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

S–2497–4003

Contents

Page

Tables

Software Enhancements [1]

Cray Linux Environment (CLE) 4.0 software update release packages 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.UP03

1.1.1 Application Level Placement Scheduler (ALPS) Flexible Communication Domain

How does this feature benefit customers?

This feature allows cooperating customer applications in different batch sessions to share and access the same memory regions through the Gemini interconnect. System administrators can also enable shared system service memory domains that are accessible to I/O daemons, for example.

How can this feature help customers be more productive?

This allows customers to expand the *protection domain* model of the Gemini interconnect so that multiple applications in different batch sessions can operate on a shared memory region.

What does this feature do?

Since ALPS allocates protection domains pTag/Cookie pairs for each user application, these applications have a private memory domain. This feature allows user applications to share protection domains and all users to share system services resources that previously could not be shared on a protection domain. You must use `apmgr` to create the shared protection domain. System dedicated protection domains are created at ALPS startup after the administrator has modified the ALPS configuration file, `/etc/alps.conf`.

Where can I learn more?

See the `apmgr`, `aprun`, and `apstat` man pages, *Managing System Software for Cray XE and Cray XK Systems*, and *Using the GNI and DMAPP APIs*.

1.1.2 Cray Data Virtualization Service (Cray DVS) Enhancements

1.1.2.1 DVS Read-only Cache Mode

Does this feature provide any performance improvements?

This may provide a performance improvement for files read multiple times on writable mount points.

What does this feature do?

Many-to-one file reads on writable mount points can impact performance of a DVS-projected file system as it requires a DVS server to process potentially many transactions to read the data. A DVS mount option, `ro_cache`, is introduced in this release to allow files opened in read-only mode on writable mount points to be treated as if they were opened on a cached read-only mount point. This prevents DVS servers from processing `read()` requests for previously cached files. If the file receives a concurrent `write()` request, the mode reverts to the non-cached mode for the current and subsequent requests.

Where can I learn more?

See *Introduction to Cray Data Virtualization Service*.

1.1.2.2 DVS Bulk RW

Does this feature provide any performance improvements?

The main improvement in this feature is a performance improvement for clustered file systems that use the `bulk_rw` option.

What does this feature do?

By default, DVS bulk read-write is not enabled. However, if a file system is mounted in `bulk_rw` mode, it may realize a performance improvement with this release.

Customer-visible changed functionality:

This feature is a performance improvement and transparent to the administrator and end-user.

1.1.3 Cluster Compatibility Mode (CCM) Custom Resources

What does this feature do?

This feature allows administrators of workload managed systems (i.e., PBS Professional, Moab and TORQUE) greater control over what is used in CCM jobs. Using resources as opposed to specialized batch queues provides flexibility and administrators can limit the amount of resources dedicated to CCM across all managed batch job queues.

Customer-visible changed functionality:

Instead of submitting to a batch queue users would use a resource option, (e.g., `-lccm=`*n*). Administrators provision custom resources in a resources file and MOM server configuration files.

Where can I learn more?

See *Managing System Software for Cray XE and Cray XK Systems*.

1.1.4 Parallel Command Tool

Note: The parallel command tool, `pcmd` was originally marked in Cray Linux Environment (CLE) 4.0.UP02 as "Deferred Implementation." That limitation no longer exists in CLE 4.0.UP03.

How does this feature benefit customers?

It allows administrators and/or, if the site deems it feasible, other users to securely execute programs in parallel on compute nodes.

Customer-visible limitations:

- Running commands on a compute node while an application is running on that node can cause the application to fail. For example, deleting an application's open file could cause it to fail. Therefore, `pcmd` should be used with caution.
- `pcmd` is root-only by default. It must be installed as `setuid root` in order for non-privileged users to use it. If `pcmd` is enabled for users, they will only be able to execute `pcmd` on the nodes which they currently have a job reservation.

Where can I learn more?

See the `pcmd`(1) man page.

1.2 Software Enhancements in CLE 4.0.UP02

1.2.1 Additional Storage System Supported as Boot RAID

How does this feature benefit customers?

The NetApp, Inc. Engenio Serial-attached SCSI (SAS) Storage System is now supported for use as the Cray system boot RAID as a lower-end alternative to the Fibre Channel boot array currently implemented in the Cray XE and Cray XK lines. Contact your Cray representative for additional information.

1.2.2 Linux Zone Moveable

How does this feature benefit customers?

The Linux zone moveable feature can improve performance of applications that are sensitive to memory fragmentation.

What does this feature do?

Zone moveable is a Linux kernel feature that separates user from kernel memory spaces. With applications that have a sensitivity to huge page fragmentation, performance is possibly degraded by this condition. One-sided PGAS (partitioned global address space) applications are likely to benefit from the inclusion of zone movable in the compute node kernel as they are typically affected by huge page memory fragmentation.

Where can I learn more?

Zone moveable is disabled by default. System administrators can enable zone moveable, see *Managing System Software for Cray XE and Cray XK Systems*.

1.2.3 Cray Data Virtualization Service (Cray DVS) Fairness of Service

How does this feature benefit customers?

The feature addresses a quality of service issue when an I/O intensive application overwhelms a DVS server and starves out other users and jobs. This may improve performance or fairness of performance between users and/or applications.

What does this feature do?

Cray DVS creates a pool of message processing threads for its servers. The Fairness of Service feature introduces a list of queues within this pool—one queue for each client and/or job. The list is processed in a circular fashion. When a message thread is available, it fetches the first queue on the list, moves that queue to the end of the list and processes the first message in the chosen queue. This helps to distribute the workload and potentially helps contending applications perform better.

Where can I learn more?

