LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

UCRL-TR-218537

Certification of Completion of Level-2 Milestone 464: Complete Phase 1 Integration of Site-Wide Global Parallel File System (SWGPFS)

S. T. Heidelberg, K. J. Fitzgerald, G. H. Richmond, H. A. Wartens

January 31, 2006

This work was performed under the auspices of the U.S. Department of Energy by University of California, Lawrence Livermore National Laboratory under Contract W-7405-Eng-48.

Introduction

This report describes the deployment and demonstration of the first phase of a Site-Wide Global Parallel File System on the open network. The report and the references herein are intended to certify the completion of the following Level 2 Milestone from the ASC due at the end of Quarter 4 in FY05:

Milestone: 464

Title: Complete Phase 1 Integration of Site-Wide Global Parallel File System (SWGPFS)

Category: Campaign 11—NA113, Advanced Simulation and Computing

ASC Program Element: Simulation and Computer Science

The milestone is defined as follows:

"At LLNL, the Lustre file system will be deployed to create a new Site-Wide Global Parallel File System (SWGPFS) for both the open and classified networks. On the open network, SWGPFS will be the primary data resource for capacity systems, BlueGene/L, and visualization resources and will have high-speed access to the HPSS archive. Deployment on the classified network will follow at a later date when appropriate multi-cluster security plans are in place. For this milestone, Phase 1 of the SWGPFS will be deployed and scalable file system functionality will be demonstrated between a minimum of two LLNL ASC platforms and archival storage on the open network. File system performance will be demonstrated using the IOR test suite to show transfers between the Lustre-enabled clusters with a minimum of 60% of the effective measured aggregate network and I/O bandwidth available and a target of 80%. Archive performance of at least one GigaByte per second will also be demonstrated using HPSS interfaces to the archive."

Milestone integration/interfaces we defined as:

"Integration of an initial SWGPFS requires continued cooperation between ICC, PSE, and VIEWS program elements at LLNL. The Lustre file system PathForward effort also requires continued tri-lab cooperation with LANL and SNL. The SWGPFS team will work closely with the HPSS project and file system and platform vendors to ensure successful early deployment of this new file system model."

The milestone was completed September 23, 2005. We demonstrated that two ASC clusters of heterogeneous architecture, sited in different buildings, can share the same parallel filesystem, and that the performance achieved from both clusters exceeds the required levels, and either exceeds, or very slightly under achieves, the desired levels. In addition we demonstrated archival data rates between the shared parallel file system and the archival system satisfied the 1GB/s archival data rate requirement.

Background

There has been substantial development of the Lustre parallel filesystem prior to the configuration described below for this milestone. The initial Lustre filesystems that were deployed were directly connected to the cluster interconnect, i.e. Quadrics Elan3. That is, the clients (OSSes) and Meta-data Servers (MDS) were all directly connected to the cluster's internal high speed interconnect. This configuration serves a *single* cluster very well, but does not provide sharing of the filesystem among clusters.

LLNL funded the development of high-efficiency "portals router" code by CFS (the company that develops Lustre) to enable us to move the Lustre servers to a GigE-connected network configuration, thus making it possible to connect to the servers from several clusters.

