

Intel® C++ Compiler 9.1 for Linux*

Installation Guide

Contents

- [Introduction](#)
- [System Requirements](#)
- [Obtaining the Compiler and Tools](#)
- [Installing the Compiler and Tools](#)
- [Setting Up the Compiler Environment](#)
- [Uninstalling the Compiler and Tools](#)
- [Obtaining Technical Support](#)
- [Disclaimer and Legal Information](#)

Introduction

This document explains how to install and configure for use the Intel® C++ Compiler 9.1 for Linux* product. Installation is a multi-step process. Please read this document in its entirety before beginning and follow the steps in sequence. For information about the product contents, including new and changed features, please refer to the separate [Release Notes](#).

If you have an older version of the Intel C++ Compiler for Linux installed, you do not need to uninstall it before installing this version. If you choose to uninstall the older version, you may do so before or after installing this version.

System Requirements

Processor Terminology

Intel compilers support three platforms: general combinations of processor and operating system type. This section explains the terms that Intel uses to describe the platforms in its documentation, installation procedures and support site.

IA-32

IA-32 (Intel Architecture, 32-bit) refers to systems based on 32-bit processors supporting at least the Pentium® II instruction set, (for example, Pentium® 4, Pentium® D, Celeron®, Intel® Core™ Duo or Intel® Xeon®), or processors from other manufacturers supporting the same instruction set, running a 32-bit operating system ("Linux x86").

Intel EM64T

Intel® EM64T (Intel® Extended Memory 64 Technology) refers to systems based on IA-32 processors which have 64-bit architectural extensions, running a 64-bit operating system ("Linux x86_64"). Systems based on the AMD* Athlon64* and Opteron* processors running a 64-bit operating system are also supported by Intel compilers for Intel EM64T-based applications.

Intel Itanium®

Refers to systems based on the Intel Itanium® 2 processor running a 64-bit operating system.

Native and Cross-Platform Development

The term "native" refers to building an application that will run on the same platform that it was built on, for example, building on IA-32 to run on IA-32. The term "cross-platform" or "cross-compilation" refers to building an application on a platform type different from the one on which it will be run, for example, building on IA-32 to run on Intel Itanium®. Not all combinations of cross-platform development are supported and some combinations may require installation of optional tools and libraries.

The following list describes the supported combinations of compilation host (system on which you build the application) and application target (system on which the application runs).

IA-32 Host

Supported target: IA-32

Intel® EM64T-based Host

Supported targets: IA-32 and Intel® EM64T

Intel® Itanium®-based Host

Supported target: Intel® Itanium®

Note: Development for a target different from the host may require optional library components to be installed from your Linux Distribution.

Note: Intel® Cluster OpenMP* is a separately licensed feature and has different system requirements from that of the compilers. Please refer to the Intel Cluster OpenMP documentation for further details.

Requirements to develop IA-32 applications

- A system based on an IA-32 processor (minimum 450 MHz Intel Pentium® II processor or greater - Intel Pentium® 4 or Intel® Core™ or Intel® Xeon® processor recommended), or a system based on an Intel processor with Intel EM64T, or a system based on an AMD* Athlon* or AMD Opteron* processor
- 256 MB of RAM (512MB recommended).
- 100 MB of disk space, plus an additional 200 MB during installation for the download and temporary files.
- Linux system with glibc 2.2.4, 2.2.5, 2.2.93, 2.3.2, 2.3.3, 2.3.4, or 2.3.5 and the 2.4.X or 2.6.X Linux kernel as represented by the following distributions. **Note:** Not all distributions listed are validated and not all distributions are listed.
 - Red Hat* Linux 7.3, 8, 9
 - Red Hat Enterprise Linux* 2.1, 3, 4
 - SUSE* LINUX 8.2, 9.1
 - SUSE LINUX Enterprise Server* 8, 9
 - Fedora* Core 4
- Linux Developer tools component installed, including gcc, g++ and related tools.

