Intel® Thread Checker 3.1 for Windows* Release Notes

Contents

Overview

Intel(R) Thread Checker is designed to help developers create threaded applications more quickly by detecting thread safety issues in programs that use Win32*, Win64* and OpenMP* threads. Source locations where shared memory or synchronization conflicts could cause non-deterministic results or deadlock are shown, including call stacks for each participating thread.

Product Contents

Intel(R) Thread Checker for Windows* plug-in to the VTune(TM) Performance Analyzer to be installed on Microsoft Windows* systems.

What's New

Improved existing features of the Thread Checker in this release include the following:

- 32-bit and 64-bit support for Microsoft Windows* Vista*.
- Support for Intel(R) VTune Performance Analyzer 9.0 for Windows*.
- Support for the latest Dual-Core Intel(R) Xeon processor series.
- Support for the latest Intel(R) multi-core processors, including
  - Intel(R) Core(TM) 2 Duo and
  - Intel(R) Core(TM) 2 Quad processors
- Performance optimizations by descriptor-based static binary instrumentation which
  - reduces the instrumented image size
  - decreases runtime overhead

System Requirements

Microsoft Windows* systems with Pentium(R) processors

Minimum Hardware Required

- Pentium(R) 4 processor
- 512 MB of RAM
- 300 MB of disk space

Recommended Hardware

- Pentium(R) 4 processor supporting Hyper-Threading Technology or Intel(R) Xeon(R) processor or newer
- 2 GB of RAM

Required Software

- Microsoft Windows* Vista* or Microsoft Windows* XP Professional x32 Edition or Microsoft Windows* Server 2003, or Microsoft Windows* XP Professional x64 Edition or newer
- VTune(TM) Performance Analyzer 8.0 or higher
- Microsoft Internet Explorer* 6.0 or higher
- Microsoft Visual Studio .NET 2003 or higher
- Adobe® Reader®*

Required Software for OpenMP* Analysis or Source Instrumentation

- Intel(R) C++ Compiler for Windows* 8.1, Package ID: w_cc_pc_8.1.023 or higher
- Intel(R) Fortran Compiler for Windows* 8.1, Package ID: w_fc_pc_8.1.023 or higher

  When the Intel(R) compilers are used with compiler source instrumentation (-Qtcheck, -tcheck), Intel(R) compiler version 9.1 or higher must be used for platforms with Intel(R) 64 architecture capability as well as for Itanium processor based platforms.

Known Issues and Limitations

Linux* RDC is no longer part of the Windows product. Linux* RDC may be available for legacy usage. Contact customer support for more information.

While upgrading from earlier versions, please use the license file rather than the serial number.

Data and results from previous versions of Intel(R) Thread Checker can still be displayed in Intel(R) Thread Checker for Windows* 3.1. However, projects from previous versions of Intel(R) Thread Checker can not be modified and rerun within this 3.1 release. A new project must be created to obtain new results with this release.

Pausing of the data collection in Intel(R) Thread Checker suppresses analysis of memory accesses only. While paused, the analysis of threading API calls continues so that memory accesses can be correctly interpreted when the data collection resumes. Consequently, the performance of the data collection is not

significantly improved when the data collection is paused.

This release does not support Unicode file names.

The binary instrumentation technology used by Thread Checker can cause some applications to change behavior or terminate abnormally. In this case, you may be able to complete the analysis after lowering the instrumentation levels of the problematic modules.

Thread Checker does not detect threading errors for threads and synchronization objects shared across processes (executables).

Thread Checker cannot analyze processes that are already running. You must either specify the executable as the application to launch, or specify it as a module of interest and then launch it manually when prompted.

Thread Checker supports analysis of native binaries and does not support intermediate executable representations intended for managed runtime environments.

If an application has data races on a global variable, the definition information of the variable will not be available unless source instrumentation is used.

Thread Checker ignores the Duration setting in any of the Activity configuration dialogs.

After uninstalling Thread Checker, the standard project wizards may not appear in the New Project dialog. In this case, restart the VTune(TM) environment to restore the standard project wizards.

Software that defines functions with names that match system API names such as ReadFile() or CreateThread() may have link failures when built with source instrumentation using /Qtcheck. The software may crash when run with Thread Checker. Software should use function names that do not match system API names to avoid these issues.

On Microsoft* Windows* XP x64 or Microsoft* Windows* Vista* x64 with compiler instrumentation, compiler switch /MD or /MDd should be used. Otherwise, some system APIs including APIs beginning with “_” may not be correctly modeled.

When using source instrumentation with functions that contain inline assembly, invalid diagnostics may be generated if data is collected outside of the VTune(TM) environment. To avoid this, compile without /Qtcheck and run your software in the VTune(TM) environment.

The first time an Intel(R) Thread Checker 1.0 project is opened with Intel(R) Thread Checker 2.0 or higher, you may see a warning dialog which states: "The following were present when this file was created but are not currently present: Intel(R) Thread Checker. There may be unexpected failures if you proceed." If this dialog appears, click **Yes** to proceed as no failures should occur.

