Cray Compiling Environment Release Overview and Installation Guide (S-5212-84)

Contents

Cray Compiling Environment 8.4 Release Overview Introduction

This publication is an overview of the Cray Compiling Environment 8.4 release for the Cray XE, Cray XK, Cray XC, and Cray CS series systems.

This publication does **not** describe hardware, software, installation of related products, or products that Cray does not provide.

Emphasis for the Cray Compiling Environment 8.4 Release

The Cray Compiling Environment (CCE) release provides the following:

- Support for the C++11 language standard.
- Support for the OpenMP 4.0 specification.
- Support for the inline assembly ASM construct for x86 processor targets.
- Support for GNU extensions by default (-h gnu option).
- Performance improvements for Cray XE, Cray XK, Cray XC, and Cray CS series systems.

Cray Compiling Environment 8.4 Release Package Description

The Cray Compiling Environment 8.4 consists of:

- Cray Fortran Compiler, version 8.4
- Cray C and C++ Compiler, version 8.4
- CrayLibs (libraries and utilities), version 8.4
- *Cray Compiling Environment 8.4 Documentation Overview*

All software is installed by means of scripts and RPM Package Manager (RPM) files. For more detail about Cray Compiling Environment 8.4 release package, refer to the *Cray Compiling Environment 8.4 Release Package Overview* on page 10.

Cray Compiling Environment 8.4 Software Enhancements

This topic describes software enhancements provided with the Cray Compiling Environment 8.4 release.

Performance Enhancements

- Performance of nested OpenMP parallel regions, OpenMP task constructs, and OpenMP workshare constructs is improved.

- Performance of the Fortran 2008 `DO CONCURRENT` construct is improved.

- Increased vectorization of high latency instructions results in improved performance.

- Performance is improved for memory sets and copies in loops with small trip counts.

- The C++ runtime library is now inline. This produces 'flatter' code for better optimization.

- Performance of 32 bit sin and cos is improved for the Haswell processor target.

- Compile time improvements for large functions have been made.

C++ Support

This release adds support for the C++11 language standard. To enable C++11 features, use the `-h std=c++11` command line option.

OpenMP Support

This release adds support for the OpenMP 4.0 specification. Features of note include:

- Target directives for accelerator support
- `simd` directives for vectorization
- Affinity control
- Task dependences and task groups
- User-defined reductions
- Cancellation

The OpenMP 4.0 standard is available on the OpenMP website, *http://openmp.org/wp/openmp-specifications/*.

Accelerator Support

This release introduces support for executing OpenMP 4.0 "target" constructs on NVIDIA accelerator targets. This release continues to support OpenACC 2.0 constructs, also for NVIDIA accelerator targets.

This release adds support for CUDA 7.0, and continues support for CUDA 6.5. CUDA 5.5 is no longer supported.

The Cray specific interfaces that have been adopted by OpenACC 2.0 have been removed in this release. See the `intro_openacc(7)` man page for the most current information regarding this implementation of OpenACC.

The OpenACC 2.0 standard is available on the OpenACC website, *http://www.openacc.org*.

PGAS Support

Support for UPC 1.3 atomic memory operations on XC systems has been added in this release. The UPC 1.3 standard is available on the UPC specification website, *http://code.google.com/p/upc-specification*. See the `intro_pgas(7)` man page for more information regarding support for PGAS programming models.

Instruction Set Support

Support has been added for the inline assembly ASM construct for x86 processor targets.

Compiler Command Line Changes

The following command line changes apply to CCE 8.4:

The `-h craylibs_arch_override` forces Craymath to honor the processor architecture specified by the `-h cpu` option. Processor architecture is typically specified by loading one of the targeting modules, e.g., `craype-sandybridge`, but can be overriden at link time by using the `-h cpu` option. If the `CRAYLIBS_ARCH_OVERRIDE` environment variable is specified, it takes precedence over this option.

The `-h std` option determines the C/C++ language standard: either c99, c89, c++03, or c++11. The default is c99/c++03.

Environment Variable Changes

The `CRAYLIBS_ARCH_OVERRIDE` run time environment variable can be used to specify that a lowest-common-denominator math library be used instead of the default library, thus ensuring that identical computations produce identical results regardless of the type of compute node CPU actually used.

Cray Compiling Environment 8.4 Compatibilities and Differences

This topic describes compatibility issues and functionality changes to be aware of when upgrading from earlier releases of this software.

In CCE 8.4, certain gnu features, which were previously disabled, have been enabled by default. The old default for the gnu command line option was -hnognu. The new default is -hgnu. This change was made to facilitate interoperability with codes developed using gcc and g++. This will have minimal effect on most applications, however, there are a few issues to be aware of:

- The order of processing of -D and -U on the command line has changed. Previously, all of the command line defines (-D) were processed first and all of the command line undefines (-U) were processed after. The new behavior is to process the -D and -U parameters in the order in which they appear.

