

*Intel(R) VTune(TM) Performance Analyzer 9.0 Update 1 for Linux**

Release Notes

Contents

[Overview](#)

[What's New and Core Features](#)

[Package Contents](#)

[System Requirements](#)

[Installation](#)

[Usage Notes](#)

[Using per CPU buffering in an Activity Collecting Sampling data](#)

[Known Limitations](#)

[Technical Support](#)

[Documentation](#)

[Additional Information](#)

[Disclaimer and Legal Information](#)

Overview

The VTune(TM) Performance Analyzer 9.0 for Linux* product provides graphical user interface and command-line capabilities that assist in finding performance bottlenecks and hotspots in application code. It enables you to collect, analyze, and display performance data for Linux* applications running on IA-32, Intel(R) 64 or Itanium (R)-based systems.

The VTune analyzer in the integrated Eclipse* platform environment provides these data collection functionalities:

- **Sampling:** Basic, non-intrusive, system-wide profiling capability with support for systems with IA-32, Intel(R) 64 and Intel(R) Itanium(R) processors, including multiprocessor systems.
- **Call graph:** Intrusive profiling capability for capturing the flow of the application and the parent-child relationships with support for systems with IA-32, Intel(R) 64 or Intel(R) Itanium(R) processors.

Additional information on the Intel Software Development Products is available at <http://www.intel.com/software/products/> .

What's New and Core Features in Version 9.0 for Linux

The following are the main new features and core features in the VTune(TM) Analyzer 9.0 for Linux*:

Feature	Benefit
New for 9.0!	
Supports latest Intel Processors	Supports the Dual-Core Intel(R) Xeon(R) processor 5100 series, Dual-Core Intel® Xeon® Processor 30xx series, Quad-Core Intel® Xeon® processor 32xx series, Intel(R) Core(TM) 2 Duo processor, Intel(R) Core(TM) Duo, Intel(R) Core(TM) Solo and Dual-Core Intel(R) Itanium(R) 2 processor 9000 sequence processors.
New events for tuning multi-core Intel(R) Core(TM) Duo processors	New events measure parallelism, core sharing of the bus & cache and modified data sharing by threads. These identify opportunities to improve threading, tune multi-core sharing of the bus & cache and optimize cache-line usage.
Sifts the Intel compiler optimization report to find the information you need	The Intel compiler optimization report contains a wealth of useful information. The problem is that the 3 lines that you care about are buried in a large multi-megabyte file and hard to find. After you locate your hotspot using the VTune analyzer, just select those lines of code and press the optimization report button. The VTune analyzer filters the report and shows you only the portions of the report that apply to the code you selected. Now you can see what the compiler did and easily choose pragmas that will dramatically improve performance.
Installation improvements	Improvements have been made to the installation to make registration and installation easier.
Instruction Filtered Events on Intel(R) Itanium(R) architecture	Instruction filtering allows you to collect events only when they occur with a specified opcode. This helps isolate events only when they occur at critical times.
Branch and Call Navigation made easy	Instantly follow a branch in disassembly by clicking a menu. No more hunting for the destination, just choose "Go to target" to scroll the display.
Faster Call Graph	Now you can selectively profile Java* or native code to improve runtime performance. By gathering data only on the modules being tuned, overhead is reduced and runtime is improved.
New Operating System support including SuSE* Linux* Enterprise Server (SLES) 10.0, SuSE* Linux* 10, and Red Hat* Fedora* Core 5	For a complete list, see: Supported Distributions
Now native on Intel(R) Itanium(R) Architecture!	

Finding your bottleneck is easier than ever.	Get a list of the top 5 time consuming functions with just one simple dialog box. One click on a function name displays the source and shows what's taking all the time. This makes it fast and easy to find your performance bottleneck. VTune analyzer is integrated with the most popular development environment on Linux - Eclipse. The new Eclipse 3.1 includes context sensitive hint panes to help the user become effective quickly with important tips and explanations.
Event ratios	Often a ratio like "clockticks / instruction retired" is a better metric than just a single event. You can use our pre-defined ratios or create your own.
Multi-Row column labels and improved display layouts	Better column labels and improved display layouts make finding the right information faster and easier.
A new "Getting Started" document	Enables users become productive quickly, with a short "hands on" tour of VTune Analyzer's three major analysis flows.
Core Product Features	
Multi-user Call Graph and Large System Support	Large system support is even better with the addition of Multi-user Call Graph and Sampling buffering per CPU. Multi-user Call Graph allows several users to share a large resource for concurrent performance testing. Buffering per CPU ensures that Sampling is accurate on large HPC systems.
Read only file system support	Performance tuning in a secure environment just became a whole lot easier. Call Graph can now analyze secure systems with read only file systems.
Large and small applications welcome	VTune analyzer is a robust solution even with large executables (100MB+). If you have a large design with hundreds of thousands of functions, bring it to VTune analyzer.
No recompile required	VTune analyzer works with your existing binary. Unlike traditional instrumented profilers that make you recompile or modify your build script for profiling, you just run VTune analyzer with your normal production build.
System-wide event based sampling	Accurately identifies where the program is spending its time with negligible overhead (typically less than 5%).
Call graph	Determines calling sequences and finds the critical path, but has a higher overhead.
Both sampling & call graph.	Unlike other offerings, VTune analyzer provides both sampling & call graph analysis tools. Even if you plan to do mostly call graph analysis, running sampling first lets you identify the modules that need it so you only pay the overhead for the modules that need to be analyzed. This can be vital on large projects. Sampling is great for analysis of "loopy" code. Call Graph is usually better for "branchy" code. You need both to get the job done right.
Op-code matching	Sometimes just choosing an event is not selective enough, because the event can occur both at critical and non-critical times. On Intel(R) Itanium(R) architecture, op-code matching allows you to collect events only when they occur with a specified op-code. This lets you to isolate problems like poor pre-fetch and poor memory alignment

Non-uniform memory architecture (NUMA) support	Sampling data is now stored in local CPU memory to minimize system bus traffic. This is critical to avoid saturating the system bus and slowing the system under test.
Enhanced CPU masking	For systems with a large number of processors you usually only want to collect data from a few of the CPUs. CPU masking lets you control exactly where data is collected, from all processors, only those in your allocation or only the processors you specify. This greatly reduces the amount of data you need to collect.
Open your source editor from Source View	You've found your performance problem in Source View and are ready to edit the source. Instead of browsing for the file and hunting for the line you need, a single menu click opens the file for edit and scrolls to the selected line
Event aliasing	When you create a custom event, it is often difficult to remember exactly what you did. Event aliasing let's you create a custom label that is meaningful to you.
Tune Inline Functions	Tune your inlined code with instance-specific event counts on the source and assembly views.
Hotspot Navigation	Find the hottest spot, or jump to the next hottest line. Just select the event you want to navigate by clicking in its column, and then click the Min, Max, Next and Previous icons to quickly browse through your hot spots.
Faster Source and Assembly Views	Source view and assembly views open and scroll much faster – over 10X faster on many large files.

See the [System Requirements](#) section below for a full list of supported Linux* distributions and Linux* JVMs.

Package Contents

The VTune analyzer for Linux* contains the following components:

- Intel Eclipse* Package for supported Linux* installations.
- The VTune analyzer for Linux* graphical user interface (GUI) integrated in the Eclipse platform environment.
- VTune analyzer for Linux* command line interface, available for both IA-32, Intel(R) 64 and Intel(R) Itanium (R) architecture systems. The command line interface includes: vtl, standalone viewers, and the ActivityController that enables separate control of the data collection.
- EntireX DCOM for Linux*.
- The VTune(TM) Performance Analyzer Driver Kit.
- Kernel modules for the supported distributions (see the [full list](#) below).
- VTune(TM) Performance Analyzer Remote Agent.
- Stand-alone viewers that allow a user to view results for IA-32, Intel(R) 64 and Intel(R) Itanium(R) architecture systems.

- Product documentation including: VTune analyzer Documentation Index, Installation Guide, Administrator's Guide, Readme, man pages, User's Guide, Eclipse Help for the VTune analyzer, HTML help for VTune analyzer views, Getting Started web-based tutorial, and these Release Notes.

System Requirements

This section details the processor, memory and operating system requirements for installing the various components of the VTune analyzer.

