and higher 10 Gigabit Ethernet port density in a smaller, more energy efficient chassis to serve the Center's current and future requirements

A key Arista component that provides improved service to our HPC users is their Multi-Chassis Link Aggregation (MLAG) feature. In traditional switched Ethernet environments, multiple Layer 2 links between switches cannot both be utilized in an active/active mode. One link is active and the other in a standby mode in the event of a failure of the primary link. MLAG removes this restriction and allows the utilization of all interconnects in an active/active mode. HPC center networks require both the highest level of network bandwidth and reliability. MLAG employs link aggregation to spread network traffic across a pair of supercomputing center switches in order to deliver system-level redundancy as well as network-level resiliency. These enhancements provide better network service to our HPC user community.

The upgrade was performed in two stages in order to minimize impact to the user community: stage one involved the HPC infrastructure, followed by stage two involving the storage network infrastructure. Throughput tests performed before and after the upgrades demonstrated that a significant improvement in network performance was realized by decreasing internal LAN latency, thereby resulting in faster data transfer speeds. Internal data transfers between the Center's HPC compute systems and mass storage have improved by as much as 50 percent over the previous environment.

Facilities

Upgrades are underway at the Navy DSRC that will provide significant improvements in the energy efficiency, flexibility, and redundancy of its chilled water plant. The existing single temperature system is being modified to be able to produce both low and medium temperature chilled water in separate cooling loops. This portion of the upgrade primarily consists of adding a 300-ton, high-efficiency magnetic bearing chiller to provide the low temperature cooling typically required by air-cooled HPC equipment, and replacing a pair of aging chillers with a pair of

Facilities engineers Will Cook and Rob Thornhill

700-ton, high-efficiency magnetic bearing chillers in order to provide the medium temperature cooling typically required by water-cooled HPC equipment. Initially, the 700-T chillers will be configured in an N+1 rotation, with either of them being fully capable of supporting the medium temperature requirement expected from our Technology Insertion 2011/2012 (TI-11/12) HPC systems.

A major project milestone was accomplished recently with the successful modification of an existing tie-in from the neighboring NAVOCEANO chilled water plant. This modification will provide the DSRC with full redundancy to support the new Low Temperature water loop.

These upgrades position the Center to accept current and forthcoming TI systems and increase overall energy efficiency. Additionally, all of these upgrades provide improved availability to all users of the Navy DSRC resources by ensuring that failure of any single component of the chiller plant will not force the Center to shutdown *EINSTEIN* (Cray XT5) or any other HPC systems supported by the plant. Work on the chilled water plant is being overseen by Rob Thornhill, Will Cook, and Leo Foster, all of the Navy DSRC.

Upgrades performed on the SUN M5000 archive server, Newton

Facilities personnel oversee implementation of Center's secondary "cold" chilled water loop that will increase efficiency and flexibility

Installation of a pair of cost-saving, high-efficiency 700-ton magnetic chillers begins

Remodeling of the computer operations room will improve system visibility and provide additional floor space

Recounting SC11

By Chuck Abruzzino, Graphic Designer, Air Force Research Laboratory DoD Supercomputing Resource Center, Wright-Patterson Air Force Base, Ohio. Additional images by Lynn Yott, Multimedia Specialist, Navy DoD Supercomputing Resource Center, Stennis Space Center, Mississippi

"Connecting Communities through HPC" served as the theme of the 24th premier international conference on high performance computing, networking, storage and analysis held at the Washington State Convention Center in Seattle on November 12-18, 2011. SC11 featured "the interdisciplinary thrust of data intensive science," according to its web page. Scott Lathrop, education director for the Blue Waters Project at the National Center for Supercomputing Applications at the University of Illinois, served as General Chair of the conference.

The Department of Defense was represented with a booth constructed and manned by team members of the DoD High Performance Computing Modernization Program and its Supercomputing Resource Centers located throughout the country. Approximately 11,000 people attended the conference, which is sponsored by the IEEE Computer Society and ACM (Association for Computing Machinery). The SC12 Conference will be held in Salt Lake City, Utah, November 10-16, 2012.


