User and Performance Impacts from Franklin Upgrades

Yun (Helen) He
National Energy Research Supercomputing Center
Lawrence Berkeley National Laboratory
Berkeley, CA 94720

ABSTRACT: *The NERSC flagship computer Cray XT4 system "Franklin" has gone through three major upgrades: quad core upgrade, CLE 2.1 upgrade, and IO upgrade, during the past year. In this paper, we will discuss the various aspects of the user impacts such as user access, user environment, and user issues etc from these upgrades. The performance impacts on the kernel benchmarks and selected application benchmarks will also be presented.*

KEYWORDS: Cray XT4, Franklin, NERSC, Quad Core, CLE 2.1, Application Performance, IO Performance, User Impacts.

1. Introduction

1.1 The Role of Franklin at NERSC

NERSC is the US Department of Energy’s (DOE) keystone high performance computing facility that serves the needs of the DOE and open science computational research community.

Franklin is the “flagship” system at NERSC serving about 400 projects and 3,100 scientific users in different application disciplines, including astrophysics, fusion, climate change prediction, combustion, energy, biology, and more [1]. It serves the needs for most NERSC users from modest (a few hundred cores) to extreme concurrencies (more than 8,000 cores). We expect a significant percentage of time to be used for capability jobs on Franklin.

1.2 Franklin before Upgrades

Before various major upgrades that began in July 2008, Franklin was a Cray XT4 dual core system, with 9,660 compute nodes (total of 19,320 processor cores). Its peak performance was about 101.5 TFlop/sec.

Each of Franklin's compute nodes consisted of a 2.6 GHz dual-core AMD Opteron processor with a theoretical peak performance of 5.2 GFlop/sec. Each compute node had 4 GBytes of memory, and the aggregate memory was 39 TBytes. Franklin’s high speed network was connected in a 3D torus configuration [2].

Franklin used two different operating systems. Full-featured SuSE Linux was run on service nodes. A light weight OS based on Linux, Cray Linux Environment (CLE), was run on each compute node. The parallel file system on Franklin was Lustre with approximately 350 TBytes of user disk space.

2. Franklin Benchmarks

2.1 Kernel Benchmarks

The kernel benchmarks selected for the Franklin procurement and long term system performance evaluation include benchmarks to measure performance in the areas of processor, memory, interconnect, and IO.

The interconnect latency is measured with the Multipong [3] benchmark. The NAS Parallel Benchmarks [4] (NPB) serial 2.3 Class B (best understood code base) and NPB parallel 2.4 Class D (Class D not available in NPB 2.3) are used for processor speed measurement. The STREAM [5] benchmark is used to measure the sustainable memory bandwidth. Finally, the IOR [6] benchmark is used for IO performance measurement.

2.2 Application Benchmarks

NERSC has a diverse user base compared to most other computing centers. Seven application benchmarks from different science disciplines were selected as Franklin benchmarks for the procurement and long term performance evaluation purposes [7]: CAM [8] (climate model), GAMESS [9] (computational chemistry), GTC [10] (fusion), MADbench [11] (astrophysics), MILC [12] (QCD), Paratec [13] (materials science), and PMEMD [14] (computational chemistry).

These seven applications represent over 85% of the NERSC workload (see Figure 1), also cover most frequently used programming libraries and programming languages, and have different performance requirements in CPU, memory, network and IO.

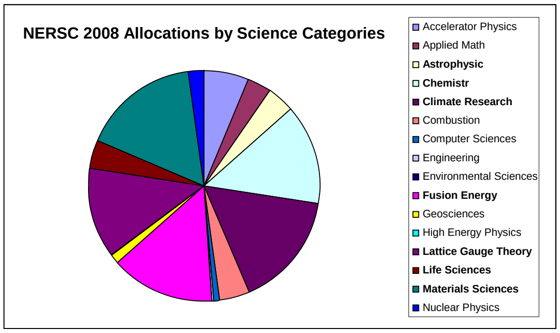


Figure 1. NERSC 2008 allocated computer resources by science categories.

