

Intel® C++ Compiler 8.1 for Linux*

Getting Started Guide

Contents

1. [Overview](#)
2. [Supported Linux Distributions](#)
3. [System Requirements](#)
4. [Installation Notes](#)
5. [Building “Hello World” Sample Application](#)
6. [Getting Started with Compiler Optimizations](#)
7. [Additional Information](#)

Overview

This document explains how to install the Intel® C++ Compiler for Linux*, build a sample "Hello World" application for IA-32, Intel® Extended Memory 64 Technology (Intel® EM64T) and Intel® Itanium® processor based systems and get started optimizing your applications.

The Intel C++ Compiler for Linux consists of the following:

- Intel C++ Compiler for IA-32 based applications: **icc** and **icpc** compiler drivers for C and C++ applications.
- Intel C++ Compiler for Intel® EM64T-based applications: **icc** and **icpc** compiler drivers for C and C++ applications.
- Intel C++ Compiler for Itanium processor-based applications: **icc** and **icpc** compiler drivers for C and C++ applications.
- Linux Application Debugger 8.1 for IA-32 based applications: **ldb**
- Linux Application Debugger 8.1 for Intel® EM64T-based applications: **ldb**
- Linux Application Debugger 8.1 for Itanium-based applications: **ldb**
- Intel Itanium Assembler 8.0 for Itanium-based applications: **ias**
- The Intel® Compilers code-coverage tool: **codecov**
- The Intel® Compilers test-prioritization tool: **tselect**
- Product documentation

Supported Linux* Distributions

This version of the Intel® compilers for Linux supports Linux distributions with the following combinations of glibc, the C language library, and the Linux kernel.

Supported IA-32-Based Systems

A Linux system with glibc 2.2.5, 2.2.93 or 2.3.2 and the 2.4.x Linux kernel, represented by the following distributions:

- Red Hat* Linux 7.3, 8, 9
- Red Hat Advanced Server 2.1
- Red Hat Enterprise Linux 3.0
- SuSE* Linux* 8.2, 9.1
- SuSE* Enterprise 8 / United Linux* 1.0

Supported Intel® EM64T-Based Systems

One of the following Linux distributions:

- Red Hat Enterprise Linux 3.0
- SuSE Linux 9.1 Professional
- SuSE Linux Enterprise Server 9

Supported Itanium® processor based systems

A Linux system with glibc 2.2.4, 2.2.5 or 2.3.2 and the 2.4.x or 2.6.x Linux kernel, as represented by the following distributions:

- Red Hat Linux 7.2
- Red Hat Advanced Server 2.1
- Red Hat Enterprise Linux 3.0
- SuSE Linux*Professional 9.1
- SuSE Enterprise 8 / United Linux 1.0

Binutils 2.14 is recommend, especially if using shared libraries as there are known issues with binutils 2.11.

System Requirements

IA-32 Processor System Requirements

- A computer based on a Pentium® processor or subsequent IA-32 based processor. A Pentium 4 or Intel® Xeon™ processor is recommended.
- 128 MB of RAM, 256 MB recommended.
- 100 MB of disk space, plus an additional 200 MB during installation for the download and temporary files.
- A supported Linux* distribution, see [Supported Linux Distributions](#) section.

Intel® EM64T Processor System Requirements

- An Intel® EM64T-based computer
- 256 MB of RAM, 512 MB recommended.
- 300 MB of disk space, plus an additional 300 MB during installation for the download and temporary files.
- A supported Linux distribution, see [Supported Linux Distributions](#) section.

Itanium® Processor System Requirements

- A computer with an Itanium® or Itanium 2 processor.

- 256 MB of RAM
- 100 MB of disk space, plus an additional 200 MB during installation for the download and temporary files.
- A supported Linux distribution, see [Supported Linux Distributions](#) section.

Installation Notes

The Intel® compilers use the Globetrotter* FLEXlm* electronic licensing technology. A valid license is required for installation. Following are the steps to install the compiler:

1. The installation requires a valid license before installing any product component. The license must remain on the system to use the Intel® C++ Compiler. If you have downloaded the compiler from Intel® Premier Support website, the license key you received with your 8.0 compiler will work with the 8.1 release provided your support services have not expired.

Here is how to setup the license file before installation.

- If you electronically downloaded the Intel C++ Compiler 8.1 for Linux*, the license will be sent via email with instructions to install the license file.
- If you have a CD version of the Intel C++ Compiler 8.1 for Linux, a valid license is included on the CD and the installation program will locate it automatically. In order to obtain technical support, including product updates, a **CD-ROM user** must do the following:
 - **Register your product:** First, locate the serial number found on the inside flap of the product box. Then, visit the web site <http://www.intel.com/software/products/registrationcenter/index.htm> and follow the instructions. After the registration you will receive an email within 24 hours containing a new license.
 - **Install the new license:** The new license in the email entitles you to one year of support services that allows you to download and execute product updates and obtain full technical support. The email also contains the instructions on how to install the license.

