

Intel® Inspector XE 2011 Update 10 Release Notes for Windows* OS

Installation Guide and Release Notes

Document number: 323608-009US

10 June 2012

Contents:

[Introduction](#)

[What's New](#)

[System Requirements](#)

[Installation Notes](#)

[Issues and Limitations](#)

[Attributions](#)

[Disclaimer and Legal Information](#)

1 Introduction

The Intel® Inspector XE 2011 helps developers identify and resolve memory and threading correctness issues in their unmanaged C, C++ and Fortran programs as well as in the unmanaged portion of mixed managed and unmanaged programs. Additionally the tool identifies threading correctness issues in managed .NET C# programs.

The Intel Inspector XE is a dynamic error checking tool for developing multithreaded applications on Windows* or Linux* operating systems. The Intel Inspector XE maximizes code quality and reliability by quickly detecting memory, threading, and source code security errors during the development cycle. You can also use the Intel Inspector XE to visualize and manage static security analysis results created by Intel(R) Composer XE 2011 in various suite products. The Intel Inspector XE is an easy, comprehensive solution that delivers rapid results by isolating memory and multithreading errors.

The Intel Inspector XE has a standalone graphical user interface (GUI) as well as a command-line interface (CLI). In addition, the Intel Inspector XE also integrates into the Microsoft Visual Studio* 2005 and later versions.

This document provides system requirements, installation instructions, issues and limitations, and legal information.

Use the Getting Started tutorial and reference documentation to learn more about the Intel Inspector XE. For documentation, open the `documentation_inspector_xe.htm` file in the

following directory: `<install-dir>\documentation\<locale>`. For example, if you choose the default installation path, you can find the `documentation_inspector_xe.htm` file in the `C:\Program Files\Intel\Inspector XE 2011\documentation\en\` directory.

If you did not register this product during installation, please do so at the Intel® Software Development Products Registration Center (<https://registrationcenter.intel.com/>). Registration entitles you to free technical support, product updates and upgrades for the duration of the support term.

For Technical Support, Product Updates, User Forums, FAQs, tips and tricks, and other support information, please visit <http://www.intel.com/software/products/support/>. **Note:** If your distributor provides technical support for this product, please contact them for support rather than Intel.

2 What's New

Intel® Inspector XE 2011 Update 10:

- Added stability and performance improvements

Intel® Inspector XE 2011 Update 9:

- C++-based Getting Started Tutorials redesigned to focus on a single task/problem
- New Fortran-based Getting Started Tutorials designed to focus on a single task/problem
- Added stability and performance improvements

Intel® Inspector XE 2011 Update 8:

- Added support for analysis of applications that use Intel® Cilk™ Plus support
- Added support for threading analysis of applications that use Qt* APIs
- Enhanced suppressions management
- Improved support for reported call stacks in .NET* applications
- Improved leak detection during memory analysis at the mi1 level
- Improved support for memory analysis of applications that use Microsoft Windows* OLE Automation
- Added interactive debug support during analysis
- Added stability and performance improvements

Intel® Inspector XE 2011 Update 7:

- Added stability and performance improvements

Intel® Inspector XE 2011 Update 6:

- Adds support for analysis of MPI programs that use the Intel® MPI library
- Simpler, more intuitive memory growth reporting - Use new Set Transaction Start and Set Transaction End buttons during dynamic analysis to detect if a block of memory is allocated but not deallocated within a specific time segment during application execution
- Simpler, more intuitive custom analysis creation - Use the new Copy button to create a new custom analysis type based on the currently selected analysis type, then fine-tune the copy to meet your needs
- More robust project support in the standalone GUI - Use the new Project Navigator pane to see a hierarchical view of your projects and results based on the directory where the opened project resides
- A new Static Security Analysis (SSA) tutorial and accompanying sample code - Use a pair of end-to-end examples to learn how to:
 - Use the Intel® Parallel Studio XE SSA feature to identify and resolve problems
 - Apply that learning to your own applications
- Analysis support for C++/CLI and C# applications
- New C# .NET and Fortran sample code
- Added stability and performance improvements

Intel® Inspector XE 2011 Update 5:

- Enhanced module inclusion/exclusion capability when configuring projects for analysis - for example, you can inspect specific modules and disable inspection of all other modules, or disable inspection of specific modules and inspect all other modules
- Added stability and performance improvements

Intel® Inspector XE 2011 Update 4:

- Added stability and performance improvements

Intel® Inspector XE 2011 Update 3:

- Performance improvement for memory analysis at the mi2 and mi3 levels

- Significant reduction in false positives during memory analysis
- Enhanced exception handling during threading analysis
- Improved diagnostic messages during threading analysis
- Enhanced warnings when cross-thread stack accesses are detected
- Enhanced warnings of false positives if the application under analysis uses the Microsoft* Parallel Patterns Library (PPL) programming model
- Added stability improvements

Intel® Inspector XE 2011 Update 2:

- Improved usability
 - Simpler and more intuitive real-time analysis views and informational messages
 - Enhanced state management and problem filtering
 - New memory gauge to help choose the optimal analysis level
- Faster memory analysis
- Added stability improvements

Intel® Inspector XE 2011 Update 1:

- Enhanced Intel® Cilk™ Plus programming model support

Intel® Inspector XE 2011:

This is the first release of the Intel Inspector XE product which is a successor to the Intel® Thread Checker. The following are some key new features in this next generation product:

- Find memory errors in addition to threading errors
- Find memory and threading errors in source code using the Static Security Analysis feature of Intel® Composer XE and then view, filter and process the results in the Intel Inspector XE when both the products are installed as part of Intel® Parallel Studio XE or Intel® C++ Studio XE
- New dynamic instrumentation implementation to accelerate analysis (only executed code is instrumented)
- New standalone GUI on all supported operating systems

- New grouping of related diagnostics, diagnostic suppression and a powerful results filtering interface
- Supports analysis of managed code (C# .NET)

3 System Requirements

Supported Architectures and Terminology

The Intel Inspector XE supports the following architectures:

- **IA-32 Architecture** refers to systems based on 32-bit processors generally compatible with the Intel® Pentium® processors (such as, Intel® Pentium® 4 processor or Intel® Xeon® processor), or processors from other manufacturers supporting the same instruction set, running a 32-bit operating system.
- **Intel® 64 Architecture** refers to systems based on IA-32 architecture processors that have 64-bit architectural extensions (such as, Intel® Core™2 processor family), running a 64-bit operating system. If the system is running a 32-bit operating system, then IA-32 architecture applies instead. Processors from other manufacturers supporting the same instruction set and running a 64-bit operating system are also supported.

Minimum System Requirements

- A system based on an IA-32 or Intel® 64 architecture processor supporting the Intel® Streaming SIMD Extensions 2 (Intel® SSE2) instructions (Intel® Pentium® 4 processor or later, or compatible non-Intel processor)
 - The Intel Inspector XE requires specific knowledge of assembly-level instructions. Its analysis may not operate correctly if a target executable contains instructions not supported by IA-32 or Intel® 64 architectures. In this case, run the analysis with a target executable that contains only supported instructions. After you finish using the Intel Inspector XE, you can revert to using unsupported instructions in the target executable.
 - For the best experience, a multi-core or multiprocessor system is recommended.
- 2GB RAM
- 4GB free disk space for all tool features and architectures
- Supported Microsoft* operating systems (embedded editions not supported):
 - Microsoft Windows XP* SP2 and SP3
 - Microsoft Windows XP Professional x64 Edition SP1 and SP2
 - Microsoft Windows Vista* SP1 and SP2
 - Microsoft Windows 7* and SP1
 - Microsoft Windows Server* 2008 R2 and R2 SP1

NOTE: In a future major release of this product, support for installation and use on Microsoft Windows Vista* will be removed.

- Supported Microsoft Visual Studio* versions:
 - Microsoft Visual Studio* 2005
 - Microsoft Visual Studio* 2008
 - Microsoft Visual Studio* 2010 and SP1
 NOTE: In a future major release of this product, support for installation and use with Microsoft Visual Studio* 2005 will be removed. Intel recommends that customers migrate to Microsoft Visual Studio* 2010 at their earliest convenience.
- Recommended Intel compilers:
 - Intel® C/C++ Compiler 11 or higher
 - Intel® Fortran Compiler 11 or higher
- Native Application coding requirements
 - Supported programming languages:
 - Fortran
 - C
 - C++
 - Supported threading methodologies:
 - Intel® Threading Building Blocks (Intel® TBB)
 - Win32* Threads on Windows* OS
 - OpenMP* (*via Intel® OpenMP compatibility library*)
 - Intel C/C++ parallel language extensions
 - Intel® Cilk™ Plus
 - Microsoft* PPL (*via Intel TBB*)
- Managed Application coding requirements
 - Supported programming languages:
 - C# (full support for .NET versions between 2.0 to 3.5 inclusive)
 - C# (limited support for .NET 4.0 – see note below).
- Adobe* Reader* 7.0 or later needed to read installed documentation

Note:

The Intel Inspector XE has limited support for .NET 4.0:

- The Task Parallel Library classes and algorithms in the System.Threading.Tasks namespace are not supported.
- The following synchronization classes in the System.Threading namespace (new in .NET 4.0) are not supported: Barrier, CountdownEvent, LazyInitializer, Semaphore, SemaphoreSlim.

