XT7? Integrating and Operating a Conjoined XT3+XT4 System

Shane Canon, Don Maxwell, Josh Lothian, Kenneth Matney, Makia Minich, H. Sarp Oral
Oak Ridge National Laboratory
Jeffrey Becklehimer, Cathy Willis
Cray Inc.

May 8, 2007

Abstract

The Center for Computational Sciences at Oak Ridge National Laboratory runs a single Cray XT system of directly connected XT3 and XT4 cabinets. We describe the processes and tools used to move production work from the pre-existing XT3 to the new system incorporating that same XT3, including novel application of Lustre routing capabilities. We also describe the ongoing operation and use of the system, including batch configuration and scheduling of the heterogeneous computing resources.

1 ORNL NCCS

The National Center for Computational Sciences (NCCS) was founded in 1992 to advance the state of the art in high-performance computing (HPC) by bringing a new generation of parallel computers out of the laboratory and into the hands of the scientists who could most use them. In 2004, the National Leadership Computing Facility (NLCF) was formed to provide the nation's most powerful open resource for capability computing, with a sustainable path that will maintain and extend national leadership for the Department of Energy's Office of Science (SC). Platforms of the NLCF are being housed in the NCCS at Oak Ridge National Laboratory.

An aggressive schedule is currently being implemented to provide a petascale machine for both the NLCF program and DOE's Innovative and Novel Computational Impact on Theory and Experiment (INCITE) program in 2008. This capability is being implemented in steps. It began with a single Cray XT3 development cabinet in January 2005, then a 10 cabinet Cray XT3 that formed the beginning of the production machine in March 2005. Soon to follow in April 2005 were 10 more cabinets at the beginning of the month and 20 more cabinets at the end of the month. The XT3 received its final 16 cabinets in June 2005 for a total of 56 cabinets. The theoretical peak of this machine was 25 Teraflops. In July 2006, the XT3 was upgraded to AMD Opteron dual-core processors bringing the theoretical peak to just over 50 Teraflops while plans were well underway to purchase a new Cray XT4. In November 2006, a 68 cabinet Cray XT4 arrived with plans to combine the XT3 and XT4 for a theoretical peak of over 100 Teraflops.

The steps and experiences in combining the XT3 and XT4 (XT7?) are outlined below as well as some of the current activities and challenges in running the machine. This somewhat heterogeneous combination required not only steps to migrate users' data but new features to provide users the ability to determine which part of the machine they were accessing in their runs. Both the Lustre data migration and the new batch features required to run the combined machine are discussed in detail along with ongoing operational issues.

2 Lustre Data Migration

Prior to transitioning users to the combined system, the existing users data stored in the XT3 Lustre file system was copied to the new system. This transfer was performed before the systems were combined, while the existing XT3 file system was still connected. Various strategies and techniques were examined for performing the data transfer. We considered providing a window for users to perform the transfer. We also considered various tools and approaches, including GridFTP, scp, and cross mounting the file systems. Ultimately it was decided that the staff would perform the transfer during a dedi-

cated time period. Furthermore, it was determined that the most effective approach was to mount both file systems on external hosts which would perform the actual copy. We will briefly discuss why this approach was chosen and some of the details on how the transfer was performed.

At the time of transition, a known performance bug existed in the version of the Lustre Client running on the XT service nodes. This bug significantly impacted read performance for small files. While this issue had been patched in newer versions of Lustre, the bug still existed in the Cray release and the patch was still undergoing testing by Cray for inclusion in a future release. As a result of this issue, performing the copy using XT service nodes would dramatically impact the time needed to perform the copy. However, it was possible to quickly apply patches to the Lustre client running on an external host in order to address this problem. Therefore, it was decided to adopt this approach. This bug also ruled out using other tools such as scp and GridFTP running on the service nodes, since they would still be affected by this bug.

