Cray Linux Environment™ (CLE) 4.0 Software Release Overview Supplement

S–2497–4001

RECORD OF REVISION

S–2497–4001 Published September 2011 Supports the 4.0.UP01 update release of the Cray Linux Environment (CLE) operating system running on Cray XE and Cray XK systems.

S–2497–3103 Published March 2011 Supports the 3.1.UP03 update release of the Cray Linux Environment (CLE) operating system running on Cray XT and Cray XE systems.

S–2497–3102 Published December 2010 Supports the 3.1.UP02 update release of the Cray Linux Environment (CLE) operating system running on Cray XT and Cray XE systems.

S–2497–3101a Published October 2010 Supports the 3.1.UP01 update release of the Cray Linux Environment (CLE) operating system running on Cray XT and Cray XE systems.

S–2497–3101 Published September 2010 Supports the 3.1.UP01 update release of the Cray Linux Environment (CLE) operating system running on Cray XT and Cray XE systems.

Contents

Tables

Software Enhancements [1]

Cray Linux Environment (CLE) 4.0 software update releases provide bug fixes and a limited set of software enhancements or features. This chapter provides an overview of software enhancements that are introduced in each update release.

Software enhancements and features that were introduced with the initial or base CLE 4.0 release are described in *Cray Linux Environment (CLE) Software Release Overview*, which is also provided with the release package.

1.1 Software Enhancements in CLE 4.0.UP01

1.1.1 Cray XK6 Hardware Support

Who will use this feature?

End users, programmers, site analysts, system administrators

What does this feature do?

The Cray XK6 system is a hybrid massively parallel processing system. Each Cray XK6 blade consists of four compute nodes with up to 64 integer cores per blade. Each compute node has an AMD Opteron 6200 Series processor with 16 or 32 GB of memory. Cray XK6 blades are available with *or* without NVIDIA Tesla-based GPGPU (General Purpose Graphics Processing Unit) processors with 6GB of memory.

Cray XK6 blades use the Gemini system interconnect and can be used within Cray XE systems. For optimal use of compute node resources in mixed Cray XE systems with Cray XK6 compute blades, the system administrator can elect to assign Cray XK6 compute nodes to a batch queue, allowing users to make reservations for either scalar-only or accelerator-based compute node pools.

Initially, Cray will provide programming environment support with compilers from NVIDIA that support the CUDA (Compute Unified Device Architecture) programming model. Cray will also provide NVIDIA's CUDA Toolkit that includes some GPU-optimized libraries relevant to scientific computing, profiling tools, and a debugger. Cray will provide support for an Alpha release of Cray Libsci that has some accelerated BLAS and LAPACK routines. In future releases, Cray will release more compiler and language support in addition to libraries optimized for use with accelerators that could provide greater performance when using applications that target accelerators.

How does this feature benefit customers?

Running applications with a Cray XK6 allows for programmers and end users to potentially enhance the performance of their applications when they adapt their code to incorporate the use of the NVIDIA GPUs. Users may also realize some performance improvement from autotuned GPU kernels generated by Cray libraries.

Does this feature provide any performance improvements?

Yes, provided that either the application is ported to use the GPUs or it uses the accelerated routines available in Cray Libsci, there is a possibility of significant performance improvements for certain applications.

1.1.2 AMD Opteron 6200 Processors

Who will use this functionality?

Users and administrators on Cray XE and Cray XK systems.

How can this support help me?

Base support of AMD Opteron 6200 series processors is provided in this update package. Performance of applications will potentially be improved with increased core counts. Furthermore the AMD Opteron 6200 series processor includes the Bulldozer architecture, which provides improved floating point support and additional support for FMA (fused-multiply add) and AVX (Advanced Vector Extensions) instruction sets. The Cray XE6 compute node has two AMD Opteron 6200 Series processors (eight-core, 12-core, and 16-core), each coupled with its own memory and a connection to the Gemini ASIC. Each Cray XE6 compute node is designed to efficiently run up to 32 MPI tasks or a hybrid of MPI and OpenMP parallelism. Cray XK6 compute nodes can run up to 16 MPI tasks per node. For more information, contact your Cray service representative.