Processor Requirements for Installing the VTune(TM) performance Analyzer

The following table lists the processor and operating system requirements for installing the various components of the VTune analyzer. The table columns are the following:

- **Processor** column lists the supported processors.
- **VTune analyzer with Eclipse*** column indicates the processors supporting the VTune analyzer integrated in the Eclipse platform environment.
- **vtl** column indicates the processors supporting the VTune analyzer command line interface.
- **RDC** column indicates the processors supporting the VTune analyzer remote data collector (RDC).

Processor	VTune analyzer with Eclipse* (vtlec)	VTune analyzer Command Line Interface (vtl)	RDC (vtserver)
Mobile			
Mobile Intel(R) Pentium(R) 4 processor - M	+	+	+
Intel(R) Pentium(R) M processor	+	+	+
Intel(R) Celeron(R) M processor	+	+	+
Intel(R) Celeron(R) D processor	+	+	+
Intel(R) Celeron(R) processor	+	+	+
Desktop			
Intel(R) Core(TM) 2 Quad processor	+	+	+
Intel(R) Core(TM) 2 Extreme processor	+	+	+
Intel(R) Core(TM) 2 Duo processor	+	+	+
Intel(R) Core(TM) Duo processor	+	+	+

Intel(R) Core(TM) Solo processor	+	+	+
Intel(R) Pentium(R) Processor Extreme Edition	+	+	+
Intel(R) Pentium(R) 4 processor Extreme Edition	+	+	+
Intel(R) Pentium(R) 4 processor	+	+	+
Intel® Pentium® 4 processor supporting Hyper-Threading Technology	+	+	+
Intel(R) Pentium(R) 4 processor with Streaming SIMD Extensions 3 (SSE3)	+	+	+
Server			
Quad-Core Intel® Xeon® processor 5300 sequence	+	+	+
Dual-Core Intel(R) Xeon(R) processor 7100 series	+	+	+
Dual-Core Intel(R) Xeon(R) processor 5300 series	+	+	+
Dual-Core Intel(R) Xeon(R) processor 5100 series	+	+	+
Dual-Core Intel® Xeon® Processor 30xx series	+	+	+
Intel(R) Xeon(R) processor MP	+	+	+
Quad-Core Intel® Xeon® processor 32xx series	+	+	+
Intel(R) Xeon(R) processor	+	+	+
Intel(R) Xeon(R) processor with Intel(R) Extended Memory 64 Technology (Intel (R) 64)	+	+	+
Intel(R) Pentium(R) D Processor	+	+	+
Intel(R) Pentium(R) 4 processor	+	+	+
Dual-Core Intel(R) Itanium(R) 2 processor 9000 sequence	+	+	+
Low Voltage Intel(R) Itanium(R) 2 Processor	+	+	+
Intel(R) Itanium(R) 2 processor	+	+	+

Memory and Disk Space Requirements

This section details the memory and disk space requirements for using the VTune analyzer. Note that memory and disk space requirements for the application you are tuning may be larger than the VTune

analyzer requirements. In this case you need to have sufficient memory and disk space for running your application in addition to the VTune analyzer application and the data collection.

When collecting data on your application with the call graph collector, your application requires more memory than for regular execution.

Memory Requirements

The following table details the memory requirements for running the VTune analyzer application:

VTune(TM) Performance Analyzer interface	RAM	Swap space
command line	256 MB	256 MB
integrated Eclipse platform environment	512 MB	512 MB

Disk Space Requirements

The following table details the disk space requirements for running the VTune analyzer:

Component	Disk Space for IA-32 Systems	Disk Space for Intel(R) 64 Systems	Disk Space for Intel(R) Itanium(R) Architecture Systems
Total (tar file, its extracted files, and all installed components)	902 MB of total available disk space	918 MB of total available disk space	617 MB of total available disk space
<i>/opt/sag</i> directory (created during install) for the included EntireX DCOM for Linux software.	41.3 MB	45.4 MB	57.2 MB
Eclipse* and JRockit* components (<i>/opt/intel/eclipsepackage/3.2.1/eclipse</i> and <i>/opt/intel/eclipsepackage/3.2.1/jrockit-jre1.5.0_04</i>)	117.8 MB	117.8 MB	142.1 MB

Operating System Requirements

This section explains the operating system requirements for running the VTune analyzer for Linux.

Operating System Requirements for Linux Command-line and Graphical User Interface

NOTE:

If you are not using a default kernel on the supported Red Hat* and SuSE* distributions listed below, use the VTune(TM) Performance Analyzer Driver Kit to compile drivers for your kernel. The VTune analyzer Driver Kit software is included with this VTune analyzer product, and can also be obtained via Intel(R) Premier Support. See [Technical Support](#) section below for more information on the Intel(R) Premier Support.

Linux* Distributions Supported

The VTune analyzer for Linux has been tested for operation on the following Linux distributions. If support is only enabled via Remote Data Collection, RDC is listed in the table entry. The kernel versions are supported for Linux systems with supported processors for both uniprocessor systems (UP) and multiprocessor systems (SMP), unless otherwise noted:

The VTune analyzer for Linux has been tested for operation on the following Linux distributions. If support is only enabled via Remote Data Collection, RDC is listed in the table entry. The kernel versions are supported for Linux systems with supported processors for both uniprocessor systems (UP) and multiprocessor systems (SMP), unless otherwise noted.

Operating System	Kernel Version	IA-32	Intel(R) 64	Itanium(R)-based systems
Red Hat* Fedora* Core 5	2.6.15-1.2054_FC5	+	+	
Red Hat* Fedora* Core 6	2.6.18-1.2798.fc6	+	+	
Red Hat* Enterprise Linux* 3.0 Update 8	2.4.21-47.EL	+	+	+
Red Hat* Enterprise Linux* 4.0 Update 4	2.6.9-42.EL	+	+	+
Red Hat* Enterprise Linux* 5	2.6.18-8.el5	+	+	+
SuSE* Linux* Enterprise Server (SLES) 9.0 Service Pack 3	2.6.5-7.244	+	+	+
SuSE* Linux* Enterprise Server (SLES) 10.0	2.6.16.21-0.8	+	+	+
SuSE* Linux* 10	2.6.13-15	+	+	
SGI Pro Pack* 4.0, Service Pack 3	2.6.5-7.244-sn2			+
SGI Pro Pack* 5.0	2.6.16.21-08			+

Red Flag* Linux* 5.0 (Data Center)	2.6.9-11.19AX	+	+	+
Miracle* Linux* 4	2.6.9-11.19AX	+	+	+
Haansoft Linux* 2006 Server	2.6.9-11.19AX	+	+	+
Turbo Linux* 10	2.6.9-5.13	+	+	+
Mandriva / Mandrake Linux* 10.2	2.6.11-6mdk	+	+	
MontaVista* Linux Carrier Grade Edition 4*	2.6.10	RDC		

The VTune Analyzer for Linux supports all compilers that follow industry standard object code formats (i.e. ELF, STABS, DWARF).

For example, the analyzer works with applications built with these compilers:

- GCC* 2.96
- GCC* 3.2, 3.3, 3.4
- GCC* 4.0
- Intel(R) C++ Compiler 8.0, Intel(R) Fortran Compiler 8.0
- Intel(R) C++ Compiler 8.1, Intel(R) Fortran Compiler 8.1
- Intel(R) C++ Compiler 9.0, Intel(R) Fortran Compiler 9.0
- Intel(R) C++ Compiler 9.1, Intel(R) Fortran Compiler 9.1

Operating System Requirements for Linux* Remote Data Collector

Please refer to the [Operating System Requirements](#) section in these release notes.

Java* Development Kits (JDKs)

The VTune analyzer in the integrated Eclipse platform environment has been tested for operation on the following JDK:

- BEA* JRockit* v5.0

The VTune analyzer has been tested for profiling under the following JDKs. The latest releases of these JDKs may be used:

On IA-32 systems:

- Sun* J2SE v5.0
- IBM* JDK 1.4.2 and v1.5

- BEA JRockit 1.4.2 and v5.0

On Intel(R) Itanium(R) architecture systems:

- IBM JDK 1.4.2
- BEA JRockit 1.4.2 and v5.0

On systems with Intel(R) 64:

- Sun J2SE v5.0
- BEA JRockit 5.0

NOTE:

The VTune analyzer will crash under Eclipse if you run it with BEA JRockit 1.5.0 under Red Hat Enterprise Linux 4.0 for Intel(R) 64. Do one of the following to avoid this situation:

- Install the errata kernel for Red Hat Enterprise Linux 4.0
- Use Red Hat Enterprise Linux 4.0 U1
- Use the Sun JVM 1.5.0

Call graph support on the different architectures

	VTune Analyzer 8.0 for Linux*		VTune Analyzer 3.0 for Linux*	
	Local	Remote data collector on Linux	Local	Remote data collector on Linux
IA-32	Yes	Yes	Yes	Yes
Intel(R) 64				
64-bit apps	Yes	Yes		
32-bit apps	Yes	Yes		
Intel(R) Itanium(R) Architecture				
64-bit apps	Yes	Yes	Yes	Yes
IA-32 apps				

Installation

Refer to the information in the [Installation Guide](#) located in the `<install_dir>/INSTALL.txt` document.