4 Installation Notes

The installation of the Intel Inspector XE removes any earlier installed minor version of this product (with the same major version number). Different major versions can co-exist with each other, however, only one version can be integrated with any one particular Microsoft Visual Studio* installation at a given time.

If you are installing the product for the first time, you need the product serial number or a valid license file to activate the product.

The product is a self-extracting executable archive with one IA-32 package that you can install on either a 32-bit or 64-bit system.

To begin the installation, double-click the `<package_name>_setup.exe` file as a user with administrative privileges. Select the full installation option to use default install settings and get the complete set of user interfaces (includes a GUI front-end for using the Intel Inspector XE as well as Microsoft Visual Studio* integration). Activation is required.

Default Installation Folders

The default top-level installation folder for this product is:

```
C:\Program Files\Intel\Inspector XE 2011
```

A different top-level folder name can be chosen during a custom installation.

Activating Your Evaluation Software After Purchasing

Users of evaluation versions of Intel Developer Products have a new tool that allows converting evaluation-licensed products to fully licensed products once the product is purchased and a serial number is obtained. The activation tool is a utility that allows users of evaluation products to enter a valid product serial number to convert the product to fully licensed status.

Click **Start > All Programs > Intel Parallel Studio XE 2011 > Product Activation**, supply a valid product serial number, and click **Activate** to convert your evaluation software to a fully licensed product.

Changing, Updating or Removing the Product

If you want to add or remove components from an installation, open the **Control Panel**, select the **Add or Remove Programs** applet, select **Intel Inspector XE 2011 Update X**, and click **Change/Remove** (or **Uninstall/Change** in the Microsoft Windows 7* operating system).

Silent (Non-interactive) Installation

Silent installation allows you to install, repair or remove the product without requiring your input during the execution of the install. Options given on the command line allow you to activate the product with license file or serial number and customize the installation location.

To install the Intel Inspector XE in silent mode, invoke the setup using the following command:

```
<package_name>_setup.exe -s -a install --eula=accept [{--  
license=<path-to-license-file> | --sn=<serial number>}] --  
output=<report-file>
```

With this command, the `<package_name>_setup.exe` self-extracting archive silently uncompresses to `%ProgramFiles%\Intel\Download\Inspector_XE_2011_setup` (the

default extract location) and executes `setup.exe`, passing all arguments following option `-a`. In turn, the application `setup.exe` installs the product into a default directory using the specified license file or serial number for activation. If no license file or serial number is specified, the existing licenses are used.

The use of `--eula=accept` means you accept all terms and conditions of the End User License Agreement. Silent installation will not continue without this option. If you have a serial number, choose the `--sn` option. If you have a license file, choose the `--license` option. If you omit both of these options, the install will look for an existing license. The `--output` option puts information reported during the install into the name of the file specified.

To get more help on self-extracting archive options, use the following command:

```
<package_name>_setup.exe -help
```

To get more help on `setup.exe` options, use the following command:

```
%ProgramFiles%\Intel\Download\Inspector_XE_2011_setup\setup.exe -  
help
```

Installing Collectors On A Remote System

You can install the Intel Inspector XE collectors on a remote system after installing and activating the full product on one system. This is intended for use on systems where space or license checking is an issue. No product activation is required to install and run the collectors on a remote system, but the analysis results may not be viewed on the remote system. Result folders must be copied or made available through the network to the system where the full product was installed and activated, since the visualization of results and command-line reporting option require an activation check.

To install the Intel Inspector XE collectors on a remote system:

- 1) Run `<package_name>_setup.exe -x` to extract the installer files into the default location: `%ProgramFiles%\Intel\Download\Inspector_XE_2011_setup`. If you need another location, use the command `<package_name>_setup.exe -x -f <location>`. **Note:** You can skip this step if you know the extraction location for the original installation.
- 2) Locate and copy the folder `Installs\cs_he_cli` to the remote machine.
- 3) On a remote machine, run `Installs\cs_he_cli\Inspector_XE.msi` and follow the installer instructions.
- 4) To install collectors silently, use:

```
msexec.exe -i Installs\cs_he_cli\Inspector_XE.msi -qn -log <log-file>
```


IMPORTANT: While no activation is required to install and run collectors on a remote machine, you are still bound by the terms and conditions of the End User License Agreement in the original install and included with the collectors remote install after you install the collectors.