1.1.3 Memory Control Groups

How does this feature benefit customers?

Memory control groups can improve the resiliency of the kernel and system services running on compute nodes while also accounting for application memory usage.

What does this feature do?

Memory control groups are a Linux kernel feature that allows an administrator to force compute node applications to execute within memory control groups.

If memory control groups are enabled, ALPS determines how much memory is available prior to application launch. It then creates a memory control group for the application with a memory limit that is slightly less than the amount of available memory on the compute node. CLE tracks the application's memory usage, and if any allocations meet or try to exceed this limit, the allocation fails and the application aborts.

Since non-application processes execute outside of the memory control group and are not bound to this limit, system services should continue to execute normally during these low memory scenarios, resulting in improved resiliency for the kernel and system services.

Where can I learn more?

To configure and use memory control groups see:

- *Managing System Software for Cray XE Systems*
- The `apmgr`(8) and `apinit`(8) man pages

1.1.4 Node Health Checker (NHC) Enhancements

How does this feature benefit customers?

Because NHC now tests the health of each node every time it is booted or rebooted, unreliable nodes are detected and taken off line before Application Level Placement Scheduler (ALPS) places a job on those nodes. This increases system reliability and serviceability by reducing the likelihood of a job failing because it was launched on an unhealthy node.

How can this feature help customers be more productive?

Testing a node's health at boot and reboot prevents losses in productivity caused by ALPS attempting to place jobs on unhealthy nodes.

Does this feature provide any performance improvements?

There should be no change in performance beyond the increased reliability mentioned earlier.

What does this feature do?

Earlier versions of NHC ran only after a job terminated. This made the first job to run on a newly booted compute node vulnerable to failure. Effective with this release, whenever a compute node is booted, a system-level script launches the NHC. If the NHC tests pass, the node is booted. If one or more tests fail, the node remains down (unbooted) and NHC writes its warnings and error messages to the console log.

As part of the health testing, NHC detects the presence of accelerators, also called Graphics Processing Units (GPUs) on the compute node and runs health tests specific to the type of GPU present. If any tests fail, the node remains down and NHC writes the errors to the console log.

System Administrators can modify the NHC configuration file to specify which tests should or should not be run at boot time, however they should be aware that the modified configuration file must be repackaged into the compute node's boot image (i.e., CPIO), if it is to be used at boot/reboot.

The ALPS subsystem will continue to launch NHC when a job terminates.

Customer-visible compatibility issues:

There should be no compatibility issues for systems that do not have GPU accelerators. In the absence of an accelerator on the node the GPU test returns a `pass` value.

Customer-visible changed functionality:

Because compute nodes that do not pass the NHC tests at boot time are not booted it may appear that fewer nodes are available on the system. However, the net effect of boot-time node health checking is to prevent jobs from failing by running on nodes that previously were incorrectly treated as if they were available.

1.1.5 CCM (Cluster Compatibility Mode) Enhancements

1.1.5.1 Cluster Compatibility Mode (CCM) Platform LSF support

How does this feature benefit customers?

Sites that use Platform LSF as a workload management system for their Cray system can now use Cluster Compatibility Mode (CCM).

What does this feature do?

Cluster Compatibility Mode (CCM) allows ISV (independent software vendor) cluster applications to run on Cray's MPP architectures. CCM is tightly coupled to the batch system. The user running an ISV cluster application makes a reservation request with the batch system for a CCM application and then runs the application using `ccmrun`. Initially CCM supported Moab with TORQUE and PBS Professional. It is now modified to work with Platform LSF.

Where can I find more information?

- The procedure for setting up Platform LSF with CCM is in *Managing System Software for Cray XE Systems*.
- The user guide *Workload Management and Application Placement for the Cray Linux Environment* provides some LSF analogues of PBS commands.