Usage Notes

Launching the VTune(TM) Performance Analyzer for Linux

This section explains how to launch the VTune analyzer for Linux in the integrated Eclipse platform environment, or with the command line user interface.

NOTE:

Launching the VTune analyzer requires that the USER environment variable be defined. The USER environment variable is usually defined during the login process.

NOTE:

For best performance when working with your Linux system remotely using an X-server, be sure the X-server supports efficient font anti-aliasing (for example, Hummingbird* Exceed* X server v10 and higher).

NOTE:

If you have ever used the Eclipse GUI from the VTune analyzer 3.x, 8.x for Linux on your machine, please remove the `.eclipse` directory from your `$HOME` directory before running the Eclipse GUI for the VTune analyzer 9.0 for Linux.

Use the following command to launch the VTune analyzer in the integrated Eclipse platform environment:

```
$ <install_dir>/bin/vtlec
```

Where: `<install_dir>` is the installation directory. The default installation directory is: `/opt/intel/vtune`.

NOTE:

The VTune analyzer projects are saved within an Eclipse workspace. The workspace location is established at the time `vtlec` is invoked. By default, the workspace directory is created in the current directory. You can use the `-data` option to specify a workspace directory in a different location.

The following example creates a workspace directory called **workspace** in the directory `/home/MyProj`. All VTune analyzer project files are saved within that workspace directory.

```
$ cd /home/MyProj  
$ /opt/intel/vtune/bin/vtlec
```

If you are in a directory other than `MyProj` and wish to invoke `vtlec` and view the project files created during the above session, use the following command line:

```
$ /opt/intel/vtune/bin/vtlec -data /home/MyProj
```

To view the Eclipse help, go to **Help > Help Contents**. Within the help, the following books contain information on the VTune Performance Tools: VTune(TM) Performance Environment, VTune(TM) Performance Analyzer, VTune(TM) Performance Analyzer Reference. For information on the various Eclipse options that can be added to the `vtlec` command line, see the **Workbench User Guide > Tasks > Running Eclipse** topic.

To change the Eclipse JVM arguments, use the standard Eclipse mechanism, the `-vmargs` command-line option. Use the following command:

```
vtlec [platform options] [-vmargs [Java VM arguments]].
```

NOTE:

`-vmargs` must be the last option on the command line.

Using the `vtl` Command Line

To use the command line version of the VTune Performance Analyzer, use this command line:

```
$ <install_dir>/bin/vtl
```

where: `<install_dir>` is the installation directory. The default installation directory is: `/opt/intel/vtune`.

Using the Pause/Resume and Stop/Start APIs

Use the Start/Stop (for sampling only) and Pause/Resume APIs to start the data collection from the application and profile only specific sections of your code. See the VTune analyzer command line interface User's Guide for more information. The VTune analyzer User's Guide is in: `/<install_dir>/doc/users_guide/index.htm`, where `<install_dir>` is the installation directory. The default installation directory is `/opt/intel/vtune`.

Sampling Data Collection and Analysis on large HPC multi-processor systems

This section explains how to use this release for sampling collection and analysis of large multi-processor systems using the `vtl` command line.

NOTE:

Due to the unique requirements for supporting large systems, if the software will be used on systems with more than 128 cores please contact us before purchase to make special arrangements.

1. Install VTune analyzer 9.0 for Linux and setup local sampling for the number of processors on the desired Linux machine.
2. Create and run an Activity with a sampling collector. An optional cpu-mask can be specified to limit collection to a subset of processors available on the system. For example, the following vtl command can be used for sampling processors 1, 4 and processors 20 to 25:

```
vtl activity -c sampling -o "-cpu-mask 1,4,20-25"
```

3. Open the sampling collection results using the command line viewer (vtl). For example:

```
vtl view -processes -cpu 1,4
```

You can use the VTune(TM) Performance Analyzer 8.0 (for Windows* operating systems) to view and analyze the sampling collection results in graphical mode using Process, Modules, Hotspot, and the Sampling Over Time (SOT) Views. To view the sampling data using GUI Viewers on Windows, perform the following steps:

1. Install the VTune analyzer 8.0 or higher, on a system with the Windows operating system.
2. Copy the .tb5 file created in step 2 above, to the Windows system.
3. Launch the VTune analyzer 8.0 and use the **File > Open File** menu command to open the .tb5 file.
4. View the sampling results in the **Process/Modules/Hotspots** views.
5. Click the hourglass icon to invoke the **Sampling Over Time (SOT)** views for the selected processes/modules.

NOTE:

Collecting sampling data on systems with a large number of processors generates a significant amount of data. The most significant factor is the sampling rate. First, try to increase the **Sampling Interval** value, then use CPU masking to further reduce the amount of collected data. CPU masking enables you to limit data collection only to the processors of interest.

VTune(TM) Performance Analyzer and the Processes Started by EntireX DCOM for Linux

The VTune analyzer uses a number of daemons which are started and stopped during the installation process. During the installation, you are prompted to select automatic startup of these daemons at system boot time, the suggested default behavior.

The administrator of the system can manually stop and start these daemons by issuing one of the following commands as root (use `su -` to log in as root before proceeding).

```
$ /etc/rc.d/init.d/ntd stop (for RedHat and SuSE based systems)
```

```
$ /etc/rc.d/init.d/ntd start
```

If the system isn't based on RedHat or SuSE the administrator of the system should use the following commands.

```
$ /etc/rc.d/init.d/ntdwrapper stop (for Turbo Linux* 10 and others)
```

```
$ /etc/rc.d/init.d/ntdwrapper start
```

Typically, about thirteen (13) processes are started as a result of the `dcom start` script. The number can increase from there, upon use. To see the number of processes currently running on your server:

```
$ ps -u vtunesag | wc -l # shows number of processes
```

Using per CPU buffering in an Activity collecting Sampling Data

Use the Per CPU buffering feature to enhance the quality of sampling results on systems with large number of processors. This feature is automatically enabled on SGI Altix Pro Pack 4 Service Pack 3, and SGI Altix Pro Pack 5 systems.

Turning on/off per CPU buffering

To manually enable Per CPU Buffering, set **SEP_PERCPU_BUFFER=1** prior to starting the VTune analyzer.

To manually disable Per CPU Buffering, set **SEP_PERCPU_BUFFER=0** prior to starting the VTune analyzer.

Adjusting the Priority of VTune(TM) Performance Analyzer process

During sampling collection, data is stored in the internal kernel buffers. When the buffers become full, the VTune analyzer flushes the data to the disk. During these brief periods, the VTune analyzer does not collect samples. To minimize the loss of samples, the VTune analyzer needs to run at a higher-than-normal priority while sampling.

To adjust the process priority, set the **SEP_PRIORITY** environment variable prior to starting the VTune analyzer. The valid range for this variable is between -20 to 19. A negative number means a higher priority will be used and fewer samples will be missed. A positive number means a lower priority will be used but with the risk of potentially missing a significant number of samples. A priority of 0 means some samples may be missed. If **SEP_PRIORITY** is not set or the value is outside of this range, then a value of -1 will be used.

Known Limitations

This section details the known limitations and possible solutions in the following areas: [Installation Limitations](#), [General Limitations](#), [Sampling Limitations](#), [Call Graph Limitations](#) [Online help Limitations](#).