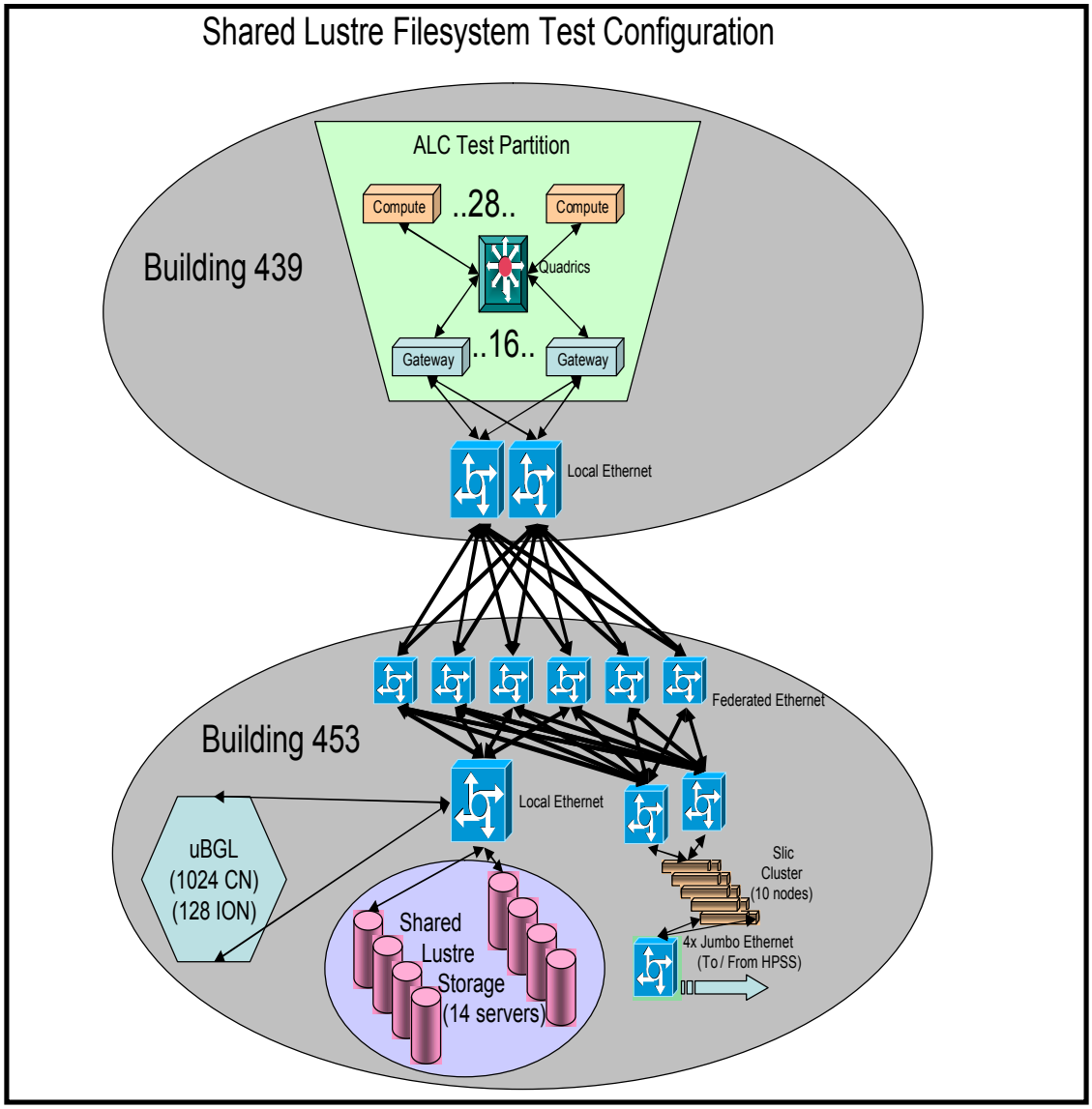
With portals routing available, here is what changes: (1) another storage-only cluster is deployed to front the Lustre storage devices (these become the Lustre OSSes and MDS), (2) this "Lustre cluster" is attached via GigE connections to a large GigE switch/router cloud, (3) a small number of compute-cluster nodes are designated as "gateway" or "portal router" nodes, and (4) the portals router nodes are GigE-connected to the switch/router cloud. The Lustre configuration is then changed to reflect the new network paths.

A typical example of this is a compute cluster and a related visualization cluster: the compute cluster produces the data (writes it to the Lustre filesystem), and the visualization cluster consumes some of the data (reads it from the Lustre filesystem). This process can be expanded by aggregating several collections of Lustre backend storage resources into one or more "centralized" Lustre filesystems, and then arranging to have several "client" clusters mount these centralized filesystems. The "client clusters" can be any combination of compute, visualization, archiving, or other types of cluster.

This milestone demonstrates the operation and performance of a scaled-down version of such a large, centralized, shared Lustre filesystem concept.

Description of the Hardware Configuration

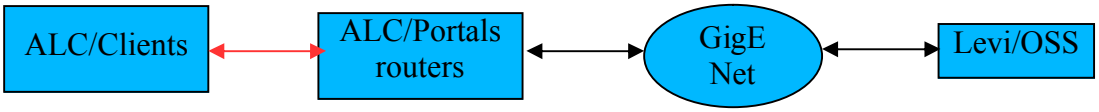
A diagram of the network and disk configuration used for the demonstration for this milestone is shown on the next page.


Shared Lustre Filesystem Test Configuration
ALC Test Partition
Compute
..28..
Compute
Quadrics
Gateway
..16..
Gateway
Building 439
Local Ethernet
Federated Ethernet
Building 453
Local Ethernet
uBGL
(1024 CN)
(128 ION)
Shared
Lustre
Storage
(14 servers)
Slic
Cluster
(10 nodes)
4x Jumbo Ethernet
(To / From HPSS)

The table below shows the role of the cluster nodes, their hardware architecture, and the version of Linux that was running on each cluster component of the milestone configuration. Note that this configuration is quite heterogeneous.

