Center for High-Performance Computing @ the University of Utah

News

University of Utah Fall 2007

Article

CHPC Supports Web Site to Explore Cone Snail Biology Interactively

by Janet Ellingson

Center for High Performance Computing, University of Utah

Cone snails (Conus) are very successful predators. These abundant small marine animals use their venom to defend themselves and to capture prey by delivering fast-acting toxins that paralyze the victim. Even though the nearly 500 living species of the cone snail cannot swim, many survive by eating live fish. Dead fish just won't do. The snails have developed sophisticated venom production apparatus and delivery systems. Some of the fish-eating snails have teeth which act as both harpoon and needle for delivering the venom to fish. Others engulf their prey with large distensible mouths, like a fisherman with a net, before delivering the fatal sting. The venom varies widely from one species to the next, each having evolved to work most effectively on the specific targets. The venom of Conus geographus, the geography cone, is extremely toxic to humans. The sting has been fatal in 70 percent of the untreated cases.

Baldomero M. "Toto" Olivera, a distinguished professor of biology at the University of Utah, grew up in the Philippines, where cone snails were a common staple in the fish markets. Olivera learned of the cone snail's poisonous venom in occasional stories of fishermen dying after being stung. For the past three decades, Olivera has been studying these venoms, composed mostly of peptides, to identify the specific peptide structures of the thousands of toxins that have evolved. Cone snails manufacture a variety of toxins each having a particular role to play in the capturing of prey, working effectively as a combination drug therapy. For example, a group of toxins at the injection site will immediately stun the prey while a second group will travel into the neuromuscular system causing paralysis so the prey remains immobilized while it is devoured. In addition, the snails produce precisely targeted toxins that work on particular receptors. The evolution of the targeting capabilities of cone snail venom has been a primary focus of Olivera's work. Drugs that have precise targeting abilities are less likely to have severe side effects, a great benefit to patients who take medicines for chronic conditions.

CHPC is working with Olivera's research group to create and maintain a web site that presents their research findings. Go to http://www.neogastropodtol.org to see videos of predatory cone snails in action as they immobilize and consume their prey.

Cone snail research has already led to the development of medicines for nervous system and cardiovascular disorders. Prialt, a drug which can be injected into fluid surrounding the spinal cord as a treatment for severe pain due to cancer, AIDS, injury and failed back surgery, was developed from cone snail research. The venomous cone snail Conus geographus produces a substance named conantokin-G. Cognetix Inc., a Salt Lake City company cofounded by Olivera in 1996, now is developing CGX-1007, a compound derived from conantokin-G, as a possible treatment to control seizures in patients with intractable epilepsy.

Understanding precisely how specific toxins work on their targets could also lead to additional methods of pain control. J. Michael McIntosh, one of Olivera's colleagues at the University of Utah Center for Neuropeptide Pharmacology, published findings in 2002 about two cone snail toxins RgIA and Vc1.1 that treat nerve hypersensitivity and pain in rats by blocking a cell molecule known as the "alpha9alpha10

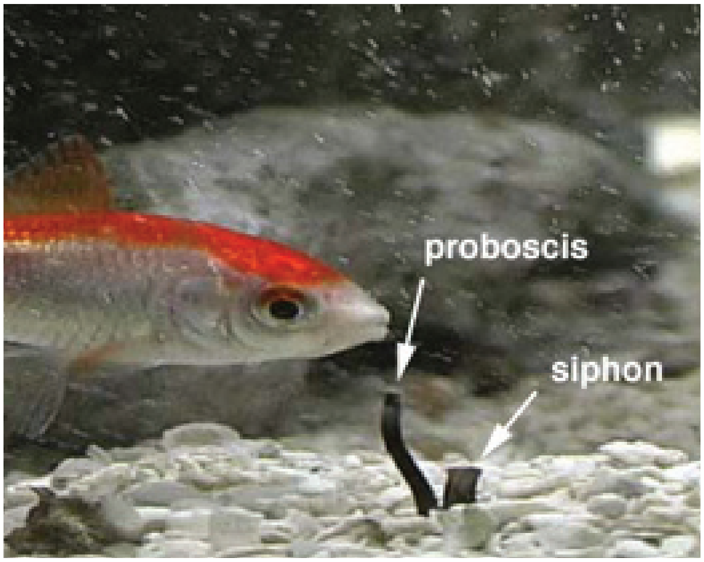

Figure 1. A goldfish is about to become the lunch of Conus Manachus - to view the capture, visit:
http://www.neogastropodtol.org/movies/Goldfish.mov