Requirements to Develop Applications for Systems with Intel EM64T or AMD Opteron Processors

- A system based on an Intel processor with Intel EM64T or based on an AMD Opteron processor
- 256 MB of RAM (512 MB recommended)
- 300 MB free hard disk space, plus an additional 300 MB during installation for download and temporary files.
- 100 MB of hard disk space for the virtual memory paging file. Be sure to use at least the minimum amount of virtual memory recommended for the installed distribution of Linux
- Linux system with glibc 2.2.93, 2.3.2, 2.3.3, 2.3.4 or 2.3.5 and the 2.4.X or 2.6.X Linux kernel as represented by the following Linux distributions, running in 64-bit mode. **Note:** Not all distributions listed are validated and not all distributions are listed.
 - Red Hat* Enterprise Linux 3, 4
 - SUSE* LINUX 9.1 Professional
 - SUSE LINUX Enterprise Server 9
 - Fedora* Core 4
- Linux Developer tools component installed, including gcc 3.3.3, g++ and related tools.
- 32-bit (IA-32) C and C++ runtime libraries: libm.so.6, libpthread.so.0, libc.so.6, libstdc++.so.5 and libgcc_s.so.1

Note: The requirement for the 32-bit (IA-32) libraries is due to the compiler and other tools being 32-bit applications that dynamically link to these libraries. If these libraries are not installed, the following error may be displayed when the compiler is invoked:

```
error while loading shared libraries: libstdc++.so.5:
cannot open shared object file: No such file or directory
```

The error message is confusing as it does not indicate that the IA-32 version of libstdc++.so.5 is required. To avoid this problem, be sure that the 32-bit (IA-32) versions of these libraries are installed. Most, but not all, Linux distributions for Intel EM64T will install these by default. Consult the documentation that came with your Linux distribution for instructions on how to install the 32-bit libraries, typically in packages named libstdc++ and libc. If you still have problems, please contact Intel® Premier Support for further assistance.

Requirements to Develop Itanium-based Applications

- A system based on an Intel® Itanium® 2 processor.
- 512 MB of RAM (1 GB recommended).
- 150 MB of disk space, plus an additional 200 MB during installation for the download and temporary files.
- Linux system with glibc 2.2.4, 2.2.93, 2.3.2, 2.3.3 or 2.3.4 and the 2.4.X or 2.6.X Linux kernel as represented by the following distributions. **Note:** Not all distributions listed are validated and not all distributions are listed.
 - Red Hat Linux 7.2
 - Red Hat Enterprise Linux AS 2.1, 3, 4
 - SUSE LINUX Professional* 9.1
 - SUSE LINUX Enterprise Server 8, 9
 - United Linux* 1.0
- Linux Developer tools component installed, including gcc, g++ and related tools.

We recommend using binutils 2.14 or later, especially if using shared libraries as there are known issues with binutils 2.11.

Notes:

- Compiling very large source files (several thousands of lines) using advanced optimizations such as `-O3`, `-ipo` and `-openmp`, may require substantially larger amounts of RAM.
- The above lists of processor model names are not exhaustive - other processor models correctly supporting the same instruction set as those listed are expected to work. Please contact Intel® Premier Support if you have questions regarding a specific processor model
- Some optimization options have restrictions regarding the processor type on which the application is run. Please see the documentation of these options for more information.

Additional System Requirements for Eclipse*

- Use of the Eclipse* Integrated Development Environment on Red Hat Enterprise Linux AS 2.1 has the following additional requirements:
 - Red Hat AS 2.1 Update 6
 - Mozilla 1.4 Xft or higher or Firefox 1.0
 - For users of the GTK* window system: version 2.2.1 of the GTK+ widget toolkit and associated libraries (GLib, Pango) should be installed.
- On Turbo Linux 10 systems, the BEA JRockit JRE that is provided with the Intel Eclipse support may not function properly, aborting unexpectedly. Another standard JRE can be substituted for use on Turbo Linux 10 systems.
- On SuSE SLES 8 IA-32 (only, not Itanium) systems, the Eclipse Integrated Development Environment support will not function properly if used with the GTK* window system. The Eclipse support requires a later version of GTK, version 2.2.1, than what is installed by default with this operating system. Note that upgrading the system to GTK version 2.2.1 is non-trivial.
- Complete, fully functioning browser support in the Eclipse Integrated Development environment requires the installation of one of these browsers:
 - Mozilla 1.4 Xft or higher or Firefox 1.0

On systems where these browsers are not installed by default or available otherwise, such as on a SGI Propack4 Itanium system, an alternate browser, e.g. Konqueror, can be used in the Eclipse Integrated Development environment. Within Eclipse, set it as the browser to be used by selecting Windows->Preferences->General->Web Browser and entering it as the external Web Browser. Note that such a browser cannot be designated as the internal Web Browser within Eclipse, and thus there will be no support available for internal web browsing with this configuration.

