ERDC MSRC

Major Shared Resource Center

RESOURCE

A Novel Approach to Computational Stream Modeling, Restoration, and Simulation Using HPC

HIGH PERFORMANCE COMPUTING FOR THE WARFIGHTER.

SPRING 2008

from the director . . .

David Stinson
Acting Director, ERDC MSRC

"The only thing constant is change." Haven't we all heard that statement time and again? This seems especially true of the Engineer Research and Development Center Major Shared Resource Center (ERDC MSRC). I have chosen this edition of the *Resource* to point out some of the changes in earlier years and especially to bring to mind the dynamics of the last year or so. The growth in our program has been painful at times, but the result has been a stronger more resilient program, better equipped to serve the Department of Defense's (DoD) force of engineers and scientists.

Our history began back in 1993 when the Waterways Experiment Station Information Technology Laboratory (WES ITL) rechartered its Army Supercomputer Center as the first DoD Major Shared Resource Center (MSRC). Things really began to change in 1995 when over 70 people representing all military services converged in Arlington, Virginia, to serve on a source selection evaluation board for the first of many large high performance computing acquisitions for the High Performance Computing Modernization Program (HPCMP). As a result of a 1-½ year effort, Nichols Research Corporation was awarded the first large integration contract to supply services to the Corps of Engineers Waterways Experiment Station (CEWES) MSRC, and Steve Adamec was chosen as the first MSRC Director. Since that time, we have had four other Directors/Acting Directors of the MSRC; our host laboratory, ITL, is now on its third Director (see inside article on Dr. Reed Mosher) and along with the rest of WES is now part of ERDC; we have acquired computers from most major vendors including IBM, SGI, Cray, and HP/Compaq; and our site support contractor has changed from Nichols Research Corporation to Computer Sciences Corporation and most recently to Lockheed Martin.

In April 2007, I became Acting Director of the ERDC MSRC and can say that this last year has been arguably one of the busiest years since the inception of the MSRC. Budget cuts, consolidation of the customer assistance centers, consolidation of the scientific visualization centers, resequencing of the technology insertion (TI) acquisitions from a 2- to a 3-year acquisition cycle, and replacing the existing Millennia Task Order contracts at the four MSRCs with a consolidated Next Generation Technical Services Contract (NGTSC) covering all four MSRCs have all been major challenges. With the March announcement of Lockheed Martin Infrastructure Services (LMIS) as the winner of the NGTSC, John West, our most recent full-time MSRC Director, announced his resignation from the Government to become the LMIS Site Technical Lead at the ERDC MSRC.

Other changes occurring over the past year include our TI-07 acquisitions, encompassing an upgrade of our 4096-node Cray XT3 to dual core and the acceptance of our 2152-node quad-core Cray XT4, which is still in progress. Construction is underway to provide scalable power and cooling along with a new computer room with 10,000 square feet of raised floor for TI-09. This new construction will position the ERDC MSRC to handle any future acquisitions coming down the pike and will be a showcase for the ITL. This is all good news for our users who will now have access to some of the largest and fastest computers on the planet. Could petaflop computing be right around the corner? Time will tell, but one thing that we can count on for sure is that t**he dynamics are certain to continue!**

About the Cover: High performance computational fluid dynamics: Velocity vectors showing flow around an embedded roughness element as part of stream restoration modeling. For related story, see page 2.

Contents

from the director . . .

Computational Stream Habitat and Flow Modeling

By Dr. Jeffrey B. Allen, ERDC Information Technology Laboratory (ITL), Dr. David Smith, ERDC Environmental Laboratory (EL), Dr. Owen Eslinger, ERDC ITL, Miguel Valenciano, ERDC ITL, Dr. John Nestler, ERDC EL, and Dr. Andrew R. Goodwin, ERDC EL

Introduction

The fields of fluid dynamics, ecology, and fluvial geomorphology all converge at a certain physical scale. Thus ecosystem analysis must address this convergent scale as well as integrate information across disciplines. This challenge is particularly evident in the field of stream restoration where engineers, ecologists, and fluvial geomorphologists all must interact during project planning and execution.

This interaction is critical to improve the success rate of individual restoration projects. For example, over the past decade, approximately 15 billion dollars have been spent on stream restoration efforts in the United States, and the pace of spending is expected to increase in the near future [1]. Such projects typically cost on the order of $100,000/km to implement [2]. Unfortunately, preliminary results indicate that there is a high failure rate of these projects, and there is a developing consensus that better planning and execution is required. These costs do not include estimates of lost economic activity (such as lost power generation) associated with establishment of environmentally based flow schedules. Predictably, the engineering and biological disciplines are deeply divided on what better planning and implementation actually entails.

Representing rivers at the scale where fluid dynamics, ecology, and fluvial geomorphology converge requires high-resolution physical and biological models. For example, fish navigation within a river is related to its ability to separate types of flow resistance such as form versus friction drag [3] [4]. While this may not be germane to a hydraulic simulation alone, it is critical to represent both sources of resistance explicitly if an ecological simulation is required.

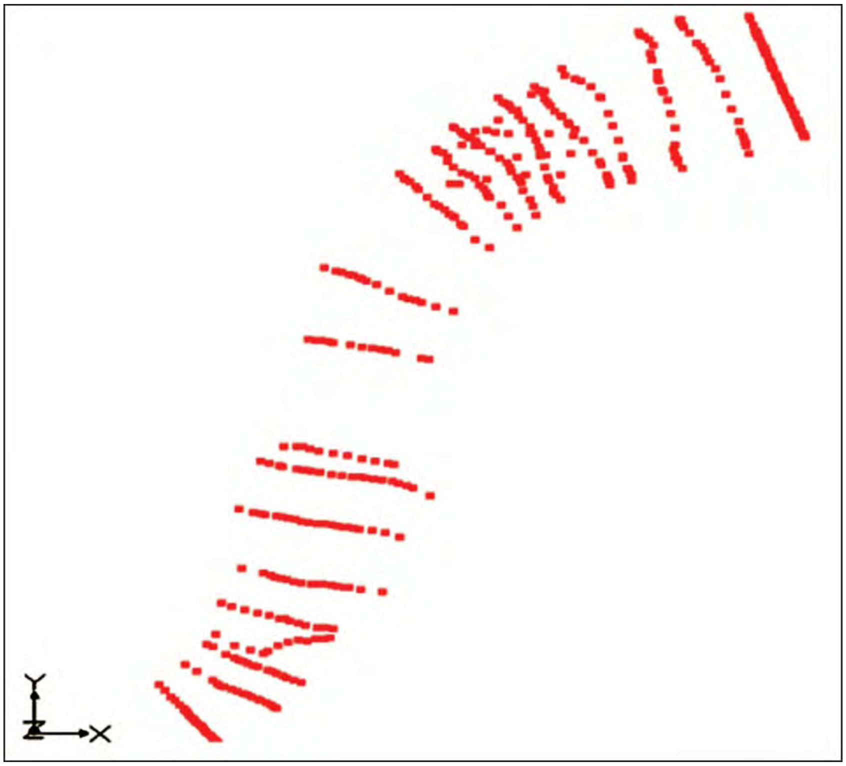


Figure 1. Merced River, initial field data

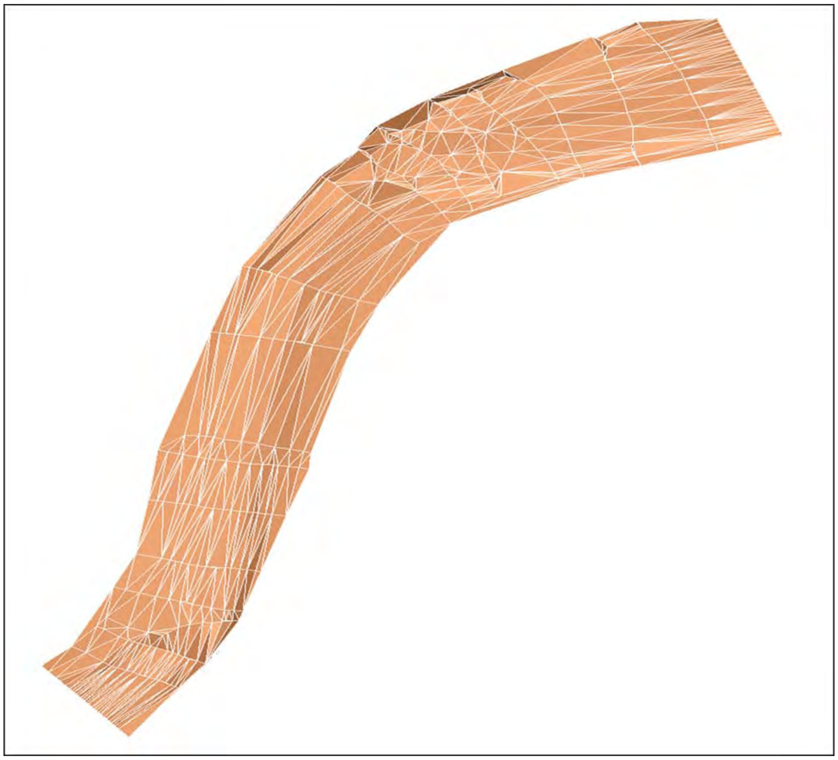


Figure 2. Initial triangulation

Fortunately, the rapid advances and availability of HPC resources, along with the increased sophistication of both in-house and commercial software, have made the creation of such models not only possible but also increasingly more efficient. The authors have identified several challenges that when overcome will improve the ability to seamlessly represent physical and ecological aspects of stream restoration:

1. Derive a realistic representation of a natural streambed from an initial coarse set of field measurements.
2. Freely deform and embed large roughness elements within the surface (boulders, large woody debris, root wads, etc.).
3. Mesh the surface and its surrounding volume in accordance with relevant physical length scales described above.
4. Obtain an accurate flow field solution.

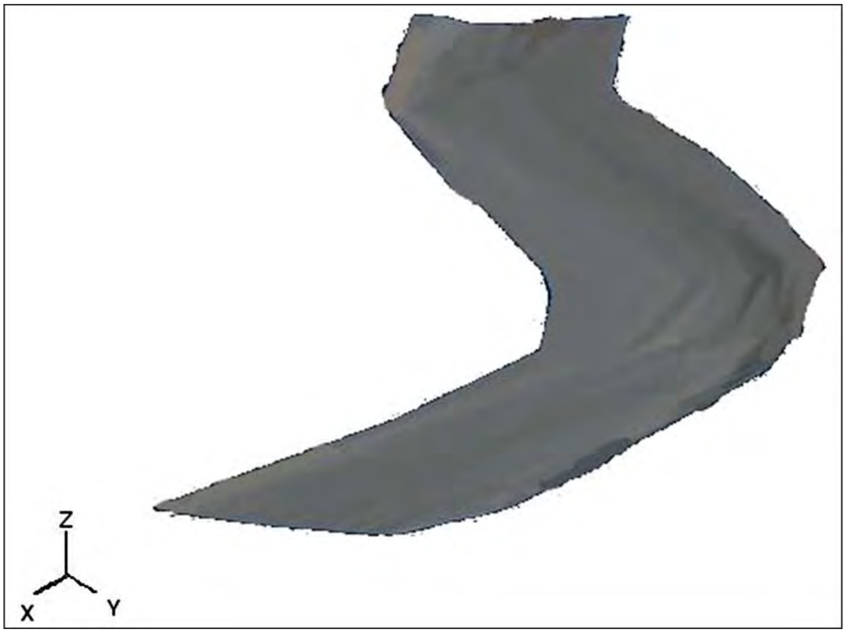


Figure 3. Merced River, refined geometry

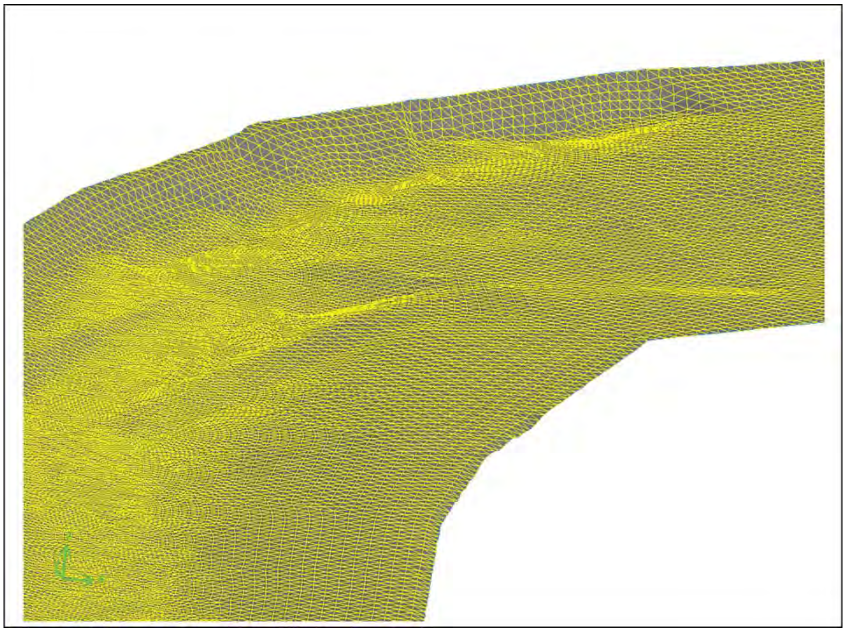


Figure 4. Merced River, refined mesh

The information below generally discusses each of these challenges and their respective solutions (as presently conducted by the authors) using field measurements taken from one of four study sites (S1) along a 1.5-mile stretch along the Robinson Restoration project of the Merced River (California) [5].

