# External Services on the NERSC Cray XT5 System

**Katie Antypas, Tina Butler, and Jonathan Carter**

***National Energy Scientific Computing Center, Lawrence Berkeley National Laboratory, Berkeley, CA 94720, USA***

**ABSTRACT:** Cray External Service offerings such as login nodes and file systems which are external to the main XT system, provide an opportunity to make High Performance Computing resources more robust and accessible to end users. This paper will discuss our experiences using external services on Hopper, a Cray XT5 system at the National Energy Research Scientific Computing (NERSC) Center. It describes the motivation for externalizing services, early design decisions, security issues, implementation challenges and production feedback from NERSC users.

**KEYWORDS:** Cray XT, external services

## Introduction

The National Energy Scientific Computing Center, located at the Lawrence Berkeley National Laboratory, is the flagship computing center of the US Department of Energy serving the whole range of science sponsored by DOE Office of Science. The current systems deployed include a more than 9000 node Cray XT4, Franklin, a 664 node Cray XT5, Hopper, and a 400 node IBM iDataplex, Carver. The systems share a common high-performance filesystem, based on GPFS, for permanent data storage and locally attached storage for temporary data.

Deployment of the next generation computing system, named after Admiral Grace Murray Hopper, is taking place in two phases. Phase 1, a Cray XT5, entered production on March 1, 2010. The Phase 2 system will be deployed in the fall of 2010 and will be provisioned with the Cray Gemini interconnect and nodes having two 12-core AMD Opteron 6100 series (Magny Cours) chips. This paper concerns our experiences in deploying external services, i.e. providing filesystem, login, and other functionality via external servers rather than through XT service blades, for the Phase 1 system.

## Motivation

While NERSC users have made great use out of the Franklin Cray XT4 system - routinely running applications to the full scale of the machine - the great variety of users, projects, applications and workflows running at NERSC have uncovered several complex failure modes and sensitivities. Many of these issues are tied to the fact that almost all user-level serial processing, e.g. compiling, running analysis, pre- or post-processing, file transfer from other

systems, etc., on the system is confined to a small number of service nodes. These nodes have limited memory (8 GB), no swap disk, and relatively old processor technology. Service nodes can be added, but must be swapped with compute blades in ratio of two compute blades for one service blade. Typical failure modes for the service nodes have involved out memory of conditions due to long and involved application compilation, analytics applications, file transfers, or many batch jobs using the node as a head or MOM node. A failure of a single node might then lead to many other processing failures, batch job termination, or sometimes a system wide outage. This problem was particularly acute on Franklin because of the large number of NERSC users logged into the system at a given time. Typically 200-300 scientists are logged into 10 Franklin service nodes during the day time hours and even if a service node failure did not occur, often the serial processing from one user would cause the node to be less responsive for other users. During the contract negotiation for the Hopper system, one of the goals was to architect a system based on a Cray technology offering that could incorporate the many advantages of a tightly coupled XT system, but that could address some of issues seen in day to day production. As a solution, Cray Custom Engineering designed a configuration consisting of a set of external servers, based on Dell hardware, to augment the tightly-coupled compute pool hosted on XT hardware.

## External Services Configuration

The Hopper Phase 1 compute portion is an XT5 that includes 664 compute nodes, each containing two 2.4 GHz AMD Opteron quad-core processors and 16 GB of memory. In addition, a set of service nodes act as MOM nodes for the Torque/Moab batch system, network nodes, data virtualization service (DVS) nodes, and Lustre router (LNET) nodes. In addition to the compute system, a set of external servers enhance this configuration. Unlike most XT5 systems in production, the Lustre filesystem is not hosted on the XT, but on a subset of the external servers (esFS nodes). An Infiniband DDR network, provisioned with QDR-capable switches, provides the connectivity between the OSS/MDS and Lustre router nodes. Eight of the external servers act as login hosts (esLogin) where users can build applications, submit batch jobs, and run serial pre- and post-processing steps. Finally, four servers act as data movers (esDM) providing offload from the login nodes for high bandwidth traffic to the NERSC mass storage system. External node characteristics and a schematic are shown in Table 1 and Figure 1. In the following subsections, we elaborate on the configuration and relevant pieces of the software stack running on each of these distinct external server types.

