

Intel® Fortran Compiler 7.0 for Linux*

Getting Started Guide

Contents

1. [Overview](#)
2. [System Requirements](#)
3. [Installation Notes](#)
4. [Building "Hello World"](#)
5. [Getting Started with Compiler Optimizations](#)
6. [Additional Information](#)

Overview

This document explains how to install the Intel® Fortran Compiler for Linux*, build "Hello World" for IA-32 and Intel® Itanium® processor based systems and how to get started optimizing your applications with the Intel compilers.

The Intel Fortran Compiler 7.0 for Linux consists of the following:

- Intel Fortran Compiler for IA-32 based applications: `ifc`
- Intel Fortran Compiler for Itanium processor-based applications: `efc`
- Linux Application Debugger for IA-32 based applications: `ldb`
- Linux Application Debugger for Itanium based applications: `ldb`
- Intel Itanium Assembler for Itanium based applications: `ias`
- Product documentation

System Requirements

The system requirements can be found on the product web-site, <http://www.intel.com/software/products/compilers/cwin/> and in the *Intel® C++ Compiler Release Notes* (also available on the product web-site.)

Installation Notes

The Intel® compilers use the GlobeTrotter* FLEXlm* electronic licensing technology. A valid license is needed to use the compilers. To install the compilers:

1. Download the compiler package or insert the product CD-ROM.
2. If you downloaded the compiler, untar the compiler package.
3. Become the root user, needed to install the license and to run `rpm`. Install the FLEXlm license: for electronic download, the license is sent via email with install instructions. For the CD-ROM, read the **README.TXT** file.

4. Execute the install script, "**install**". It's possible to install without root access by unpacking the RPM files with rpm2cpio and editing the compiler environment and configuration files, described below, but this installation method is not supported.
5. Select the compiler or debugger to install. The default RPM options "**-U --replacefiles**" are recommended to update existing files. The recommended installation directory is /opt/intel.
6. After installation, the installed packages are listed. Enter "x" to exit.
7. Execute the appropriate script to setup the compiler environment variables:
 - `source <install-dir>/compiler70/ia32/bin/ifcvars.sh(.csh);`
 - `source <install-dir>/compiler70/ia64/bin/efcvars.sh(.csh).`
8. Run the compiler.

Compiler Environment and Configuration Scripts Information

The compilers use the environment variables and the install script creates script files to correctly set the compiler environment. The names and locations are:

`<install-dir>/compiler70/ia32/bin/ifcvars.sh(.csh)` for IA-32 application, and
`<install-dir>/compiler70/ia64/bin/efcvars.sh(.csh)` for Itanium processor-based applications.

The installation creates configuration files,

`<install-dir>/compiler70/ia32/bin/ifc.cfg` and

`<install-dir>/compiler70/ia64/bin/efc.cfg`. They contain common settings and can be used to add additional default options. Note, if you install a compiler update package, you need to rename the config files to avoid being overwritten.

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

Become the root user, needed to run rpm, and run the uninstall script:

- `<install-dir>/compiler70/ia32/bin/uninstall` on IA-32 based systems.
- `<install-dir>/compiler70/ia64/bin/uninstall` on Itanium based systems.

Building "Hello World"

Building the classic "Hello World" program is described. After successful installation with a valid FLEXlm license, the configuration scripts, `ifcvars.sh(csh)` or `efcvars.sh(csh)`, must be [sourced as described above](#). Execute these steps:

1. Create a simple "Hello World" Fortran program in a text editor and save as 'hello.f'.


```
PROGRAM HELLO_WORLD
      PRINT *, 'Hello World!'
END PROGRAM HELLO_WORLD
```

2. Set up Intel Fortran Compiler environment variables as discussed above:
`ifcvars.sh` for IA-32 applications or `efcvars.sh` for Itanium based applications.
3. Compile `hello.f` :
 - `ifc hello.f -o hello`. Creates IA-32 application.
 - `efc hello.f -o hello`. Creates Itanium-based application
4. Run the executable: `./hello`, it should display "Hello World!".

Getting Started with Compiler Optimizations

The Intel Fortran Compiler enables programmers to take full advantage of the advanced performance enhancement features of Intel's latest IA-32 and Itanium processors and includes advanced optimizations. These optimizations are intended for use in product-release builds of applications, not necessarily for earlier phases of application development cycles. 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 Fortran Compiler.

During application development, the `-g -O0` switches are recommended to allow fast compile times and full debugging with no optimization. To start to optimize, the default optimization `-O2` is recommended. Then, `-O3` enables advanced optimizations. 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. Of course, not all optimizations are beneficial for all applications. -For additional details on optimizing, the paper, “**Optimizing Applications with the Intel C++ and Fortran Compilers**” is available at <http://www.intel.com/software/products/compilers/flin>. For complete information on the individual optimizations, please refer to the Intel Fortran Compiler User's Guide at `<install-dir>/compiler70/docs/f_ug_lnx.pdf`.

Remember to always measure the performance of your application after each optimization added to verify the benefits. The 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

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/performancetools/fortran/v7>. Note, if you already have access to Intel® Premier Support and the “Intel® Fortran Compiler, 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/performancetools/fortran>.

The product release notes contain additional information and are located at <install-dir>/compiler70/docs/FortranReleaseNotes.htm. The Compiler User's Guide is located at <install-dir>/compiler70/docs/f_ug_lnx.pdf

Copyright and Legal Information

Intel, Itanium, 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) 2001, 2002 Intel Corporation. All Rights Reserved.