Creation of Realistic Streambed Representations from Coarse Field Data

The field data, consisting of 448 spatial, streambed measurements, were taken over a mean length, width, and depth of approximately 1019 ft, 152 ft, and 5.4 ft, respectively. The field data coordinates as well as the corresponding initial triangulation (surface mesh) are shown in Figures 1 and 2, respectively.

The relative coarseness of the initial field data clearly prohibits physical length scales appropriate to the requirements of computational fluid dynamics (CFD) or, for that matter, any of the integrated disciplines described above. By way of refinement, the authors utilized the conform utility of 3ds Max [6] to create a completely new geometry that retains the geometric contours of the original without being limited to the coarse number of original data points. The process involves overlaying an initial, planar surface over the contoured surface and "fitting" or conforming it to match the desired contours of the original geometry. Indeed, the number of points corresponding to the new geometry can be further refined in accordance with a user-specified level of tolerance. The resulting streambed geometry and surface mesh (consisting of approximately 1.0E6 nodes) are shown in Figures 3 and 4, respectively.

Free Form Deformation (FFD) of Streambed Surfaces

The general technique of FFD, first developed by Sederberg and Parry [7], allows surface primitives or volumes of any type or degree to be deformed. The technique is based on cubic Bezier basis functions or trivariate Bernstein polynomials. Object deformation is conducted via the manipulation of a prescribed set of control points.

Figure 5 shows the results of applying an in-house developed, FFD algorithm to a set of 28,869 Light Detection and Ranging (LIDAR) points using Bezier basis functions and 64 control points to dictate the relative amount of surface displacement.

Because of its inherent utility, FFD functionality is available in a variety of commercial products [2],[4]. Its utility allows for the creation of any number of global or localized deformations including depressions, elevations, stream embankments, and many other needed stream habitat formations. Figure 6 illustrates several of these deformations applied to the newly refined mesh of S1 using 3ds Max [6] (Figure 6, inset shows S1 prior to deformations).

Addition of Large Roughness Elements and Meshing

The utility of 3ds Max [6] also allows for the creation and embedment of large roughness elements (e.g., rocks, large woody debris, root wads) into the newly refined surface geometry. The elements themselves may come from any number of sources including commercial or in-house database libraries [6],[8], three-dimensional laser scan images, or simple freehand creations. These imported objects may be further deformed and manipulated as desired (using FFD) and made to conform to the underlying structure of the streambed.

From 3ds Max [6], the surface geometry is exported in one of several industry standard geometry formats (IGES, STEP, Parasolid, ACIS, STL, …) to GAMBIT

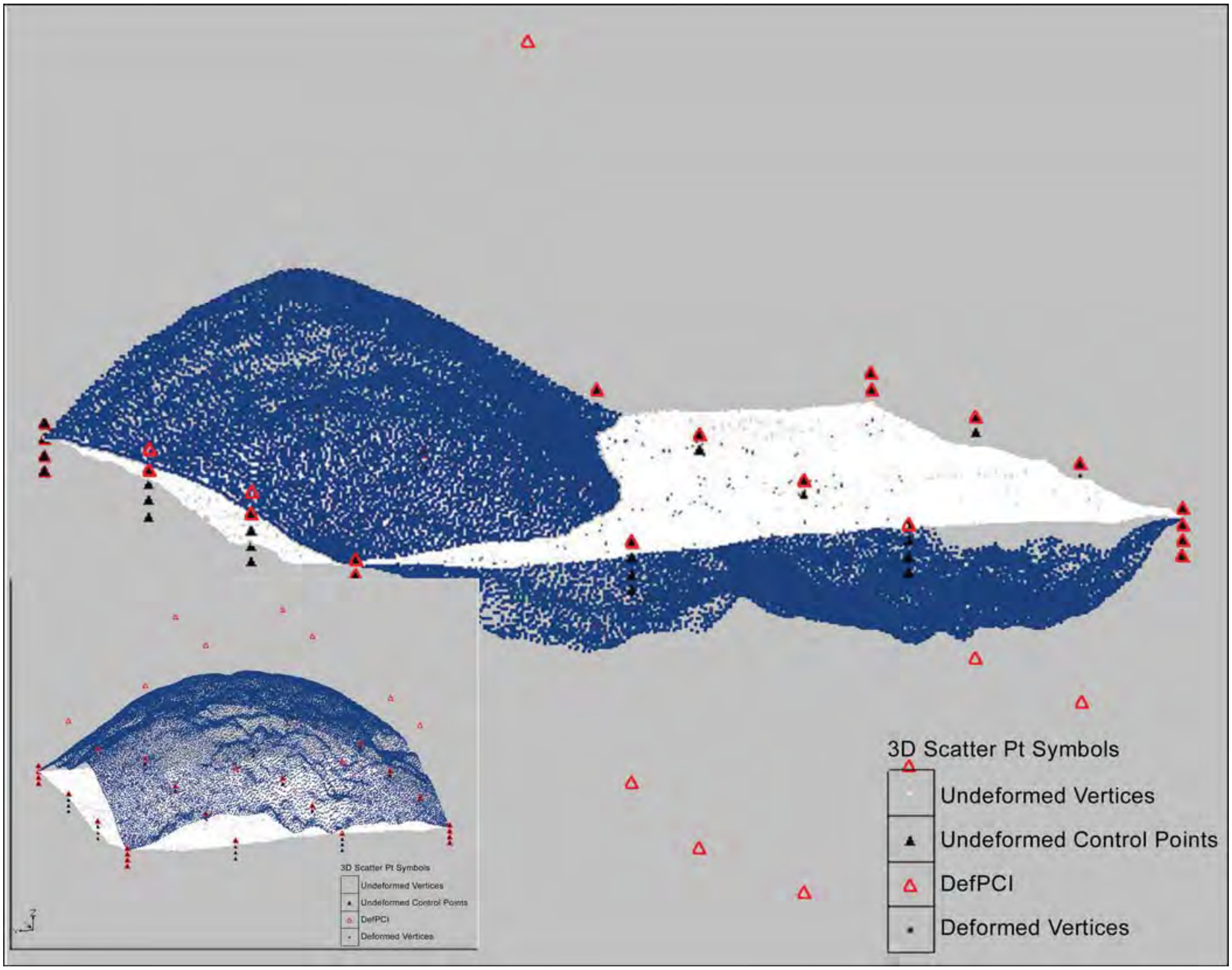


*Figure 5. FFD of generalized LIDAR surface (in-house so*fware development)

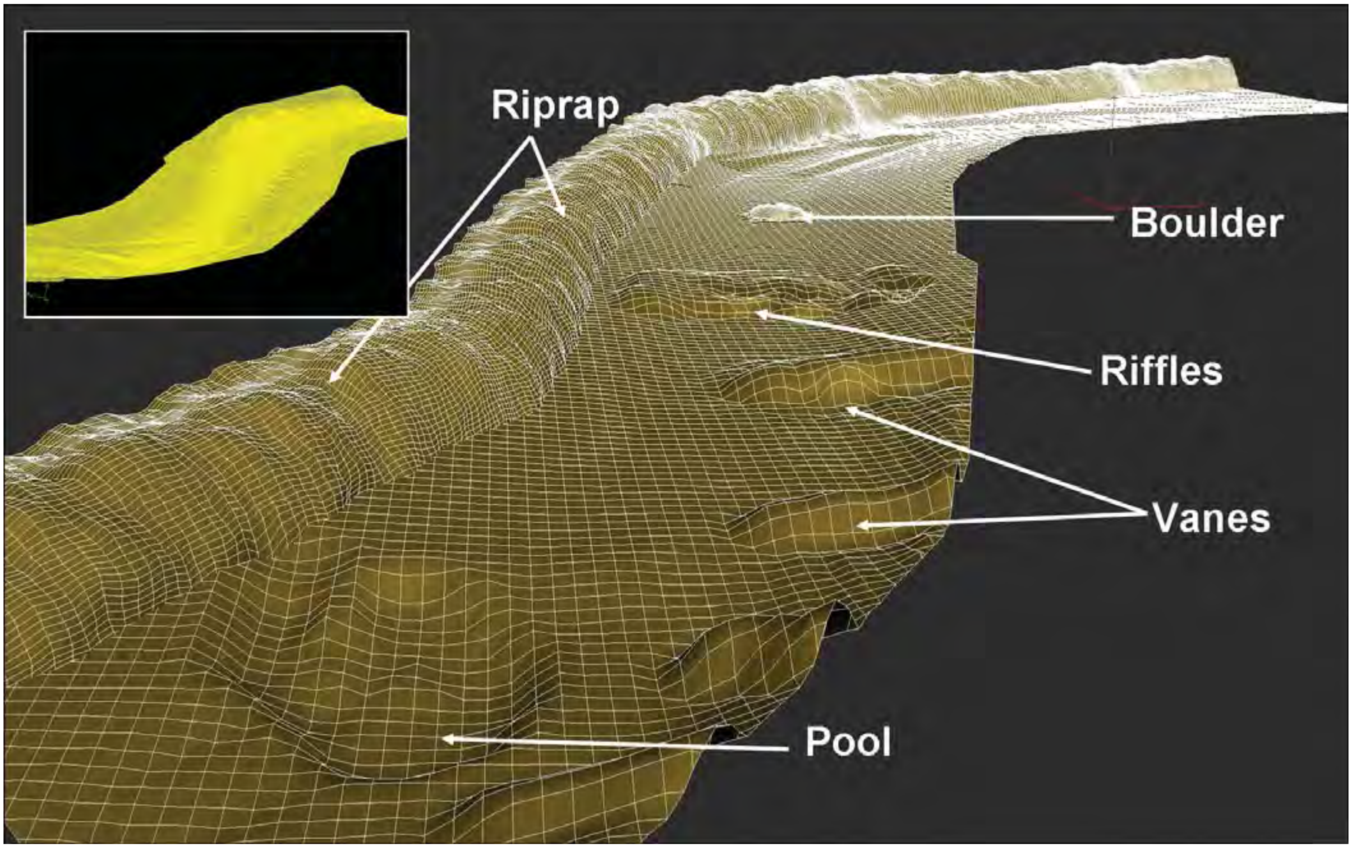


Figure 6. FFD of Merced River via FFD (3ds Max)

[9]. Once imported, a set of "cleanup" tools may be utilized to correct defects that may exist in the geometry. Here, the surface and surrounding volume (also created in GAMBIT [9] using a comprehensive set of Boolean operator tools) are meshed with hexagonal, tetrahedral, or hybrid elements. The utility of GAMBIT's [9] size function application allows the modeler to specify localized regions of mesh refinement, such that surrounding cells gradually coarsen through a functional approach. In this manner, face geometries having very small length scales govern the initial size function parameter. Figure 7 illustrates the above process via the incorporation of a root wad system (Figure 7A, representation only) that is embedded within the underlying surface geometry (Figure 7B) of S1 and meshed with over 1.5E6 tetrahedral elements (Figure 7C).

Flow Field Solution

Obtaining (and visualizing) an accurate flow field solution represents the final stage of the process. Because of the potential for substantial levels of detail within the original or deformed streambed geometry leading to the requirement of several millions of grid elements, the use of HPC resources is essential. At present, results from two parallel solvers, Fluent [10] and the Adaptive Hydraulics Model (ADH) [11], are being evaluated and compared for result accuracy and performance. ERDC's Cray XT3 supercomputer (Sapphire), using upwards of 16 nodes (32 processors) and runtimes of up to 12 hours were typical for the present simulations.

Some challenges inherent to the solution process include proper boundary and initial condition assignment (including fully developed flow inlet velocities), establishment of grid-independent solutions, adequate flow relaxation factors assignment (as applicable to the solver), procurement of steady-state/transient solution convergence (as applicable), and assignment of appropriate and sufficiently high-order discretisation schemes. Studies are underway to evaluate the effects of various surface roughness models and turbulence closure models.

Figure 8 shows Fluent [10] velocity results of the S1 geometry after embedding a root wad and boulder. The effects on the flow field because of the presence of the embedded features are clearly distinguishable. The simulation was conducted using a Reynolds Averaged Navier Stokes (RANS), k-epsilon turbulence closure model and a fully developed inlet velocity condition (with a maximum inlet velocity of 1 m/s).