| Server Type | Count | Model | CPU | Memory |
|---|---|---|---|---|
| esLogin | 8 | Dell R905 | 4 x AMD Opteron Quadcore 2.4 GHz | 128 GB |
| esFS (OSS + 3 MDS) | 48+3 | Dell R805 | 2 x AMD Opteron Quadcore 2.6 GHz | 16 GB |
| esDM | 4 | Dell R805 | 2 x AMD Opteron Quadcore 2.6 GHz | 16 GB |
| esMS | 1 | Dell R710 | 4 x Intel Xeon Quadcore 2.67 GHz | 48 GB |

Table 1: External server configuration

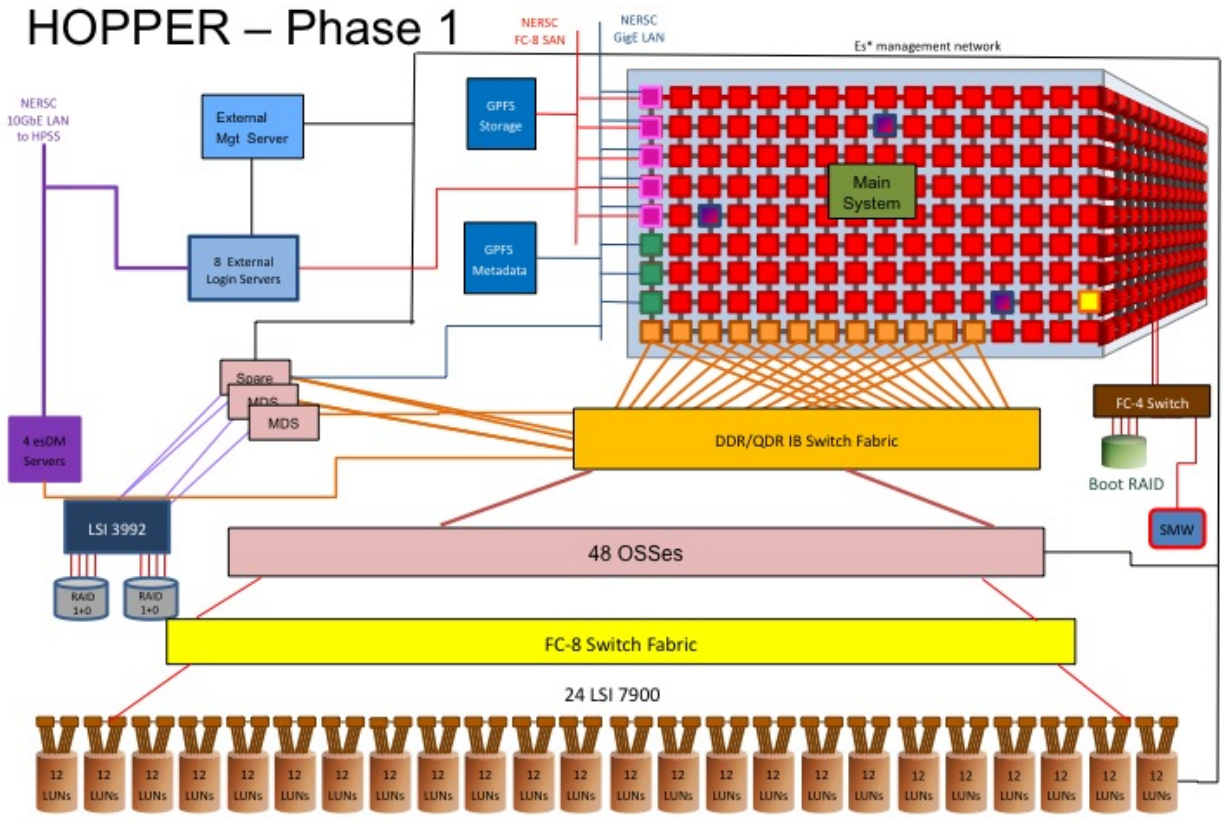


Figure 1: Schematic of Phase 1 system

**esLogin Nodes**

The esLogin nodes must provide all the functionality of an internal login node as transparently as possible, including: developing and building applications; obtaining system status and statistics; running jobs and interacting with the batch system. Developing and building applications for the XT is taken care of via the CADE/CADES package available from Cray that provides cross-compilers and libraries that can be used on many Linux servers [1]. Similarly, obtaining system status and node use statistics is enabled by the set of commands such as xtstat, etc., that are bundled into the Cray eswrap package. These commands contact the XT compute system and display results as if they were run on an internal service node.