The following documentation is provided with Platform LSF software:

- Administering Platform LSF Guide
- Platform LSF Command Reference Guide

Also see: http://www.platform.com for more information.

1.1.5.2 ISV Application Acceleration

Who will use application acceleration?

End users and application developers.

How can application acceleration help me?

When using ISV applications in CCM, application acceleration can potentially improve performance of the program.

What do application acceleration do?

Application acceleration allows third-party MPI-based ISV applications to use the OpenFabrics Enterprise Distribution (OFED) interconnect protocol over the Gemini high-speed network. Previously, CCM only supported applications over TCP/IP protocol, which can inhibit application performance on Cray systems. Application acceleration uses OFED over the Gemini interconnect to leverage the communication advantages found therein.

Customer-visible limitations:

- Only Open MPI is presently supported.
- 32-bit applications are not supported.

Where can I find more information?

The changes visible to the end user are documented in *Workload Management and Application Placement for the Cray Linux Environment*. Typically ISV applications come with their own pre-packaged MPI. However, if system administrators are supplying a site wide implementation, they must follow the instructions in *Managing System Software for Cray XE Systems*.

1.1.6 FUSE (File system in Userspace) Support in CLE

How does this feature benefit customers?

The benefit of this feature is the ability to make use of any available FUSE file system.

How can this feature help customers be more productive?

Any productivity or performance benefits would be provided by and depend on what FUSE file system is made available.

What does this feature do?

FUSE is now supported in the Cray Linux Environment (CLE) for Gemini and future interconnects. Cray only supports FUSE if persistent mount points are created by a privileged (`root`) user with the `/usr/bin/fusermount` utility. This prevents non-root users from mounting file systems that may interfere with other users or leave resources on nodes after their application has exited. Permissions for the `fusermount` utility are set in `/etc/permissions.local`. FUSE mounts are supported on service nodes and compute nodes; however, compute node support requires the DSL environment.

Where can I find more information?

- The FUSE home page at http://fuse.sourceforge.net/ provides an introduction, documentation, FAQ, and example FUSE file systems.

1.2 Software Enhancements in CLE 4.0.UP00

For information about feature content in the initial or base CLE 4.0 release (CLE 4.0.UP00), see *Cray Linux Environment (CLE) Software Release Overview* (S–2425–40).

1.3 Previously Documented Limitations that are Now Resolved

The following features or capabilities were identified as limitations with an earlier CLE 4.0 release and were described in the *CLE 4.0 Limitations* document. These issues are now resolved in the update release indicated.

- CCM support for Platform LSF was previously marked as Deferred Implementation in CLE 4.0.

1.4 Bugs Addressed Since the Last Release

The primary purpose of each CLE 4.0 update package is to provide fixes for the release. The list of customer-filed bug reports that were closed with CLE 4.0 releases is included in the *CLE 4.0 Errata*; this document is provided with the release package.

1.5 Compatibilities and Differences

The *README* document that is included with the release package describes compatibility issues and functionality changes that you should be aware of after you install a CLE 4.0 update release on a Cray system that was running an earlier version of the CLE 4.0 release.

The *README* document also includes additional documentation or changes to the documentation identified after the documentation for this release was packaged.

Support Requirements [2]

2.1 Supported Cray System Hardware Platforms

The CLE 4.0.UP01 update release supports Cray XE6, Cray XE6m, Cray XE5, Cray XE5m, and Cray XK systems.

2.2 Supported Software Upgrade Path

The CLE 4.0.UP01 release supports initial system installations and migration/upgrade installations from CLE 3.1.UP03 and 4.0.UP00.

2.2.1 System Management Workstation (SMW) Requirements

You must be running the SMW 6.0.UP01 release or later before you install the CLE 4.0.UP01 update release package. For additional information, see the SMW *README* document included with the SMW release package.