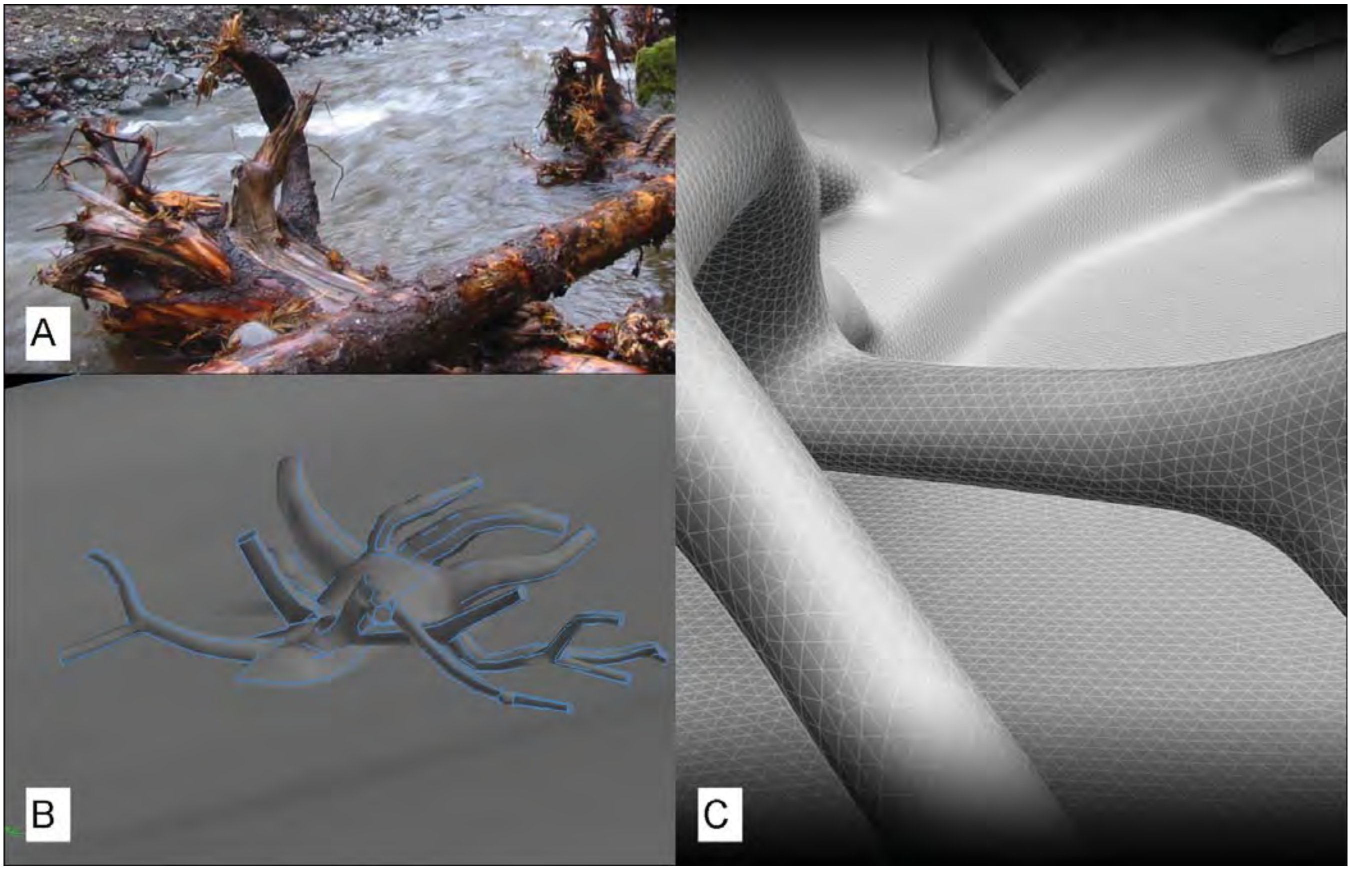


Figure 7. Root wad embedment and mesh

Summary and Future Vision

A major hurdle for developing high-resolution studies is the lack of data. Data are needed to define discharge, channel bathymetry, substrate size, and distribution and ecological attributes of a system without which modeling and simulation cannot occur. Detailed design and planning needs may justify time and expense associated with developing such data. However, early planning and decision making often occur without the benefit of detailed data assimilation and collection.

Many rivers already have some data available. For example, high-resolution aerial photographs provide information on channel planform and reach scale habitat types (riffles, pools, runs, etc). Channel slope can be estimated from digital elevation models, and channel discharge can be obtained or estimated from the network of stream gages. Estimates of the substrate types can often be found. Data on habitat quality such as the amount of large, woody debris or the number of pools can be located for some rivers. Aerial photos also may reveal the locations of large, woody debris or large rocks. With this information, it is possible to assemble a three-dimensional representation of a river that when coupled to a biological or ecological model provides a convincing representation of a river. Further, planners know the types of restoration options that need evaluation. Is a channel realignment being contemplated or a change in discharge from altered dam operations in the works? What about the addition of engineered log jams, cabled woody debris, or bioengineered streambank stabilization? All of these options could be computationally evaluated through the processes of mesh manipulation and addition of large roughness elements described above.

Just as data are expensive to collect, detailed modeling can also be expensive. The authors envision a tool, the Stream Habitat Analysis Package or SHAPE, that would provide simulation capabilities without programming by the end user. A set of predefined computational meshes of river channel types coupled with a library of three-dimensional logs, rocks, engineered log jams, bioengineered banks, and other interesting objects would be available. The user would choose predefined channel geometries, change the channel depth and width as desired, and then supplement with realistic habitat features. The predefined channel combined with habitat features would then be suitable for hydrodynamic modeling and subsequent analysis using an ecological model such as the Numerical Fish Surrogate [4]. The goal is to develop a technology to make this quick and easy enough to be done by end users.

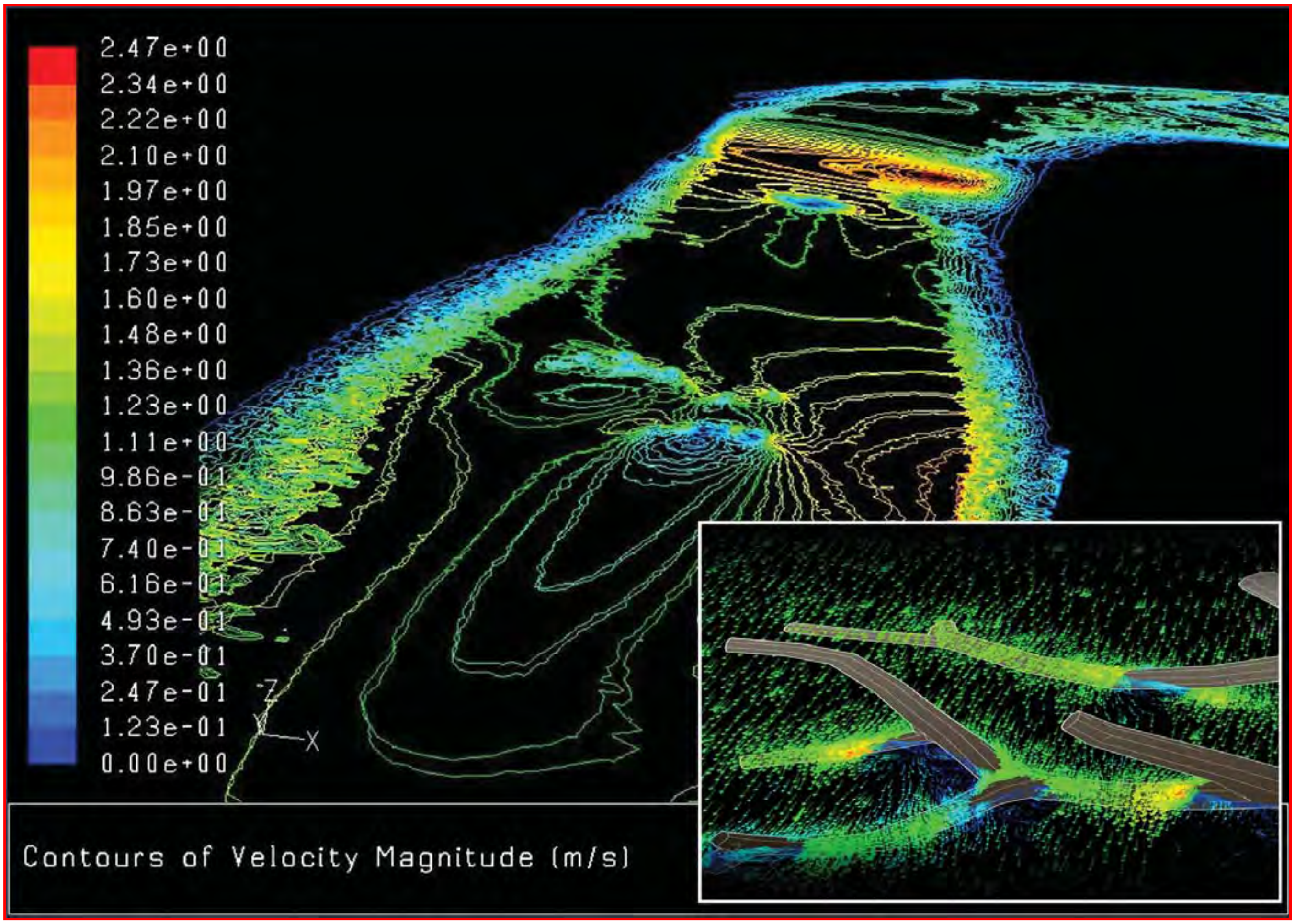


Figure 8. Fluent flow field solution of the Merced River (S1) incorporating large roughness elements

References

[1] Science. 2005. Vol. 308: 636-673.

[2] Science. 2004. Vol. 305: 937-939.

[3] Nestler, J. M., Goodwin, R. A., Smith, D. L., Anderson, J. J., and Li, S. 2008. "Optimum fish passage and guidance designs are based in the hydrogeomorphology of natural rivers." *River Research and Applications* 24: 148-168.

[4] Goodwin, R. A., Nestler, J. M., Anderson, J. J., Weber, L. J., and Loucks, D. P. (2006). "Forecasting 3-D fish movement behavior using a Eulerian-Lagrangian-agent method (ELAM)." *Ecological Modelling:* 192, 197-223.

[5] Energy Planning and Instream Flow Branch, USFWS, SFWO, "Monitoring of Restoration Projects in the Merced River using 2-Dimensional Modeling Methodology," *Merced River Restoration Modeling Final Report,* 2005.

[6] 3ds Max, www.autodesk.com/3ds Max, 22 February 2008.

[7] Sederberg, T. and Parry, S., "Free-Form Deformation of Solid Geometric Models," ACM SIGGRAPH Computer Graphics, Vol. 20, No. 4,1986, pp. 151–160.

[8] Maya, www.autodesk.com/maya, 22 February 2008.

[9] *Gambit Users Guide*, Lebanon NH, 2007.

[10] *Fluent 6.3 Users Guide*, Lebanon NH, 2006.

[11] Howington, S. E., Berger, R. C., Hallberg, J. P., Peters, J. F., Stagg, A. K., Jenkins, E. W., and Kelley, C. T., "A Model to Simulate the Interaction Between Groundwater and Surface Water," Proceedings of the 1999 High Performance Computing Users Group Meeting, Monterey, CA, USA, Washington, DC: DoD HPC Modernization Office, 1999.

A Coupled Watershed-Nearshore Model Using the ESMF and DBuilder

By Bobby Hunter and Dr. Ruth Cheng, ERDC MSRC, and Dr. Pearce Cheng, ERDC Coastal and Hydraulics Laboratory

In 2005 the DoD High Performance Computing Modernization Program started the Battlespace Environment Institute (BEI) to migrate existing DoD climate/weather/ocean modeling and simulation, environmental quality modeling and simulation and space weather applications to the Earth System Modeling Framework (ESMF) (Figure 1). Two models that comprise one of the BEI subtasks are pWASH123D and ADCIRC (Advanced Circulation). The work to couple these models with the ESMF is carried out in collaboration between the Engineer Research and Development Center (ERDC) in Vicksburg, Mississippi, and the Naval Research Laboratory at the Stennis Space Center. This particular BEI task has been named COSM, which stands for Coupled Ocean nearShore Model.

Software Components

The ADCIRC code is a finite element hydrodynamic model for coastal oceans, inlets, rivers, and floodplains. It solves time-dependent, free-surface circulation and transport problems in two and three dimensions. Typical ADCIRC applications within the Navy include simulations of circulation in coastal and riverine waters, wave-current interaction, forecasting hurricane storm surge, and flooding. The model implements the continuous Galerkin finite element method based on the Generalized Wave Continuity Equation (GWCE). The code is written in Fortran 90.

A parallelized version of WASH123D or pWASH123D is designed to solve watershed systems involving a coupled system of 1-D channel networks, 2-D overland regimes, and 3-D subsurface media. The interactions between different media (1- and 2-D, 2- and 3-D, and 1- and 3-D) impose flux continuity and state variable continuity on the medium interfaces. The pWASH123D aims to efficiently simulate the regional scale of real-world problems on HPC machines. Different parallel algorithms and partitioning strategies are implemented in different components in order to maintain load balance and reduce communication overhead. This application is a mixed C, Fortran, C++ code.