Success Stories
BLUE ARC
Collaboration at its finest

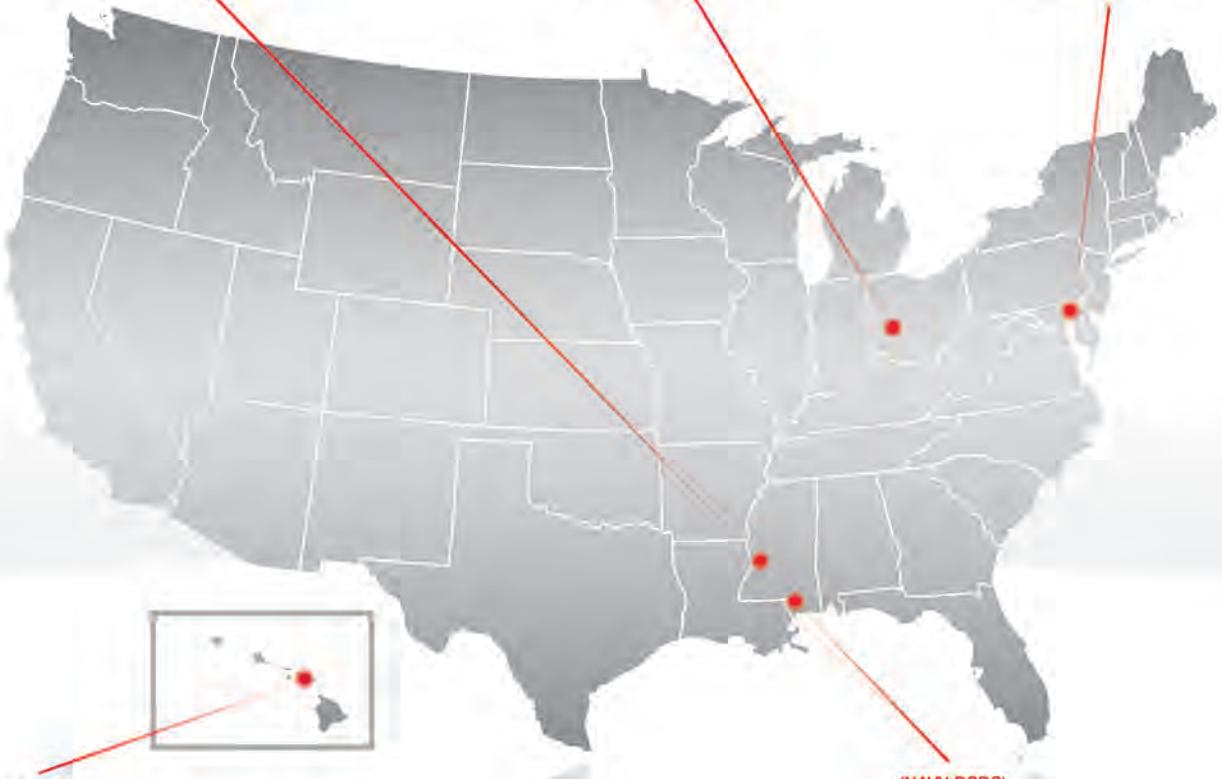
DOD HPC

DEPARTMENT OF DEFENSE
HIGH PERFORMANCE COMPUTING
MODERNIZATION PROGRAM

(ERDC DSRC)
US Army Engineer Research and Development Center
Vicksburg, MS,
http://www.erdc.hpc.mil/

(AFRL DSRC)
US Air Force Research Laboratory
Wright-Patterson AFB, OH,
http://www.afrl.hpc.mil/

(ARL DSRC)
US Army Research Laboratory
Aberdeen Proving Ground, MD,
http://www.arl.hpc.mil/



(MHPCC DSRC)
Maui High Performance Computing Center
Kihei, Maui, HI,
http://www.mhpcc.hpc.mil/

(NAVY DSRC)
Navy DoD Supercomputing Resource Center
Stennis Space Center, MS,
http://www.navo.hpc.mil/

Scientists and engineers throughout the U.S. leverage the capabilities of the High Performance Computing Modernization Program (HPCMP) to solve the most time-consuming computational problems. They know that the HPCMP's supercomputing centers continually architect, deploy, and sustain their equipment to deliver world-class network and supercomputing capabilities, resulting in simulations with greater resolution and faster results than achievable on conventional workstations. You too can gain access to these powerful resources by calling to register for an account!

CCAC Accounts Center
1-877-222-2039

http://www.hpc.mil