In order to allow the two file systems to be mounted on external hosts, Lustre routing was configured. A detailed description of how to configure Lustre Routing can be found in [1], but we will briefly describe the configuration changes required for our circumstance. Lustre routing works by defining *networks* and *routes*. For the data transfer, there were three networks and four routes. The three networks included the two Portals networks on the XT3 and XT4 and a 10G Ethernet network. The XT3 routers, XT4 routers, and external transfer nodes were all attached to this Ethernet network. Since there were two Portals networks, it was important that the Lustre Networking (LNET) software could distinguish between the two. Therefore, on one system, the XT3, the Portals Lustre Networking Device was configured as `ptl1` versus the default `ptl` network id. The XT4 kept the default network. This allowed LNET to distinguish the two portals networks and route RPC traffic to the appropriate system. In addition to defining the networks, the routes had to be defined. LNET routes work by defining NIDS that can act as routers between two networks. It is also possible to create multi-hop routes, but that was not required for the data transfer. The final configuration is shown in Fig. 1. In the example below, two routers are defined for each XT system. In actuality there were eight routers defined for the XT4 and two routers for the XT3.

Once the routes were defined on all systems, then it was simply a matter of starting up lustre on the two systems. On the XT3, it was necessary to manually bring up LNET in order to force it to use the correct portals network id. This was done by issuing the following command to all Lustre OSS and MDS nodes on the XT3.

```
modprobe lnet;lctl net up ptl1
```

Furthermore, the Lustre configuration on the XT3 had to be modified to include `@ptl1` after each network identifier (NID) in the Lustre configuration script. This also required regenerating the XML configuration file and running the following command,

```
lconf --write_conf ./config.xml
```

to update the configuration of the file system. During this time, the file system was not actually mounted on the XT3. This could have been done, but would have required changing the mount commands to reflect the new Portals network number.

Once Lustre had been started on both systems, the file systems could be mounted on the transfer nodes. This was a typical Lustre mount with the exception that the Portals network had to be included along with the NID (i.e. `15952@ptl1`). An example mount command is shown here.

```
/bin/mount -t lustre
 15952@ptl1:/jaguar-mds/client /lustre/jaguar
```

Several custom utilities and scripts were used to perform the data transfer. To recreate the directory structure on the target file systems a specialized `find` utility was used which we term `xt-find`. This utility walks a directory tree much like the UNIX find command but in addition to dumping standard metadata information (i.e. permission, modification times, etc) it also prints out the Lustre striping information. This output was piped into another utility that would create the directory and file structure on the target file system.

Since both the source and target file systems were directly mounted on the transfer nodes, the standard UNIX copy utility, cp, could be used to transfer the data. Furthermore, since a stub file had been created on the target file system with the same striping parameters, all of the file system attributes could be preserved.

After some trial and error, the best method for maintaining a high sustained throughput was obtained by using a master process which tasked 24 worker threads. This was accomplished with relatively simple PERL scripts. The master process

modprobe.conf

```
options lnet ip2nets="
  ptl1        192.168.*.*          # XT3 linux nodes;\
  ptl0        192.168.*.*          # XT4 linux router;\
  tcp0(eth0)  160.91.205.[215,218] # XT4 router;\
  tcp0(eth0)  160.91.205.[210,211] # XT3 router;\
  tcp0(eth2)  160.91.195.[60-80]   # transfer nodes"\
 routes="\
  ptl0  1      160.91.205.[215,218]@tcp0 # XT4 <- IP;\
  ptl1  1      160.91.205.[210,211]@tcp0 # XT3 <- IP;\
  tcp0  1      [8,12]@ptl0     # IP <- XT4;\
  tcp0  1      [24,12824]@ptl1 # IP <- XT3 ;"
```

Figure 1: Routing configuration for LNET.

listened on a socket and sent the name of a single file. The worker threads requested a file, perform the copy, and continued. This approach insured the workload was relatively balanced across the transfer nodes.

Sustained transfer rates of around 750 MB/s were observed using this method. This corresponds to 375 MB/s per router on the XT3 which was the likely bottleneck. This was achieved using five transfer nodes each running five slave threads. Higher rates may have been possible by adding additional transfer nodes. However, the achieved rate was sufficient to complete the process in the allowed time. After completion of the data transfer, the same framework was adopted to perform a comparison. This comparison was performed using the standard UNIX diff command. Another approach would have been to perform MD5 checksums on each file. However, this would require reading the same amount of data, so there was no strong advantage to that approach.

Using this process roughly 20 TB of user data was transfered and verified over a weekend time period. This experience also illustrated the need for a parallel copy utility that would reliably perform the same service as the scripts and could be used by regular users. LCF is nearly finished developing this utility. Rather than creating a custom run system, the new utility uses MPI to perform the orchestration between the master thread and the worker threads. The system also distributes the block copy for large files over multiple threads to insure better load balancing.