2.3 Binary Compatibility

The language in the binary compatibility statement in *Cray Linux Environment (CLE) Software Release Overview* remains accurate. Applications targeted for AMD Opteron 6100 series processors will work without modification on AMD Opteron 6200 processors. However, there is no backward compatibility for applications targeted for AMD Opteron 6200 processors. Cray will provide support for systems that have mixed processor types in a future update release package.

2.4 Advance Notice of Change in Default Programming Environment

Note: This change is effective in a future update release package.

In CLE 4.0.UP02 the default programming environment, as configured in the `/etc/*rc.local` files, will change from `PrgEnv-pgi` to `PrgEnv-cray`. For systems without a CCE compiler the default will remain unchanged.

2.5 Additional Software Requirements

2.5.1 Release Level Requirements for Other Cray Software Products

- You must upgrade the Cray Application Developer's Environment (CADE) to the most current version (at time of release, this is 6.01) to ensure compatibility with the CLE 4.0.UP01 release package. For CADE release information, see *Cray Application Developer's Environment Installation Guide* (S–2465).

Support for other Cray software products is provided in the form of updates to the latest released version only. Unless otherwise noted in the associated release documentation, Cray recommends that you continue to upgrade these releases as updates become available.

2.5.2 Third-party Software Requirements

Cray Linux Environment (CLE) Software Release Overview (S–2425–40) includes a section that lists third-party software requirements for the CLE 4.0 release; this information applies to CLE 4.0 update releases with the following exceptions:

- You must upgrade the PGI Compiler to version 10.9 or later to ensure compatibility with the CLE 4.0 release. PGI release information is available from The Portland Group, Inc. at http://www.pgroup.com.
- Updated information regarding supported and certified batch system software release levels is available on the CrayPort website at http://crayport.cray.com. Click on **3rd Party Batch SW** in the menu bar.

Cray recommends that you continue to upgrade these products as new versions become available.

Documentation [3]

3.1 Cray-developed Books Provided with This Release

Table 1 lists the books provided with the CLE 4.0.UP01 release and indicates which books are new or revised with this update release. The most recent version of each book is provided with the release package.

For information about additional documentation resources and accessing documentation, see *Cray Linux Environment (CLE) Software Release Overview* (S–2425–40), which is also provided with the release package.

Table 1. Books Provided with This Release

Book Title	Most Recent Document	Updated
Cray Linux Environment (CLE) Software Release Overview Supplement (this document)	S–2497–4001	Yes
Cray Linux Environment (CLE) Software Release Overview	S–2425–40	No
Installing and Configuring Cray Linux Environment (CLE) Software	S–2444–4001	Yes
Managing System Software for Cray XE Systems	S–2393–4001	Yes
Managing Lustre for the Cray Linux Environment (CLE)	S–0010–4001	Yes
Introduction to Cray Data Virtualization Service	S–0005–3102	No
Writing a Node Health Checker (NHC) Plugin Test	S–0023–40	No
Workload Management and Application Placement for the Cray Linux Environment	S–2496–4001	Yes
Using the GNI and DMAPP APIs	S–2446–3103	No
CrayDoc Installation and Administration Guide	S–2340–411	No
Repurposing Compute Nodes as Service Nodes on Cray XE and Cray XT Systems	S–0029–3101	No
Lustre Operations Manual	S–6540–1815	N/A

3.2 Changes to Man Pages

3.2.1 New Cray Man Pages

3.2.1.1 CLE 4.0.UP01

- `xtfsck`(8): checks file systems or a subset of file systems for system set(s) defined within `/etc/sysset.conf`.

3.2.2 Changed Cray Man Pages

3.2.2.1 CLE 4.0.UP01

- `aprun`(1): added `PGAS_ERROR_FILE` environment variable description.
- `apmgr`(8): added the `mcgroup -M` option for memory control group per-core memory limit.
- `apinit`(8): added the `-M` option to enable memory control group application execution.
- `basil`(7): added several changes to reflect support of BASIL 1.2
- `cnselect`(1): Added `numcores`option to enable users to find compute node resources based on the decimal value.

  Note: The `coremask` option is deprecated and will be removed in a future release.