For details about the support service license, please see

<http://www.intel.com/software/products/compilers/clin/pricelist.htm>

2. If you downloaded the compiler, untar the compiler package (`tar -zxvf l_ccxxx.tar.gz`).
3. Login as the root user and execute the install script, "`./install.sh`". Please note that you may not be able to run the 'rpm' command successfully if you are not logged in as the root user. It is possible to install the compilers without root access by unpacking the RPM files with rpm2cpio and editing the compiler environment and configuration files.
4. Select the compiler or debugger to install. The default RPM options, "`-U --replacefiles`", are recommended to update existing files. The default

installation directory is `/opt/intel_cc_80` for the compiler, and `/opt/intel_idb_80` for the debugger.

5. After the installation process, all the packages that are successfully installed are listed. Please confirm that this list is correct and Enter "x" to exit.
6. Execute the appropriate script, for sh or csh type shells, to setup the compiler environment variables:
`source <install-dir>/bin/iccvars.sh (or .csh for csh type shells)`
7. Run the compiler.

Compiler Environment and Configuration Scripts Information

The compilers environment variables are set by the following scripts.

`source <install-dir>/bin/iccvars.sh (or .csh for csh type shells)`

The compiler configuration files for C and C++ are `<install-dir>/bin/icc.cfg` and `<install-dir>/bin/icpc.cfg`, respectively, and contain the default compiler options. The configuration files can be used to add additional system-wide default compilation options. Please note that you will need to save and restore any custom options in the configuration files, when you install a new version of the Intel compiler.

Installation Warning for RPM 4.0.2

RPM 4.0.2 cannot install to a non-default directory. This has been resolved in RPM 4.0.3.

Uninstalling the Compiler

Login as root and run the uninstall script:

- `<install-dir>/bin/uninstall`

Building "Hello World" Sample Application

Building the classic "Hello World" program is described. After successful installation with a valid FLEXlm license, the configuration scripts, `iccvars.sh(csh)`, must be sourced as described above. Then, execute these steps:

1. Create a simple "Hello World" C++ program in a text editor 'hello.C':

```
#include <iostream>
int main() {
    std::cout << "Hello World !" << std::endl;
    return 0; }
```
2. Set up Intel® C++ Compiler environment variables as discussed above:
`source <install-dir>/bin/iccvars.sh`
3. Compile C++ application hello.C with the C++ compiler driver, `icpc`:
 - `icpc hello.C -o hello.`
4. Run the executable: `./hello`, it should display "Hello World!".

Getting Started with Compiler Optimizations

The Intel® C++ Compiler enables programmers to take full advantage of the advanced performance enhancement features of Intel's latest IA-32, Intel® EM64T and Itanium®

processors and includes advanced optimizations. In general, increasing the degree of optimization done by the compiler leads to an increase in compile-time and reduced debugging capability. This section describes an optimization methodology with the Intel C++ Compiler.

To start to optimize, the default optimization "-o2" is recommended. Interprocedural optimization allows the compiler to optimize across different compilation units and can have large performance improvements. Profile guided optimization uses information from running an instrumented executable that allows the compiler to rebuild the application knowing where the majority of the computations are. Processor specific optimizations can also yield large performance improvements, for example, "-axP" on IA-32 to optimize for Pentium® 4 processor with Streaming SIMD Extensions 3 support. Of course, not all optimizations are beneficial for all applications. During the application development process, the "-g -oo" switches could allow faster compile times and full debugging with no optimization.

Please see the paper, “**Optimizing Applications with the Intel C++ and Fortran Compilers**”, available at <http://www.intel.com/software/products/compilers/clin>, for additional details. For complete information on the individual optimizations, please refer to Intel C++ Compiler User's Guide at <install-dir>/docs/c_ug_lnx.pdf. The product release notes contain additional information is located at <install-dir>/docs/C++ReleaseNotes.htm.

Remember to always measure the performance of your application after each optimization added to verify the benefits. The Intel VTune™ Performance Analyzer can be a great help for measuring the performance benefits of each, as well as giving advice on further tuning opportunities, additional information is available at <http://www.intel.com/software/products/vtune/>.

Additional Information

Additional information is available at the compiler product web site at <http://www.intel.com/software/products/compiler>.

Your feedback is very important to us. To receive technical support for the tools provided in this product and technical information including FAQ's and compiler updates, you need to be registered for an Intel® Premier Support account on our secure web site, <https://premier.intel.com/>. You can register for an Intel Premier Support at <http://support.intel.com/support/performance/c/linux>. Note, if you already have access to Intel Premier Support and the “Intel(R) C++ Compiler for Linux” product you do not need to re-register. Compiler support information, including top technical issues and known issues is available at <http://support.intel.com/support/performance/c/linux>.

Copyright and Legal Information

Intel, Itanium, Intel Xeon, Pentium, and VTune are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries

* Other names and brands may be claimed as the property of others
Copyright (C) 2003, 2004 Intel Corporation. All Rights Reserved.