For the batch system, each esLogin node runs a Torque job submission server with a set of routing queues that normally forward the job on to the central Torque server running on the system database node of the XT system. In the event that the central server is unavailable, the jobs are routed to local queues on each node and held. When the central server is available, these local queus are started and the jobs are routed to the central server. Since the numerous Torque servers can be confusing to users, a set of batch command wrappers were written by NERSC staff and provide batch environment continuity for the users by hiding the implementation details. For example, users see the same list of jobs from any esLogin node or internal node when executing the *qstat* command.

**esLogin Node Advantages: Usability and Productivity**

The added computational power and large memory available on the esLogin nodes has elicited the most positive response from users. One esLogin server has 4 quad-core processors and 128GB of memory compared internal service nodes which have two dual-core processors and 8GB of memory. This represents and increase of 4 times the computational power and 16 times more available memory than an internal service blade could offer. On the Franklin system some 250 users are logged in during a given week day. When the Hopper Phase II system enters production we expect the same number of users on Hopper. Figure 2 shows the average amount of memory per user on the login nodes for both Franklin and Hopper, assuming typical user load. Users on the Hopper system are able to compile applications, run python scripts, and post-processing applications such as IDL with much less interference from other users.

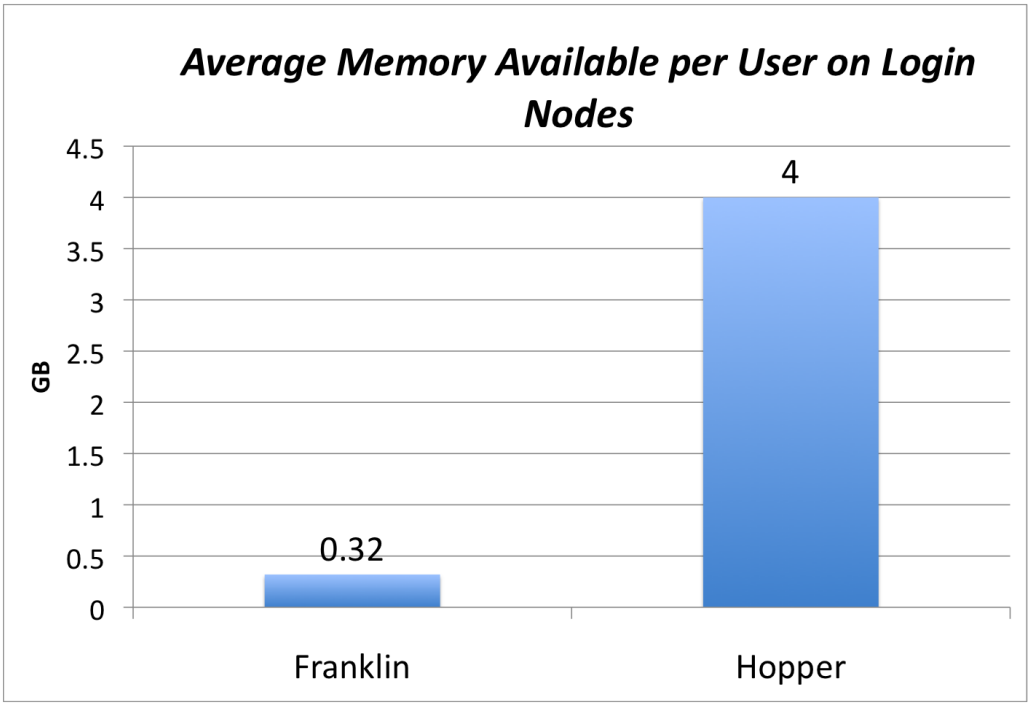


Figure 2: Memory available per user on Franklin (internal) and Hopper (external) Login nodes

Additionally, because the esLogin nodes are external to the main XT system, they are often kept available when Hopper is taken down for maintenance. This provides continuity for users who can login, compile applications and submit jobs which are held locally on the esLogin nodes until the XT becomes available again.