Establishing the Intel Inspector XE Command-Line Environment

To easily access the command-line interface, use a command prompt window with the proper environment settings. You can open such a window from the **Start** menu, using a shortcut under **All Programs > Intel Parallel Studio XE 2011 > Command Prompt**.

Alternatively, you can open a command prompt window of your choosing and use the following command to establish the proper environment:

```
C:\Program Files\Intel\Inspector XE 2011\inspxe-vars.bat
```

Known Installation and Configuration Issues

- In very rare situations, the installer may hang during the phase *Configuring integration with Microsoft* Visual Studio 20xx software...* phase. If the installer hangs for more than 20 minutes, locate process `devenv.exe` and shut down the process. Make sure that you do not have any running instances of Microsoft Visual Studio* software. Reboot the machine and then install the product again.

5 Issues and Limitations

General Issues

- Future generations of the Intel Inspector XE product may not support processing of results generated by the legacy Intel® Thread Checker product. However, such Intel® Thread Checker generated results can still be processed and converted by the Intel® Inspector XE 2011 product (either Standalone or Visual Studio* integrated version of the product).
- Intel does not guarantee that the Intel Inspector XE will detect or report every memory and threading error in an application.
 - Not all logic errors are detectable.
 - Heuristics used to eliminate false positives may hide real issues.
 - Highly correlated events are grouped into a single problem.
- You can use the Intel Inspector XE to analyze applications in debug and release modes. To learn more about options necessary to produce the most accurate, complete results, refer to the following related resources:
 - Memory error analysis: <http://software.intel.com/en-us/articles/compiler-settings-for-memory-error-analysis-in-intel-inspector-xe/>
 - Threading error analysis: <http://software.intel.com/en-us/articles/compiler-settings-for-threading-error-analysis-in-intel-inspector-xe/>
- If no symbols are found for a module in which a problem is detected, the Intel Inspector XE displays the call stack and observation source code of the first location where it can find symbols. If it cannot find any location in the call stack with symbols, it displays the module name and relative virtual address (RVA) for the location.

- Applications that crash when run outside the Intel Inspector XE may crash or hang the Intel Inspector XE runtime analysis engine. For example, a corrupt return address on an application call stack crashes the runtime analysis engine. If a crash occurs, problems detected prior to that time can be viewed, but memory leaks will not be reported.
Recommendation: Review the reported problems – it is likely one of them caused the crash.
- The Intel Inspector XE uses a socket to communicate between the graphical user interface and the runtime analysis engine. Preventing an application from opening a socket prevents an analysis of the application from being started by the graphical user interface. The command-line interface can be used to run an analysis in this case and the results can subsequently be viewed using the graphical interface.
- The Intel Inspector XE may report an incorrect call stack following an interruption of normal call flow, such as when an exception is thrown and caught. While the Intel Inspector XE recognizes and attempts to correct result data when this situation occurs, it is possible for a threading or memory problem to be reported before the call stack is fully corrected.
- If the Intel Inspector XE reports insufficient memory errors while analyzing OpenMP* applications, try setting OMP_NUM_THREADS to limit the number of OpenMP* threads. In most cases, OMP_NUM_THREADS=2 is sufficient.
- If the Intel Inspector XE reports insufficient memory errors during analysis, try analyzing your application in sections by running several analyses and excluding a different set of modules for each run.
Note: The Intel Inspector XE does not detect or report issues in excluded modules.
- To analyze applications using Boost* on a Windows* operating system, you must compile your application and Boost libraries (for example, boost_thread-vc80-mt-gd-1_42.dll etc.) with debugging information enabled (/Zi), optimization disabled (/Od), and the C runtime library dynamically linked in. To make sure you have debugging information enabled and optimization disabled, you can download the Boost source and build the libraries yourself. The .pdb files need to be placed next to the binaries or at the location of pdb paths coded in the binaries. To learn more, see: <http://software.intel.com/en-us/articles/building-boost-for-applications-analyzed-by-intel-inspector-xe/>.
- If a child application is selected for analysis, the Intel Inspector XE analyzes only the first instance of that application, even if multiple instances of that application occur.
- If the Intel Inspector XE is integrated with Visual Studio* software, it is not possible to remove the last module from the include/exclude module list in the Target tab of the Project Properties dialog box. [200217183]
Recommendation: Un-checking “Inherit system environment variables” check box on the same tab will allow removal of the last module from the list.