Each application has a Medium test case (run on 64 processors, except CAM on 56 processors for technical reason) and a Large test case (run on 256 processors, except CAM runs on 240 processors for technical reason and GAMESS runs on 384 processors for compatibility with a DOD procurement benchmark). There is also an Xlarge case for MADbench (runs on 1,024 processors) and an Xlarge case for MILC (runs on 2,048 processors).

3. Quad Core Upgrade

3.1 Upgrade

NERSC upgraded Franklin to a quad-core XT4 between July and October 2008 [15]. The 2.6 GHz AMD Opteron dual core compute nodes were replaced with 2.3 GHz single socket quad core nodes (Budapest) with improved 128-bit floating point units. The theoretical peak for each compute core is 9.2 GFlop/sec (4 flops/cycle). The memory on each node was also doubled to 8 GB, keeping the same average of 2 GB/core. The new memory speed is 800 MHz, an improvement over the old 667 MHz chips. The theoretical peak performance of Franklin after the upgrade is about 356 TFlops/sec. Table 1 shows the Franklin configurations before and after the quad core upgrade.

Table 1. Dual Core and Quad Core Franklin Configurations.

	Dual Core	Quad Core
Compute nodes	9,660	9,660
Cores per node	2	4
Total compute cores	19,320	38,640
Processor core type	Opteron 2.6 GHz dual core	Opteron 2.3 GHz quad core
Theoretical peak per core	5.2 GFlop/sec	9.2 GFlop/sec
System theoretical peak	101.5 TFlop/sec	356 TFlop/sec
Physical memory per node	4 GB	8 GB
Memory usable by applications per node	3.75 GB	7.38 GB

The quad core upgrade was designed to be done in multiple phases in order to have maximum system availability and job throughput for the users. The goal was to deliver more than 75% of the original computing power on the production system throughout the upgrade. The number of upgraded columns was increased gradually over the phases to reduce risk of problems. The columns to be upgraded were migrated into a separate "test environment" system (called “Gulfstream”) where the hardware was physically replaced. NERSC selected friendly users stress-tested the quad core nodes during the testing and “burn in” time to check out the failed nodes, those columns were then integrated back into the production system. There was also a 7-day full system production stabilization time between each upgrade phase.

3.2 User Impact and Programming Environment Changes

The upgrade was done in four phases to minimize resource commitments and interruption for users. The production environment experienced very brief periods of system unavailability while the migration happened.

During various phases, all users had access to the Franklin "production environment," which was a mixture of dual core and quad core nodes. A job could be run on either set of nodes via specific settings in batch job scripts. A single job could not run on a mixture of nodes of differing core size. Franklin was set to be quad core default when majority of compute cores were quad core. Users were able to experiment more with the hybrid MPI/OpenMP paradigm with quad cores.

Quad core nodes were free of charging first, then they had the same charging factor as dual core nodes, i.e., charged only 2 cores per node for the allocation year 2008 and finally full charging. Due to the reduced number of nodes available during the upgrade, average queue wait time was longer. The situation became better after the upgrade completed.

Uncorrectable Memory Errors (UMEs) rates were higher than normal for the Franklin upgrade phase 3. User jobs had more failure rate due to these UMEs. The "bad" nodes have since being gradually swapped out and the UME rates have decreased as expected.

Although the CPU clock rate was reduced, memory speed improved. Overall application performances (NERSC Sustained System Performance) are about the same (~1% difference).

3.3 Performance Impact

Please note that the Franklin inter-node network topology was not a complete 3D torus during the course of the quad core upgrade. Some applications experienced some performance slowdowns and variation depending on job placement.