Obtaining the Compiler and Tools

Before installing the compiler and tools, you should check the *File Downloads* section of Intel® Premier Support to see if a newer version or update is available. The version on CD or as listed in your electronic download license letter may not be the most current. In order to

download and install a compiler from Intel® Premier Support, you will first have to register for support as described below in the topic [Obtaining Technical Support](#).

Installing the Compiler and Tools

If you encounter difficulty with the initial installation or registration process, please visit <https://registrationcenter.intel.com/support> to request help from Intel.

The default installation directories, referred to elsewhere in this document as <install-dir> and <idb-install-dir>, are:

- /opt/intel/cc/9.1.xxx (for IA-32 and Intel Itanium)
- /opt/intel/cce/9.1.xxx (for Intel EM64T)
- /opt/intel/idb/9.1.xxx (for IA-32 and Intel Itanium)
- /opt/intel/idbe/9.1.xxx (for Intel EM64T)

Note that the path includes the full version number including update number. This means you can keep multiple versions of the compiler installed on the system.

If you are installing both the Intel C++ Compiler and the Intel Fortran Compiler, they each provide the Intel Debugger. If the update numbers are different between the C++ and Fortran installations, the Intel Debugger update numbers will also differ. If you are installing compilers for both languages which were released at the same time, we recommend that you choose a Custom Install for the second compiler and don't install the Intel Debugger a second time, as the debuggers will be the same.

Perform the following steps to install the compiler.

1. Unpack the compiler package in a directory to which you have write access.
> tar -xvf l_cc_p_9.1.xxx.tar
or
> tar -zxvf l_cc_p_9.1.xxx.tar.gz
2. Run the installation script
Execute the install script in the directory where the tar file was extracted.
> cd l_cc_p_9.1.xxx
> ./install.sh
3. If you are not logged in as root, you will be asked if you want to install as root, install as root using sudo, or to install without root privileges. Installing as root (using sudo if you have that privilege) is recommended, as that will update the system RPM database. Select `root`, `sudo` or `ignore` - the last says to install as not root. Use the not-root install if you want to install to a private area.
4. The install script will display a series of options, allowing you to begin installation or view documentation. Type 1 and press enter to begin the compiler installation.
5. You will then be prompted to enter your Intel C++ Compiler for Linux serial number. The serial number was provided to you when you purchased the product, either in an e-mail from the reseller or on a sticker attached to the CD-ROM package. You may also choose to enter a path to an existing license file. If you choose the license file option, you must give the complete path to the license file, including the file name and type. Make your selection and then follow the prompts.

6. The install script then does some prerequisite checking and displays which Intel software development tools are installed, if any, and then offers a choice of a Typical Install or a Custom Install. Selection of a Typical Install is recommended - type `1` and press enter.
7. Press enter again to display the license agreement. After the license agreement is displayed, you are prompted to accept or reject the license. If you accept the license, type `accept` and press enter. If you reject the license, type `reject` and press enter to end the install.
8. The install will then continue - you may be prompted to accept further license agreements, specify install paths or to press enter to proceed through install steps.
9. At the end of the installation, you will be prompted to register for Intel® Premier Support. Registration gives you full access to Intel® Premier Support for the length of your support term (typically one year for licenses purchased with support), including all updates and new versions. Without registering, you will be unable to install or use product updates. Enter your e-mail address, when prompted, to register. (If you already have an Intel® Premier Support account, enter its registered e-mail address.) You will then receive an e-mail with registration information including an initial password. If you do not wish to register, or if you have already registered, press `x` and press enter to exit.
10. After registration, the install script exits.