The ESMF provides and defines a software architecture for composing complex, coupled modeling systems and includes data structures and utilities for developing individual models. The ESMF consists of a superstructure that can be assembled into user applications and

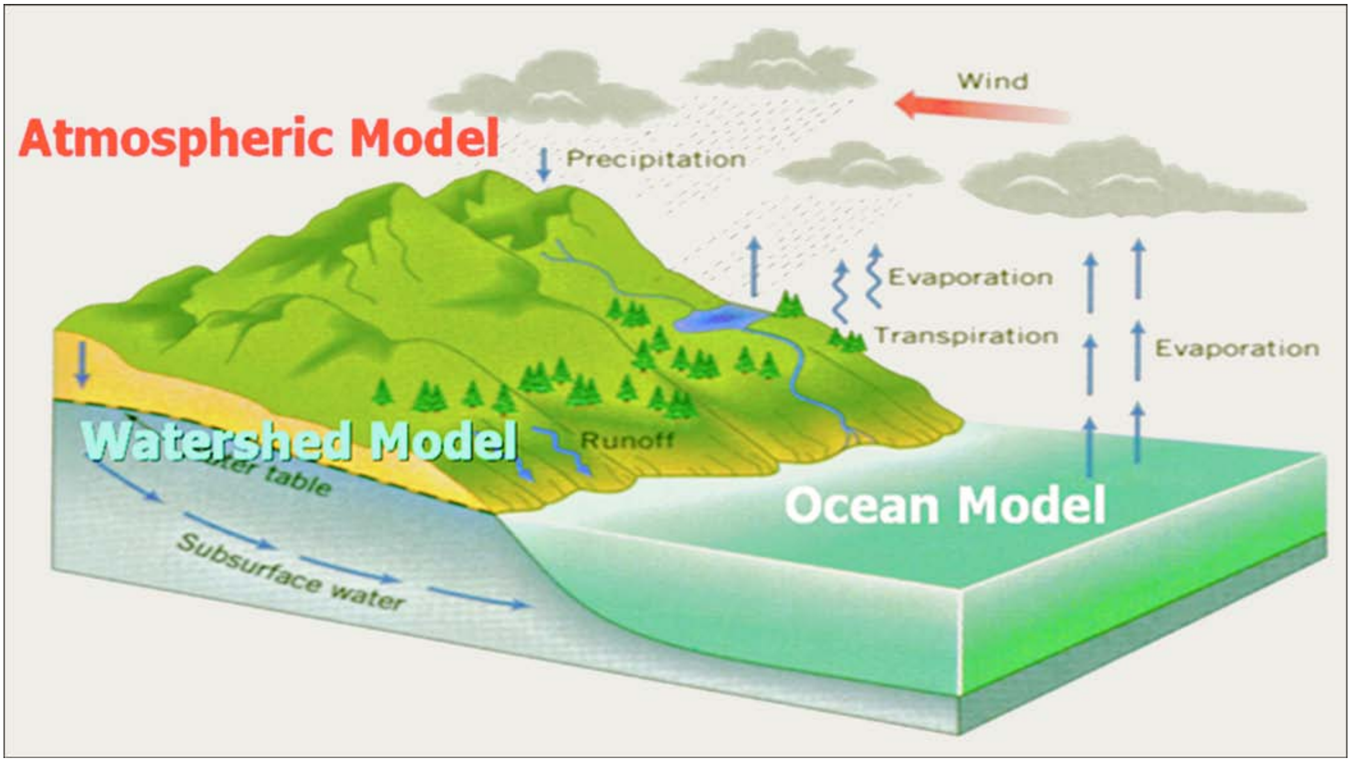


Figure 1. Overview of the Battlespace Environment Institute

an infrastructure for building model components. It was originally funded by the National Aeronautics and Space Administration to support climate and weather modeling. The ESMF effort then brought together different areas of research funding to extend its support to diverse modeling works. Sponsored by BEI, the ESMF will include unstructured mesh functionality.

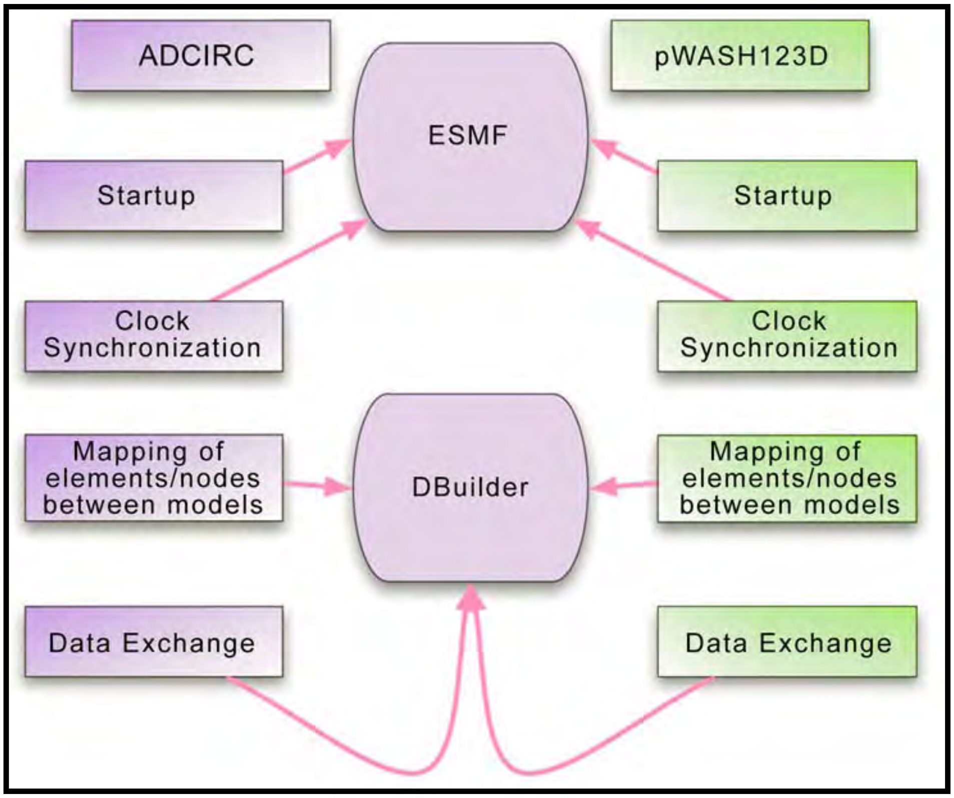


Figure 2. Current implementation strategy

One last component that was used in the development of COSM is the parallel software library DBuilder. DBuilder is a parallel data management library for scientific applications, which is currently used to facilitate the data exchange between the two models. At the time of initial development, the ESMF lacked support for unstructured meshes. Because DBuilder supports coupling independent domains in a single model (i.e., pWASH123D's 1-D, 2-D, and 3-D domains), adapting it to exchange data between models was not a large task. However, the ESMF is still used for startup, oversight of model runtime, and completion of the models (Figure 2).

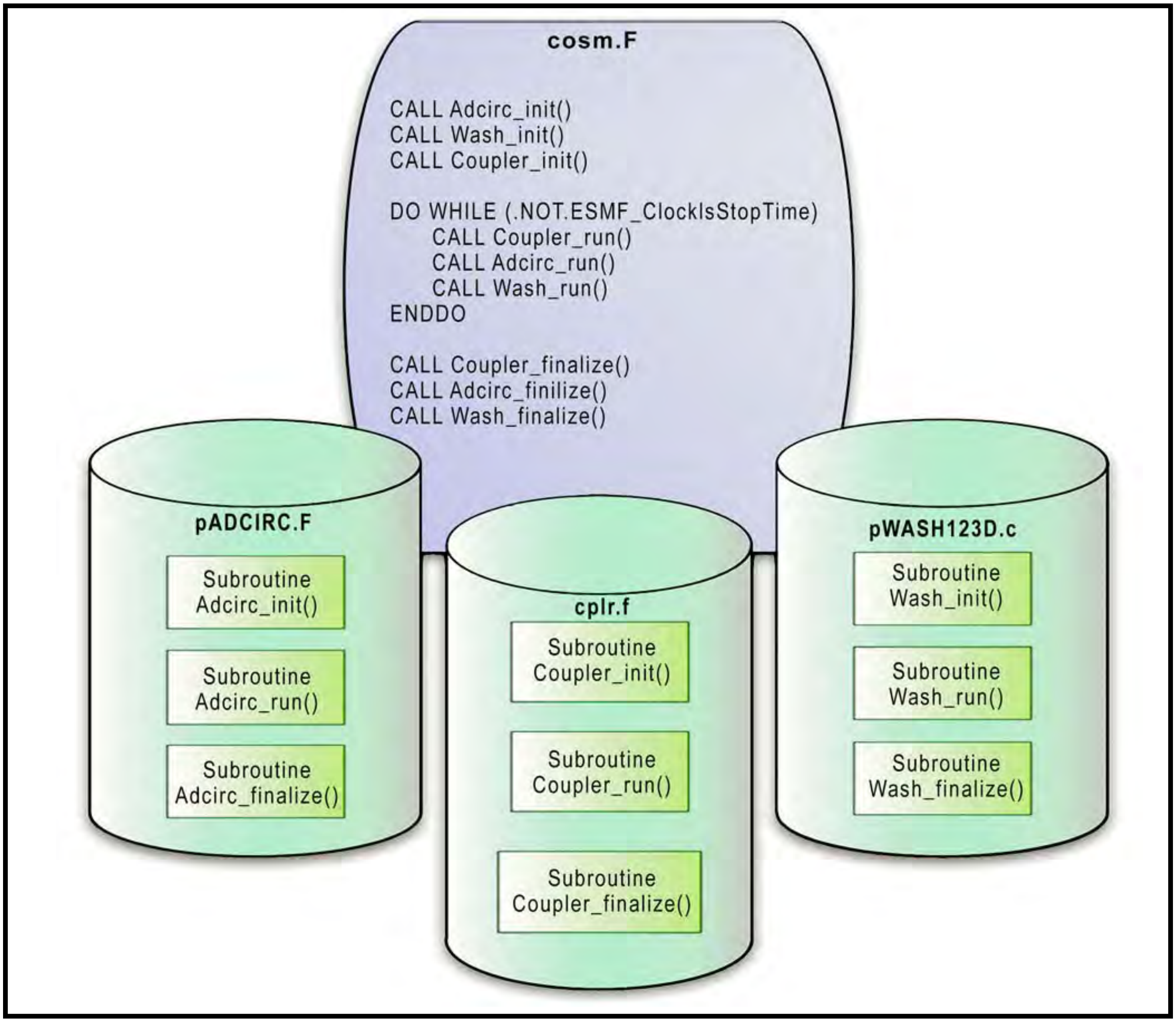
Since development of the alpha version of COSM began, the ESMF has added support for regridding, which is synonymous with coupling, for unstructured meshes. When COSM moves to the beta development phase, ESMF will be incorporated into the COSM coupling component in addition to DBuilder.

The ESMF, as aforementioned, requires that a model application code consists of three distinct phases: initialize, run, and finalize phases. The application developer needs to implement a main program including the three phases, in which some ESMF functions are called (Figure 3). The authors will call this file cosm.F. A user would then create an interface file that contains the three functions (initialize, run, and finalize) for a particular model. For this example, the authors will call them pWASH123.c and pADCIRC.F. The last piece the user needs to implement is a coupler component. This component contains coupler initialization routines, which may be used to create import and export states containing scalars and vectors for data exchange. The coupler component also contains the run routine for the coupling and a finalize routine.

Coupler Component

In the current alpha implementation of COSM, the coupler component relies on the functionality of DBuilder for implementation of the coupler "init" and coupler runtime routines. The coupler contains a key component provided by DBuilder, which is an element-searching routine. When the models are run, there is no static information with regards to the mapping of nodes to elements along the boundary interface of the two models. This interface boundary may also be an overlapped region. At runtime each model determines its interface nodes based on its own boundary condition values. Then in the coupler initialization routine, each model passes the geometric coordinates of its interface nodes to the other model. The element searching routine is then called on each model to build a list of elements containing the coordinates from the other model. One should keep in mind this is all done in parallel over already partitioned meshes in both models. The element searching algorithm is constructed using an Alternating Digital Tree (ADT) with complexity of O(log N), where N is the number of elements.