nicotinic acetylcholine receptor." The toxins were particularly effective in alleviating pain in rats with severe sciatic nerve damage. McIntosh plans to use these findings to develop a treatment for severe pain in humans that would similarly target the alpha9alpha10 nicotinic receptors found in nerve cells.

The deadly geography cone snail engulfs its prey. See it in action at http://neogastropodtol.org/movies/Geographus.mov.

Cone snail research has also provided provocative insights into human evolution. Olivera and his colleagues, biologists Pradip Bandyopadhyay and James E. Garrett, published the results of their research on a gene found in humans, fruit flies and cone snails that makes gamma-glutamyl carboxylase or GGC. Composed of "junk DNA" -- portions of the genetic code that are within genes but have no apparent function -- this gene is not only present in all three, it is also located in the same place in the genetic sequence, indicating that the gene and the enzyme it produces originated very early in the evolutionary chain. This discovery provides an interesting possibility that "junk DNA" is not a relatively recent addition to the human gene pool as many scientists argue. Nor is it "junk." Olivera speculates that it may have played a developmental role in the growth of embryos, giving chemical signals that prompted embryonic cells to differentiate into the types of cells needed within a living organism.

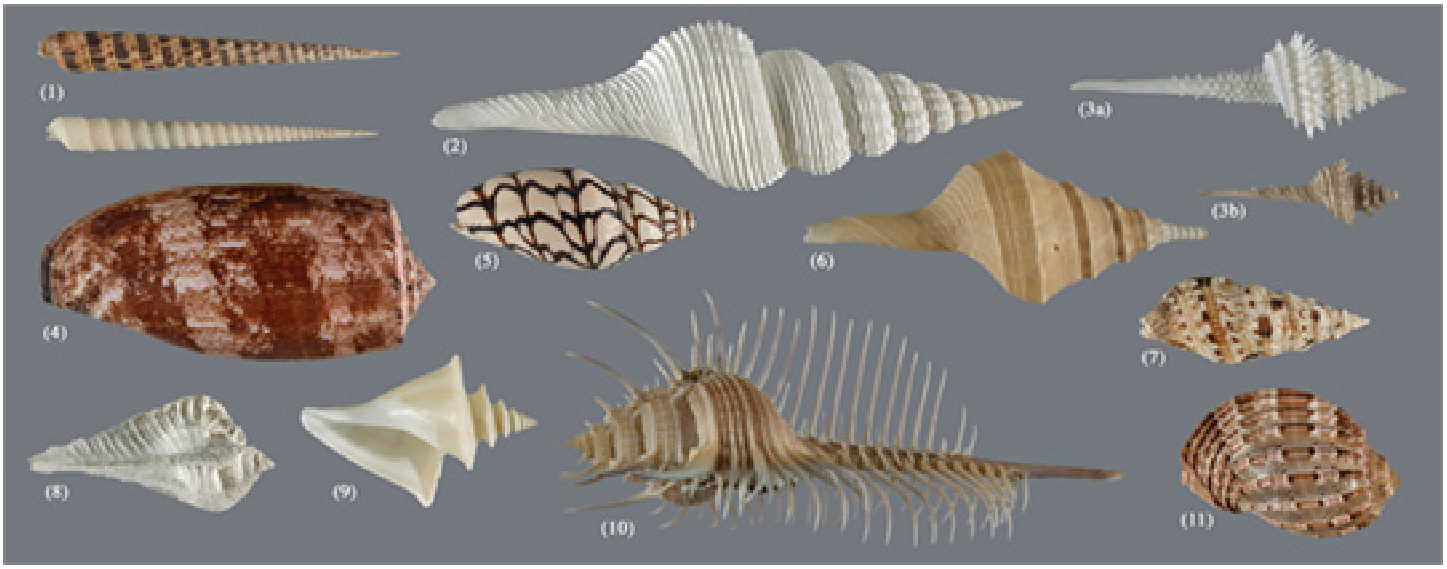

Figure 1: Neogastropoda Diversity. Neogastropods are among the most dazzling examples of the diversity that evolution has produced. Sculpturally the Neogastropoda include some of the most extreme shell morphology found in any living group, ranging from the exceedingly long, thin, many-whorled auger snails (family Terebridae, (1)), the amazing spines of some Muricids (family Muricidae, (10)), the elegance of the wonder shell, Thatcheria (family Turridae s.l. (9)), and Syrinx aruanus, the largest living snail that grows to be 1 meter long (family Melongenidae, (6)). The life-style diversity represented is similarly remarkable, ranging from the deadly geography cone that kills 70 % of the people that it stings (family Conidae, (4)) to marine vampires waiting