**esLogin Node Challenges: Software consistency**

The greatest challenge we encountered installing and configuring the esLogin nodes was maintaining software consistency both across the pool of eight esLogin nodes, as well as between the EsLogin nodes and the internal MOM nodes. esLogin node configuration is not managed by the "shared root" feature of the internal compute portion of the XT and instead is managed by a tool called *SystemImager* [2]. An esLogin node is designated as the master image and then that image is synced to the other esLogin nodes. System Imager is a new tool for Cray and so the primary challenge was getting familiar with a new tool and deciding which components needed to be the same across all esLogin nodes and which components needed to be unique to each esLogin node.

In addition to software consistency between the esLogin nodes, it is also important for the software versions and installation paths to be the same on the esLogin nodes as on the internal service nodes. This is because while users compile applications on the esLogin nodes, jobs are launched from the internal service nodes and users rightly assume that software versions will be the same. Currently there is no supported way to verify that the esLogin node versions are compatible with the internal service node software. Software for the esLogin nodes and internal service nodes are packaged and distributed separately and are installed individually, and in addition must be verified by hand. While different software versions is less of a problem with

statically built applications, now that the XT platform is supporting dynamic and shared library applications, software that is available on the esLogin nodes must also be available on the internal service nodes.

Another issue NERSC scientists have faced when running on the Phase I Hopper system is the problem of importing the esLogin node environment to the internal service nodes. Many scientists use the -V flag in PBS batch scripts or as a qsub command-line argument to instruct the system to pass the environment from the esLogin nodes to the batch system. However, when the paths to software is different on the esLogin and internal service nodes this can cause commands and applications to fail. One simple example is the Lustre *lfs* command. On the esLogin nodes, *lfs* is installed in the traditional Linux location in /usr/bin/. On the internal service nodes however *lfs* is installed with Cray Lustre software in: /opt/xt-lustre-ss/[version]/usr/bin. If both the *lfs* command and -V flag are used in a batch script, the *lfs* command will fail since the system is looking for the path assigned on the esLogin node and not the internal service node. Cray and NERSC staff are discussing possible solutions for these cases as it may not be possible to get software install paths identical in all cases.

Additionally, there are some ALPS commands that worked on the internal service nodes that now require passwords from the esLogin nodes. The *apstat, xtnodestat* and *xtprocadmin* commands are useful for users and administrators of the system. These tools all use the ssh protocol to authenticate when used on the esLogin nodes and subsequently require passphrases to be typed whenever they are executed.

**esFS Nodes**

The esFS nodes are divided between 3 MDS and 48 OSS nodes. Filesystem storage is provided by 24 LSI 7900 storage subsystems, each with dual controllers and 120 TB of disk configured as 12 RAID6 (8+2) LUNs for a total of more than 2 PB of user accessible storage. In the current Phase 1 system, the OSS are connected via DDR Infiniband fabric to 24 internal service nodes serving as Lustre routers. The esFS serves two independent, identically-sized scratch filesystems. While the esFS nodes run Sun/Oracle supported Luster 1.8.1.1 GA servers, the internal router nodes run Cray supported Lustre 1.6.5 clients.

**esFS Benefits**

One of the key benefits of the esFS is to provide highly reliable data storage, and to complement the benefits of the esLogin nodes. The combination of the two enables a consistent, highly reliable interface to be presented to users. In the event that the compute portion of the system is down for maintenance or failure, users can still access data, compile applications, and submit jobs.

The esFS is configured with OSS failover pairs that allow LUNs to be served by a single OSS while the other is down; this, and the fact that each external node can reliably be booted separately, means that rolling upgrades can be performed, i.e. a new version of Lustre can be installed on the servers one at a time until the whole set is upgraded, minimizing downtime. In

fact this was performed successfully, upgrading Hopper's esFS OSSes from Lustre 1.8.0.1 to 1.8.1.1 with the system in production.