Memory Error Analysis Issues

- An Intel(R) Cilk(TM) Plus program that does not run to completion when serialized will not run successfully under memory analysis. The Inspector XE memory analysis works with Intel® Cilk™ Plus programs that have well defined semantics. These semantics are defined with respect to the program's serialization (see the following resource for more details: http://software.intel.com/sites/products/documentation/hpc/composerxe/en-us/cpp/mac/cref_cls/common/cilk_serial.htm). [200174735]

- The Intel Inspector XE may report false positives when the analyzed application uses custom memory allocators. Use of `_itt_notify` to annotate your source code can reduce these false positives.
- If the semantics of standard C runtime allocators are changed, the behavior of the Intel Inspector XE is unknown and could lead to abnormal analysis termination. For instance, if the application is using non-standard versions of these allocators where the memory returned by the allocator is initialized when it would normally be uninitialized.
- Memory error analysis of applications that use Qt* APIs may report false positives.
- The Intel Inspector XE ignores an include/exclude module directive that uses a forward slash (“/”) in the module pathname. [200264013]

Threading Error Analysis Issues

- The Intel Inspector XE does not capture the main thread creation site if:
 - The binary is built without debug symbol information.
 - The `.pdb` symbol file is not in the location specified within the `.exe` or `.dll` executable file or in the location containing the `.exe` or `.dll` executable file.
- The Intel Inspector XE does not detect deadlocks or potential deadlocks created with:
 - Some types of locks via Intel C/C++ parallel extension (`__critical`) provided by the Intel® Compiler Professional Edition 11.0.
 - Some types of locks in Intel® TBB (`spin_mutex`, `spin_rw_mutex`)
 - Non-exclusive ownership synchronization objects involved, for example, condition variables, semaphores and events and reader/writer locks.
- The Intel Inspector XE does not detect inter-process data races or deadlocks.
- The Intel Inspector XE may report false positives for analyzed applications using customized synchronization primitives. Use of `_itt_notify` to annotate your source code can reduce these false positives.
- The Intel Inspector XE may report false positives when analyzing applications that use the Microsoft* Concurrency Runtime framework (Parallel Patterns Library, Asynchronous Agents Library, etc.).
- The Intel Inspector XE may report false positives if you have `Microsoft C runtime` statically linked in the application.
Recommendation: Build your application with `Microsoft C Runtime` dynamically linked in.
- To enable correct analysis of Intel® TBB applications, set the following required macros before compiling:
 - **TBB_USE_DEBUG** (which sets **TBB_USE_THREADING_TOOLS**) if you use Intel® TBB debug libraries
 - **TBB_USE_THREADING_TOOLS** if you use Intel® TBB release libraries
- Using multiple versions of the Microsoft .NET* framework in the same application may cause the Intel Inspector XE to hang.

6 Attributions

ICU License - ICU 1.8.1 and later

COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1995-2009 International Business Machines Corporation and others

All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, provided that the above copyright notice(s) and this permission notice appear in all copies of the Software and that both the above copyright notice(s) and this permission notice appear in supporting documentation.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

wxWindows Library

This tool includes wxWindows software which can be downloaded from <http://www.wxwidgets.org/downloads>.

wxWindows Library Licence, Version 3.1

=====

Copyright (C) 1998-2005 Julian Smart, Robert Roebing et al

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

WXWINDOWS LIBRARY LICENCE TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Library General Public Licence as published by the Free Software Foundation; either version 2 of the Licence, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Library General Public Licence for more details.

You should have received a copy of the GNU Library General Public Licence along with this software, usually in a file named COPYING.LIB. If not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA.