Although Get Tuning Advice (F8 or View menu) may appear to be available when viewing Intel(R) Thread Checker results, it should not be used because it may cause the VTune(TM) environment to become unstable.

If you have integrated the VTune(TM) analyzer into Microsoft* Visual Studio* 2005 on systems running Windows* 2003 Server SP1, some of the links in the on-line Help may not work properly and a

warning message may appear. To correct this problem, adjust the Security settings for Microsoft* Internet Explorer as follows:

1. In Microsoft* Internet Explorer, click on Tools, Internet Options.
2. Click the Security tab.
3. Click the Custom Level button.
4. Scroll down to the Miscellaneous section.
5. Scroll down to the "Web sites in less privileged web content zone can navigate into this zone" item and select the Prompt radio button.
6. Click OK.

When opening help topics that contain a Related Topics button link, you may see an Internet Explorer* warning message that reads: "An ActiveX control on this page might be unsafe to interact with other parts of the page. Do you want to allow this interaction?" You can safely click **Yes** to continue. This problem occurs due to registry errors caused by installing a Windows* SP. To avoid seeing the warning, you can reregister the HTML Help ActiveX control. To do this, execute the following two commands:

```
regsvr32 /u %windir%\system32\hhctrl.ocx

regsvr32 %windir%\system32\hhctrl.ocx
```

Microsoft Windows* Analysis

When using Thread Checker from within the Microsoft Visual Studio* environment, the Cancel and Stop commands will not shut down the process under analysis as expected.

When using Thread Checker from within the Microsoft Visual Studio* environment, some output messages may not be generated.

If you use Microsoft* Visual Studio* 2005 or Intel® C++ Compiler for Windows* 9.1 or Intel® Visual Fortran Compiler for Windows* 9.1 integrated into Microsoft* Visual Studio* 2005 you should not remove the manifest file as it contains important information that is required for any application which is built using the VS Studio 2005 infrastructure. If you remove the manifest file then your application will not run and you will not get Intel® Thread Checker results.

Microsoft DirectX DirectShow* SDK: When using versions prior to DirectShow* 9.0, you may need to manually raise the instrumentation level for the QUARTZ.DLL from **Module Imports** to **API Imports**. To do this, right-click on your Activity and select **Modify Collectors**. Click **OK** when prompted to modify the selected Activity. From the Configure Intel(R) Thread Checker dialog, click the **Instrumentation** tab. Locate the QUARTZ.DLL module from the list of module names. Change the instrumentation level by clicking on the instrumentation level drop down combo box for this DLL and then select API Imports.

If you build and link your application with /Qtcheck /fixed:NO and receive an error like the following:

```
  myfile.obj : error LNK2017: 'ADDR32' relocation to 'main' invalid
without /LARGEADDRESSAWARE:NO

   LINK : fatal error
```

Then please add the following option to the link command: **/LARGEADDRESSAWARE:NO**

Technical Support

The product support web site (http://support.intel.com/support/performancetools/threadchecker/) contains frequently asked questions, product documentation, product errata, as well as solutions to common issues.

To receive technical support for this product or product updates, you need to register for an Intel(R) Premier Support account at the Intel(R) Registration Center (http://www.intel.com/software/products/registrationcenter/).

When submitting an issue to Intel(R) Premier Support (https://premier.intel.com/), be sure to select **Intel(R) Thread Checker** from the **Product Name** drop down list.

When submitting an issue please provide the product build number. This information can be found in the **ThreadCheckerSupport.txt** file. To open this file, go to Start, Programs, Intel(R) Software Development Tools, Intel(R) Thread Checker, View Support and Build Ids.

Once you have contacted us with your suggestion or problem using your Premier Support account, a technical support engineer will respond within one Intel business day.

If you have not received or have lost your Premier Support login ID or password, or are having trouble with access, please visit https://registrationcenter.intel.com/support for assistance.

Related Products and Services

Information about Intel(R) Software Development Products is available at http://www.intel.com/software/products.

Some of the related products include:

- Intel(R) Thread Profiler is a performance tuning tool for parallel programs that use Win32*, Win64*, POSIX*, OpenMP* or custom synchronization.
- VTune(TM) Performance Analyzer enables you to evaluate how your application is utilizing the CPU and helps you determine if there are modifications you can make to improve your application's performance.
- Intel(R) Compilers are an important part of making software run at top speeds with full support for the latest Pentium(R) and Itanium(R) processors.
- Intel(R) Cluster Tools can help developers create, analyze and optimize high-performance applications on clusters of Intel(R) processor-based systems.
- Intel(R) Performance Library Suite provides a set of routines optimized for various Intel processors.
- Intel(R) Software College provides training for developers on leading-edge software development technologies. Training consists of online and instructor-led courses covering all Intel architectures, platforms, tools, and technologies.

Disclaimer and Legal Information