

# Performance Profiler

## Intel® VTune™ Amplifier XE 2017

### Intel® Software Development Tools

“Intel® VTune™ Amplifier XE analyzes complex code and helps us rapidly identify bottlenecks. By using it and other Intel® Software Development Tools, we were able to improve PIPESIM\* performance up to 10 times compared with the previous software version.”

Rodney Lessard  
Senior Scientist  
Schlumberger

### Tune Applications for Scalable Multicore Performance

- **Faster code.** Tune with accurate data and low overhead.
- **Fast answers.** Easy analysis turns data into insight.
- **More data.** CPU, FPU, GPU, threading, memory access, and more.
- **Local and remote collection.** Command line and graphical interface.

### What's New

- **Profile both Python\* and native code.** Low overhead, source-line detail.
- **Quickly profile three critical metrics for modern hardware performance:** CPU utilization (threading), memory access, and FPU utilization (FLOPS).
- **Tune Intel® Xeon Phi™ processors** including in-package MCDRAM.
- **Storage analysis.** Tune interplay of I/O and compute.

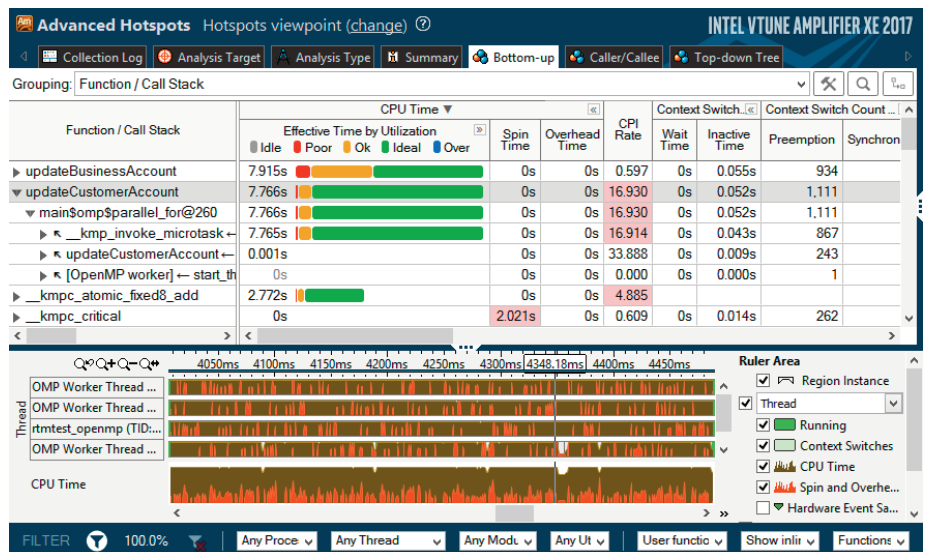


Figure 1. In addition to hotspot profiling (shown), Intel® VTune™ Amplifier can give a detailed analysis of OpenMP performance showing the potential performance gain and common causes of poor performance such as imbalance, lock contention, forking, scheduling, and reduction.

“We achieved a significant improvement (almost 2X), even on one core, by optimizing the code based on the information provided by Intel VTune Amplifier XE.”

**Alexey Andrianov**  
R&D Director Deputy  
Mechanical Analysis Division  
Mentor Graphics Corporation

## Accurate Data plus Powerful Analysis to Interpret It

Whether you're tuning for the first time or doing advanced performance optimization, Intel® VTune™ Amplifier provides accurate profiling data—collected with very low overhead. But good data isn't enough. VTune Amplifier gives you the tools to mine it and interpret it.

## Modern Processor Performance Analysis

Performance on modern processors requires much more than optimizing single-thread performance. High-performing code must be:

- **Threaded and scalable** to utilize multiple CPUs
- **Vectorized** for efficient use of multiple FPUs
- **Tuned** to take advantage of non-uniform memory architectures and caches

Media applications also need to tune OpenCL\* and the GPU. You get all these advanced profiling capabilities with a single, friendly analysis interface.

## Integrated With Intel® Parallel Studio XE Analysis Tools, Runtimes, and Compilers

Additional Intel® Parallel Studio XE analysis tools complement Intel VTune Amplifier XE:

- **Intel® Advisor** optimizes vectorization and prototypes threading.
- **Intel® Trace Analyzer and Collector** examines MPI applications and tells Intel VTune Amplifier XE which loops will benefit most from threading optimization.