in the shadows of coral reefs to suck blood from fish (family Colubraridae, (7)). The shells of most of the different products of the neogastropod radiation are illustrated; spindle shells (Fasciiolariidae, (2)); pagoda shells, genus Columbarium (3a,b)); volutes (family Volutidae, (5) harp shells (family Harpidae, (11)), and two fossils, a winged muricid (Muricidae, (8)) and a pagoda shell (Columbariidae, (3b). Note the similarity of living and fossil pagoda shells, and the morphological divergence of the Muricids (8 vs. 10). (Photographs by K. S. Matz)

Article

Product evaluations at CHPC – latest quad-core processors from AMD and Intel

Martin Čuma

Center for High Performance Computing, University of Utah

Evaluation of the computing technology plays an important role in high performance computing as it helps us to stay informed on the industry developments and define strategic directions. Part of this assessment is acquiring demonstration samples of the latest technology and benchmarking them with both generic and user specific applications. Recently, we have performed one such review.

Both AMD and Intel have released server quad-core CPUs. The Intel Xeon is offered in various clock speeds up to 3.0 GHz, with the mainstream chip, E5345, being clocked at 2.33 GHz and costing $455. AMD released its quad-core chip in September 2007; the fastest, Opteron 2350, is clocked at 2.0 GHz and costs $389. We have several dual-processor servers with the Xeon E5345 chips and obtained a demo machine from Dell with two Opteron 2350 CPUs. We ran a series of benchmarks to compare these two competing eight core servers.

In terms of raw performance of a single core, both processors have SSE instruction units capable of performing four dual precision floating point operations per clock cycle. The Intel processor should thus have about 16% advantage over the AMD chip (2.33 GHz vs. 2.0 GHz).

Raw processor speed is important, but in the era of multi-core processors and multi-processor servers, performance of the memory subsystem becomes crucial. The Intel and AMD solutions are quite different in this respect. Intel's strategy has been to keep the memory controller off the CPU. This makes the system design easier, but it hurts