Once the element has been determined, weights are calculated for each associated node of the element. These weights are the nodal contribution from each node to the value calculated at the geometric coordinate. The computed value is then shipped back to the model processor that owns the node.



```
CALL Adcirc_init()
CALL Wash_init()
CALL Coupler_init()

DO WHILE (.NOT.ESMF_ClockIsStopTime)
     CALL Coupler_run()
     CALL Adcirc_run()
     CALL Wash_run()
ENDDO

CALL Coupler_finalize()
CALL Adcirc_finilize()
CALL Wash_finalize()
```

Figure 3. User support routines

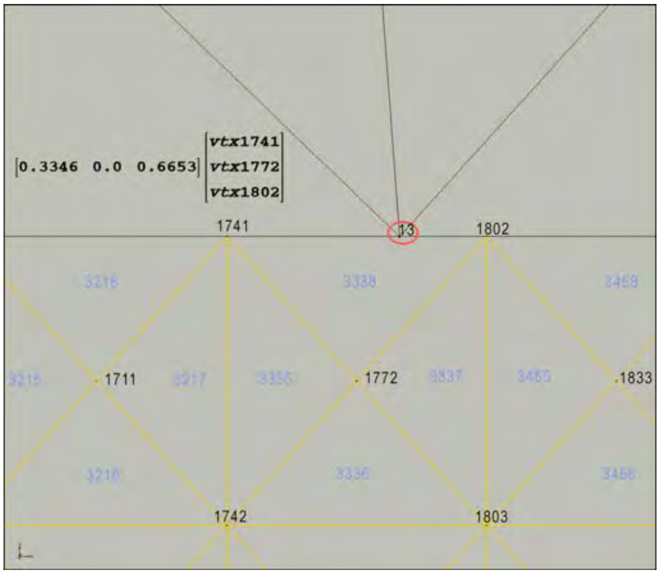


Figure 4. Weight calculation example

Experimental Results

The test example (Figure 5) is a computational mesh of the coupled models constructed from topographic and bathymetric data in the vicinity of the lower Biscayne Bay, Florida. In Figure 6, for the pWASH123D model, the northern and western boundaries were cut along two major canals in south Florida. Therefore, the observed canal stage can be used to set up head-type boundary conditions there.

The eastern boundary of pWASH123D is also the western boundary of ADCIRC, and it is the interface boundary through which the two models exchange water flow and water elevation data in the coupling process. The southern and the eastern boundaries of ADCIRC are the boundary of Elliot Key and Key Largo, where the no-flow boundary condition is applied. The northern boundary of ADCIRC has periodic tides applied as a forcing function for the model simulation run. Note that the canal waters coming in from the two inlets merge at the canal junction and flow eastward until entering the Biscayne Bay through the outlet.

Various land-use types, e.g., urban, cropland, rangeland, and wetland, are specified on the surface domain. The mesh resolution of pWASH123D along the interface boundary can be different from that of ADCIRC. The interface boundary, both from the pWASH123D side and from the ADCIRC side, falls onto the interface arc for the current implementation of the coupler. It will be generalized in the beta development.

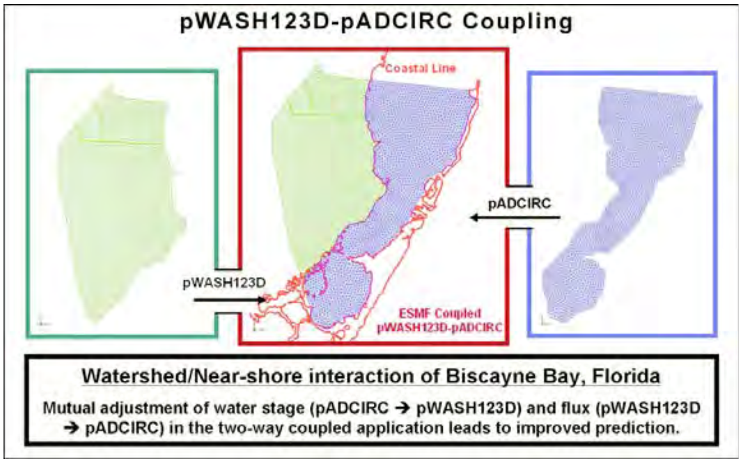


Figure 5. Computational meshes

Criteria for the alpha testing of COSM require one-way data exchange for software integration along with portability, accuracy, and scalability. So in pWASH123D, the water elevation data on the interface boundary were obtained from ADCIRC results. The alpha test was performed on the Cray XT3 machine (Sapphire) at ERDC and the IBM P575+ system (Babbage) at the Naval Oceanographic Office. All the metrics were well passed.

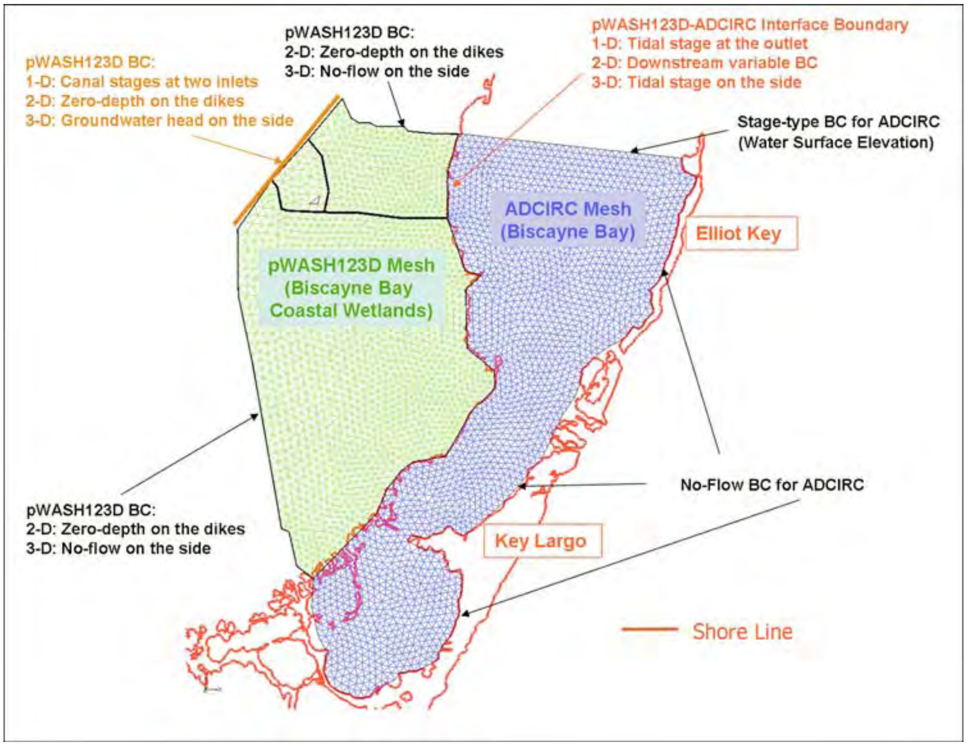


Figure 6. Problem setup

Figure 7 shows the water depth difference at time 4.5 hours between the results from the pWASH123D simulations without coupling and with coupling with ADCIRC. Running longer simulation and further testing scalability using larger meshes would be worthwhile.

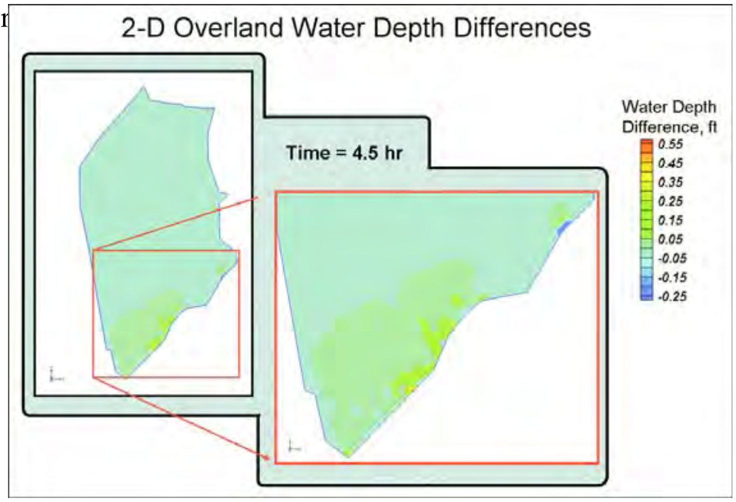


Figure 7. Water-depth differences at time = 4.5 hours

A sequential communication paradigm should be implemented, in addition to the current concurrent communication paradigm, in the coupler for the beta development.

For additional information on the topics discussed and the tools used, visit the following sites: http://www.erdc.hpc.mil/customerService/CS_E/Tools—home, http://www.esmf.ucar.edu, and http://www.nd.edu/~adcirc.

Increase Multicore Code Performance with Loop Blocking

By Tyler Simon

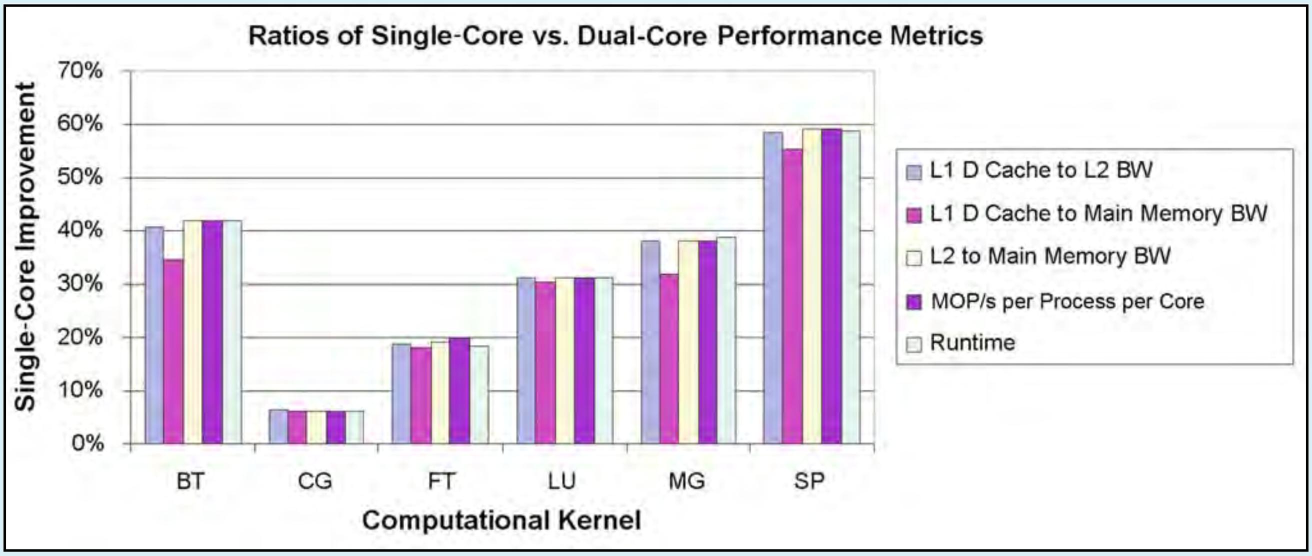


Figure 1. Ratios of on chip and off chip bandwidth values, runtime, and MOP/s/processor for single- and dual-core runs

Problem

Some HPC users may have noticed that their applications may not be running as fast on dual-core processors as they were on single-core processors; this is a common problem with no common solution. Unfortunately, there is no single technique or tool that can alleviate this performance gap. Clearly understanding how an application runs on multiple CPUs within a node and focusing on the cause of the delay in code execution are what is needed. This article provides a study of multicore memory contention and how a user or developer can address and circumvent this performance dilemma.

Identifying the Cause

On Sapphire the cause of increased application runtimes submitted with "yod –VN" is due to bandwidth contention at the processor cache level and from cache to main memory. Figure 1 demonstrates clearly this memory contention "bottleneck" on Sapphire using the NAS (Numerical Aerodynamic Simulation) parallel benchmark kernels [1]. Each one of the NAS kernels performs a computationally intensive numerical simulation that is representative of a "scientific computing" workload. For Figure 1, single-core results were obtained by submitting jobs using "yod -SN"; and dual-core results were obtained with the "yod –VN" command on Sapphire. Results were obtained using the Craypat performance analysis tool. There is virtually no variation in the bandwidth ratios with the runtime, and one can safely conclude that there is a correlation.

Identifying a Solution

Identifying that memory bandwidth contention is causing increased application runtime is not enough. An example of how structured memory access can increase application runtime is presented here. Often the largest performance gains can be made by focusing on the most computationally intensive and memory-intensive aspects of an application. This is usually contained within looping constructs. Figure 2 is a matrix multiplication loop consisting of C = A*B where A and B are 1024 × 1024 matrices.

```
DO J = 1, 1024
        DO K = 1, 1024
            DO I = 1, 1024
            C(I,J) = C(I,J) + A(I,K) * B(K,J)
            END DO
        END DO
    END DO
```

*Figure 2. C = A*B; matrix mutiply in Fortran*

Manually blocking the loop in Figure 2 works well for multicore chips because of the inherent memory bandwidth contention, as loop blocking limits redundant calls to memory both in the cache and to main memory off the chip. Blocking a loop structures the data in memory into chunks or "blocks" that may reside in larger portions or the entire cache. For computationally intensive loops, blocking forces cache reuse that limits the amount of data requests and actual data sent between L1and L2 caches and main memory. Thus each core retains a copy of data in its local cache. A blocked version of the matrix multiply loop is shown in Figure 3. In this case, KLBOCK and IBLOCK can be chosen manually based on the known cache size of the system processor cache size. Choosing an ideal block size is not a trivial task, and tools such as ATLAS [2] exist to automatically choose these values for a particular BLAS kernel.