## Get the Data You Need

- **Hotspot** (statistical call tree), call counts (statistical)
- **Thread profiling** with locks and waits analysis
- **Memory access**, cache miss, bandwidth, NUMA analysis
- **FLOPS and FPU** utilization
- **Storage accesses** mapped to source, latency histogram, I/O wait
- **OpenCL** program kernel tracing and GPU offload

## Easy to Use

- **No special compiles:** C, C++, C#, Fortran\*, Java\*, Python, Go\*, ASM\*
- **Microsoft Visual Studio\* IDE** integration
- **Graphical interface** and command line
- **Local and remote data collection**, multi-rank setup for MPI applications
- **Analyze Windows** and Linux data from OS X\*

## Find Answers Fast

- **View results** on the source/assembly.
- **OpenMP\* scalability analysis** and graphical frame analysis
- **Memory analysis:** Tune data structures and optimize NUMA latency.
- **Filter out extraneous data** and organize data with viewpoints.
- **Visualize thread and task activity** on the timeline.

“Where Intel VTune Amplifier XE really shines is profiling OpenMP code. We were able to achieve almost linear scaling in our application thanks to Intel VTune Amplifier XE.”

**Jaroslav Šindler**  
Researcher  
Czech Technical University  
Faculty of Mechanical Engineering

## Low-Overhead/High-Resolution Hardware Profiling

Intel® processors have an on-chip performance monitoring unit (PMU). In addition to basic hotspot analysis that works on both Intel and compatible processors, Intel VTune Amplifier XE has advanced hotspots analysis that uses the PMU on Intel processors to collect data with very low overhead. System-wide analysis lets you analyze drivers. Increased resolution (approximately 1 MS versus approximately 10 MS) can find hotspots in small functions that run quickly.

### Product Details

**Quickly locate code taking a lot of CPU time.** Hotspots analysis gives you a sorted list of the functions using a lot of CPU time. This is where tuning gives you the biggest benefit. Click [+] for the call stacks. Double-click to see the source.

| Function / Call Stack                              | Effective Time by Utilization | Spin Time | Overhead Time |
|--|-------------------------------|-----------|---------------|
| std::basic_ifstream< char,struct std::char_traits< | 3.287s                        | 0s        | 0s            |
| FireObject::ProcessFireCollisionsRange             | 2.450s                        | 0s        | 0s            |
| FireObject::FireCollisionCallback← Parallel        | 2.155s                        | 0s        | 0s            |
| FireObject::EmitterCollisionCheck← FireO           | 0.295s                        | 0s        | 0s            |
| CBaseDevice::Present                               | 2.200s                        | 0.180s    | 0s            |
| D3DXCompileShader                                  | 2.010s                        | 0s        | 0s            |

**See the results on your source— now available for both Python and Go.** A double-click from the function list takes you to the hottest spot in the function: C, C++, Fortran, assembly, Java, and now Python and Go. See line-level profiling details on the source.

| Source Line | Source                             | Effective Time by Utilization | Spin Time | Overhead Time |
|-------------|------------------------------------|-------------------------------|-----------|---------------|
| 1,456       | for( u32 j = rangeBegin; j < range | 0.5%                          | 0.0%      | 0.0%          |
| 1,457       | {                                  | 0.0%                          | 0.0%      | 0.0%          |
| 1,458       | FireObject *pfo = m_pFireObj       | 0.4%                          | 0.0%      | 0.0%          |
| 1,459       | if ( checkCollision(ttp, ttp,      | 5.4%                          | 0.0%      | 0.0%          |
| 1,460       | {                                  | 0.0%                          | 0.0%      | 0.0%          |
| 1,461       | // if it passes this test          | 0.0%                          | 0.0%      | 0.0%          |

**Tune threading with locks and waits analysis.** Quickly find a common cause of slow performance in parallel programs waiting too long on a lock while the cores are under-utilized during the wait.

| Sync Object / Function / Call Stack | Wait Time by Thread Concurrency | Wait Count | Object Type        |
|-------------------------------------|---------------------------------|------------|--------------------|
| Manual Reset Event 0xf04628bd       | 71.808s                         | 1,072      | Manual Reset Event |
| Auto Reset Event 0xcc18b37c         | 41.789s                         | 2,540      | Auto Reset Event   |
| Thread Pool                         | 38.303s                         | 1          | Constant           |
| Sleep                               | 38.212s                         | 3,815      | Constant           |
| Manual Reset Event 0xba2e95f3       | 35.302s                         | 505        | Manual Reset Event |
| Auto Reset Event 0x38cd6d85         | 0.737s                          | 298        | Auto Reset Event   |

**Tuning OpenMP is easier with the right data.** See the cause of OpenMP inefficiencies sorted by potential impact for accurate data and low overhead.