3 Software Changes for the XT7?

The XT3 and XT4 are built with the same basic design. The two primary differences are memory and interconnect. The XT3 uses DDR memory while the XT4 uses DDR2 memory. As for the interconnect, the Cray SeaStar chip is used for the XT3 while the Cray SeaStar2 chip is used for the XT4. These architectural differences have the potential to impact application performance due to differences in memory bandwidth and interconnect injection bandwidths. It became clear early in the planning stages for the merging of the machines that a mechanism for being able to ask for and determine which architecture an application was accessing would be necessary.

3.1 Transition to TORQUE from PBSPro

NCCS had already successfully deployed Cluster Resources, Inc. (CRI) MOAB Workload Manager for batch scheduling on the XT3 with Altair's PBSPro resource manager in December 2005 [3]. In discussions with CRI, transitioning the resource manager from PBSPro to the open source TORQUE product primarily maintained by CRI seemed to be the correct course to follow going forward. The integration between MOAB and TORQUE is much tighter due to the fact that the same company controls both products. In the MOAB/PBSPro model, MOAB provides a set of nodes a job should use only to have that list thrown away by the PBSPro Machine Oriented Mini-server (MOM) which does the actual node allocation by creating a Compute Processor Allocator (CPA) partition. When designing TORQUE to run on the XT3, CRI made the decision to allow MOAB to do the actual CPA partition creation

and then provide that information to the TORQUE MOM for job launch. This meant that the set of nodes allocated by MOAB were the actual nodes used to run the job.

The MOAB/TORQUE design provides several advantages. With the MOAB/PBSPro design, there was a constant synchronization that had to occur to provide MOAB with the nodes it thought individual jobs were using. Early on, this caused problems for reservations and other time-critical scheduling activities during periods when the scheduler and resource manager disagreed. As time went on, fixes were provided to alleviate the problems, but it's rather easy to understand that having the scheduler and resource manager agree on the nodes allocated from the beginning is a better solution. Having the two disagree would not provide the ability to target individual architectures as was required. The possibilities for a placement algorithm to take advantage of topology also a reality with the MOAB/TORQUE design. There can be performance advantages gained when considering the topology of the interconnect as evidenced by prior studies, so it is essential that the scheduler and resource manager agree on the nodes to be used for a particular job.

Once the initial CPA code was integrated into MOAB, the steps to bring MOAB/TORQUE into production on the machine were minimal. One key feature that Cray had provided in PBSPro was the ability to spool both standard output and standard error to the PBS working directory defined by the environment variable PBS_O_WORKDIR. This feature allows the administrator to configure spooling for all jobs to use PBS_O_WORKDIR to avoid the pitfalls of using the /var filesystem for spooling which is the default behavior in PBS. With /var being a NFS-mounted filesystem in the ORNL XT configuration, there is potential for performance impacts both to the system which relies on the filesystem and to any jobs which heavily utilize standard output for results. Experience with PBS on other platforms has also resulted in downtimes when a user job filled up /var rendering the system unusable. Through a collaborative effort, CRI and ORNL made the necessary modifications to bring this feature to TORQUE. It is currently being used in production.

Other necessary modifications included minor changes such as importing the feature specification from TORQUE into the virtualization layer used by MOAB on the XT platform. The feature specification provides the interface to the user to request nodes based on node attributes or features in MOAB terminology. This will be discussed in detail in the next section. Additionally, one formatting change was necessary in TORQUE to accommodate five-digit node counts.

3.2 MOAB/TORQUE Node Attribute Implementation

Once the software was ported to the XT, the next step was implementation. While the typical implementation of node attributes involves the resource manager - TORQUE in this case, the XT platform creates challenges for that model since the compute nodes run the Catamount microkernel operating system which has very limited outside communication. MOAB imports information about each node through its virtualization layer, so MOAB features were defined for each node through the MOAB configuration file. Using an `#INCLUDE` parameter to keep the configuration file less cluttered, two files were included to specify XT3 or XT4 as appropriate based on node number (e.g., `NODECFG[0] FEATURES+=xt3`). Users may then specify a feature on the qsub command line based on their needs.

```
qsub -l feature=xt3|xt4 job.pbs
```

Features have also been used to target areas of the machine for hardware troubleshooting, analysis of different memory DIMMs, etc. MOAB provides the ability to create reservations based on features, so a targeted set of nodes can be associated with a feature, and a reservation with an access control list (ACL) for an individual user or group of users can be established to provide exclusive access to that set of nodes over a given period of time. This capability gives the administrator the ability to easily manage different pieces of the machine whether for hardware testing purposes or production use.