Installation Limitations

- The VTune(TM) Analyzer installation may fail if all users, including root, are not allowed to add new users and groups to their systems except through NIS. In this case you should create an NIS user "vtunesag" and an NIS group "vtunesag" with the following parameters:
\$ groupadd -g 28888 vtunesag
\$ useradd -u 28888 -g vtunesag -r -s /bin/false -d /sbin vtunesag
After that, try to re-install VTune Analyzer. [SCR #27637]
- If you wish to use the VTune(TM) analyzer for Linux* Remote Data collector (vtserver) to profile an application in a firewall environment, you need to do the following:
 1. Set: VTUNE_RDCPORT_IN environment variable before running the VTune analyzer. For example:
set VTUNE_RDCPORT_IN=55555. This setting is communicated to vtserver so that it will initiate callback connections only to this port and the next nine ports above it.
 2. The network administrator must reserve a range of 10 ports for lab outgoing connections. In the above example: 55555-55564
- The VTune(TM) Analyzer 9.0 for Linux* can be integrated only with the 32 bit version of Eclipse 3.2.1 on Intel(R) 64 systems. Integration into a 64-bit version of Eclipse is not supported. [SCR #25119]
- If you are installing the vtl software on a network file system (such as NFS or AFS), you may experience security-related installation failures, depending on how the network file system was set up. In the event of such a failure, you should first install the needed software locally (by default, the /opt/intel and /opt/sag directories, and their subdirectories), and then, after successful completion, put the software on the shared drive. To complete this process, you need to create symbolic links pointing from the local directories to the network directories. [SCR #13717]
- During installation, do not enter an installation path that contains a space; if you do, the installation will fail. [SCR# 14544]
- If you install the Intel(R) C++ Compiler 8.1 with Eclipse integration, then install the VTune analyzer with Eclipse integration and try to run vtlec, it fails to start. You should set these environment variables prior to launching vtlec: OTHER_JVM_BINDIR=Your **JVM/bin** directory ("**/opt/intel/eclipsepackage/3.1.1/eclipse**" by default) and OTHER_ECLIPSE_BIN=Eclipse binary file for VTune analyzer ("**/opt/intel/eclipsepackage/3.1.1/eclipse**" by default).[SCR #23917]
- VTune Analyzer 9.0 for Linux* **requires** Eclipse* version 3.2.1. It will not work properly with another version of Eclipse.
 - Intel compilers version 9.0 and earlier install an earlier version of Eclipse, therefore during the VTune analyzer installation, you must choose to **install a separate version**. You should not merge with an existing version of Eclipse* unless it is version 3.2.1.
 - If you have installed any other tools that require an earlier version of Eclipse*, you **must install a separate version** of Eclipse for the VTune analyzer for Linux.
- The VTune(TM) Performance Analyzer does not support the cluster file systems Linux kernel under SuSE*

General Limitations

- Unaligned access messages are due to compilation issues with the VTune(TM) analyzer binaries, and have no effect on any user data collection or activity.
- Eclipse* does not work on SGI Altix* machines with more than 256 CPUs. [SCR #29379, 29863]
- For some optimization phases (e.g. IPO, HLO, PGO), the Intel(R) Compiler may not generate line number information. In these instances the VTune analyzer will display the OptReport for the entire source file even if the user selects a specific range in Source View. [SCR #28908]
- The VTune(TM) analyzer cannot pack files that are larger than 2 GB. You may be able to mitigate this limitation by compressing that data. This feature can be set using the global-options [SCR #29391]
- If you try to use the online help, you may see an error message similar to this: **Could not launch external Web Browser for http://127.0.0.1:57399/help/index.jsp**. If this error occurs, check your browser proxy settings and make sure the address 127.0.0.1 is in the proxy exceptions list. [SCR #26760].

All non-standard GTK IM modules (like SCIM on Red Flag* Linux* 5.0 DC) will be disabled and will not run after starting Eclipse. This is the result of SCIM bug #1502121 (Crash caused by SCIM (C++ ABI problems),

http://sourceforge.net/tracker/index.php?func=detail&aid=1502121&group_id=108454&atid=650539).

The problem is caused by a C++ runtime conflict between the SCIM and VTune analyzer libraries which occasionally causes Eclipse to crash.

The issue could not be fixed on the VTune analyzer side due to incorrect STL symbols versioning in SCIM. You may download and install a fixed SCIM package from <http://www.asianux.com/tsn/>. After installing the fix, or to enable another GTK IM module, set the VTUNE_ENABLE_SCIM environment variable to enable SCIM as follows:

```
$ export VTUNE_ENABLE_SCIM=yes
```

and re-run VTune(TM) Performance Analyzer.[SCR# 27259]

- Do not use the “New Editor” option in the context menu of all VTune analyzer for Linux results viewers and the same option in the “Window” menu in Eclipse. After selecting one of these options the viewer may not function correctly. “New Editor” is an Eclipse feature, and the VTune analyzer for Linux does not support it. If you select “New Editor”, close and reopen the VTune analyzer. [SCR# 26209]

- The VTune analyzer may crash under Eclipse if you run it with BEA JRockit 1.5.0 under Mandriva 10.2 Linux on Intel(R) 64 and Intel(R) IA-32 systems. Install the 32-bit Sun JRE v5.0 and set the path to the JRE in the environment variable OTHER_JVM_BINDIR to Sun's JVM directory as a workaround for this issue [SCR #27182]
- If you have a counted (also known as floating) license file, install the FLEXlm server on supported IA-32 or Intel (R) Itanium(R) processor family system. FLEXlm servers are not supported on Intel(R) Itanium(R) Processor Family systems with Red Hat* Enterprise Linux* 4.0, SuSE Linux* Enterprise Server 9, or other Linux distributions with an equivalent kernel and glibc installed. If you install the VTune(TM) performance Analyzer on an Intel(R) Itanium(R) Processor Family system with Red Hat Enterprise Linux 4.0 or SuSE Linux Enterprise Server 9 the license checkout will succeed as long as the FLEXlm server is running on a supported operating system and architecture, and the license file is copied appropriately on the system where the VTune analyzer is installed. Please refer to the FLEXlm User's Guide (flex_ug.pdf) for more information on how to setup the licensing environment. This information is available on the Intel(R) Premier Support website under the **Intel SW Dev Tools** product. [SCR #25212]

- The VTune Analyzer for Linux supports SELinux (new kernel security extensions from the SELinux project at the NSA) if the current policy is "Targeted". Otherwise you may receive the error messages like "cannot restore segment prot after reloc: Permission denied" when launching the VTune Analyzer on the newer Linux distributions such as Red Hat* Fedora* Core 4. You may disable SELinux by:

- setting the line “SELINUX=disabled” in your */etc/sysconfig/selinux* file
- adding "selinux=0" kernel argument in lilo.conf or grub.conf files

[SCR #26525, #26745]

- Java profiling may crash under BEA JRockit* 1.5.0_04 since the JDK communicates the wrong address range for jitted functions. You should perform Java profiling under JRockit versions which do not suffer from this issue, for example:
 - BEA JRockit* 1.5.0_02 and 1.5.0_08 on Intel(R) IA-32 system

- BEA JRockit* 1.5.0_02 on Intel(R) 64 and Itanium(R)-based systems

[SCR #27068, #27228]

- After you have installed the product and are using it, if you get an error message similar to: "Error: NTD is not running at the moment, please start it before using NT services", and if, after trying to restart the ntd service, you get a message asking you to create a password for "User vtunesag" followed by an error message similar to: "`vtunesag:vtunesag': invalid user", please uninstall and reinstall the product. For more information on how to uninstall the product please refer to the information in the [Installation Guide](#) located in the `<install_dir>/INSTALL.txt` document.
- Java* may dynamically generate components during execution. As a result, a high percentage of samples (>5%) may be attributed to the **Other32** pseudo module since no executable modules are associated with the dynamically generated code. This generated code does not relate to the Java programming language, but relates only to the JVM implementation. Java developers may safely ignore samples in the **Other32** pseudo module and focus only on the compiled Java code that is found in the java.jit module. [SCR #24897]
- BEA JRockit* 1.5.0_06 system reports the wrong address range for jitted functions during execution. As a result, about ~30% of the samples may be attributed to the Other32 pseudo module. You should perform java profiling under a BEA JRockit* version that does not suffer from this issue, for example, BEA JRockit* 1.5.0_08. [SCR #29439]
- The VTune analyzer will crash under Eclipse if you run it with BEA JRockit 1.5.0 under Red Hat Enterprise Linux 4.0 for Intel(R) 64. The problem occurs if the "**Modify Activity before running**" check box is checked at the end of the wizard. Do one of the following to avoid this situation:
 - Install the errata kernel for Red Hat Enterprise Linux 4.0
 - Use Red Hat Enterprise Linux 4.0 U1
 - Use the Sun JVM 1.5.0[SCR #24369]
- Eclipse 3.1.1 may hang on an Intel(R) IA-32 system with BEA* JRockit* 1.5.0 and Red Hat* Enterprise Linux 4.0. Install the 32-bit Sun JRE v5.0 and set the path to the JRE in the environment variable OTHER_JVM_BINDIR to Sun's JVM directory as a workaround for this issue [SCR#24524].
- You may get an error message: "An error has occurred. See error log for more details" if you have uninstalled the analyzer and reinstalled it. Delete the `$HOME/.eclipse` directory after the uninstall is finished. [SCR #24141, 26740].
- When you open an Eclipse workspace created with a previous version of Eclipse, you may get an error message. This is a general Eclipse issue and will be considered by eclipse.org for a future version of Eclipse. The error message is only displayed the first time you open the Eclipse workspace. [SCR #24203]
- The evaluation version of the analyzer contains all of the features of the commercial version, except you do not have the ability to open existing projects, save projects, or the ability to create or unpack a **Pack and Go**