Installation Warning for RPM 4.0.2 and RPM 4.1

RPM 4.0.2 cannot install to a non-default directory. This has been resolved in RPM 4.0.3.
RPM 4.1 cannot install to a non-default directory. This has been resolved in RPM 4.11 to 4.2.

Eclipse* Installation Notes

The 9.1 versions of the Intel C++ Compiler for IA-32 and the Intel C++ Compiler for Intel Itanium® optionally install the Eclipse* Integrated Development Environment (IDE) version 3.1.1 with C/C++ Development Tools (CDT) version 3.0.1, and the associated components required to use the Intel C++ Compiler with Eclipse. If you wish to use the Intel C++ Package with your existing version of Eclipse, CDT and/or Java* Runtime Environment (JRE), you can, but it is your responsibility to ensure that you are using compatible versions of Eclipse (version 3.1.1), CDT (version 3.0.1) and a compatible JRE (version 1.4.2 or higher).

If you wish to use the Intel C++ Package with your existing, compatible versions of Eclipse, CDT and/or a JRE, there are two ways to do so:

1. Setup the environment to invoke Eclipse with the Intel Eclipse launcher, `iccec`.
2. Setup the environment to invoke Eclipse directly

Also, you can follow approach 2 to bypass using the Intel Eclipse launcher, `iccec`, to invoke Eclipse even when you are using the Intel provided versions of Eclipse, CDT and/or a JRE.

1. Using the Intel Eclipse Launcher, `iccec`

If you want to use the `<install-dir>/bin/iccec` file, which is the Intel supplied Eclipse launcher used to start Eclipse and setup the development

environment for use with the Intel software development tools, (`<install-dir>` is the location where the Intel C++ Compiler is installed), you will need to do the following steps before invoking your pre-installed Eclipse with the Intel Eclipse launcher, `iccec`:

- Make sure that you have installed both the Intel C++ Compiler for 32-bit applications or the Intel C++ Compiler for Itanium applications and the Plugins/Features for Integration into Eclipse*. When you choose Plugins/Features for Integration into Eclipse*, the Intel C++ Compiler features and plugins for integration into Eclipse* CDT development environment will be installed automatically. After the installation of these plugins and features, in the next screen, you can choose the option Integrate Intel(R) C++ Compiler Version 9.1 into Eclipse* installed at different location. You will be asked to provide the location where Eclipse is installed. Provide the location and exit the installation.
- Set the value of the variable `OTHER_JVM_BINDIR` to the full path of the folder of the java file from the JRE installed on your system. If you are using the bash shell, make sure that you export this environment variable.
- Set the value of the variable `OTHER_ECLIPSE_BIN` to the full path of the eclipse binary in the Eclipse installation folder. If you are using the bash shell, make sure that you export this environment variable. For example, if you have installed Eclipse in `/opt/intel/eclipse`, then `OTHER_ECLIPSE_BIN` should be set to `/opt/intel/eclipse/eclipse`. (Make sure that this file exists.)
- You can then execute the `iccec` script and your chosen versions of Eclipse, CDT and a JRE will be used. If you get an error regarding the loading of libraries, make sure that you have set `LD_LIBRARY_PATH` to include the appropriate folder in which Eclipse is installed.

2. Invoking Eclipse directly

If you want to launch Eclipse directly, without using the Intel Eclipse launcher, `iccec`, you will need to do the following before invoking Eclipse:

- Make sure that you have installed both the Intel C++ Compiler for 32-bit applications or the Intel C++ Compiler for Itanium Applications and the Plugins/Features for Integration into Eclipse*. When you choose Plugins/Features for Integration into Eclipse*, the Intel C++ Compiler features and plugins for integration into Eclipse* CDT development environment will be installed automatically. By choosing default options during installation, the Intel C++ compiler plugins will be integrated with the versions of Eclipse and CDT provided on the compiler kit.

If you would like to integrate the Intel C++ compiler plugins into your own, pre-existing compatible versions of Eclipse and CDT, follow this additional step: After the installation of these plugins and features, in the next screen, you can choose the option Integrate Intel(R) C++ Compiler Version 9.1 into Eclipse* installed at different location. You will be asked to provide the location where Eclipse is installed. Provide the location and exit the installation.