3.3.1 Latency

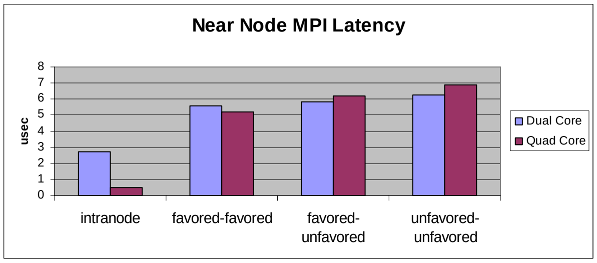


Figure 2. Comparison of near node MPI latency between Franklin dual core and quad core compute nodes.

Figure 2 shows the nearest node MPI latency for dual core and quad core nodes. There is one favored core, and one unfavored core per node for each dual core. And there is one favored core and three unfavored cores for a quad core node. Although the latency difference for each of the three pairs (favored-to-favored, favor-to-unfavored, and unfavored-to-unfavored) is not significant, the possibility of having unfavored-to-unfavored pair communication for the quad core nodes (9 in16 possibilities) is much higher than for the dual core nodes (1 in 4 possibilities).

The intranode improvement is mainly from using Message Passing Toolkit (MPT) version 3 instead of MPT2. Far node latency would be about 1.9 usec (35 hops with 0.053 usec per hop) extra on top of the near node latency with the 3-D torus Franklin full configuration.

3.3.2 STREAM

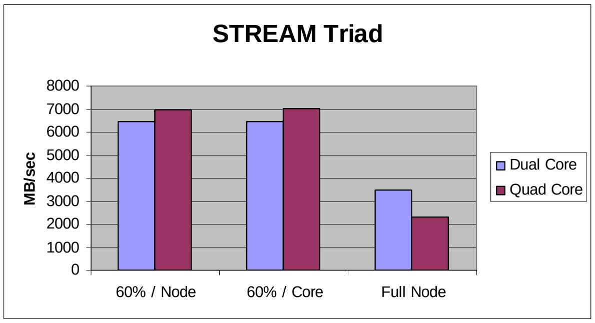


Figure 3. Comparison of STREAM Triad benchmark performance between dual core and quad core nodes on Franklin.

Figure 3 shows the STREAM Triad benchmark which measures sustained memory bandwidth on Franklin dual core and quad core nodes. The quad core performance of single core, using 60% node memory, and single core, using 60% core memory is higher than the dual core performance. And the dual core performance of all cores, using 60% node memory is higher than the quad core performance.

3.3.3 NPB benchmarks

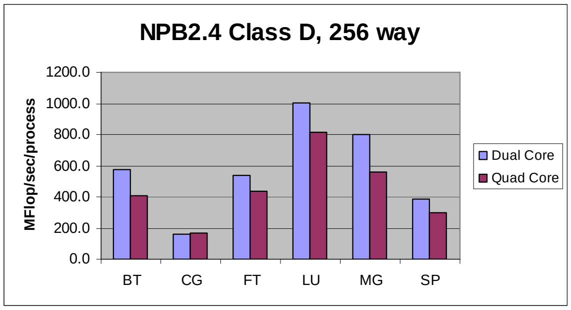


Figure 4. Comparison of NPB 2.4, Class D, 256 way benchmark performance between dual core and quad core nodes on Franklin.

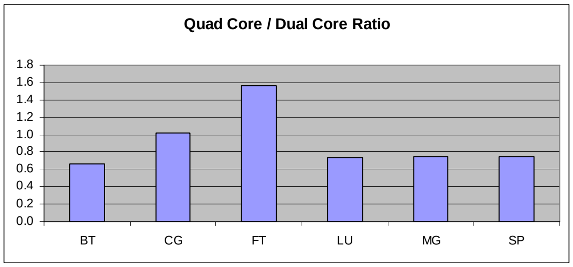


Figure 5. Quad core to dual core performance ratio of NPB 2.4, Class D, 256 way benchmark on Franklin.