Figure 4 displays the effect of block size on runtime of the matrix multiply loop in Figure 3 for both dual-core machines Jade and Sapphire. The near equivalence in processors is apparent: each is AMD Opteron and has a 65k L1 cache and a 1024k L2 cache. Figure 4 shows two processes running on two nodes and shows that 32 bytes is an ideal block size for this processor with the default compiler options set.

```
DO J = 1, 1024
      DO KOUT = 1, 1024, KBLOCK
         DO IOUT = 1, 1024, IBLOCK
            DO K = KOUT, KOUT+KBLOCK-1
               DO I = IOUT, IOUT+IBLOCK-1
               C(I,J) = C(I,J) + A(I,K) * B(K,J)
               ENDDO
            ENDDO
         ENDDO
      ENDDO
   ENDDO
```

Figure 3. Matrix multiply with blocking

Does blocking really help?

Yes, the results displayed in Figure 5 quantify the benefits of loop blocking on Sapphire. These tests were run over several compiler optimization levels with "O3 –fastsse" providing the fastest runtimes for single-core "yod –SN" and dual-core "yod –VN" jobs. The loop in Figure 3 was run for block sizes ranging from 4 bytes to 8192 bytes; the best blocked times are shown in Figure 5 with the nonblocked results. The optimal block size for the "O3 –fastsse" runs was 32 bytes for single-core mode and 512 bytes when running in dual-core mode. This difference in block size is an effect of

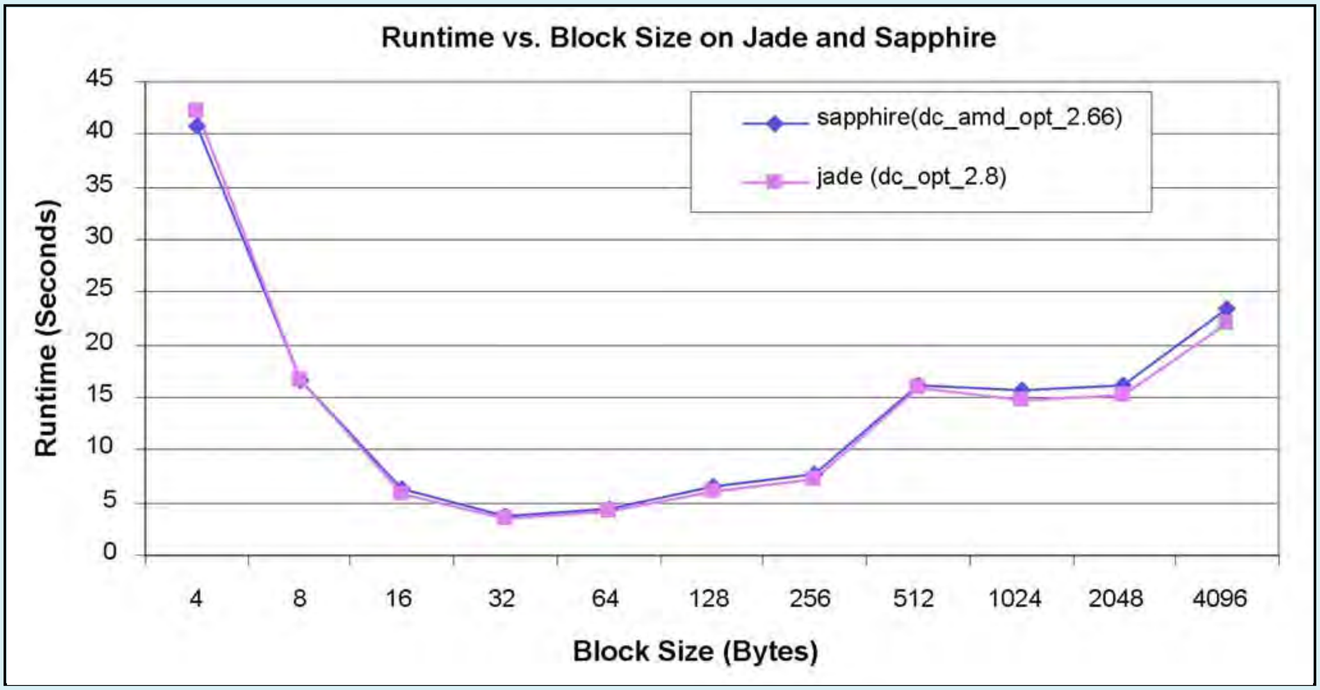


Figure 4. Effect of block size on runtime of matrix multiply loop on Jade and Sapphire

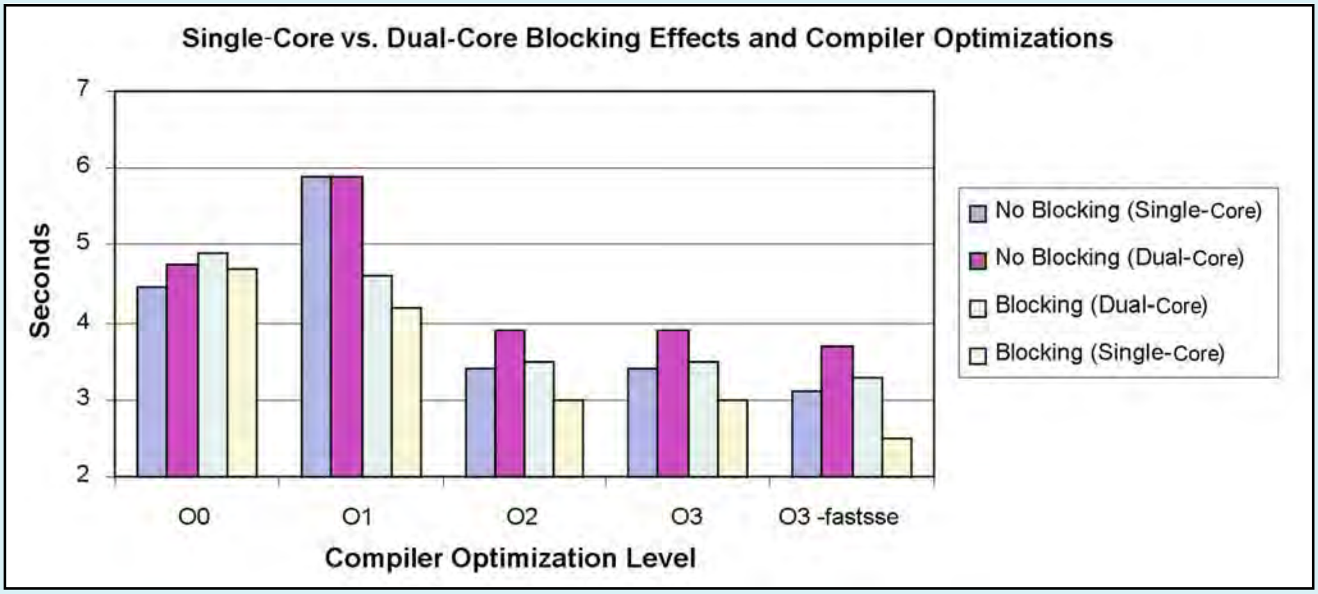


Figure 5. Runtime of blocked vs. nonblocked matrix multiply for single- and dual-core nodes

bandwidth contention; dual-core nodes naturally experience more cache contention. Larger block sizes offset contention by permitting more reuse of the available cache, but as a tradeoff, dual-core nodes with larger block sizes cannot fit the data into L1 cache. Therefore, the runtimes will always be slower, as fetching from L2 is more costly than fetching from L1. Heavy L1 cache use is primarily behind the performance improvement of single-core runs. In this case, the improvement is 75 percent.

Conclusions

Multicore processors are not going away anytime soon, and the promise of performance gains without code modifications is an illusion. By focusing on the real issue of memory contention both on and off chip, one can help the multicore transition to be a little less painful for code performance. By working with newer compilers, load balancing, and process placement strategies coupled with intelligent loop blocking, one can squeeze some better runtimes out of the codes. Hopefully, users will find the information presented in this article beneficial and share any optimizations that work. Good luck.

References

[1] D. Bailey and J. Barton. The NAS Kernel Benchmark Program. Technical Report 86711, NASA Ames Research Center, Moffett Field, California, August 1985.

[2] R. Clint Whaley, A. Pettit, J. Dongarra, "Automated Empirical Optimization of Software and the ATLAS Project" Parallel Computing vol. 27, No. 1-2, pp 3-35 2001.

"Mr. Force Protection"

Himself Named New ERDC Information Technology Laboratory Director

By Rose J. Dykes

Dr. Reed L. Mosher was recently named the new Director of the ERDC Information Technology Laboratory, which is home to the ERDC MSRC. The MSRC feels fortunate having a leader of his caliber with firsthand appreciation for high performance computing (HPC), as he has guided numerous projects that used HPC to help find answers for protection against terrorism.

Dr. Mosher discussed one such project on a broadcast of the CBS weekly news magazine "60 Minutes II." He talked about the section of the Pentagon that, just prior to September 11, 2001, had been renovated and fitted with blast-resistant windows designed in part by blast simulation performed on DoD HPCMP supercomputers. After the hijacked plane crashed into the Pentagon, this section was left relatively intact for a time, saving many lives, while other sections were obliterated.

After the Pentagon Renovation Team evaluated future actions, Dr. Mosher led the team that recommended the use of the latest protective technologies, many of which the use of HPC helped determine. The Pentagon is now outfitted with products of Dr. Mosher's research, making much of the renovation 10 years more advanced than was originally planned.

Dr. Mosher comes from the ERDC Geotechnical and Structures Laboratory (GSL) where he most recently served as the technical director for Survivability and Protective Structures, head of the ERDC task force for Homeland Security, and the lead technical director for Military Engineering in GSL—no wonder he was dubbed "Mr. Force Protection" in an article by ERDC PAO Acting Director Wayne Stroupe. Of course, Dr. Mosher shies away from any such title and says,

"Force protection is no one person's responsibility; it's a team effort. There is no one single force protection 'Bubba' in the Army. We (ERDC) want to be part of the solution, and we've gotten nothing but great responses on our research products."

Just prior to coming to ITL as its director, Dr. Mosher received the DoD Distinguished Civilian Service Award, the highest award given by the Secretary of Defense to a career employee, and the Army Engineer Association's Bronze de Fleury Medal for his leadership in research that has led to the development of innovative products for force protection of U.S. military and civilian personnel worldwide from terrorist bombings and conventional weapons.

Dr. Mosher, a native of Maine, earned his bachelor's degree in civil engineering from Worcester Polytechnic Institute in Worcester, Massachusetts, master's degree in civil engineering from Mississippi State University in Starkville, Mississippi, and doctorate degree in civil engineering from Virginia Polytechnic Institute (Virginia Tech) and State University in Blacksburg, Virginia. He has served as an adjunct professor at Mississippi State University, University of Puerto Rico, Virginia Tech, and Louisiana State University.

Communicate with Your Data

By Dr. Michael Stephens

As part of the Department of Defense High Performance Computing Modernization Program's (HPCMP) new initiatives, the visualization components were combined to form the Data Analysis and Assessment Center (DAAC). Previously, these components were located at the four Major Shared Resource Centers—the Army Research Laboratory (ARL), Aeronautical Systems Center, U.S. Army Engineer Research and Development Center (ERDC), and Naval Oceanographic Office, as well as two Allocated Distributed Centers—the Arctic Region Supercomputing Center and the Maui High Performance Computing Center.

With the new DAAC, the data analysis and visualization efforts are now run at the Program level rather than the Center level. Furthermore, operationally, the DAAC is split into two operating units depending on the classification of the data being generated by the projects. Classified data are handled by the CDAAC, collocated with the MSRC at ARL. Unclassified data are handled by the UDAAC, hosted by the ERDC MSRC. So that is the organizational nuts and bolts of the new DAAC; but it does not really tell you what you really want to know—

"What is the DAAC to me?"

The answer is the DAAC is a resource whose mission is to help you to communicate with your data. To communicate with your data has multiple meanings. For practical purposes, this communication happens at two distinct levels. One level is the personal "conversation" you, the researcher, have with your data. In this conservation, you interrogate or ask questions of the data using visualization tools. The answers you get, the visual images, are absolutely the best way to distill your data into valuable information that further guides and advances your research with the ultimate goal of obtaining insights into your problem's solution. The other level occurs when you wish to communicate your findings to a broader audience—for instance, to research peers, research sponsors, or the interested public. Again, the visual image of your data is the best way to communicate to others. The DAAC has the resources and expertise to help you accomplish both of these levels of communication.

The gateway to your DAAC resource is through the Web site: http://daac.hpc.mil/. Here you will find information about the DAAC resources such as the available computing hardware and the supported visualization software. This Web site also contains two unique features that allow you, the user, to be directly connected to the DAAC staff as well as other DAAC users: a Community Forum and a Wiki. On the Community Forum, not only can you seek help with your visualization problems but you can also share your expertise with the DAAC community and provide answers or comments for others in the community. The other novel feature is the Wiki, which provides a wealth of information and self-education materials such as tutorials, a gallery of past projects, and like all Wikis, allows you to provide comments and share your experiences and expertise.