4 Physical Creation of XT7?

As plans were being made for the petaflop machine in 2008, space became a critical resource. The decision was made to house the future machine in the first floor 20,000 ft2 computer room. With that in mind, new space had to be found to house the current XT3 residing on the first floor in addition to the XT4 which had not arrived yet. The second floor 20,000 ft2 computer room was the obvious choice, but the raised floor was only two feet which would not accommodate XT fan cages. Furthermore, business systems, robotic laboratories and other clusters occupied the space. A massive moving and construction effort began in June 2006. In November 2006,

the XT4 arrived and went directly to the second floor. After the XT4 was accepted, users were transitioned from the XT3 to the XT4 to prepare the XT3 for moving day. The data transfer described above was performed prior to the shutdown and relocation of the XT3.

The move of the machine to the second floor and reassembly took approximately one week. Part of the activity involved the replacement of every voltage regulator module (VRM) in the XT3 as problems for particular applications were found a few months earlier. The XT3 was cabled up as a separate machine after the move for an initial stabilization period. During this time, the XT4 was running production, but a problem soon became evident as certain applications began experiencing uncorrectable DIMM errors. The DIMM errors were primarily on the Samsung DDR2 parts in the machine. The Micron DDR2 parts were not experiencing the same failure rate. Once the XT3 had been stabilized, the XT4 was taken down and the machines were cabled together. A decision was made to replace the Samsung DIMMs during the downtime for merging the machines.

5 Acceptance of XT7?

Both the XT3 and XT4 had their own acceptance tests individually which involved a suite of functionality, performance and stability tests, so the decision was made to only do a 72 hour stability test for the merged machine. The biggest problem encountered during acceptance was a Lustre issue. The S3D application code which is part of the acceptance suite uses a file per process model of checkpointing, so a job the size of the entire machine attempted to create approximately 23,000 sixteen megabyte files in lustre at once. This caused problems primarily for the Lustre metadata server.

Using a code developed for the functionality portion of the acceptance test called simpleio which also creates a file per process but in a much shorter timeframe, the failure was repeated. Cray has been working on a SMP kernel for the service nodes for some time, and it was undergoing testing at another site. Simpleio was provided for testing to determine if a SMP kernel might be the solution to the problem, but it caused other failures.

Multiple workarounds were then provided to increase timeouts for both clients and servers and to reduce the amount of output being generated by Lustre on the client side. One final problem with the metadata server being falsely marked down in the database due to missed heartbeats was temporarily solved by simply marking it back up as a real solution was being developed. Each of these problems were reported to Cray and several of these issues now have commits in a future release of the operating system.

The only other problems encountered during acceptance were two interconnect link failures which voided the currently running test. Acceptance of the XT7? was completed on April 3, 2007.

6 Current Activity on XT7?

There are currently three separate activities utilizing the 124 cabinet machine. First, production jobs are running with some codes scaling higher than ever before achieved. A wide variety of scientific fields are covered in the INCITE program including fusion, astrophysics, climate modeling, materials sciences, etc. Currently, the machine is running version 1.5.31 of the Cray Unicos/LC operating system with three patches. This particular release has proven to be very stable for the ORNL workload. The key driver to an upgrade will be the release of the much anticipated SMP kernel for the service nodes. This will provide the ability to take advantage of the dual-core AMD Opterons available on each service node. With the current kernel, only one core is available for processing. The hope is that this will provide a much needed boost to the performance of the Lustre servers.

Cray is using the XT7? for scaling and testing of the Compute Node Linux (CNL) operating system which will be the basis of the quad-core AMD Opterons scheduled to be available for XT4 in the fall of 2007. ORNL has plans to upgrade the XT4 to quad-core processors at the end of 2007 bringing the theoretical peak of the machine to approximately 250 teraflops. This testing is being accomplished by alternating use of the full machine and only the XT3 side two weekends per month. Recently, there have been some performance successes on codes important to ORNL. Other presentations at this conference will cover the results of CNL testing.