Figures 4 and 5 show the performance comparison of NPB 2.4, Class D, 256 way benchmark between dual core and quad core nodes. Quad core node performance is mostly slower than dual core node performance, except for Conjugate Gradient (CG) benchmark. It is same for NPB 2.3 Serial, Class B and NPB 2.4 Parallel, Class D 64-way benchmarks (not shown). The dual core version of CG is highly tuned.

3.3.4 Application Benchmark

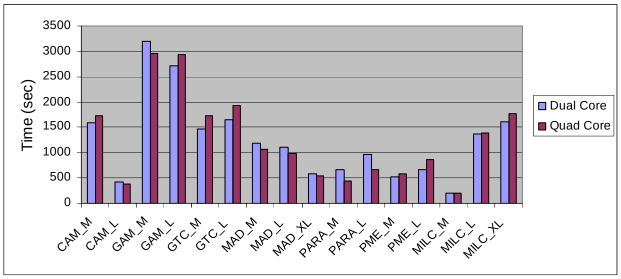


Figure 6. Comparison of application benchmarks run time between dual core and quad core nodes on Franklin.

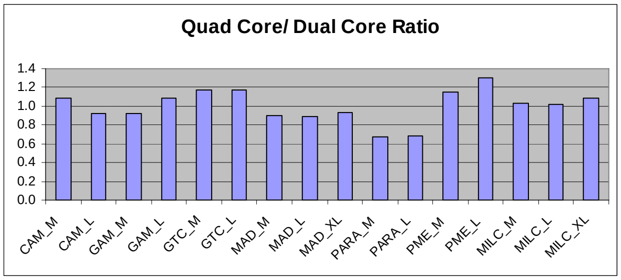
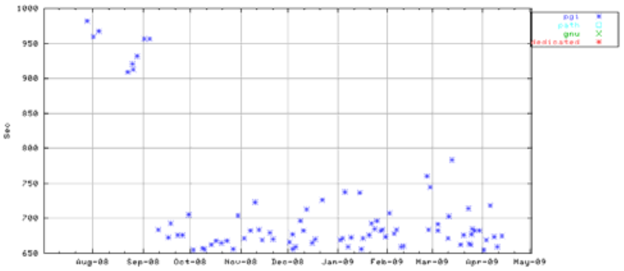


Figure 7. Quad core to dual core run time ratio of application benchmarks on Franklin.

Figures 6 and 7 show the performance comparison of application benchmarks between dual core and quad core nodes. Some applications are faster (Madbench, PARATEC), some are slower (GTC, PMEMD) on quad core nodes. Most applications differ within 20% except Paratec Large benchmark is 30% slower (see the run time increase from Sep-08 in Figure 8) and PMEMD Large benchmark is 30% faster (see the run time decrease from Sep-08 in Figure 9). PMEMD has large amount of short communication messages, so it is sensitive to latency and memory caching effect. Paratec takes advantage of SSE128 optimization on quad core nodes.

The overall application performance, the NERSC Sustained System Performance, which is measured by some geometric means of these seven applications, is about the same.



.

Figure 8. Run time of Paratec Large benchmark (run on 256 procs) on Franklin.

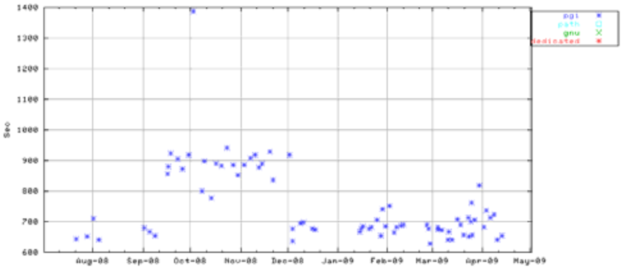


Figure 9. Run time of PMEMD Large benchmark (run on 256 procs) on Franklin.

4. CLE 2.1 Upgrade

4.1 Upgrade

The quad core upgrade migration system "Gulfstream" was used by Cray, NERSC staff and selected friendly users as the CLE 2.1 test bed before the upgrade. No major issues were found via the testing.