file. You do not need to re-install the product when you convert from the evaluation version to the commercial version. Simply copy your commercial license file into the directory specified by the INTEL_LICENSE_FILE environment variable. The default location is ***/opt/intel/licenses***.

You must install the 32-bit packages to enable the 32-bit Eclipse package on an Intel(R) 64 system from x86_64 distros to guarantee co-existence of 32-bit and 64-bit packages. The exception here is the Mandrake Linux 10.2 x86_64 distro which doesn't contain the required 32-bit packages. You can install the 32-bit packages from the Mandrake Linux 10.2 x86 distro as a workaround for the issue.

-
- Importing an Eclipse project which contains a tuning project requires using the **Pack and Go...** and **Unpack...** options in the Tuning Browser pop-up menu for projects that include only a tuning project. [SCR #20550]. For Eclipse projects with a tuning project and other items, follow these steps to complete this process:
 1. In Eclipse, select your project and choose **File > Export**. Follow the wizard steps to export the project. When asked what files to include, deselect the .vtproject. data directory.
 2. In the Tuning Browser view, right-click your project or Activity and select **Pack and Go...** from the pop-up menu. This creates a .vxp file.
 3. Import the exported Eclipse project into a new workspace. If asked to clear the tuning project, select **yes**.
 4. In the Tuning Browser view, right-click and select **Unpack...** . Unpack the packed VTune analyzer project (the .vxp file) into the empty VTune analyzer project. The VTune analyzer project has the same name as the Eclipse project.
- **Copy/paste** in Eclipse may not work as expected. To overcome this issue, try one of the following: ctrl-insert to copy and shift-insert to paste or, copy to a scratch file and then paste to your Eclipse location. You can also use third party tools for cutting and pasting to and from Eclipse [SCR #20768].
- The HTML documents delivered with this product are designed for use with the latest 4.0 HTML-compliant browsers. Older browsers may not display some pages correctly. To view the full features of the HTML documentation, upgrade your browser to the latest release. [SCR #17614]
- The Flash movies in the Getting Started tutorial may not work properly in Mozilla 1.4 browser. To view the animation, use the recommended browsers. [SCR #19723].
- To access the context-sensitive help on GTK Eclipse, press Shift-F1.
- The Macromedia* Flash* player, required for viewing the web-based tutorial, is not available for Intel(R) Itanium (R) architecture. To view the animated sections of the tutorial, use a 32-bit browser.
- Working with *.tb5 and *.vxp files make sure they are created with the VTune Performance Tools, otherwise, you may experience serious problems with your current project. [SCR#20450].
- When running the VTune analyzer on Intel(R) Itanium(R) processor you may receive the following messages: "Unaligned access to: 0xxxxxx ip:0xxxxxx". This is a Linux kernel warning message. [More information](#) on how to disable the warning [SCR#22433].
- Call graph profiling on a remote system may not function properly, when launched in parallel with 'nscd' daemon

in the network with unconfigured DNS, NIS, NIS+. To avoid this issue, configure these services or stop the 'nscd' daemon. [SCR# 20330].

- When querying available data collectors on a remote system, be aware that output will contain a list of proxy collectors that potentially can be used for data collection, but they might not be necessarily physically available on that remote host. [SCR# 19905].
- On distros for the Intel(R) Itanium(R) processor family, enable 32-bit application support in the kernel. Otherwise you get a warning message "cannot execute <binary name>".
- You cannot convert the .rsf and .rmf files collected on an Intel XScale(R) technology platform into a .tb5 file using the buildTbxXsc utility because it is not available for Linux* hosts. [SCR #24677]
- If the VTune analyzer terminates abnormally, there is a high probability that the workspace remains locked. Subsequent invocations of the VTune analyzer show the warning message **Workspace in use, choose a different one**. To unlock the workspace, go into the workspace directory and delete **.metadata/lock** file. [SCR #26042]
- Dynamic Help loses context sensitive related topics in the About section after the section is sequentially collapsed and expanded. It affects dynamic help for the Preferences, Call Graph activity configuration and Sampling activity configuration dialogs. Don't click on the About section to collapse the entry to avoid this problem. [SCR #25831]

Sampling Limitations

- In order to build the VTune analyzer Driver Kit driver for Red Flag 5.0 DC kernel compiled with the Intel(R) C++ Compiler you must install icclibs rpm which provides Intel(R) C++ Compiler runtime libraries [SCR #27206].
- Sampling collection under Eclipse may take a longer time if Dynamic Help indexing is occurring during the sampling activity. As a workaround for this issue, wait until indexing is finished, then run your sampling activity. [SCR #27782]
- Sampling on kernels configured with non-consecutively numbered CPUs (as seen by "cat /proc/cpuinfo") is not supported [SCR #28550]
- The VTune(TM) performance Analyzer 9.0 for Linux* client is not compatible with the VTune analyzer 8.0 for Linux* vtserver. If you try to collect data using this configuration you will get an error message: **Remote agent on the remote system is incompatible with this version of the VTune Performance Analyzer**. Be sure you install the VTune(TM) performance Analyzer 9.0 for Linux* remote agent on the remote machine to avoid this situation. The reverse is also true. The VTune analyzer 8.0 for Linux client is not compatible with the VTune analyzer 9.0 for Linux vtserver. [SCR #25992]
- After installing the VTune(TM) performance Analyzer 8.0 for Windows* Remote Data Collector on an Intel(R) 64 system, the VTune analyzer 9.0 for Linux* Sampling fails with the error message: **Could not connect to remote machine 127.0.0.1**. There is no workaround for this issue. Do not install the Remote Data Collector from the VTune(TM) performance Analyzer 8.0 for Windows* package on an Intel(R) 64 system. [SCR #26556]
- The VTune analyzer cannot recognize module names for applications built with the Intel(R) Fortran Compiler for Linux* 9.0 using Red Hat* Fedora* Core 2 on IA-32. The workaround for this issue is to use the newer version of the Intel(R) Fortran Compiler for Linux*, version 9.0. [SCR #24720]

- Eclipse may crash if you have insufficient disk space when you run a sampling activity. Be sure you have sufficient disk space before running your activity, as specified in the disk space requirements section of these Release Notes. [SCR #25239]
- You may not be able to open up sampling results if you do not have sufficient free disk space. Be sure you have sufficient disk space before running your activity, as specified in the disk space requirements section of these Release Notes. [SCR #25240]
- Sampling does not work on Dual-Core Intel(R) Itanium(R) 2 processor 9000 sequence systems with Asianux 2.0 (kernel 2.6.9-11.24AX). You should upgrade to a newer kernel which supports sampling on Dual-Core Intel (R) Itanium(R) 2 processor 9000 sequence systems. [SCR #27085]
- Sampling with BEA JRockit 1.4.2 incurs overhead > 50%. Add the following command line options to the java program to solve the problem. This is not needed for BEA JRockit 1.5.0.