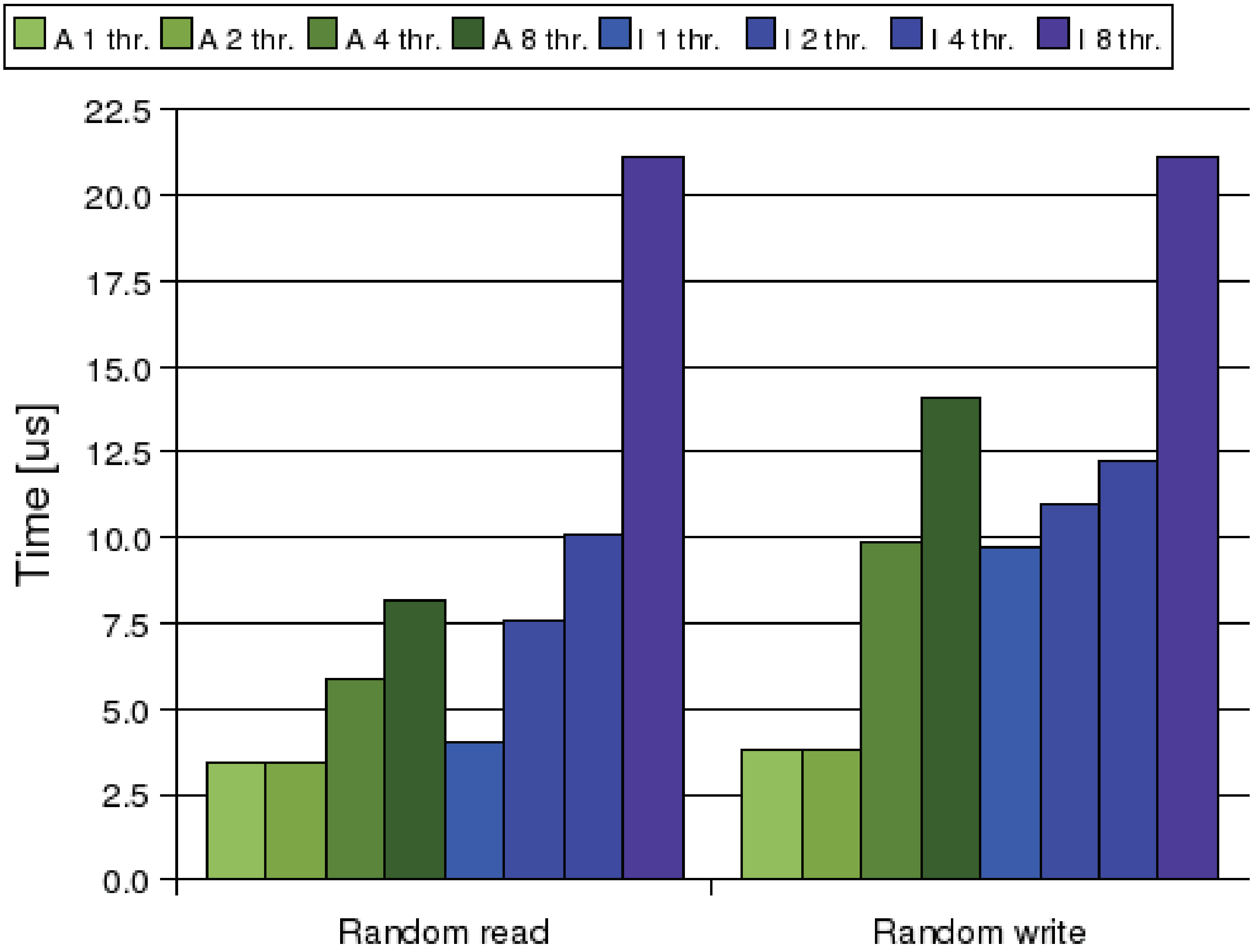

Figure 1. Random memory read and write time, A denotes AMD, I denotes Intel

performance when more cores try to access the memory at the same time. The AMD solution incorporates a memory controller on the processor and as such multiple processors can access the memory independently. The difference between these two approaches is evident in the results of various memory benchmarks. Here we present results of random read and write tests. We read or write randomly into a very large array, consecutively on 1, 2, 4 and 8 threads. Figure 1 shows the times that it takes to do 300,000 of these reads or writes on both the AMD and Intel processors. We notice that the AMD takes the same time to read or write for one or two threads, while the Intel's performance already deteriorates at two consecutive threads. By the time eight consecutive threads are used – which is what most users would want to do to fully utilize the server – the AMD system's memory access is about three times faster for reads and 50% faster for writes. Other types of memory benchmarks exhibit similar trends.

From the raw performance and memory benchmarks, we thus have a conflicting result, the former favors the Intel CPU, the latter favors the AMD. There are several benchmark suites that try to model performance of a wider array of scientific algorithms. One such benchmark suite is the NASA Advanced Supercomputing (NAS) Parallel Benchmarks (NPB). NPB is a set of programs derived from computational fluid dynamics (CFD) applications, that use generic algorithms very common in scientific computing, such as differential equation solvers, Fast Fourier transforms or random number generation algorithms. We have run a set of NPB benchmarks with OpenMP parallelization to get an idea of real-world applications performance on the two quad-core systems. For the Intel system, we built NAS with the latest Intel 10.0 compilers and aggressive optimization flags. For the AMD system, we have used the Pathscale 3.0 compiler with similarly aggressive optimization flags. Note that using different compilers and compilation flags can seriously affect performance. Intel compiler has consistently performed the best on the Intel CPUs. Pathscale compiler gave the best performance for the two benchmarks we present below.