The CLE 2.1 upgrade was performed on Franklin on Dec 3-4, 2008. Major enhancements from CLE 2.0 to CLE 2.1 [16] are:

- SUSE Linux Upgrade: OS on the service nodes upgraded to SLES10 Service Pack 1.
- Lustre file system upgraded from release version 1.4 to 1.6.

- Comprehensive System Accounting (CSA) open source software is supported.
- NUMA Kernel: Kernel includes updates to include Non-Uniform Memory Access.
- Huge Pages: OS now supports 2 MB huge pages as well as the default 4KB small pages.
- System Resiliency Enhancements: System admin tools include new feature to recover from system or node failures.

The CLE 2.1 upgrade provided more potential system functionalities: Data Virtualization Service (DVS) for having compute nodes to have access to non-Lustre file systems and Checkpoint/Restarting capabilities for more flexibility in system handling.

4.2 User Impact

Users were asked to completely recompile their application codes. All the NERSC supported applications and libraries were also rebuilt. However, it was not obvious that the rebuilds needed to use MPT3 library. We encountered user codes built with MPT2 library that caused several system outages. The NERSC supported NWCHEM [17] binary also needed to be rebuilt based on the MPT3 compiled Global Array (GA) version 4.1. A complete post-mortem analysis was provided in Craw *et. al.* [18].

NERSC implemented an aprun wrapper that would test the user parallel code to see if it was built with MPT3 library. The launch was rejected if the parallel executable was built with MPT2.

4.3 Performance Impact

4.3.1 Latency

There were significant latency changes resulting from underlying portals software change. Under CLE 2.0 quad core, within each quad core node, there is one favored core and three unfavored cores.

Measured latency between two nodes could land in three different buckets: favored/favored core pairs with the average of 5.46 usec, favored/unfavored core pairs with the average of 6.09 usec, and unfavored/unfavored core pairs with the average of 6.74 usec. Under CLE 2.1, there are no more favored/unfavored cores in each quad core node, the latency between different cores is much more uniform and averaged out to be 6.46 usec.

4.3.2 STREAM

There are no significant performance differences in the memory benchmark STREAM TRIAD operation using three different configurations: 60% memory of each node, 60% memory of each core, or full node.

4.3.3 NPB Benchmarks

There are no noticeable performance differences for most NPB benchmarks, except the NPB 2.4, 64-way SP, which increased from 287 MFlop/sec/process with CLE 2.0 to 306 MFlop/sec/process with CLE 2.1. The SP performance has been seen to be very sensitive to compiler options and user environment changes (see Figure 10). Performance swings between 306 and 287 Mops/sec/process with the OS level and compiler version changes.

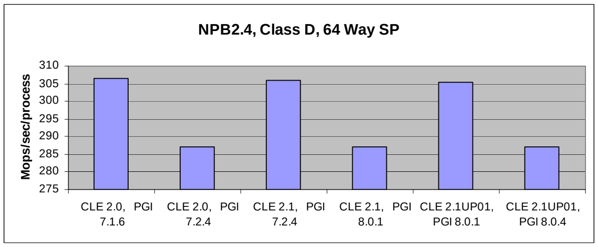


Figure 10. Run time of NPB 2.4, Class D, 64 way SP benchmark on Franklin.

4.3.4 Application Benchmarks

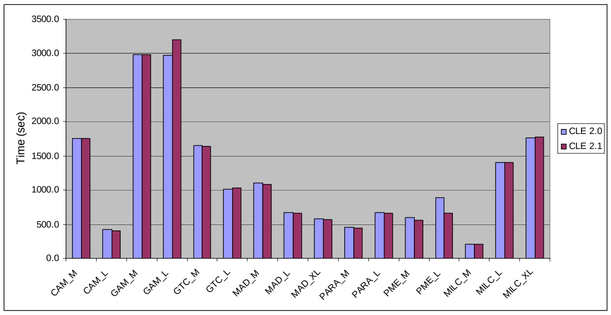


Figure 11. Comparison of application benchmarks performance between CLE 2.0 and CLE 2.1 on Franklin.