**Optimize multi-rank hybrid MPI/OpenMP.** Profile multiple MPI ranks selected in Intel® Trace Analyzer and Collector. Sort by impact of improved OpenMP performance.

| OpenMP Region / Function / Call Stack | Imbalance | Lock Contention | Creation | Scheduling | Reduction | Other  | OpenMP Potential Gain (% of Colle ...) |
|---------------------------------------|-----------|-----------------|----------|------------|-----------|--------|--|
| conj_grad_\$omp\$parallel:24@         | 3.944s    | 0s              | 0.000s   | 0.002s     | 0.000s    | 0.010s | 34.7%                                  |
| MAIN_\$omp\$parallel:24@/h            | 0.086s    | 0s              | 0s       | 0s         | 0s        | 0.000s | 0.8%                                   |
| [Serial - outside any region]         |           |                 |          |            |           | 0s     | 0.0%                                   |

## Product Details (continued)

**Optimize data structures:**

- Attribute cache misses to data structures (not just to code lines)

**Optimize NUMA latency and scalability:**

- Tune true and false sharing
- Analyze inter-socket bandwidth
- Intel® Xeon Phi™ processor MCDRAM (high bandwidth memory) analysis.

**Bandwidth Utilization Histogram**

Bandwidth Domain: DRAM, GB/sec DRAM, GB/sec MCDRAM Flat, GB/sec

**Top Memory Objects by LLC Misses**

| Memory Object                         | L2 Miss Count <sup>®</sup> |
|---------------------------------------|----------------------------|
| <a href="#">stream.c:100 (381 MB)</a> | 2,000,060                  |
| <a href="#">stream.c:98 (381 MB)</a>  | 2,000,060                  |

## Included in Intel Parallel Studio XE

Intel VTune Amplifier XE is available standalone or as part of [Intel Parallel Studio XE](#), a comprehensive software development suite for high-performance software.

## Specifications At a Glance

|  |  |
|--|--|
| <b>Processors</b>                              | Intel® and compatible processors and coprocessors including Intel® Xeon Phi™ processors  |
| <b>Languages</b>                               | C, C++, C#, Fortran, Java, Python, Go, ASM, and more. Works with compilers from Microsoft, GCC, Intel, and others that follow standards. |
| <b>Compilers</b>                               | Works with compilers from Microsoft, GCC, Intel, and others that follow standards  |
| <b>Development Environment</b>                 | Integrated with Microsoft Visual Studio or runs standalone   |
| <b>Host Operating Systems</b>                  | Windows, Linux, OS X viewer available for Windows or Linux data  |
| <b>Target Operating Systems</b>                | Windows and Linux  |
| <b>Basic Threading Analysis</b>                | Intel® OpenMP, Intel® Threading Building Blocks, Intel® Cilk™ Plus and native threads  |
| <b>Extended Threading Performance Analysis</b> | Intel OpenMP, Intel Threading Building Blocks  |
| <b>MPI Parallelism</b>                         | Integration with Intel® Trace Analyzer and Collector MPI Profiler  |
| <b>GPU</b>                                     | OpenCL™ program and media application tuning on newer Intel processors   |

[Learn more about Intel VTune Amplifier XE >](#)

[Get a Free 30-Day Evaluation >](#)

Intel's compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice.

Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. For more complete information visit <http://www.intel.com/performance>.

No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document. Intel disclaims all express and implied warranties, including without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement, as well as any warranty arising from course of performance, course of dealing, or usage in trade. This document contains information on products, services and/or processes in development. All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest forecast, schedule, specifications and roadmaps. The products and services described may contain defects or errors known as errata which may cause deviations from published specifications. Current characterized errata are available on request. Copies of documents which have an order number and are referenced in this document may be obtained by calling 1-800-548-4725 or by visiting [www.intel.com/design/literature.htm](http://www.intel.com/design/literature.htm). For more information regarding performance and optimization choices in Intel® Software Development Products, see our Optimization Notice. [software.intel.com/en-us/articles/optimization-notice#opt-en](http://software.intel.com/en-us/articles/optimization-notice#opt-en)

Copyright © 2016, Intel Corporation. All rights reserved. Intel, the Intel logo, Intel Inside, Intel Atom, Intel Cilk, Intel Core, Intel Iris, Intel Quark, Intel VTune, and Intel Xeon are trademarks of Intel Corporation in the U.S. and/or other countries. \*Other names and brands may be claimed as the property of others.

Printed in USA

0716/SS

Please Recycle