DAAC also prepares and distributes a semiannual publication called *enVision*, which contains information about visualization procedures, algorithms, tools, and projects. *enVision* can also be found on the DAAC Web site.

If you have problems or questions regarding data analysis or visualization, there are several ways to contact the DAAC for help. The first way to contact us is to send an e-mail to support@daac.hpc.mil. This message will be relayed to various members of the DAAC for review and response accordingly. Another way to contact us is to post a message in the Forums on the Web site to be answered by DAAC team members or other users.

So start communicating with your data today!

The HPCMP Data Analysis and Assessment Center assists users in visualizing and analyzing their simulation outputs through high-quality visualization, support of end-user applications, and training and reference materials.

Research

The DAAC continually researches new software and technology to better provide capabilities to our users. These include remote visualization, virtual environments, renderwalls, new visualization algorithms, and the latest in computing hardware. The DAAC is investigating the use of IP Video System's V$_2$D as a hardware solution for remote visualization for DoD researchers and scientists.

Community Services

The DAAC Web site and enVision magazine establish a user community, allowing users to help each other and help themselves. The DAAC Web site, http://daac.hpc.mil, contains educational and system information for all data analysis resources within the HPCMP. A Wiki and Forum accept user contributions on topics ranging from basic analysis algorithms to software tutorials. enVision is a semiannual publication distributed by the DAAC containing information about visualization procedures, algorithms, tools, and projects.

Collaboration

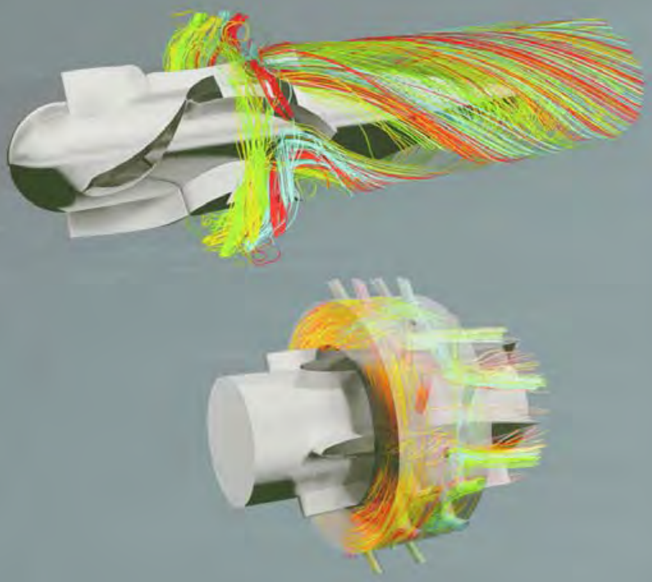
Collaborative visualization is used to provide technical assistance to enable researchers to create the visualizations themselves. This involves teaching the user which data formats are preferred for using visualization tools, how to get them into those formats, and how to use those tools.

ezviz

Developed and maintained by the ERDC MSRC, ezVIZ is a command-line tool for batch processing of large-scale data directly on the supercomputers. This prevents the user from duplicating data onto multiple systems for specialized processing and enables visualization while simulations are still running.

Custom Services

Custom visualization is used to produce high-quality animations of the data that allow researchers to communicate their results to others. This involves the creation of narrated DVD movies, posters, or movies that are suitable for presentations. Additionally, these projects assist users in discovery of physical and numerical phenomena.



ERDC MSRC and HPCMP Announce Release of ezHPC v.2.0

By Scotty Swillie

Efforts to make high performance computing (HPC) easier took a big step forward recently with the announcement of the release of ezHPC v.2.0. As you may remember, ezHPC began as a CHSSI (Common High Performance Scalable Software Initiative) project several years ago at the Engineer Research and Development Center Major Shared Resource Center (ERDC MSRC). The research project focused on the development of an easy-to-use, secure graphical user interface to local HPC systems. The end product became known as ezHPC, and it has now flourished into a program-wide resource for all High Performance Computing Modernization Program (HPCMP) users.

ezHPC v.2.0 is a combination Web service and Java(tm) Applet-based client front-end used to allow simplified access to HPCMP HPC resources. The Web service provides an application programming interface (API) for accessing and manipulating HPC resources. The client uses the Web service API to allow all HPC users with a Web browser, a Java(tm) runtime environment, and proper credentials to access and manipulate their HPC data.

"ez"
just got a lot
"ez-er"

To accomplish its objective of providing users easy-to-use access to HPC resources, the ezHPC interface offers users of all experience levels a complete toolkit that includes the ability to obtain target system status, run jobs, monitor job progress, move files between HPC systems, move files from local systems to HPC systems, edit and run scripts, and access mass storage systems. Because ezHPC is Web based, users can potentially access their information from anywhere.

Location /home/rappold

Name	Date	Size	Permissions	Owner	Group
Desktop	Tue Mar 04 15:09:47 CST 2008	4096	drwxrwxrwx		
Examples	Tue May 30 19:52:57 CDT 2006	4096	drwxrwxrwx		
apache-tomcat-5.5.17	Tue Nov 28 09:38:47 CST 2006	4096	drwxrwxrwx		
javastuff	Tue Mar 04 14:31:03 CST 2008	4096	drwxrwxrwx		
testing2	Tue Nov 27 21:08:35 CST 2007	4096	drwxrwxrwx		
testing3	Wed Nov 28 14:43:47 CST 2007	4096	drwxrwxrwx		
testing5	Tue Mar 04 14:34:52 CST 2008	4096	drwxrwxrwx		
tomcat	Tue Nov 28 09:38:47 CST 2006	4096	drwxrwxrwx		
.ICEauthority	Mon Feb 04 09:14:03 CST 2008	2059	-rwxrwxrwx		
.Xauthority	Mon Feb 04 09:14:01 CST 2008	206	-rwxrwxrwx		
.bash_history	Tue Mar 18 09:51:59 CDT 2008	10701	-rwxrwxrwx		
.bash_logout	Fri Aug 11 10:33:28 CDT 2006	220	-rwxrwxrwx		
.bash_profile	Fri Aug 11 10:33:28 CDT 2006	414	-rwxrwxrwx		
.bashrc	Fri Aug 11 10:33:28 CDT 2006	2227	-rwxrwxrwx		
.dmrc	Mon Aug 14 10:57:04 CDT 2006	26	-rwxrwxrwx		
.esd_auth	Mon Aug 14 10:57:04 CDT 2006	16	-rwxrwxrwx		
.gksu.lock	Fri Sep 08 09:25:23 CDT 2006	0	-rwxrwxrwx		
.gtk-bookmarks	Wed Nov 28 16:31:28 CST 2007	58	-rwxrwxrwx		

Copy/Move Progress

From	To	Source	Destination	Message	Progress	StartTime
LOCAL::L...	ARSC::MIDN...	/home/rappold/....	/import/home/u2/wes/krapp...	Completed.	100	Thu Mar 20 10:33:32 C...

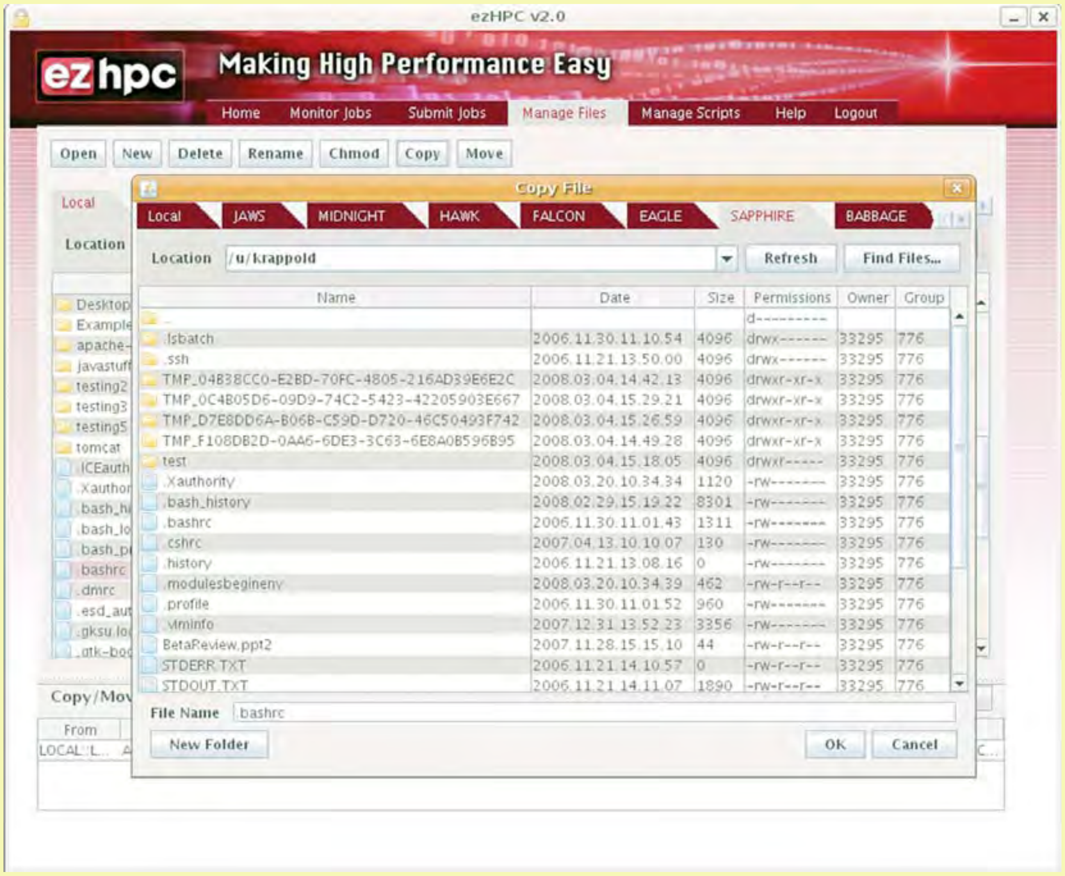
Figure 1. ezHPC manage files screenshot

The ezHPC v.2.0 interface design is more intuitive than the previous version. This is the result of leveraging many Program resources in the design, production, and testing phases. HPCMP security was consulted at every step to ensure that the product met with current security constraints. An ezHPC advisory users group was formed to help provide requirements and feedback for the overall site redesign. Additionally, a usability expert was also brought into the development process to help provide a more user-friendly interface. Alpha and beta testing were conducted to get as much user feedback as possible, which was then used to refine the client.

One of the results of these exhaustive efforts is a clean, effective design that runs fast and is easy to use for everyone – novice to power user. However, the greatest benefit users will encounter when using ezHPC v.2.0 is the power and freedom they will find in the range of tasks offered within the interface. Here are a few ways ezHPC v.2.0 is making users' lives easier:

- Moving an entire directory of data is just three clicks away.
- Machine status can quickly be viewed graphically.
- The interface no longer has to be configured; it knows on which machines users have accounts and automatically populates the screen with their files when they login.
- Seamless integration of the PC, remote HPC, and archive file systems occurs.
- Users can easily monitor jobs running on all machines to which they have access.
- The batch script generator is improved.
- Fields are sortable for quick location of files.

Overall – "ez" just got a lot "ez-er." If you are already using ezHPC v.2.0 – great! If not, what are you waiting for? Visit our Web site and take it for a spin – https://ezhpc.hpc.mil. With these upgrades, we believe you will like what you see.