- o Setup the Intel C++ compiler related environment variables by executing the `iccvars.sh` (or `.csh`) script prior to starting Eclipse:

```
source /opt/intel/cc/9.1.xxx/bin/iccvars.sh  
(where 'xxx' is the version number)
```

If you installed the Intel idb debugger integration into Eclipse and would like to use idb within Eclipse, you should setup the Intel idb environment variables, in a similar fashion, by invoking the `idbvars.sh` (or `.csh`) script.

- o Since Eclipse requires a JRE to execute, you must ensure that an appropriate JRE is available to Eclipse prior to its invocation. You can set the `PATH` environment variable to the full path of the folder of the java file from the JRE installed on your system or reference the full path of the java executable from the JRE installed on your system in the `-vm` parameter of the Eclipse command, e.g.:

```
eclipse -vm /JRE folder/bin/java
```

- o Invoke the Eclipse executable directly from the directory where it has been installed. To invoke the Intel provided Eclipse executable directly, for example:

```
/opt/intel/eclipsepackage/v.v/eclipse/eclipse  
(where v.v is the version of Eclipse)
```

Installing on Fedora Core 4 Systems

If the Intel C++ Compiler for Linux is installed on an IA-32 Fedora Core 4 system as a "local" installation, i.e. not installed as root, the installation may fail to properly execute the Eclipse graphical user interfaces to the compiler or debugger, as typically launched via the `iccec` and `idbec` commands. The failure mechanism will typically be displayed as a `JVM Terminated` error. The error condition can also occur if the software is installed from the root account at the system level, but executed by less privileged user accounts.

The cause for this failure is that a more granular level of security has been implemented on Fedora Core 4, but this new security capability can adversely affect access to system resources, such as dynamic libraries. This new *SELinux* security capability may require adjustment by your system administrator in order for the compiler installation to work for regular users.

Note that this issue affects IA32 systems only, as Fedora Core 4 is not available on Itanium systems and the Intel C/C++ Compiler integration into Eclipse is not currently supported on Intel EM64T-based systems.

Installing the Intel License Server

If you have a floating, counted or node-locked license, the license must be installed in conjunction with the FLEXlm* license server for Intel software (Intel License Server), which is available for many popular platforms. The server may be installed on any supported platform

accessible on your local network. The compiler CD contains license servers for several Linux distributions. If you do not have the CD, or need a license server for an additional platform, you can find all available license servers in the *File Downloads* section of your [Intel® Premier Support](#) account - select product Intel SW Dev Tools License Servers to find the server for your platform. Please note that the Intel License Server is not available for all platforms where the compiler is supported.

Installing the Intel License Server on SGI* Altix* Systems Running SGI ProPack* 3

If you will be installing the Intel License Server on an SGI* Altix* system running SGI ProPack* 3, please follow these alternate instructions to ensure correct operation in the partitioned cluster environment:

- Obtain the appropriate license server from Intel® Premier Support File Downloads at <https://premier.intel.com/>. On the File Downloads page, select product Intel SW Dev Tools License Servers. The file to use is `flexlm.Linux.ia64.EL3_SGIAltix.tar.Z`.
- Install the license server following the instructions in the `flexlm_ug.pdf` documentation file in the compiler package.
- To determine the host ID, required to retrieve the license file, log in to the partition from which the license server is to be run, set default (`cd`) to the directory where the Intel License Server is installed (default is `/opt/intel/flexlm`) and perform the following command:

```
./lmhostid
```

- If you do not already have your license, go to the Intel Registration Center and register your product as described in the section above. If you already have a license with an incorrect host ID, submit an issue to [Intel® Premier Support](#), providing the serial number of the license file, the results of running `lmhostid`, an indicator of whether this is a floating license or a node-locked license, and a request that a new license be generated for the new host ID. Be sure to specify that this is for an Altix system by selecting **SGI* Altix* ProPack* 3.0** in the Linux* Operating System dropdown list. The Intel customer support team will generate a new license for you with the corrected host ID.