On the weekend that only the XT3 side of the machine is used, the XT4 side of the machine continues running production. This activity requires taking the entire machine down and routing each side separately so that each machine can run independently. This is accomplished by using two separate system management workstations (SMW) attached to two separate DataDirect Network RAID storage controllers. While the partitioning feature available

on the SMW could potentially accomplish the same task using only one SMW, a decision was made to keep the environments separate to allow SMW software upgrades and access to remain completely independent. Production could not be impacted by upgrades required for CNL testing. To this end, a separate Ethernet switch is now used to accomplish splitting the cabinets between SMWs. Each SMW is connected to two separate switches along with any cabinets that the particular SMW is responsible for managing and booting. While this requires a physical move of Ethernet cables from one switch to another to partition the machine, the benefits to production justify the inconvenience.

Finally, the XT7? is being used to test the N-way Catamount operating system. This is an alternative to CNL. The current Catamount operating system being used on the compute nodes of the machine will only operate on single or dual-core machines. The N-way effort underway at Sandia National Laboratories is an attempt to support multi-core processors and multi-processor nodes. ORNL is funding some of this effort as a risk mitigation strategy for CNL. The first test of N-way Catamount at ORNL using the XT7? was completed on April 23. More tests are scheduled in the coming months.

7 Problems

By far, the predominant problem being seen is hardware failures. Links in the interconnect go down causing the machine to fall apart quickly as messages continue to try to be sent across a dead link and cause congestion in the network. Several of the failures have been attributed to VRMs failing on the mezzanine cards which house the SeaStar chips particularly on the XT4 side of the machine. Recent power fluctuations due to lightning storms and running the High-Performance Linpack Benchmark may have contributed to the problem but are definitely not the full story as a high failure rate of mezzanine VRMs persists. Cray is investigating the high rate of link failures in an attempt to find the root cause.

Lustre failures are the second leading cause of problems and downtimes with the machine. As evidenced in the acceptance section, Lustre particularly the metadata server - can be overwhelmed at this scale. Some of the problems can be attributed to other failures in the machine such as skipped heartbeats and portals bugs. However, many of the most recent issues have been problems with the Lustre code itself. The most recent problem causes the object storage servers (OSS) to completely leave the Portals network with no logs left behind. This is obviously a very serious problem that has a critical status with Cray.

Finally, particular jobs can cause many compute nodes to be marked down. Typically, those nodes can be warmbooted and put back in service, but the root cause of the initial problem is still being investigated.

8 Conclusion

The recent upgrades of the Cray XT system at the Leadership Computing Facility have pushed ORNL to the forefront of high performance computing. With these upgrades, scientist have access to the most powerful open resource for scientific research. Researchers have already successfully scaled a variety of codes to run on the new system. As expected with any leap in capability such as this, problems and challenges have and continue to be encountered. However, the engagement of the many vendor partners (Cray, CFS, and CRI) have enabled LCF to overcome these obstacles. We look forward to further collaboration with these partners as we progresses towards the deployment of a Petaflop system in the near future.

Acknowledgments

The authors would like to thank the staff and colleagues who have contributed material to this paper. Also, we wish to acknowledge the work of Cray, Inc, Cluster File Systems, Inc. and Cluster Resources Inc, for their contributions towards this project. Thanks to Trey White for coming up with the title of the paper. Research sponsored by the Mathematical, Information, and Computational Sciences Division, Office of Advanced Scientific Computing Research, U.S. Department of Energy, under Contract No. DE-AC05-00OR22725 with UT-Battelle, LLC.

About the Authors

Shane Canon is the Group Leader for Technology Integration in the National Center for Computation Sciences (NCCS) at Oak Ridge National Laboratory (ORNL). He can be reached by E-Mail: canonrs@ornl.gov. Don Maxwell is a member of the HPC Operations Group in the NCCS. He can be reached by E-Mail: mii@ornl.gov. Josh Lothian, Ken Matney, and Sarp Oral are staff members in

NCCS at ORNL. Jeffrey Becklehimer is a Principal Engineer and Cathy Willis is an on-site System Analysts with Cray Inc.

References

[1] Cluster File Systems, Inc. Lustre manual. Web page. http://www.lustre.org/manual.html.

[2] R. S. Canon. A Center-wide File System using Lustre. In *CUG Proceedings*, 2006.

[3] D. Maxwell *et al. Moab Workload Manager on Cray XT3 In* CUG Proceedings*, 2006.*