-Xjvmpi::allocs=off,monitors=off,entryexit=off

[SCR #22551]

- When performing a remote sampling session for a Java* application, make sure that vtserver is up and running on the remote host before analyzing the collected data. Otherwise, Eclipse may crash when displaying the source view for this Activity results. [SCR# 22141].
- In order for sampling to work on a Linux* 2.6 kernel, profiling support must be enabled in the kernel. Some IA-32 UP kernels such as those in Red Hat* Fedora* Core 2 and Core 3 distributions and in Red Hat* Enterprise Linux* 4.0 do not have profiling support enabled. If you are using such a kernel, we recommend that you use the IA-32 SMP kernel instead since profiling support is enabled there. The IA-32 SMP kernel is usually provided on the Linux vendor's install CDs. Once you're booted into the kernel, then you may either use the default sampling driver (if it is available) or build one using the VTune analyzer driver kit. See **<installdir>/vdk/src/README** for details.
- In some cases when a binary is compiled with a high optimization level (-O3) there may be a problem displaying sampling event statistics for inlined functions (SCRs# 23257, 24924). The sum of events in the hotspot view and in the source view may differ. If you use the Intel(R) compiler version 9.0 you should either switch off inlining at all (via "mlPOPT_ninl_debug_info=F compilation option) or explicitly switch on generation of debug info related to inline functions (via "-debug inline-debug-info" option).
- When performing remote data collection on a kernel 2.6.9-based SMP system (for example, Red Hat Enterprise Linux 4.0), if an application is launched by the VTune analyzer, **"/bin/bash"** may appear in Process View results instead of the application's name. If the results are opened in Module View, the application's name should appear and subsequent drilldown to source should work. [SCR #24977]
- By default, the duration of data collection is 20 seconds. To change the default duration, use one of the following configuration options:
 - Set the **Stop Collection** setting in the integrated Eclipse platform environment wizard.
 - Set the duration of the Activity using the `vtl activity -d` option.
 - Set the default duration using the `vtl global-options activity-duration` option.

- When running java sampling with BEA 1.4.2, the user should add the following JVM flags during java invocation, in order to reduce JVM overhead: **-Xjvmpi::allocs=off,monitors=off,entryexit=off -Xgc:gencon**
- The Sampling usage model is "one user at a time". Multiple users may not configure and run concurrent sampling activities on the same system.
- Interactive console applications can create issues when used with the `vt1` command. The following conditions should be avoided:
 - Running more than one interactive console application started for the same Activity.
 - Running an interactive console application and selecting the **Don't terminate when done collecting** option.
 - Running any interactive console application that does not properly clean up terminal settings when terminated by `vt1`.
 - Running any interactive console application that does not terminate on receipt of SIGTERM.
- If your sampling collection Activity is stuck, check whether there is still space available on the disk where the VTune analyzer project resides. If there is no more space available, the VTune analyzer hangs without notice. You need to clean up some disk space and run your Activity again.
- Per CPU buffering in sampling collection has been tested for operation on the following Linux distributions: SuSE* Linux* Enterprise Server (SLES) 9.0 Service Pack 3, SGI Pro Pack* 4.0 SP3, SGI Pro Pack* 5.0.
- If the remote Linux* application to launch is a command-line program that expects user input from standard input (stdin), then that application may not be directly launched by vtserver. The following alternatives are recommended:
 - a. Run the command-line program manually on the Linux* system (not launched through vtserver).
 - b. If the Linux* system is running X Windows*, and vtserver has access to \$DISPLAY, then the command-line program can be launched indirectly through an xterm. In the "Application/Module Profile Configuration" dialog box, specify the remote application to launch as follows:

Filename: ***/usr/bin/X11/xterm***

Command line arguments: ***-e /path/to/command-line/program arg1 arg2...***

See the man pages on xterm for further details. [SCR #25290]

- Sampling 32-bit Java* applications on an Intel(R) 64 system with a 64 bit Linux* operating system using newer kernels is not supported. [SCR #25950]
- The VTune analyzer Driver Kit driver does not compile on SuSE Linux Enterprise Server 10, Beta 11 on Itanium (R) 2-based systems.
Please see ***/opt/intel/vtune/vdk/doc/HOWTO-BUILD-VTUNE-DRIVER*** for special instructions on building the sampling driver on this platform. [SCR # 27108]
- Sampling hotspots may show up in "Other32" for Sampling activities on MontaVista Linux* based systems. This happens because the sampling driver requires kernel profiling support to be enabled and the

default MontaVista Linux* kernel is not configured with profiling support. To resolve this issue, either rebuild the kernel with profiling support, or start the application prior to starting the sampling activity and then start collecting data. [SCR #25854]

Call Graph Limitations

- The same user cannot run simultaneous remote Call Graph activities on a single vtserver session. [SCR #23894]
- Call graph may appear to provide incorrect results for calling sequences in some cases. The reason for this is that call graph support is based on binary instrumentation, and therefore only instrumented functions are reported in the call graph results. There can be several reasons why a function may not be instrumented:
 - There is no symbol for the function in the binary (the image is stripped).
 - The first basic block of the function is smaller than five (5) bytes.
 - The function is inline and therefore doesn't exist in the binary.
 - The function is called via jump instruction rather than call instruction

In instances where functions that were not instrumented are in the call stack, the call trace in the call graph result will not be complete and it will not include these functions. It may appear as if the function has the wrong siblings. [SCR #29430]

- In Call Graph view, when trying to sort by a hierarchical column, the grid's behavior is to sort by the internal column, and not the hierarchical one - in this case simply set hierarchy off, sort by needed column, and set hierarchy on again. [SCR #23082]
- If you are upgrading your version of the VTune analyzer, after installation you may discover it is not possible to create a call graph project. To correct this issue, remove the `/tmp/vtune_username/` directory, which ensures that the files in it were not created by an older version of the VTune analyzer.
- Although Red Hat* Enterprise Linux* 5 based systems are not officially supported in this release, some of the VTune™ analyzer features may work. Call Graph, however, fails to instrument some of the system modules on these systems and when you run the activity, you get a warning message that the data collection was halted and that no module of interest was loaded for the activity. There is no workaround for this issue. [SCR #29710]
- The LD_LIBRARY_PATH environment variable must not contain `"/lib"`. Including `"/lib"` in the LD_LIBRARY_PATH environment variable may cause call graph to load the wrong shared object with the application. [SCR #19374]
- The LD_LIBRARY_PATH environment variable must always contain an absolute path, not a relative path or Call Graph may fail to profile the application. [SCR #24322]
- If you try to collect Call Graph data for any application that is run through Intel(R) MPI , the VTune analyzer fails to collect Call Graph data and reports "Internal error". One possible workaround is for Intel MPI to run its daemon and application together under the VTune analyzer with a simple wrapper script as shown below. This workaround may not work for a more complex execution pipeline unless the user starts the MPI daemon just before running the application.

```
#run.sh
```

```
mpdboot -f ../mpd.hosts
```

```
mpiexec -n 2 ../mympiapp
```

mpdallexit

#vtl activity -c callgraph -app ./run.sh -moi ../mympiapp

Other possible workarounds are:

1. Instrument the Intel MPI application in advance and pass it to mpiexec in a profiling run.
2. Don't use mpiexec and pass the profiled application parameters to the MPI daemon directly [SCR #26721]

- Two options are available for launching the application: direct execution, and by using an application launcher. When using an application launcher, make sure to do the following:
 - Define the `-moi` parameter.
 - This loader capability does not support running user setuid applications.
 - The launcher application may alter the running environment of the application. Therefore, the module information in the property page for vtl, or the Advanced Activity Configuration dialog box for GUI users, may change during the Activity run.
 - If the profiled application uses the `LD_PRELOAD` env variable, the VTune analyzer saves and concatenates the profiled application value after the VTune analyzer value. The original value is restored after completing the Activity.
- Call graph support is based on binary instrumentation, and therefore only instrumented functions are reported in the call graph results. There can be several reasons why a function may not be instrumented:
 - There is no symbol for the function in the binary (the image is stripped).
 - The first basic block of the function is smaller than five (5) bytes.
 - The function is inline and therefore doesn't exist in the binary.
- When an instrumented application calls the `exec()` system call, the whole image context is replaced with the new image. If the application calls `exec()` with the original name of the instrumented image, then the instrumented image is used. If the application calls executables that are not in the list of modules of interest, the original (non-instrumented) image is called, and no results are generated from this point in the run.
- Functions that are located in the data segment will not show up in the Call Graph results. This does not affect the **critical path** or the **total time** for the caller function in the Call Graph results, but you will be unable to analyze your code at the function level within the call tree of the functions located in the data segment. [SCR #28227]
- Setuid images are not supported by call graph. The setuid mechanism is used to give a user process the effective user ID of another user, usually root. You can use call graph and run the setuid executable only if you are logged in as the same user as the owner of the setuid executable. [SCR #13308]
- Call graph results are written during the regular termination procedure of the process. This means that if the process did not terminate normally, no call graph results are generated. The application may terminate abnormally due to the following reasons:
 - Some type of crash (such as an access violation) occurred during the termination procedure.
 - The application caught an unhandled termination signal.