Name	Lustre role	Architecture	Linux Version
ALC	Clients, portal routers	X86	2.4.21
uBGL	Clients	ppc32	2.4.19
Levi	Servers (OSS/MDS)	X86-64	2.6.9
Slic	Clients	Itanium	2.4.21

ALC: ALC clients are *indirectly* connected to the IP network to which the Lustre storage servers are connected. In this configuration, data flows first over ALC's Elan3 network from clients to nodes that are assigned to be "portals router nodes," and then the data flows from the portals router nodes over gigabit-ethernet networks to the Lustre storage server nodes. The "portals router nodes" are required to "gateway" the portals-over-Elan3 streams to/from the portals-over-GigE streams. The protocol underlying the portals-over-Elan streams is native Quadrics Elan comms, while the protocol underlying the portals-over-GigE streams is TCP. In this Lustre configuration, the netperf client processes are run on the ALC portals router nodes (there are 16 of these) and communicate with netperf server processes running on the Levi Lustre storage server nodes – OSSes (there are 14 of these).



uBGL: The BGL architecture connects 64 compute nodes (CNs) via an IBM-proprietary tree network to each IO node (ION). IONs are invisible to user applications running on the CNs (but essential to their operation). The IONs serve as the Lustre clients, and are *directly* connected to the IP network to which the Lustre storage servers are connected. In this Lustre configuration, there is no need for "portals router nodes". The netperf client processes are run on the uBGL IONs (there are 128 of these) and communicate with netperf server processes running on the Levi Lustre storage server nodes – OSSes (there are 14 of these). The protocol used here is TCP.

Levi: The back-end storage devices used by the Lustre storage servers are Data Direct Networks (DDN) S2A-8500 raid storage units. The "Lustre cluster" used in this milestone is named "Levi." Each Levi OSS node connects via a single 2-gigabit FibreChannel connection to a single port on the S2A-8500. These storage units utilize SATA disk drives. Extensive testing has shown that these S2A-8500-SATA units can sustain, for each FC port, 150 MB/sec for writes, and 110 MB/sec for reads. Since there are 14 Lustre OSSes in this configuration, and each OSS is connected to a single DDN FC port, the system maximum disk I/O bandwidth is calculated as: *14*single_port_rate.*

SLIC: The "slic" cluster is a special purpose Lustre client designed as an interface to the archival (HPSS) environment. In this instance the Slic nodes are *directly* connected to the IP network to which the Lustre storage servers are connected. The protocol used here is TCP.

Demonstration Details

The network bandwidths of the cluster connections, the disk bandwidth of the back-end Lustre storage devices, and the resulting milestone performance goals (derived from the milestone description), are shown in the table below. Units are in MB/sec.

Table of Bandwidths, Requirements, and Targets				
	ALC	uBGL	Slic	Comments
Network b/w				
Network write Mb/s	3,158	3,301	2,003	Measured by netperf
Network read Mb/s	3,030	3,288	2,189	Measured by netperf
Disk b/w				
Disk write b/w	2,100	2,100	2,100	14*150MB/s
Disk read b/w	1,540	1,540	1,540	14*110MB/s
Min (network_bw, disk _bw)				
write	2,100	2,100	2,100	Disk limited
read	1,540	1,540	1,540	Disk limited
Required IOR performance				
write	1,260	1,260	N/A	60%
read	924	924	N/A	60%
Target IOR performance				
write	1,680	1,680	N/A	80%
read	1,232	1,232	N/A	80%

As the table shows, our application-level test, IOR, *cannot hope to achieve performance better than the:*

minimum (network_bandwidth, backend_disk_bandwidth)

The network bandwidths were determined by running a variant of the well-known

"netperf" performance test, from all pairs of network interface end-points, and summing the individual performance reports to an aggregate network bandwidth value. Appendix A gives examples of netperf execution lines.

The IOR test program is LLNL's standard I/O test. IOR has been used for I/O related testing for some time, and it supports many options for selecting the I/O model and the details of the I/O pattern being emulated. For this milestone, we use IOR in the simple "file-per-process" mode, which is the predominant mode used by LLNL application programs. In this mode, the various MPI tasks of the IOR job synchronize only at the beginning and end of their testing. IOR gathers information from each task at the end of the run, and computes the I/O performance values. *Appendix B* shows some details of the IOR command lines and input-scripts used in the milestone testing. Actual logs from this testing are available upon request.

Archival Interface

The Lustre global parallel file system is designed to act as fast *temporary* storage for our supercomputing resources. But our users need a way to transfer data to or from archival storage. In the past this was accomplished by running the archival interface codes on the interactive (or login) nodes associated with the various compute resources. The interactive nodes could access the local file system and were also attached to the high-speed archival network. With the advent of the global (or site wide) filesystem we are able to move this archival interface function to a special purpose cluster designed to perform the archival interface function. Part of this milestone was to demonstrate 1GB/s between the Lustre filesystem and the HPSS based archive.