  Consider the program:

  ```
  #include <stdio.h>
  int main()
  {
  #ifdef TEST
   printf ("TEST is defined\n");
  #else
   printf ("TEST is NOT defined\n");
  #endif
  }

  >cc-U TEST -D TEST test.c
  >a.out
  ```

  Old behavior: TEST is NOT defined

  New behavior: TEST is defined

- Trigraphs are not supported. Trigraphs are sequences of three characters used to represent a single character not appearing on some early computers. For example ??= would be used to represent a # character, ??(would represent a [character and so forth. It is unlikely that trigraphs would be found in any but the oldest codes.

- Cray cc now follows the g++ name decoration convention. This is generally benign as g++ generally follows the C++ standard in this regard. There is, however, one exception to be aware of.

  Consider the following:

  ```
  extern "C" {
  typedef void (*fp1)
   ( void );
  }

  typedef void (*fp2)
  ```

```
  ( void );

void all_reduce( fp1 i, fp2 j )
{
}
```

GCC has a long outstanding (present since at least 2.97) bug (*https://gcc.gnu.org/bugzilla/show_bug.cgi?id=2316*) that affects name decoration in that it does not distinguish between function types based on linkage as the C++ standard requires. Thus, GCC, and now Cray CC, considers both types the same, which causes improper name decoration for fp1. For compatibility, CCE duplicates this behavior (including the bad name decoration.

If any of these are a problem, cray cc and CC compilers can be reverted to the previous behavior by specifying -hnognu on the compile line.

Cray Compiling Environment 8.4 Documentation Overview

This topic describes the documentation that supports the Cray Compiling Environment 8.4 release.

Access Product Documentation

With each software release, Cray provides publications and man pages, and in some cases, third-party documentation. These documents are provided in the following ways:

CrayPort   CrayPort is the external Cray website for registered users that offers documentation for each product. CrayPort has portal pages for each product that contains links to all of the documents that are associated to that product. CrayPort enables the quick access and search of Cray books, man pages, and in some cases, third-party documentation. Access CrayPort by using the following URL: *http://crayport.cray.com*.

CrayDoc   CrayDoc is the Cray documentation delivery system. CrayDoc enables the quick access and search of Cray books, man pages, and in some cases, third-party documentation. Access the HTML and PDF documentation via CrayDoc at the following locations:

- The local network location defined by the system administrator
- The CrayDoc public website: *http://docs.cray.com*

Man pages   Man pages are textual help files available from the command line on Cray machines. To access man pages, enter the `man` command followed by the name of the man page. For more information about man pages, see the `man(1)` man page by entering:

```
% man man
```

Third-party documentation   Third-party documentation that is not provided through CrayPort or CrayDoc is included with the third-party product.

Cray Developed Publications Provided with this Release

The publications provided with this release are listed in the table below, which also indicates whether each publication was updated. Publications are provided in HTML and PDF formats.

Table 1. Publications Provided with this Release

Book Title	Number	Updated
Cray Compiling Environment Release Overview and Installation Guide (this document)	S-5212-84	Yes

Book Title	Number	Updated
Cray C and C++ Reference Manual	S-2179-84	Yes
Cray Fortran Reference Manual	S-3901-84	Yes

Additional Documentation Resources

The below table lists additional resources for obtaining documentation not included with this release package.

Table 2. Additional Documentation Resources

Product	Documentation Source
GNU compilers	Documentation for the GNU C and Fortran compilers is available at *http://gcc.gnu.org/onlinedocs/*
`glibc`	`glibc` documentation is available at *http://gcc.gnu.org/onlinedocs*
GLIB	GLIB documentation is available at *http://developer.gnome.org/glib/stable*
RPM	RPM documentation is available at *http://www.rpm.org*

Removed Cray Man Pages

Cray man page removed with this release:

- `numarg(3i)`

Changed Cray Man Pages

Cray man pages changed with this release:

- `craycc(1)`
- `crayCC(1)`
- `crayftn(1)`

Other Related Documents Available

The following publications contain additional information to help set up the Cray Compiling Environment 8.4; they are not provided with this release but are supplied with other products purchased from Cray:

- *Cray Programming Environments Installation Guide (S-2372)*
- *Cray Programming Environment User's Guide (S-2529)*
- *Managing System Software for the Cray Linux Environment (S-2393)*
- *Installing the Cray Programming Environment for Cray CS Systems (S-2800)*

Cray Compiling Environment 8.4 Release Package Overview

Hardware and Software Requirements

The Cray Compiling Environment 8.4 release is supported on the following software releases:

Table 3. Required OS Versions

System	OS
Cray XE	CLE 5.2
Cray XK	CLE 5.2
Cray XC	CLE 5.2 or CLE 6.0
Cray CS	CentOS 6.4/6.5 and RedHat 6.4/6.5

The Cray Compiling Environment 8.4 release requires the following supporting asynchronous software products. Required versions are listed below.