EXCEPTION NOTICE

1. As a special exception, the copyright holders of this library give permission for additional uses of the text contained in this release of the library as licenced under the wxWindows Library Licence, applying either version 3.1 of the Licence, or (at your option) any later version of the Licence as published by the copyright holders of version 3.1 of the Licence document.
2. The exception is that you may use, copy, link, modify and distribute under your own terms, binary object code versions of works based on the Library.
3. If you copy code from files distributed under the terms of the GNU General Public Licence or the GNU Library General Public Licence into a copy of this library, as this licence permits, the exception does not apply to the code that you add in this way. To avoid misleading anyone as to the status of such modified files, you must delete this exception notice from such code and/or adjust the licensing conditions notice accordingly.
4. If you write modifications of your own for this library, it is your choice whether to permit this exception to apply to your modifications. If you do not wish that, you must delete the exception notice from such code and/or adjust the licensing conditions notice accordingly

Boost Software License – Version 1.0 – August 17th, 2003

Permission is hereby granted, free of charge, to any person or organization obtaining a copy of the software and accompanying documentation covered by this license (the "Software") to use, reproduce, display, distribute, execute, and transmit the Software, and to prepare derivative works of the Software, and to permit third-parties to whom the Software is furnished to do so, all subject to the following:

The copyright notices in the Software and this entire statement, including the above license grant, this restriction and the following disclaimer, must be included in all copies of the Software, in whole or in part, and all derivative works of the Software, unless such copies or derivative works are solely in the form of machine-executable object code generated by a source language processor.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Libxml2

Except where otherwise noted in the source code (e.g. the files hash.c,list.c and the trio files, which are covered by a similar license but with different Copyright notices) all the files are:

Copyright (C) 1998-2003 Daniel Veillard. All Rights Reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE DANIEL VEILLARD BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHERIN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of Daniel Veillard shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from him.

All trademarks and registered trademarks mentioned herein are the property of their respective owners.

Apache

Apache License - Version 2.0 – January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by

combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and

customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

7 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.

A "Mission Critical Application" is any application in which failure of the Intel Product could result, directly or indirectly, in personal injury or death. SHOULD YOU PURCHASE OR USE

INTEL'S PRODUCTS FOR ANY SUCH MISSION CRITICAL APPLICATION, YOU SHALL INDEMNIFY AND HOLD INTEL AND ITS SUBSIDIARIES, SUBCONTRACTORS AND AFFILIATES, AND THE DIRECTORS, OFFICERS, AND EMPLOYEES OF EACH, HARMLESS AGAINST ALL CLAIMS COSTS, DAMAGES, AND EXPENSES AND REASONABLE ATTORNEYS' FEES ARISING OUT OF, DIRECTLY OR INDIRECTLY, ANY CLAIM OF PRODUCT LIABILITY, PERSONAL INJURY, OR DEATH ARISING IN ANY WAY OUT OF SUCH MISSION CRITICAL APPLICATION, WHETHER OR NOT INTEL OR ITS SUBCONTRACTOR WAS NEGLIGENT IN THE DESIGN, MANUFACTURE, OR WARNING OF THE INTEL PRODUCT OR ANY OF ITS PARTS.

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 information here is subject to change without notice. Do not finalize a design with this information.

The products described in this document may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Contact your local Intel sales office or your distributor to obtain the latest specifications and before placing your product order.

Copies of documents which have an order number and are referenced in this document, or other Intel literature, may be obtained by calling 1-800-548-4725, or go to:
<http://www.intel.com/design/literature.htm>

Intel processor numbers are not a measure of performance. Processor numbers differentiate features within each processor family, not across different processor families. See http://www.intel.com/products/processor_number for details.

BlueMoon, BunnyPeople, Celeron, Celeron Inside, Centrino, Centrino Inside, Cilk, Core Inside, E-GOLD, Flexpipe, i960, Intel, the Intel logo, Intel AppUp, Intel Atom, Intel Atom Inside, Intel Core, Intel Inside, Intel Insider, the Intel Inside logo, Intel NetBurst, Intel NetMerge, Intel NetStructure, Intel SingleDriver, Intel SpeedStep, Intel Sponsors of Tomorrow., the Intel Sponsors of Tomorrow. logo, Intel StrataFlash, Intel vPro, Intel XScale, InTru, the InTru logo, the InTru Inside logo, InTru soundmark, Itanium, Itanium Inside, MCS, MMX, Moblin, Pentium, Pentium Inside, Puma, skool, the skool logo, SMARTi, Sound Mark, Stay With It, The Creators Project, The Journey Inside, Thunderbolt, Ultrabook, vPro Inside, VTune, Xeon, Xeon Inside, X-GOLD, XMM, X-PMU and XPOSYS are trademarks of Intel Corporation in the U.S. and/or other countries.

* Other names and brands may be claimed as the property of others.

Microsoft, Windows, Visual Studio, Visual C++, and the Windows logo are trademarks, or registered trademarks of Microsoft Corporation in the United States and/or other countries.

Java is a registered trademark of Oracle and/or its affiliates.

Copyright © 2010 - 2012, Intel Corporation. All rights reserved.