The NPB results are often reported in mega operations per second per thread (MOps/sec/thread). In case of ideal parallel scaling using this metric, we would get the same performance value on multiple threads as we get on one thread. This is the case in Figure 2 for the EP (Embarrassingly Parallel) benchmark. This benchmark generates many pairs of random numbers, typical for Monte Carlo calculations. This algorithm puts the most stress on the CPU. As such, we see a very good parallel scaling on both systems; therefore, the Intel processor's higher raw computing power makes it a better choice than the AMD.

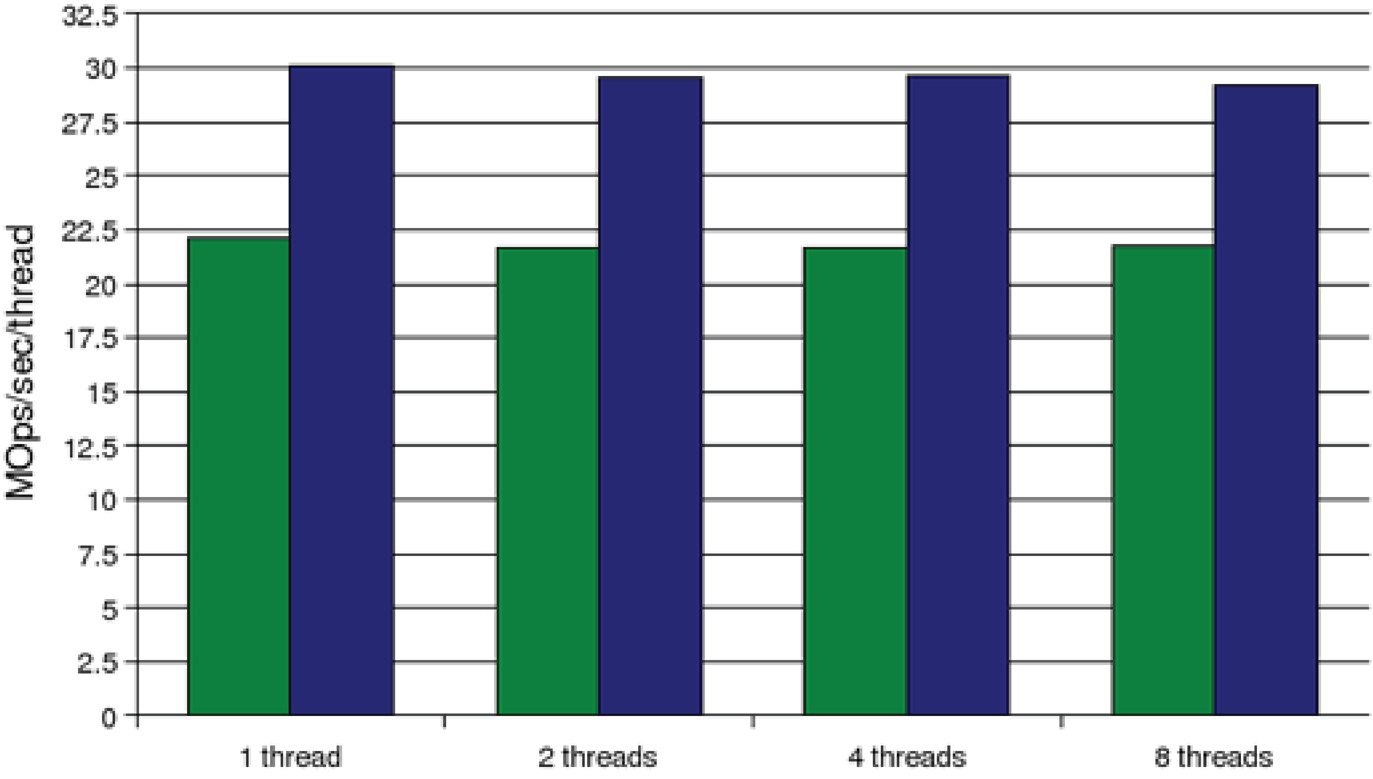

Figure 2. EP benchmark performance for Class B problem size, AMD in green, Intel in blue