### esFS Challenges

Automated failover on the esFS is still a work in progress. Failover at the disk controller level using RDAC multi-pathing has worked well, and manual failover between OSS pairs has been robust and beneficial in dealing with Lustre bugs and a rolling upgrade. However, automated failover between OSS and MDS nodes is not in service yet. As a consequence, it was necessary to schedule the MDS upgrade to Lustre 1.8.1.1 during dedicated time.

As mentioned in the previous section, the Lustre software on the esFS nodes is the Oracle/Sun GA release, so updates and patches come from Oracle, not from the Cray Lustre support group. This has lead to confusion in procedures for dealing with security patches, tuning, and other issues.

### esDM Nodes

The final set of external servers are configured as data movers. In the case of the NERSC XT4 Franklin system, transfers to the HPSS mass storage system were initiated by clients running on internal login nodes or internal MOM nodes and often created problems from consuming large amounts of memory or cpu resource. A transfer would be initiated on an internal MOM node when users include a file transfer inside a batch script. The objective in deploying the esDM nodes is to provide a resource where the these data transfers can be off-loaded thereby freeing up login nodes for other work.

### esDM Benefits

At this time on the Hopper Phase I system the esDM nodes do not appear to be providing faster transfer times to HPSS than the esLogin nodes. The Phase I system however, is small compared to the final Hopper Phase II configuration arriving later this year and so we have yet to measure the all of the potential uses and benefits of the esDM nodes.

### esDM Challenges and Opportunities

The third-party transfer agent is a new feature for HPSS and, as such, required additional development and debugging by the NERSC Mass Storage group. Possible additional functionality for the esDM nodes is to provide a platform for grid-based file staging and to interface with a true hierarchical storage management system.

### esMS Node

The esMS node is the management server for all the external servers - Login, FS, and DM. The esMS is used for standard cluster management functions such as node provisioning, power control, and unified error logging, for the external server nodes and the LSI 7900 storage subsystems.

### esMS Benefits

NERSC chose to use a separate server node for these functions rather than connecting all the administrative networks to the XT's SMW. This provides an additional layer of security and further reduces dependencies between the XT compute partition and the external services.

### esMS Challenges

The cluster management solution for the esMS is still a work in progress. System Imager is primarily in use for synchronizing and backing up the esLogin nodes.

## Summary

The external services deployed with the Hopper Phase 1 system make a very strong case that augmenting a standard Cray XT in this way produces a much more usable and reliable HPC system. For example, many of the disadvantages present with the initial deployment of our previous XT4 system discussed in previous sections, such as out of memory conditions, availaibility of data when the system is down, etc. have been completely remedied.

While there have been many advantages in deploying external servers, a number of issues have arisen that require comment. Since external services are not developed by Cray Scalable Systems, but designed and deployed for each customer individually by Cray Custom Engineering, the support model is different from other Cray products. For example, as external servers are relatively new, a full cluster configuration management suite was not included as part of the deployment for the Phase 1 system. This resulted in several inconsistencies between nodes for the first few weeks. In addition, the documentation for the external servers is much less polished than other Cray documentation, and many systems procedures were originally poorly documented. Specifically in the case of the esFS nodes, the security model for applying patches is different from the rest of the system as Cray relies on standard Sun/Oracle Lustre server software support which is available only for approved kernels. Taken together, these issues create concern over the ongoing support of the external servers we have. Finally, unrelated to the issue of two support models, software synchronization could be improved across external and internal login nodes having a single software release for CADE/CADES which can be installed on either MOM or esLogin nodes, rather than as two distinct releases as it is done currently. This would help ensure the software versions are consistent across internal and external nodes.

In summary, we think that the advantages offered by external service nodes are so compelling that a standardized offering of this kind would be extremely popular with many customers. A product offering designed and supported by Cray Scalable Systems would make an improved and uniform set of external servers available as addition to the Cray XT product.

## References

[1] Cray Application Developer's Environment Supplement Installation Guide, S-2485-17, Cray Inc.

[2] SystemImager: http://wiki.systemimager.org/index.php/Main_Page

## Acknowledgments

This work was supported by the Director, Office of Science, Office of Advanced Scientific Computing Research, of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231. The authors would like to thank Wendy Lin for writing the batch script wrappers described in the esLogin section. The authors would like to thank the Cray on-site and Cray Custom Engineering staff for valuable discussions.