Milestone Performance Results and Conclusion

The table below shows our IOR performance results and the percentages that those results represent, of the required and target rates:

Table of IOR results		
	ALC	MCR
Required Performance		
write MB/s	1,260	1,260
read MB/s	924	924

Target Performance		
write MB/s	1,680	1,680
read MB/s	1,232	1,232
Measured IOR Performance rates		
write MB/s	1,810	1,671
read MB/s	1,197	1,268
Percent of required performance		
write	143.7%	132.6%
read	129.5%	137.1%
Percent of target performance		
write	107.7%	99.5%
read	97.2%	102.9%

These results show that the two ASC clusters of heterogeneous architecture, sited in different buildings, can share the same Lustre filesystem, and that the performance achieved from both clusters exceeds the required levels, and either exceeds, or very slightly under achieves, the desired target levels.

Archival data rates between the shared Lustre file system and the open HPSS archival system were tested using nine nodes of the “slic” cluster. The test was designed to duplicate a user’s behavior moving data to/from the archival storage system. The tester ran nine concurrent ftp sessions on the nine “slic” nodes and was able to achieve an aggregate data rate of 1.042 gigabytes per second. This data rate satisfies the 1GB/s archival data rate requirement contained in the milestone.

Appendix A – netperf execution command lines

```
Netperf servers on levi:
/usr/bin/nohup /usr/local/netperf/netmonns -p 12866 > /dev/null 2>&1 &

Netperf command line on ubgl: (for ubgl there needs to be some initial
setup
all this stuff was taken care of in a script.  Note that this command
line is run
on each client and we map each client to an interface on the server as
fairly as possible)

/bgl/ion/bin/nohup /netperf/netmonnp -H levi1-eth2 -B -l 600 -f M -P 0
-- -m 1M -M 1M -S 2M -s 2M  >
/home/wartens2/testing/netperf/logs/netperf.20050919161744/ubglio1_128.
log/ubglio1 2>&1 &


Netperf command line on alc 2 clients started:
/usr/bin/nohup /home/wartens2/.bin/i386/netmonnp -H levi4-eth3 -B -l
600 -f M -P 0 -- -m 1M -M 1M -S 2M -s 2M  >
/home/wartens2/testing/netperf/logs/netperf.20050919161744/alc4_19.log/
alc4 2>&1 &

/usr/bin/nohup /home/wartens2/.bin/i386/netmonnp -H levi5-eth2 -B -l
600 -f M -P 0 -- -m 1M -M 1M -S 2M -s 2M  >
/home/wartens2/testing/netperf/logs/netperf.20050919161744/alc4_19.log/
alc4.2 2>&1 &
```

Appendix B – IOR execution command lines and input scripts

```
ior command line on alc:

srun --core=light -t 120 -W 60 -l -O -N140 -n280 -pltest ior -f
/home/wartens2/.bin/ior-scripts/posix.fpp.survey.ior
contents of posix.fpp.survey.ior (note that our good results were when
numtasks == 56):

IOR START
        intraTestBarriers=1
        writeFile=1
        readFile=1
        useExistingTestFile=0
        keepFile=0
        checkWrite=0
        fsync=1
        reorderTasks=1
        quitOnError=1
        transferSize=512k
        blockSize=32m
        intertestdelay=5
        verbose=1


        testFile=/p/gbtest/wartens2/lustre-test/ior/iorData
```

```
        filePerProc=1
        api=POSIX
        numTasks=14
        RUN
        numTasks=28
        RUN
        numTasks=56
        RUN
        numTasks=84
        RUN
        numTasks=112
        RUN
        numTasks=140
        RUN
        numTasks=168
        RUN
        numTasks=196
        RUN
        numTasks=224
        RUN
        numTasks=252
        RUN
        numTasks=280
        RUN
IOR STOP

ior command line on ubgl:

/usr/local/bin/mpirun -cwd /home/wartens2/testing -exe
/home/auselton/bgl/ior -args "-f /home/wartens2/.bin/ior-
scripts/posix.fpp.survey.stripe1.ior"
&> /home/wartens2/testing/64m/ubgl-64m-01.log
contents of posix.fpp.survey.ior:

IOR START
        intraTestBarriers=1
        writeFile=1
        readFile=1
        useExistingTestFile=0
        keepFile=0
        checkWrite=0
        fsync=1
        reorderTasks=1
        quitOnError=1
        transferSize=512k
        blockSize=32m
        intertestdelay=25
        verbose=1

        testFile=/p/gbtest/wartens2/lustre-test/ior-stripe1/iorData
        filePerProc=1
        api=POSIX

        numTasks=1024
        RUN
IOR STOP
```