Table 4. Required Asynchronous Products

Product	Cray XE	Cray XK	Cray XC	Cray CS
craype	2.4.0	2.4.0	2.4.0	2.3.0
PMI	5.0.1	5.0.1	5.0.1	N/A
LibSci	13.0.0	13.0.0	13.0.0	13.0.3
FlexNet	11.12.1	11.12.1	11.12.1	11.12.1
gcc	4.8.1	4.8.1	4.8.1	4.8.1

The Cray Compiling Environment 8.4 release requires the following minimum versions of these **optional** products:

Table 5. Optional Asynchronous Products

Product	
HDF5	1.8.13
NETcdf	4.3.2
parallel-NETcdf	1.4.1

Product	
MPT	7.0.0
GA	5.1.0.5
LibSci_acc	3.1.2
TPSL	1.4.1
PETSc	3.4.4.0
Trilinos	11.8.1.0
fftw3	3.3.4.0
fftw2	2.1.5.7
Perftools	6.2.0
Reveal	1.4

Contents of the Release Package

The release package includes:

- Cray Fortran Compiler, version 8.4

- Cray C and C++ Compiler, version 8.4

- CrayLibs (libraries and utilities), version 8.4

- CrayDoc documentation, described in the *Cray Compiling Environment 8.4 Documentation Overview* topic.

Licensing

The Cray Compiling Environment 8.4 is licensed under a software license agreement which is specific to the Cray Compiling Environment software. Upgrades to this product are provided only when a software support agreement for this Cray software is in place.

The software license agreement is enforced by FlexNet license manager software. The table below shows the licensing scope for Cray Compiling Environment 8.4. A new license key is required for initially installing Cray Compiling Environment software and when upgrading from a previous release to a new major release. For information on installing the FlexNet Server Software, see *Install and Manage FlexNet*.

To request new FlexNet license manager keys for Cray Compiling Environment, contact *license_keys@cray.com*.

Table 6. Licensing Scope for Programming Environment Products

License for Product	Host Platform	Concurrent Users
Cray Compiling Environment 8.4	Cray XE, Cray XK, Cray XC, Cray CS series systems	Unlimited or 5

For more information about contractual licensing and pricing, contact the Cray sales representative, or send an e-mail to *crayinfo@cray.com*.

Install Cray Compiling Environment 8.4

Cray Compiling Environment 8.4 is installed on the shared root. `root` permissions are necessary in order to install this software. Cray Compiling Environment 8.4 requires that asynchronous products are installed on the system.

The CCE software release consists of the downloadable `rpm` file:

- `cce-8.4.0-N.x86_64.rpm` or later

Please refer to the most recent version of the *Cray Programming Environments Installation Guide (S-2372)* available at *http://docs.cray.com/*. For installations on Cray Development and Login (CDL) nodes (formerly esLogin node) also refer to *Installing CLE Support Package on a Cray Development and Login (CDL) Node (S-2528)*. For Cray CS systems, please refer to the most recent version of *Installing the Cray Programming Environment for Cray CS Systems (S-2800).*

This release includes *Coarray C++*, a template library that implements the coarray concept in C++. The template library specifications are contained on a set of `*.html` pages that the CCE installation copies to `/opt/cray/cce/version/doc/html/` on the Cray platform; they may be copied to any location which provides HTML web content for the site, or any location that can be accessed by site local web browsers.

Install the License Key

Prerequisites
To activate the software license, insert the FlexNet software license key information provided by Cray into a FlexNet license file on the system. The FlexNet license file contains data that is used to determine whether a licensed software product is allowed to run.

The license file contains the following information:

- FlexNet software license key for the Cray product
- Initial installation instructions
- Update instructions
- License manager utilities
- Technical Support information

Cray recommends the license file be named `/opt/cray/cce/cce.lic`. These instructions assume that the FlexNet license manager is running, that the license file is located in the directory `/opt/cray/cce`, and that the file is named `cce.lic`.

If the FlexNet license manager isn't installed on the network, see *Install and Manage FlexNet*.

1. Log in to the license server as `admin` or `root`.

2. Locate the existing license file, if any.

```
# ls /opt/cray/cce
```

If the directory does not exist,

```
# mkdir -p /opt/cray/cce
```

3. Create the plain text file named `cce.lic` in `/opt/cray/cce`.

    a. Copy the FlexNet license key received from Cray (typically in an e-mail message) to `cce.lic`.

4. Set the file access permissions to 644.

```
# chmod 644 /opt/cray/cce/cce.lic
```

5. Update the FlexNet license server to use the new key. Verify that the license server is running.

```
# lmstat
```

If the server is not running, see *Install and Manage FlexNet*.