Setting Up the Compiler Environment

The programs in the Intel C++ Compiler 9.1 for Linux product rely on the environment variables `PATH` and `LD_LIBRARY_PATH`. The installation script (`install.sh`) creates compiler environment script files (`iccvvars.sh/idbvars.sh`) that set these variables. It is strongly recommended that you add those script files into your login script (`.login` file). Once the variables are set in the `.login` file there is no need to run the script files for each session.

`source` the script to setup the compiler environment:

- > source <install-dir>/bin/iccvars.sh(.csh)
to use icc
- > source <install-dir>/bin/idbvars.sh(.csh)
to use idb

The installation program also creates compiler configuration files named <install-dir>/bin/icc.cfg that contain common settings for all compilations. You can edit these files to add additional default options. **Note**, if you install a compiler update package, you need to save the configuration file, if you have modified it, to another filename so that the installation doesn't overwrite your modified file.

If you have not already done so, please register for support after you install this product. See the topic [Obtaining Technical Support](#) below for registration instructions.

Uninstalling the Compiler and Tools

Please follow the steps below to uninstall the Intel Compiler and Debugger.

1. If you installed as root, you will need to log in as root
2. To uninstall the compiler:
 <install-dir>/bin/uninstall.sh
 or if you've installed the compiler to the default directory, use
 /opt/intel/cc/9.1.xxx/bin/uninstall.sh or
 /opt/intel/cce/9.1.xxx/bin/uninstall.sh on Intel EM64T-based systems
3. To uninstall the debugger:
 <idb-install-dir>/bin/uninstall.sh
 or if you've installed the debugger to the default directory, use
 /opt/intel/idb/9.1.xxx/bin/uninstall.sh or
 /opt/intel/idbe/9.1.xxx/bin/uninstall.sh on Intel EM64T-based systems

Obtaining Technical Support

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 product updates, you need to be registered for an Intel® Premier Support account on our secure web site, <https://premier.intel.com>. Please register at <https://registrationcenter.intel.com/>.

- Registering for support varies for release products or pre-release products (alpha, beta, etc) - only released products have support web pages on <http://support.intel.com/>.
- If you are having trouble registering or are unable to access your Intel® Premier Support account, please let Intel know of the problem at <https://registrationcenter.intel.com/support>.

Note: If your distributor provides technical support for this product, please contact them for support rather than Intel.

For information about the Intel C++ Compiler Users Forums, FAQ's, tips and tricks, and other support information, please visit: <http://support.intel.com/support/performance/c/linux/>.
For general support information please visit <http://www.intel.com/software/products/support/>.

Disclaimer and Legal Information

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL® PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER, AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT. Intel products are not intended for use in medical, life saving, life sustaining, critical control or safety systems, or in nuclear facility applications.

Intel may make changes to specifications and product descriptions at any time, without notice.

Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them.

The software described in this document may contain software defects which may cause the product to deviate from published specifications. Current characterized software defects are available on request.

This document as well as the software described in it is furnished under license and may only be used or copied in accordance with the terms of the license. The information in this document is furnished for informational use only, is subject to change without notice, and should not be construed as a commitment by Intel Corporation. Intel Corporation assumes no responsibility or liability for any errors or inaccuracies that may appear in this document or any software that may be provided in association with this document. Except as permitted by such license, no part of this document may be reproduced, stored in a retrieval system, or transmitted in any form or by any means without the express written consent of Intel Corporation.

BunnyPeople, Celeron, Centrino, Centrino logo, Chips, Dialogic, EtherExpress, ETOX, FlashFile, i386, i486, i960, iCOMP, InstantIP, Intel, Intel logo, Intel386, Intel486, Intel740, IntelDX2, IntelDX4, IntelSX2, Intel Core, Intel Inside, Intel Inside logo, Intel NetBurst, Intel NetMerge, Intel NetStructure, Intel SingleDriver, Intel SpeedStep, Intel StrataFlash, Intel Viiv, Intel XScale, IPLink, Itanium, MCS, MMX, MMX logo, Optimizer logo, OverDrive, Paragon, PDCharm, Pentium, Pentium II Xeon, Pentium III Xeon, Performance at Your Command, skool, Sound Mark, The Computer Inside., The Journey Inside, VTune, Xeon, and Xircom 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 © 2006, Intel Corporation.