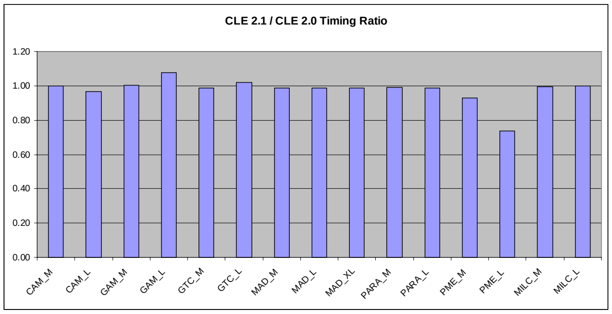


Figure 12 CLE 2.1 to CLE 2.0 run time ratio of application benchmarks on Franklin.

Figures 11 and 12 show the performance comparisons for these application benchmarks under CLE 2.0 and CLE 2.1. Most applications see within 3% performance differences, except GAMESS Large is 8% slower (see the run time increase from Dec-08 in Figure 13), PMEMD Medium is 7% faster and PMEMD Large is 26% faster (see the run time decrease from Dec-08 in Figure 9). GAMESS slowdown may be affected by a message passing library used in the application that is not quad core optimized. The speedup of PMEMD may be explained by large amount of short communication messages in the code being able to take advantage of latency changes and the memory caching improvement in CLE 2.1. The overall application performance, the NERSC Sustained System Performance, is about the same.

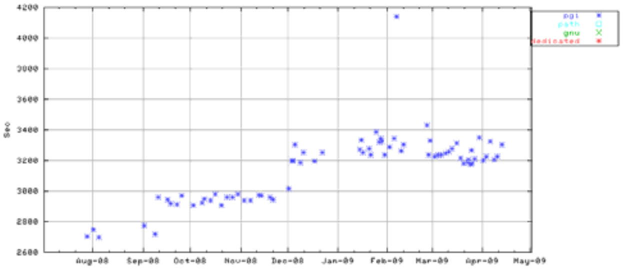


Figure 13. Run time of GAMESS Large benchmark (run on 384 processors) on Franklin.

5. IO Upgrade

5.1 Upgrade

The Franklin IO upgrade was completed during mid March to early April 2009 [19]. The upgrade included:

- Upgrade the interactive network adapters (PCI to PCI-e) to improve network performance between the interactive nodes and other NERSC systems, including NERSC Global File System (NGF /project).
- Double the number of I/O service nodes and upgrade their networking cards (PCI to PCI-e) to improve scratch IO performance.
- Separate the batch management (MOM) nodes from the login nodes.
- Reformat the /scratch file system. Introduce a new /scratch2 file system.
- Install service nodes for Data Virtualization Services (DVS) to be able to export NGF (/project) directly to compute nodes later this year.

Table 2 lists the Franklin configurations before and after the IO upgrade.

Table 2. Franklin Before and After IO Upgrade Configurations

	Before IO Upgrade	After IO Upgrade
Compute Nodes	9,660	9,572
Login Nodes	10	10
MOM Nodes	16 (also serve as login nodes)	6 (distinct MOM nodes)
I/O Server Nodes	32	56
DVS Server Nodes	0	20
File Systems	/scratch	/scratch and /scratch2
Storage	346 TB	420 TB (210 TB each)

5.2 User Impact

There were some day long outages and weekly maintenances during the IO upgrade. Users’ original /scratch data was removed during the reformatting (users were given enough advance notices to archive data). At the end, users saw significant IO performance improvement, especially for heavy IO applications. Interactive response on the login nodes was also improved.

Having two file systems allows less impact on one file system when there is IO contention on the other one. Users are free to choose which file system they would like to use, but having two copies of files are discouraged through a NERSC implemented job submission filter which rejects user jobs if the combined /scratch and /scratch2 usage is over the user quota.