To generate call graph results in the above cases, you need to cause a proper termination before it is improperly terminated by itself. Use one of the following methods:

- Use the ActivityController command line to stop the Activity, which also stops the application.
 - Send a special signal to the instrumented application. The number of the signal should be the `_Bistro_Exit_Signal_` environment variable. By default, the SIGUSR2 environment variable is used.
- Call graph uses SIGUSR2 as a special signal to produce results when the application terminates abnormally (except for Java applications). If your application uses this signal, please direct the `Bistro_Exit_Signal_` environment variable to another unused signal (SIGUSR1 for example).
- Statically linked executables are not fully supported. You may see problems, especially when C++ exception handling is used. To avoid this problem, dynamically link your application for call graph profiling.
- Call graph implements the same search algorithm as the standard loader for locating executables and shared objects. Therefore, if you use a private version of the loader with a different search algorithm, call graph may not find the required files.
- When specifying a module of interest for call graph profiling on remote machines and systems with Intel(R) 64 processors, make sure the module already exists. For local call graph profiling, you may specify a module of interest that does not exist initially, but will be generated during the Activity run. [SCR #26457]
- For IA-32 processors, the call graph information of an application is kept in memory during the run. By default, the call graph memory buffer limit is half of the physical RAM. If your application is using a large amount of memory, this can cause an extreme slowdown. To overcome this slowdown, reduce the size of the memory buffer used for call graph. The size of the buffer is defined in the property pages in the `.ini` file. Setting other value to the buffer size can be done on the property pages through the command line tool or in Eclipse. To set the value in Eclipse, go to: **Window > Preferences > VTune(TM) Performance Tools > Call Graph > Collector** and set the **Limit collection buffer size** to option to the preferred number of KB.
- For Intel(R) Itanium(R) architecture processors, when your application exceeds the maximum RAM it does not dump the memory to a file, as described in the previous bullet. Therefore, when profiling on these processors, you need to make sure that you have enough swap space in order to complete data collection [SCR #21908].
- If you use the same short `<name>` for your application to profile and the module of interest, but the application and module of interest are in different directories, the VTune analyzer will give you an error message: **<name> is not a valid module**. If you are profiling on an IA-32 system, you will get the error message, but data collection completes successfully. On Intel(R) 64 systems, the data collection will fail. An example command that will create this situation is: **`vtl activity -c callgraph -app /tmp/ls -moi /bin/ls`**. [SCR #25145]
- The buffer-size property name is **buffer size**. By default, the buffer size is set to 128MB, unless your system has less than 256MB, in this case, it is half of the physical memory. More information on the configuration file can be found in the VTune analyzer command line interface User's guide. By default, the VTune analyzer User's Guide is located in `/opt/intel/vtune/doc/users_guide/index.htm`
- Call graph does not support applications that use two different shared objects with the same name, even if they are located in different directories. This may cause an abnormal termination of the VTune analyzer. [SCR #14986].
- Call graph supports the POSIX* threading methodology (either NPTL or linuxthreads). If your application uses the clone system call directly, it is bypassing POSIX* threads. All non-POSIX* multithreading environments and direct use of clone system calls are not supported. [SCR #13552].
- Call graph adds the pthreads library to the instrumented application in order to profile it. Call graph profiling does not support applications that cannot link to the pthread library into the executable. [SCR # 20583]
- Call graph measures Wait Time values using a heuristic model rather than absolute calculations. Because of

this, Wait Time values should be considered as approximations only, not as quantitative results. [SCR #14526].

- On Intel(R) 64 platforms, *.jit files are not included in the .vxp file when you pack a Call Graph activity. This may result in symbol resolution errors when the activity is unpacked; for example: 'ERROR: no symbol match for function with RVA = 6C : java.jit'. A possible workaround may be to store the *.jit files in the /**tmp/vtune_<username>/Cache/ar<number>/JITFiles** folder separately and place them in the directory where the .vxp file was unpacked. [SCR #26637]
- The overhead of call graph data collection can be reduced by changing call graph runtime settings. The overhead varies depending on application characteristics. For example, applications with many small functions run slower. Refer to the **man callgraph** man page, or the integrated Eclipse platform help on the VTune analyzer, for more information on call graph.
- Using **Function selection** with a remote call graph Activity before running the Activity cause instabilities in the Eclipse GUI. To avoid this issue, run the Activity before using this functionality. [SCR# 20191].

Online Help Limitations

- The Online help may not display correctly on some browsers. For the list of supported browsers go to: <http://www.adobe.com/cfusion/knowledgebase/index.cfm> and search on "Webhelp browser support"

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(R) Premier Support account on our secure web site. Please, register at the [Intel Registration Center](#)

- If you have forgotten your password, please email a request to: quad.support@intel.com.

You will find a list of support resources at: <http://www.intel.com/software/products/support>. This web page will direct you to the support resources that are available to you.

Submitting Issues

This version of the VTune Analyzer for Linux includes a utility that gathers information about your specific system and VTune Analyzer for Linux configuration. The information helps our engineering team analyze your issue and may lead to a faster resolution of the issue. Please run this utility and attach the resulting log file to your support issue on [Intel Premier Support](#). You can run the utility by entering the following command: /**opt/intel/vtune/bin/vtqfagent**. The utility specifies the name and location of the generated log file after it runs.

Steps to submit an issue:

1. Go to <https://premier.intel.com/>. Java* and Javascript* and cookies must be enabled in your web browser to submit an issue.
2. Type in your Login and Password. Both are case-sensitive. (If you have trouble registering or are unable to access your Premier Support account, go to <https://registrationcenter.intel.com/support>. There is a link to technical support near the bottom of the page).
3. Click **Submit**.
4. Read the Confidentiality Statement and click the **I Accept** button.
5. Click on the **Submit Issue** link in the left navigation bar.
6. Choose **Development Environment (tools,SDV,EAP)** from the **Product Type** drop-down list.
7. Run `/opt/intel/vtune/bin/vtqfagent` to generate a log file that includes information about your specific environment and VTune Analyzer for Linux* configuration.
8. Enter your question, complete the fields in the windows that follow, and attach the log file generated by running `/opt/intel/vtune/bin/vtqfagent` to successfully submit the issue.

Guidelines for problem report or product suggestion:

- Describe your difficulty or suggestion.
For problem reports, please be as specific as possible, so that we may reproduce the problem. For VTune analyzer problem reports, please include the VTune analyzer's pack and go file, or a small test case.
- Describe your system configuration information.
Be sure to include specific information that may be applicable to your setup: operating system, versions of glibc and kernel, name and version number of installed applications, and anything else that may be relevant to helping us address your issue.
- To report an installation-related issue, include the contents of the installation log file with your report.
The installation log file is created in this file: `/var/log/vtune_install.log`.
- If you have an error installing the VTune analyzer for Linux package, create a Customer support issue on the Intel(R) Premier Support website and select **install error** for the issue type. See the [Technical Support and Feedback](#) section for instructions on how to register for support.

Documentation

The documentation for the VTune analyzer for Linux is presented in these formats:

- Documentation Index
- Readme file
- Installation Guide
- Administrator's Guide
- Readme file for the VTune analyzer Remote Agent
- command-line help
- man pages

- HTML reference, User's Guide and Getting Started tutorial for vtl
- Integrated Eclipse platform environment help

See [Online Help Limitations](#) for information on known issues with the documentation.

Command-line Help

- Enter `vtl -help` to view command syntax and options for vtl.
- Enter `vtl -help -c <collector_name>` where `<collector_name>` is either callgraph or sampling, for detailed help on collecting data.
- Enter `vtl -help -v <view_name>` where `<view_name>` is either callgraph, sampling or source, for detailed help on viewing data.
- Enter `ActivityController -help` to view command syntax and options for the ActivityController.