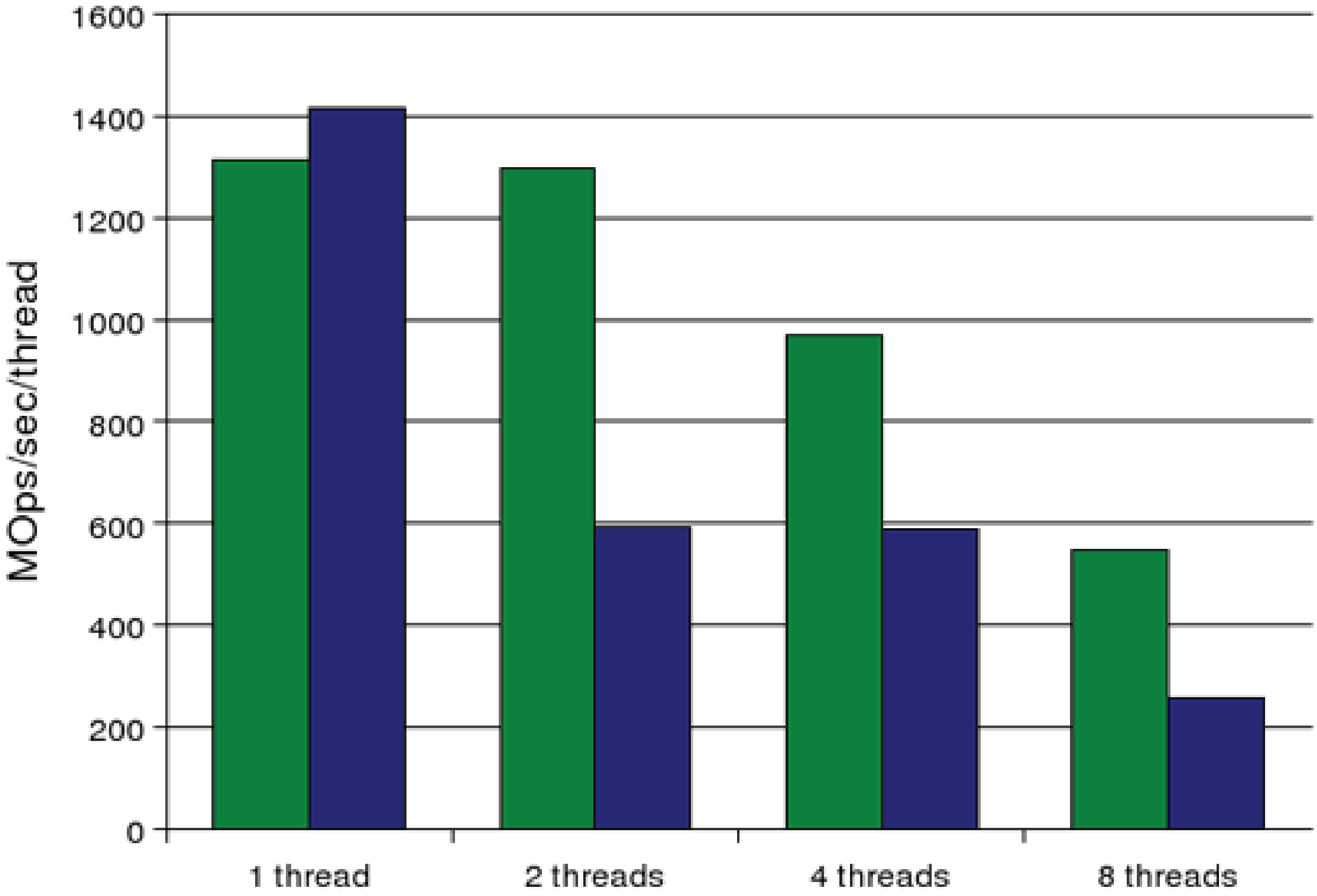

Figure 3. MG benchmark performance for Class B problem size, AMD in green, Intel in blue

However, most of the NPBs, as well as many scientific algorithms, are very demanding on all the computer subsystems, including the processor, memory and network. A good representative of this category of programs is the MG (Multi-Grid) benchmark, which represents a simple 3D multigrid solver. In Figure 3, we see that the Intel processor performs better than the AMD at one thread, but, for more threads it significantly lags behind. This is mainly because of its inferior memory subsystem. The AMD platform would be the best choice for this particular application.