The separation of login and MOM nodes helps to prevent user jobs failures due to login node crashes. When a login node also serves as a MOM node, all the jobs launched the MOM node will die if the many user processes (compiling, visualization applications, data transfer) overwhelm the login node's memory limit and cause the node crash.

5.3 Performance Improvement

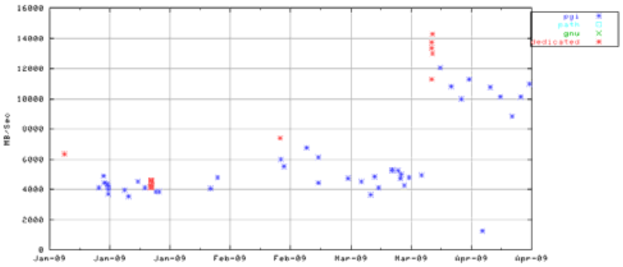


Figure 14. IOR benchmark aggregate read performance on Franklin.

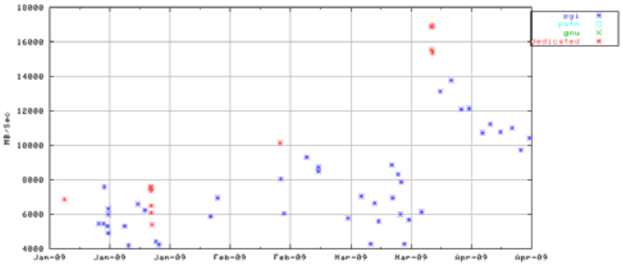


Figure 15. IOR benchmark aggregate write performance on Franklin.

Figures 14 and 15 show the IOR benchmark aggregate read and aggregate write performances. Both the dedicated and production performances improved significantly. Dedicated aggregate read rate improved from 7 to 14 GB/sec, and dedicated aggregate write rate improved from 10 to 17 GB/sec.

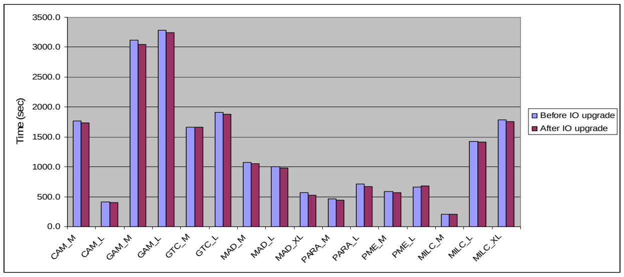


Figure 16. Comparison of application benchmarks performance before and after IO upgrade on Franklin.

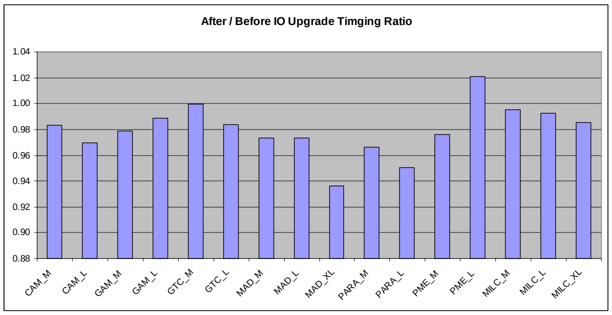


Figure 17. After and before IO upgrade run time ratio of application benchmarks on Franklin.

Figures 16 and 17 show the performance comparisons for these application benchmarks before and after IO upgrade. Most applications see slight (1~3%) performance improvement. MADBench Xlarge, which is a heavy IO code, is 6% faster (see the run time decrease from mid Mar-09 in Figure 18). Paratec Large is ~5% faster. PMEMD Large is 2% slower.