Assuming the server is running, reread the license file.

```
# lmreread
```

The license is now ready to use.

Use of the Cray Compiling Environment 8.4

After the Cray Compiling Environment 8.4 `rpm` files are installed and the license is activated, load the `PrgEnv-cray` module to use CCE.

Load the `craype-accel-nvidia`*version* module to set the necessary compiler options and targets to use the accelerator.

Use either `ftn` or `cc` command to compile. The module environment forces dynamic linking.

Because of the multiple compiling environments potentially available on Cray systems, the `ftn(1)`, `cc(1)`, and `CC(1)` man pages provide basic introductions to the compiler environment. For more information about the Cray compiler command-line options, see the `crayftn(1)`, `craycc(1)`, and `crayCC(1)` man pages.

For more detailed information about the Cray compiler options, directives, pragmas, and optimizations, see *Cray Fortran Reference Manual* and *Cray C and C++ Reference Manual*.

The Cray Compiling Environment 8.4 compilation targets are `istanbul`, `mc8`, `mc12`, `interlagos`, `interlagos-cu`, `abudhabi`, `abudhabi-cu`, `ivybridge`, `sandybridge`, and `haswell`.

The targeting modules (`craype-ivybridge`, or `craype-interlagos`, for example) set `target_system`. If the `target_system` is set during compilation of any source file, the same `target_system` must also be specified during linking and loading. For example, if users are compiling code for use on `ivybridge` nodes, they must load the `craype-ivybridge` module and use the same module at link and load time.

Install and Manage FlexNet

The Cray Compiling Environment requires FlexNet license manager.

If FlexNet license manager is not installed on the system, download the following packages provided in the Cray Compiling Environment 8.4 package:

- `cray-flexnet-installation-instructions.txt`
- `cray-flexnet-daemon-11.12.1-1.0000.9037.2.1.ari.x86_64.rpm`
- `cray-flexnet-manager-11.12.1-1.0000.9037.2.1.ari.x86_64.rpm`
- `cray-flexnet-publisher-switch-11.12.1-1.0000.9037.2.1.ari.x86_64.rpm`
- `cray-flexnet-utils-11.12.1-1.0000.9037.2.1.ari.x86_64.rpm`

New licenses for CCE or PerfTools are not required when upgrading the FlexNet Server Software only.

See the FlexNet Publisher License Administration Guide at *https://www.synopsys.com/support/LI/Licensing/Documents/Dockeys/enduser.pdf*.

Follow the instructions in the `cray-flexnet-installation-instructions.txt` file.

License Management Utilities and Files

Use the following commands to administer the license manager software:

`lmdown`	Shut down the licensing daemons
`lmgrd`	Invoke the licensing daemon
`lmhostid`	Display the host ID of a system
`lmremove`	Return a license to the license pool
`lmreread`	Update license daemons with new license data
`lmstat`	Report current status of the license daemon
`lmver`	Display the FlexNet version being used

These commands are arguments passed to `lmutil`. If any of these commands do not exist on the license server, create them by making a symbolic link to `lmutil`. For example:

```
% ln -s lmutil command name
```

Combine License Files

If using three-server redundancy, use separate license files with different TCP/IP ports for the licensed software products from each vendor. (Licensed software products from the same vendor can share license files.)

If using a single server, license files may be combined. To do so, edit all license files that have matching SERVER lines into one file, and delete the extra SERVER lines (the port number does not have to match). Place all DAEMON lines after the SERVER line.

Resolve Problems

Verify the installation first. Then examine the log file:

- Verify that the license file is /opt/cray/cce/cce.lic. If this is not the location or name of the license file, each user must have CRAYLMD_LICENSE_FILE set to the correct path and name.

- Verify that the license file has a FEATURE line for the product that must be licensed.

- Use the ps command and the lmstat utility to verify that the servers specified in the license file are running.

Examine the log file. Look for the following messages:

- Inconsistent encryption code for <*name*>: The information encoded in the encryption code for the specified feature, server, or daemon is inconsistent with the information provided in the license file. Recheck the associated FEATURE, SERVER, or DAEMON line in the license file.

- license daemon: execl failed: The lmgrd path specified on the DAEMON line in the license file is not valid. Verify that the license file contains the correct path.

- Retrying socket bind: Either the TCP port number is already in use by another process, two lmgrd daemons were started with license files that specify the same TCP port, or the port is waiting to timeout after a recently executed lmgrd command. This message typically appears after stopping and then immediately trying to restart lmgrd. In this case wait a few minutes to allow TCP time to relinquish the port.

If this fails to correct the problem, check the SERVER line in the license file. If another process needs the specified port, edit the license file to specify a different port. If no port is specified, the default port is 27000.