Location /u/krappold

Name	Date	Size	Permissions	Owner	Group
..			d---------		
.lsbatch	2006.11.30.11.10.54	4096	drwx------	33295	776
.ssh	2006.11.21.13.50.00	4096	drwx------	33295	776
TMP_04B38CC0-E2BD-70FC-4805-216AD39E6E2C	2008.03.04.14.42.13	4096	drwxr-xr-x	33295	776
TMP_0C4B05D6-09D9-74C2-5423-42205903E667	2008.03.04.15.29.21	4096	drwxr-xr-x	33295	776
TMP_D7E8DD6A-B06B-C59D-D720-46C50493F742	2008.03.04.15.26.59	4096	drwxr-xr-x	33295	776
TMP_F108DB2D-0AA6-6DE3-3C63-6E8A0B596B95	2008.03.04.14.49.28	4096	drwxr-xr-x	33295	776
test	2008.03.04.15.18.05	4096	drwxr-----	33295	776
.Xauthority	2008.03.20.10.34.34	1120	-rw-------	33295	776
.bash_history	2008.02.29.15.19.22	8301	-rw-------	33295	776
.bashrc	2006.11.30.11.01.43	1311	-rw-------	33295	776
.cshrc	2007.04.13.10.10.07	130	-rw-------	33295	776
.history	2006.11.21.13.08.16	0	-rw-------	33295	776
.modulesbeginenv	2008.03.20.10.34.39	462	-rw-r--r--	33295	776
.profile	2006.11.30.11.01.52	960	-rw-------	33295	776
.viminfo	2007.12.31.13.52.23	3356	-rw-------	33295	776
BetaReview.ppt2	2007.11.28.15.15.10	44	-rw-r--r--	33295	776
STDERR.TXT	2006.11.21.14.10.57	0	-rw-r--r--	33295	776
STDOUT.TXT	2006.11.21.14.11.07	1890	-rw-r--r--	33295	776

File Name .bashrc

Figure 2. ezHPC copy files screenshot

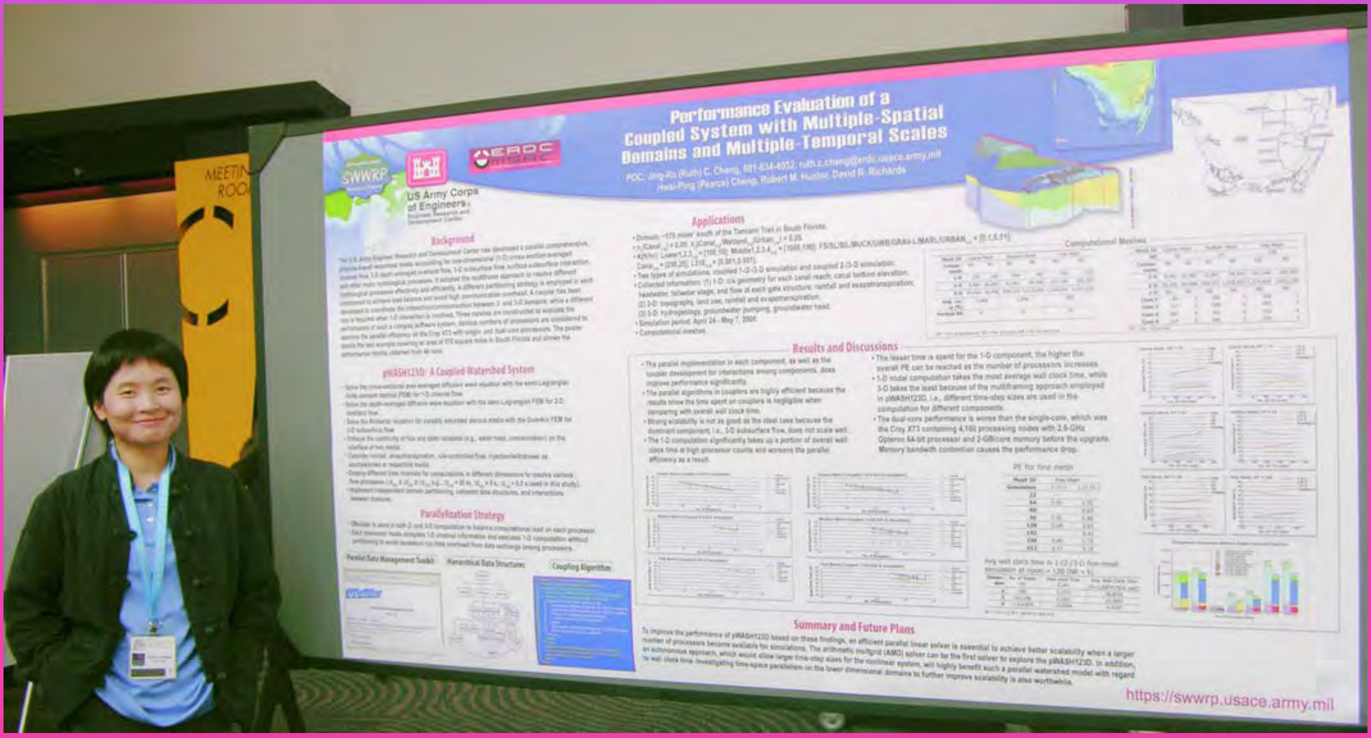
ERDC MSRC Puts Expertise on Display at SC07

By Rose J. Dykes

Several ERDC MSRC team members attended Supercomputing 2007 (SC07) in Reno, Nevada, November 10-16, to highlight the expertise of ERDC scientists and engineers, as well as ERDC's extensive supercomputing expertise, in the DoD HPC Modernization Program's booth. ERDC HPC posters and technical publications were also available for the VIP tour at the first night opening gala and each day for the remainder of the week. SC07, the international conference for HPC, networking, and storage and analysis, provides a forum of the highest quality for scientists and engineers to present their latest research findings in one of the most rapidly changing technical fields. Over 9,000 specialists from all over the world attended the conference.

Dr. Ruth Cheng participated in the Poster Division of the conference with her entry entitled "Performance Evaluation of a Coupled System with Multiple-Spatial Domains and Multiple-Temporal Scales." The poster presented the outcome of performance evaluation on the parallel algorithms developed for and implemented in the ERDC watershed model.

Paul Adams and Richard Walters presented work that the Data Analysis and Assessment Center had performed for the DoD HPCMP users. Additionally, they showcased the new *enVision* magazine, which offers tutorials on scientific visualization topics and programs such as ParaView and EnSight.



Dr. Ruth Cheng presents her poster at the Conference Poster Session

Scotty Swillie (center) and Charles Ray (far right) were part of the team that constructed the DoD HPCMP booth for the Conference

(From left) Dr. Jerry Morris and David Stinson work at the HPCMP booth during the Conference

John West Selected Mississippi State University Distinguished Fellow

By Rose J. Dykes

On March 6, the James Worth Bagley College of Engineering, Mississippi State University (MSU), honored John E. West as one of its Distinguished Fellows. The Distinguished Fellows program was initiated in 1992 as part of the College of Engineering Centennial and recognizes graduates who have made significant contributions to their field.

Upon completing degrees in electrical and computational engineering at MSU, West rejoined the staff of the ERDC Information Technology Laboratory (ITL), where he has a long history of service in the Major Shared Resource Center, starting as a contract student and later becoming its Director. He has also served as the ITL Director of the Scientific Computing Research Center and as the ITL Acting Deputy Director, supporting R&D for the Corps of Engineers in IT-related fields. In April, West accepted a position with Lockheed Martin as the ERDC Site Technical Lead for the HPCMP's Next Generation Technical Services Contract.

HPCwire selected him as one of its

"People to Watch for 2006"

Among the many awards that West has received is the Department of the Army's R&D Award in 1997 for his computational research accomplishments. In 2007, he received the Department of the Army's Commander's Award for leadership in the development of technology to assist DoD researchers in effectively utilizing new HPC architectures and exemplary leadership and vision in soliciting and acquiring the authority to manage a significant portion of the scientific visualization capability throughout the entire HPCMP. *HPCwire* selected him as one of its "People to Watch for 2006." He also often writes and speaks on supercomputing technology and on leadership and career directions for young technologists.

(From left) Dr. Julia Hodges, Professor and Chair of the Department of Computer Science and Engineering, MSU; John West; and Dr. Glenn Steele, Professor of Mechanical Engineering and Acting Dean of the Bagley College of Engineering, MSU

(From left) Dr. Deborah Dent, ERDC Information Technology Laboratory (ITL) Deputy Director, and Bernd "Bear" McConnell, Director of Interagency Coordination, North American Aerospace Defense Command and U.S. Northern Command, Peterson Air Force Base, Colorado, April 22, 2008

(From left) COL Al Lee, U.S. Army Engineer (USAE) District, New Orleans; Dr. Reed Mosher, ERDC ITL Director; and LTC Murray Starkel, USAE District, New Orleans, April 8, 2008

(From left) Dr. Dent; Dr. Mosher; John E. West, ERDC MSRC; and Dr. Thomas H. Killion, Deputy Assistant Secretary for Research and Technology, Chief Scientist, Assistant Secretary of the Army Acquisition, Logistics and Technology, Washington, D.C., April 3, 2008

(From left) Randall Hand, Data Analysis and Assessment Center (DAAC); Phil Stewart, Office of Technology Transfer and Outreach, ERDC; Marti Elder, TechLink, Washington, D.C.; and Dr. Jerry Morris, ERDC MSRC, March 31, 2008

(From left) MG Ronald L. Johnson, Deputy Commander, U.S. Army Corps of Engineers, Washington, D.C., and Dr. Dent, February 20, 2008

University of Notre Dame, Notre Dame, Indiana, students and David Stinson (far right), ERDC MSRC Acting Director, November 30, 2007

(From left) Greg Rottman, ERDC MSRC Assistant Director, and Tina Ballard, Deputy Assistant Secretary for Policy and Procurement, U.S. Army, Washington, D.C., November 19, 2007

(From left) Dr. Michael Stephens, DAAC Lead; Professor Robert Curl, Nobel Laureate (Nobel Prize in Chemistry 1996); and Dr. Bob Welch, ERDC ITL Executive Office, November 1, 2007

(From left) LTG (R) Robert B. Flowers, Chief Executive Officer, International, and Vice Chairman, Federal Services; Agnes Otto, Federal Technology Practice Leader, Associate Vice President, HNTB; John West, September 21, 2007

acronyms

Below is a list of acronyms commonly used among the DoD HPC community. These acronyms are used throughout the articles in this newsletter.

ADCIRC	Advanced Circulation
ADH	Adaptive Hydraulics Model
ADT	Alternating Digital Tree
API	Application Programming Interface
ARL	Army Research Laboratory
BEI	Battlespace Environment Institute
CDAAC	Classified DAAC
CEWES	Corps of Engineers Waterways Experiment Station
CFD	Computational Fluid Dynamics
CHSSI	Common High Performance Scalable Software Initiative
COSM	Coupled Ocean nearShore Model
CPU	Central Processing Unit
DAAC	Data Analysis and Assessment Center
DoD	Department of Defense
EL	Environmental Laboratory
ERDC	Engineer Research and Development Center
ESMF	Earth System Modeling Framework
FFD	Free Form Deformation
GSL	Geotechnical and Structures Laboratory
GWCE	Generalized Wave Continuity Equation
HPC	High Performance Computing
HPCMP	HPC Modernization Program
ITL	Information Technical Laboratory
LIDAR	Light Detection and Ranging
LMIS	Lockheed Martin Infrastructure Services
MOP/s	Millions of Operations per Second
MSRC	Major Shared Resource Center
MSU	Mississippi State University
NAS	Numerical Aerodynamic Simulation
NGTSC	Next Generation Technical Services Contract
RANS	Reynolds Averaged Navier Stokes
SC07	Supercomputing 2007
SHAPE	Stream Habitat Analysis Package
TI	Technology Insertion
UDAAC	Unclassified DAAC
USAE	U.S. Army Engineer

training schedule

For the latest on training and on-line registration, one can go
to the User Productivity Enhancement and Technology
Transfer (PET) Online Knowledge Center Web site:

https://okc.erdc.hpc.mil

Questions and comments may be directed to PET
at (601) 634-3131, (601) 634-4024, or
PET-Training@erdc.usace.army.mil

ERDC MSRC *Resource* Editorial Staff

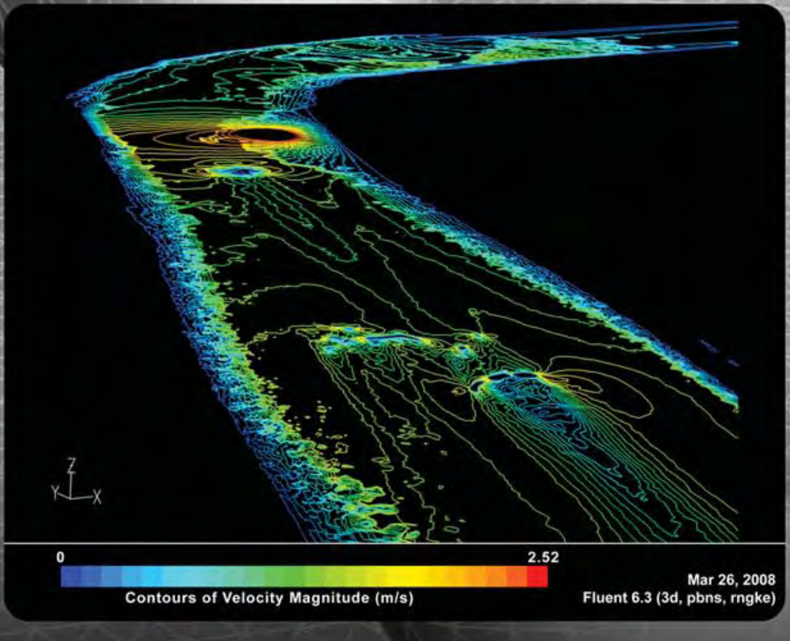
Chief Editor/Technology Transfer Specialist
Rose J. Dykes

Visual Information Specialist
Betty Watson

ERDC MSRC HPC Service Center
Web site: *www.erdc.hpc.mil*
E-mail: *msrchelp@erdc.hpc.mil*
Telephone: 1-800-500-4722

The ERDC MSRC welcomes comments and suggestions regarding the *Resource* and invites article submissions. Please send submissions to the above e-mail address.

Design and layout provided by the Visual Production Center, Information Technology Laboratory, U.S. Army Engineer Research and Development Center.



U.S. ARMY ENGINEER RESEARCH
AND DEVELOPMENT CENTER
INFORMATION TECHNOLOGY LABORATORY