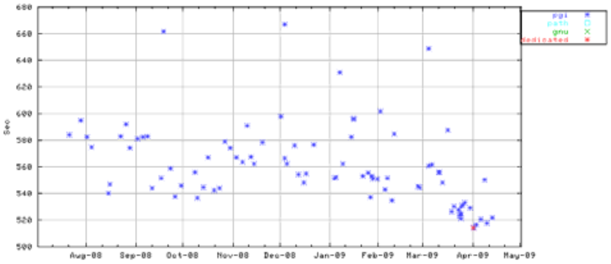


Figure 18. Run time of Madbench Xlarge benchmark (run on 1024 processors) on Franklin.

6. Summary

Franklin has undergone three major upgrades during the last year. With the collaborative effort and thoughtful planning from both Cray and NERSC, service interruptions were minimized during the upgrades. Users had free to half charging discounts for the earlier quad core stages.

Although users had to adapt to some of the programming environment changes and there were times that system was not very stable, the end results of these upgrades are quite worthwhile: With the quad core upgrade, we doubled the system size and deliverable computing cycles; With the CLE 2.1 upgrade, we have the potential to deploy DVS and Checkpoint/Restarting in the near future; With the IO upgrade, we more than doubled the aggregate IO performance; and overall we are having a more stable system.

Acknowledgments

I would like to thank Cray teams (remote and on site) and also our NERSC colleagues (NERSC management, User Services Group, Computational Systems Group, File Systems Group, and N5 Procurement Team) for their hard work on all aspects of Franklin. I would also like to thank NERSC users for their valuable feedback.

The author is supported by the Director, Office of Science, Advanced Scientific Computing Research, U.S. Department of Energy under Contract No. DE-AC02-05CH11231. This work used resources of the National Energy Research Scientific Computing Center, which is supported by the Office of Science of the U.S. Department of Energy.

References

1. Y. He, W.T.C. Kramer, J. Carter, and N. Cardo. Franklin: User Experiences. Cray User Group Meeting 2008.
2. NERSC Franklin Home Page: http://www.nersc.gov/nusers/systems/franklin/about.php
3. D. E. Skinner, Multipong benchmark.
4. NAS Parallel Benchmarks (NPB): http://www.nas.nasa.gov/Software/NPB/
5. STREAM Benchmark: http://www.cs.virginia.edu/stream/
6. IOR Benchmark: http://sourceforge.net/projects/ior-sio
7. W. T.C. Kramer, Y. He, J. Carter, J. Glenski, L. Rippe, and N. Cardo. Holistic Evaluation of\Lightweight Operating Systems using the PERCU Method. Draft, April 2008.
8. CAM Home Page: http://www.ccsm.ucar.edu/models/atm-cam/
9. GAMESS Home Page: http://www.msg.ameslab.gov/GAMESS/
10. Z. Lin, G. Rewoldt, S. Ethier, et. al, Particle-in-cell simulations of electron transport from plasma turbulence: recent progreess in gyrokinetic particle simulations of turbulent plasmas. Journal of Physics: Conference Series 16 (2005) 16-24.
11. L. Oliker, J. Borrill, J. Carter, D. Skinner, R. Biswas. Integrated Performance Monitoring of a Cosmology Application on Leading HEC Platforms. International Conference on Parallel Processing: ICPP 2005.
12. MILC Home Page: http://www.physics.indiana.edu/~sg/milc.html
13. Paratec Homepage: http://www.nersc.gov/projects/paratec/
14. PMEMD Home Page: http://amber.scripps.edu/pmemd-get.html
15. Franklin Quad Core Upgrade Web Page: https://www.nersc.gov/nusers/systems/franklin/quadcore_upgrade.php
16. Cray XT System Software 2.1 Release Overview. CrayDoc web site. http://docs.cray.com
17. NWCHEM Home Page: http://www.emsl.pnl.gov/docs/nwchem/nwchem.html
18. J. Craw, N. P. Cardo, Y. He, and J. Lebens. Post-Mortem of the NERSC Franklin XT4 Upgrade to CLE 2.1. Cray User Group Meeting. May 2009.
19. Franklin Stability and IO Upgrade Web Page: https://www.nersc.gov/nusers/systems/franklin/IO_upgrade.php