The reality is most often somewhere between the two NPB benchmarks shown above. Since every program puts different amount of stress on different system components, it is advisable to obtain performance data on competing platforms for specific programs. For most user applications that we tested (NAMD, VASP, DLEVB), the Intel system performed slightly better than the AMD. For some (Amber) they were about even. Neither of the systems showed a major advantage over each other, so, in the final purchasing decisions, other factors, such as price and power consumption, should to be taken into account.

CHPC has been doing evaluations like this for many years in effort to get the best value for our researchers and the University. If any University faculty or staff have a computer system purchasing decision, be it a single server or a supercomputer, we will be happy to share our expertise to help you to get the best value.

FYI

CHPC maintains on its web site a listing of publications and talks that acknowledge the use of CHPC's resources. You can find the current listing at the following address:

http://www.chpc.utah.edu/docs/research/CHPCBibliography.pdf

If you utilize CHPC resources in your research, please include an acknowledgement in your publications and presentations. Also, please give us a copy for our records.

Article

New diagnostic tool for troubleshooting cluster jobs

by Erik Brown

Center for High Performance Computing, University of Utah

We've added a new diagnostic tool called `delve.pl` *for misbehaving cluster jobs. Users who have a job that is exhibiting unusual behavior or is failing to start and they want to submit a problem report can now execute the* `delve.pl` *program as an alternative to just submitting the job number to the problem tracking system. This is expected to increase the ability of the staff to troubleshoot problems with the clusters in that it gives a snapshot of the job at the time the problem is noticed. Many times, especially when submitting a problem after hours or on weekends, the job clears out before the systems people have a chance to observe it in the problematic environment.*

The `delve.pl` *program performs many queries, from the queue state to the availability of scratch space and allocation allotment. These data, along with a brief synopsis from the user of the perceived problem is sent to the problem reporting system with a copy sent to the user submitting the problem. The data is also entered into the problem system with the potential for correlation with future problems. The program can be found in the UUFS file structure at:*

```
  /uufs/arches/sys/pkg/arches_diagnostic/std/bin/
```

To run the script, `cd` to the above path and then enter:

```
  delve.pl <clustername> <jobnumber>
```

Here is an example of how you might use the script on a sanddunearch cluster job number 12345:

```
> cd /uufs/arches/sys/pkg/arches_diagnostic/std/bin/
> delve.pl sanddunearch 12345

  Briefly explain the trouble you're experiencing for
the problem report. Hitting the enter key will submit
your problem

 --> My job number 12345 has stopped producing output.
Please check on it.

 Thanks for the input, we will gather the needed infor-
mation and you will get a copy in your email shortly
It may take up to a half minute or so to complete.

Thanks for your submission. Your information has been
sent to problems.
```

We hope this will be a valuable tool for both users and systems staff to treat problems quickly as they arise in the cluster environment.

SC07 is the premier international conference on high performance computing, networking, storage and analysis.

CHPC has regularly participated in this conference for over a decade. This year we will again show off some of the research supported by the computational resources at CHPC. SC07 is to be held November 10th - 16th, 2007 in Reno, Nevada. If you would like your research showcased next year at SC08 in Austin, Texas, please let us know!