Fairness of Service is enabled by default. To learn how to disable it, see *Introduction to Cray Data Virtualization Service*.

1.2.4 Application Level Placement Scheduler (ALPS) CPU List Affinity Enhancement

How does this feature benefit customers?

This feature allows customers to have better control over how threaded applications are placed on and within compute nodes.

Does this feature provide any performance improvements?

In certain applications and hardware configurations, this may result in a performance improvement. For example, this may be especially useful in "hybrid" OpenMP and MPI codes.

What does this feature do?

This feature modifies the `aprun -cc` *cpu_list* option so that users may specify multiple *cpu_lists*. Prior to the introduction of this feature, users could only specify a single list, which all processing elements (PEs) and threads shared for CPU affinity. This could lead to unpredictable CPU affinity, especially for "hybrid" OpenMP and MPI codes. With this feature, a *cpu_list* is specified for each PE and its threads. Each PE is separated by a colon (:).

Where can I learn more?

For details on how to use this feature see the `aprun`(1) man page.

1.3 Software Enhancements in CLE 4.0.UP01

1.3.1 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.

1.3.2 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.

1.3.3 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 and Cray XK Systems*
- The `apmgr`(8) and `apinit`(8) man pages

1.3.4 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.

1.3.5 CCM (Cluster Compatibility Mode) Enhancements

1.3.5.1 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 and Cray XK 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.

Customer-visible limitations:

- CCM has been qualified with Platform LSF 8.0 and later releases.
- CCM does not set `LSB_HOSTS` and `LSB_MCPU_HOSTS` with the current 8.0 LSF release.
- LSF on Cray requires the `-n` value to be a node count, not a core count
- LSF 8.0 interactive shell must be invoked with `bsub -n` *nodes* `-I bash`.

1.3.5.2 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 Platform MPI and HP-MPI are 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 and Cray XK Systems*.

1.3.6 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.

1.4 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).

1.5 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.

1.6 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.UP03 update release supports Cray XE6, Cray XE6m, Cray XE6m Series 200 (Cray XE6m-200), Cray XK6m, Cray XK6m Series 200 (Cray XK6m-200), Cray XE5, Cray XE5m, and Cray XK6 systems.

2.2 Supported Software Upgrade Path

The CLE 4.0.UP03 release supports initial system installations and migration/upgrade installations from CLE 3.1.UP03, 4.0.UP02, 4.0.UP01, and 4.0.UP00.

2.2.1 System Management Workstation (SMW) Requirements

You must be running the SMW 6.0.UP03 release or later before you install the CLE 4.0.UP03 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.

Note: AMD has deprecated support for 3DNow! instructions. This means applications that contain 3DNow! instructions will fail with an illegal instruction fault on AMD Opteron 6200 series processors.

2.4 Additional Software Requirements

2.4.1 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 6.08) to ensure compatibility with the CLE 4.0.UP03 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.

2.4.2 Third-party Drivers Provided

Accelerator driver levels supplied in the CLE 4.0.UP03 release update package are:

- NVIDIA kernel driver version: 285.05.36
- NVBIOS version: 70.10.64.00.01
- CUDA SDK version: 4.0.17a

2.4.3 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 packages with the following exceptions:

- You must upgrade the PGI Compiler to version 11.0 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.UP03 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–2497–4003	Yes
Cray Linux Environment (CLE) Software Release Overview	S–2425–40	No
Installing and Configuring Cray Linux Environment (CLE) Software	S–2444–4003	Yes
Managing System Software for Cray XE and Cray XK Systems	S–2393–4003	Yes
Managing Lustre for the Cray Linux Environment (CLE)	S–0010–4002	No
Introduction to Cray Data Virtualization Service	S–0005–4003	Yes
Writing a Node Health Checker (NHC) Plugin Test	S–0023–40	No
Workload Management and Application Placement for the Cray Linux Environment	S–2496–4003	Yes
Using the GNI and DMAPP APIs	S–2446–4003	Yes
CrayDoc Installation and Administration Guide	S–2340–411	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.UP03

- `pcmd`(1): Executes commands on compute nodes in parallel.

3.2.1.2 CLE 4.0.UP02

- `xtgpus2db`(8): Converts a text file to the SDB `gpus` table.
- `xtdb2gpus`(8): Converts the `gpus` table in the Service Database (SDB) to a text file.

3.2.1.3 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.UP03

- `CLEInstall`(8): Removed references to non-XML `xthwinv` file. Only the XML `xthwinv` file is used (`/etc/opt/cray/sdb/attr.xthwinv.xml`) and that file gets generated via `xtbootsys` and updated automatically on the boot node whenever the system boots; added `--Lustreversion` switch, which allows administrators to specify the version of Lustre to be installed; added the `--bootimage-only` switch, which recreates the `shell_bootimage_`*LABEL*`.sh` script, but performs no other installation or upgrade related tasks.
- `CLEinstall.conf`(8): Added the `NHC_pcmd_suid` parameter, which allows administrators to set the `setuid` flag for the `pcmd` command to `root`.
- `aprun`(1): Added the `-p` option to allow a user to launch an application attached to a shared protection domain.
- `apstat`(1): Added the `-P` option which displays the pTag/Cookie pairs for user and system service protection domains.
- `apmgr`(8): Added the `pdomain` option, which allows system users to create shared protection domains for applications.
- `dvs`(5): Added the `ro_cache` option to enable read-only caching for files on writable mount points.

3.2.2.2 CLE 4.0.UP02

- `CLEInstall`(8): Added the `--xthwinvxmlfile` option to specify the hardware inventory XML file to use in place of the output from the `xtwinv` command with the `-x` option.

3.2.2.3 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.