Manual Pages

To view manual entries for the VTune analyzer, enter the appropriate command:

- `man vtl` for the vtl command man page
- `man vtlec` for the vtlec command man page
- `man ActivityController` for the ActivityController man page
- `man sampling` for the sampling collector man page
- `man callgraph` for the call graph collector man page
- `man source` or `man 7 source` or `man code` for the source view man page

NOTE:

To view the man pages in the local language, set the system variables LESSCHARSET and LANG as follows:

```
LESSCHARSET=utf-8  
LANG=zh_CN.utf8
```

HTML Documentation

The HTML documentation for the VTune analyzer consists of these components:

- Documentation Index. Open this document in the browser to view the list of available documentation. This document is available at `<install_dir>/doc/doc_index.htm`.
- VTune(TM) Performance Environment User's Guide. To access the guide, launch the index.htm file from the `<install_dir>/doc/users_guide` subdirectory.

- Reference Guide. To access the reference, launch the index.htm file from the `<install_dir>/doc/reference` subdirectory.
- Getting Started Tutorial. To access the tutorial, launch the index.htm file from the `<install_dir>/training/guide_vtl` subdirectory.
- Java* Managed Run Time Environment Frequently Asked Questions. To access the document, launch java_faq.htm file from the `<install_dir>` directory.
- These Release Notes.

Where: `<install_dir>` is the installation directory. The default installation directory is: `/opt/intel/vtune`.

Viewing HTML Documentation

The HTML documentation can be viewed with any web browser, although a browser that supports HTML 4.0 is recommended. Older versions of browsers (especially Netscape* and Konqueror*) may not display some of the HTML documents correctly.

Integrated Eclipse Platform Environment Documentation

The VTune analyzer documentation in the integrated Eclipse platform environment is available from the Eclipse user interface. Go to **Help > Help Contents** and click to expand the following books: VTune (TM) Performance Environment, VTune(TM) Analyzer, or VTune(TM) performance Analyzer Reference.

Additional Information

Related Products and Services

Information on Intel software development products is available at <http://www.intel.com/software/products>.

Some of the related products include:

- The [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.
- The [Intel\(R\) C++ and Intel\(R\) Fortran Compilers](#) are an important part of making software run at top speeds with full support for the latest Intel IA-32 and Intel(R) Itanium(R) processors.
- The [Intel\(R\) Performance Library Suite](#) provides a set of routines optimized for various Intel processors.
- The [Intel\(R\) Math Kernel Library](#), which provides developers of scientific and engineering software with a set of linear algebra, fast Fourier transforms and vector math functions optimized for the latest Intel Pentium and

Intel(R) Itanium(R) processors.

- The [Intel\(R\) Integrated Performance Primitives](#) consists of cross-platform tools to build high performance software for several Intel architectures and several operating systems.

GNU libstdc++ Runtime Library Installed with the Product

This release includes a specially built version of the GNU libstdc++ runtime library, which is required for the proper execution of the product. If you need to reproduce these library binaries you must follow the procedure below. However, this is not necessary in order to use the product.

These instructions assume that the gcc-core and gcc-g++ are unpacked in gcc-3.3.3, the build happens in the parallel directory build-3.3.3, and that the build configuration has already been done.

No gcc source file, makefile, or other file either provided in the original tarfiles nor any such file generated during the build process is opened or altered by the following procedure.

1. Build the library:
CD build-3.3.3
make bootstrap
2. Save a log of the way the files are rebuilt by default, before rebuilding them:
CD ../gcc-3.3.3/libstdc++/libsupc++
touch tinfo.cc tinfo2.cc # make them rebuild
CD -
make &> out.txt
3. In the file out.txt, copy the four lines of output that have "tinfo.cc" and "tinfo2.cc" in them into the build-3.3.3/script.sh file.
Edit the script and add -D__GXX_WEAK__=0 just after the "-c" once for each line.
This changes the build of these files so that type_info is compared by string instead of pointer. You can see the effect in typeinfo, where the
__GXX_WEAK__=0 definition results in this definition:
__GXX_MERGED_TYPEINFO_NAMES=0, which causes type_info::before() and type_info::operator== () to be declared in typeinfo and their definitions to be compiled in tinfo.cc and tinfo2.cc.
4. Touch the files again:
CD ../gcc-3.3.3/libstdc++/libsupc++
touch tinfo.cc tinfo2.cc
CD -
5. Run the script created above. The compiler will complain about overriding a builtin.
The value of __GXX_WEAK__ is built in to the compiler by gcc-3.3.3/gcc/cppinit.c.
You get warnings like this:
<command line>:1:1: warning: "__GXX_WEAK__" redefined
CD i686-pc-linux-gnu/libstdc++-v3/libsupc++

\$OLDPWD/script.sh

6. Relink dependencies of the files built above:

CD -

make

Install the library:

make install

Disclaimer and Legal Information

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL(R) 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.

MPEG is an international standard for video compression/decompression promoted by ISO. Implementations of MPEG CODECs, or MPEG enabled platforms may require licenses from various entities, including Intel Corporation.

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 manual 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.

Developers must not rely on the absence or characteristics of any features or instructions marked "reserved"

or "undefined." Improper use of reserved or undefined features or instructions may cause unpredictable behavior or failure in developer's software code when running on an Intel processor. Intel reserves these features or instructions for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from their unauthorized use.

BunnyPeople, Celeron, Celeron Inside, Centrino, Centrino logo, Chips, Core Inside, 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. Leap ahead., Intel. Leap ahead. logo, Intel NetBurst, Intel NetMerge, Intel NetStructure, Intel SingleDriver, Intel SpeedStep, Intel StrataFlash, Intel Viiv, Intel vPro, Intel XScale, IPLink, Itanium, Itanium Inside, MCS, MMX, MMX logo, Optimizer logo, OverDrive, Paragon, PDCharm, Pentium, Pentium II Xeon, Pentium III Xeon, Performance at Your Command, Pentium Inside, skool, Sound Mark, The Computer Inside., The Journey Inside, VTune, Xeon, Xeon Inside 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 © 2006-2007, Intel Corporation.

This product includes software developed by the Apache Software Foundation (<http://www.apache.org/>). The following software license applies to the software developed by the Apache Software Foundation.

Copyright (c) 2000-2005, The Apache Software Foundation. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistribution of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributing in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:

"This product includes software developed by the Apache Software Foundation (<http://www.apache.org/>)."

Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.

4. The names "Apache" and "Apache Software Foundation" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org.

5. Products derived from this software may not be called "Apache", nor may "Apache" appear in their name, without prior written permission of the Apache Software Foundation.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The OpenSource GNU C++ runtime library source files can be downloaded at <ftp://ftp.gnu.org/pub/gnu/gcc> under the terms of the GNU General Public License or the GNU Lesser General Public License as published by the Free Software Foundation. Specific terms are available online at <http://www.gnu.org/> or alternatively from the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA. These links are provided in the hope that they will be useful, but the linked sites are not under the control of Intel(R) and Intel (R) is not responsible for the content of any linked site, or any link contained in a linked site. If you decide to access any of the third party sites linked to this site, you do so entirely at your own risk.

Copyright (c) 2004, Microsoft Systems Journal, All rights Reserved.

Portions copyright (c) 2004, eHelp Corporation, All rights Reserved.

Portions copyright (c) 1988-2001, Sequiter Software Inc. All rights Reserved.

Copyright (c) 2003, Trolltech AS, All Rights Reserved.

Copyright (c) 1989, 1991 by Jef Poskanzer.

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation. This software is provided "as is" without express or implied warranty.

- QImage::smoothScale
-

Copyright (c) 1999 Mizi Research Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- QEucKrCodec
-

Copyright (c) 1999 Serika Kurusugawa, All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES

OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- QEucJpCodec
- QJisCodec
- QSjisCodec
- QJpUnicodeConv

Copyright (c) 1996 Daniel Dardailler.

Permission to use, copy, modify, distribute, and sell this software for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Daniel Dardailler not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. Daniel Dardailler makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

Modifications Copyright 1999 Matt Koss, under the same license as above.

- Drag and Drop

Copyright (c) 2000 Hans Petter Bieker. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- QTsciiCodec

Copyright (c) 2000 TurboLinux, Inc. Written by Justin Yu and Sean Chen.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS "AS IS", AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- QGbkCodec

Qt supports GIF reading if it is configured that way during installation (see qgif.h). If it is, we are required to

state that "The Graphics Interchange Format (c) is the Copyright property of CompuServe Incorporated. GIF(sm) is a Service Mark property of CompuServe Incorporated."

- QImageDecoder
-