The UTAH exhibit from SC06

CHPC Staff Directory

Administrative Staff	Title	Phone	Email	Location
Julio Facelli	Director	585-3791	julio.facelli@utah.edu	410 INSCC
Julia D. Harrison	Associate Director	652-0019	julia.harrison@utah.edu	430 INSCC
Guy Adams	Assistant Director, Systems	554-0125	guy.adams@utah.edu	424 INSCC
Joe Breen	Assistant Director, Networking	550-9172	joe.breen@utah.edu	426 INSCC
DeeAnn Raynor	Administrative Officer	581-5253	dee.raynor@utah.edu	412 INSCC
Janet Ellingson	Admin. Program Coordinator	585-3791	janet.ellingson@utah.edu	405-2 INSCC

Scientific Staff	Expertise	Phone	Email	Location
Martin Cuma	Scientific Applications	587-7770	m.cuma@utah.edu	418 INSCC
Byron L. Davis	Statistics	585-5604	byron.davis@utah.edu	416 INSCC
Julio Facelli	Molecular Sciences	556-2426	julio.facelli@utah.edu	410 INSCC
Robert McDermott	Visualization	581-4370	robert.mcdermott@utah.edu	420 INSCC
Anita Orendt	Molecular Sciences	231-2762	anita.orendt@utah.edu	422 INSCC
Ron Price	Software Engineer & Grid Architect	560-2305	ronald.charles.price@gmail.com	405-4 INSCC

Systems/Network Staff	Title	Phone	Email	Location
Irvin Allen	System Administrator	231-3194	irvin.allen@utah.edu	405-40 INSCC
Thomas Ammon	Network Assistant	674-9273	thomas.ammon@utah.edu	405-22 INSCC
Wayne Bradford	System Administrator	243-8655	wayne.bradford@utah.edu	405-41 INSCC
Erik Brown	System Administrator	824-4996	erik.brown@utah.edu	405-29 INSCC
Steve Harper	System Administrator	541-3514	s.harper@utah.edu	405-31 INSCC
Brian Haymore	Lead, Comp. Cluster Admin.	558-1150	brian.haymore@utah.edu	428 INSCC
Samuel T. Liston	Digital Communication & Visualization	232-6932	sam.liston@utah.edu	405-39 INSCC
Jimmy Miklavcic	Multimedia, Telematic & Digital Communication	585-9335	jimmy.miklavcic@utah.edu	296 INSCC
David Richardson	Computer Technician	550-3788	david.richardson@utah.edu	405-38 INSCC
Steve Smith	System Administration	581-7552	steve.smith@utah.edu	405-25 INSCC
Neal Todd	System Administrator	201-1761	neal.todd@utah.edu	405-30 INSCC
Alan Wisniewski	Network Engineer	580-5835	alan.wisniewski@utah.edu	405-21 INSCC

User Services Staff	Title	Phone	Email	Location
Jason Duhaine	Systems Assistant	N/A	jason.duhaine@utah.edu	405-28 INSCC
Derick Huth	Technical Assistant	N/A	derick.huth@utah.edu	405-19 INSCC
Beth Miklavcic	Multimedia Design, Digital Video	585-1067	beth.miklavcic@utah.edu	111 INSCC
Paul Vandersteen	Technical Assistant	N/A	N/A	

The University of Utah seeks to provide equal access to its programs, services, and activities to people with disabilities. Reasonable prior notice is needed to arrange accommodations.

UNIVERSITY OF UTAH
Center for High Performance Computing
155 South 1452 East, RM #405
SALT LAKE CITY, UT 84112-0190

Welcome to CHPC News!

If you would like to be added to our mailing list, please fill out this form and return it to:

Janet Ellingson
UNIVERSITY OF UTAH
Center For High Performance Computing
155 S 1452 E ROOM 405
SALT LAKE CITY, UT 84112-0190
FAX: (801)585-5366

(room 405 of the INSCC Building)

Name:
Phone:

Department or Affiliation:
Email:

Address:
(UofU campus or U.S. Mail)

Thank you for using our Systems!

Please help us to continue to provide you with access to cutting edge equipment.

ACKNOWLEDGEMENTS

If you use CHPC computer time or staff resources, we request that you acknowledge this in technical reports, publications, and dissertations. Here is an example of what we ask you to include in your acknowledgements:

"*A grant of computer time from the Center for High Performance Computing is gratefully acknowledged.*"

If you use the NIH portion of Arches (delicatearch, marchingmen or tunnelarch), please add: "partially supported by NIH-NCRR grant # 1S10RR17214."

Please submit copies of dissertations, reports, preprints, and reprints in which the CHPC is acknowledged to: Center for High Performance Computing, 155 South 1452 East, Rm #405, University of Utah, Salt Lake City, Utah 84112-0190