

THREADING AND PERFORMANCE PROFILER

Intel® Parallel Amplifier 2011

Product Brief

Intel® Parallel Amplifier 2011



Optimize Performance and Scalability

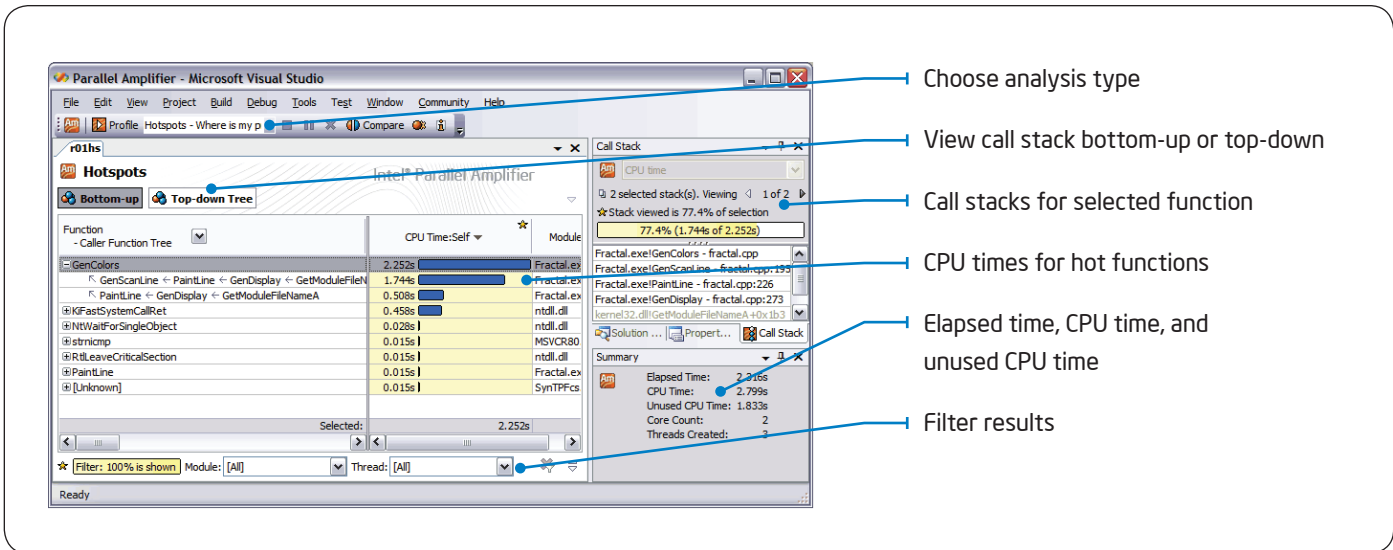
Intel® Parallel Amplifier 2011 makes it simple to quickly find multicore performance bottlenecks—without needing to know the processor architecture or assembly code. Intel Parallel Amplifier eliminates the guesswork and analyzes performance behavior in Windows* applications, providing quick access to scaling information for faster and improved decision making.

Fine-tune for optimal performance, ensuring cores are fully exploited and new capabilities are supported.

- Make significant performance gains that impact customer satisfaction.
- Increase application headroom for richer feature sets and next-gen innovation.
- Find performance problems quickly and easily.
- Scale applications for multicore.

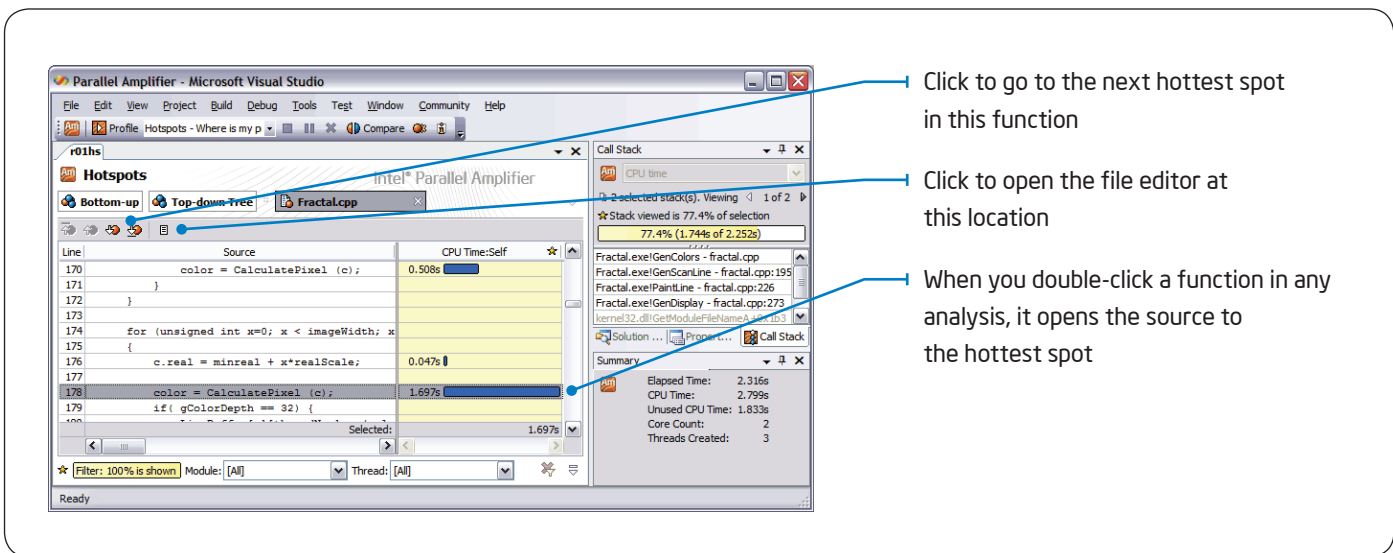
“Thanks Intel, you guys rock! I decided to give Intel® Parallel Amplifier a run. I was delighted when it pointed me to the right source line that was taking much of the time. I made the change, and voilà, our app is now almost 10 times faster. The GUI is very easy to use, in my opinion.”

*Dat Chu
Research Assistant
Computational Biomedicine Lab
University of Houston*



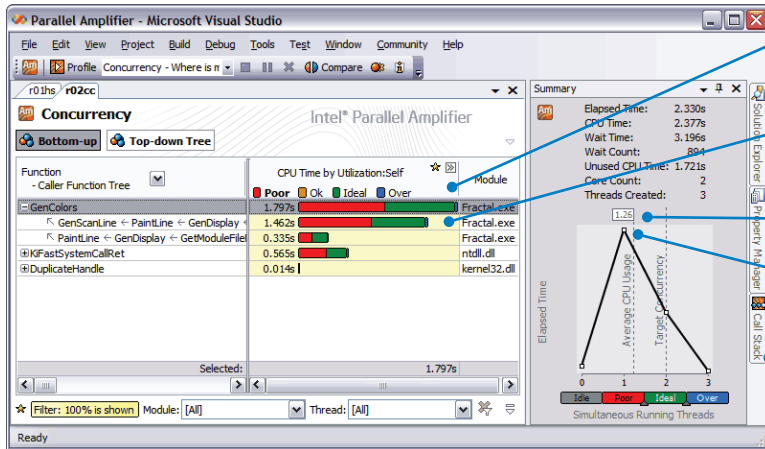
Hotspot analysis: Where is my application spending time?

Find the functions in your application that consume the most time. This is where you'll want to tune or add threading to make your program faster. Intel Parallel Amplifier also shows the stack, so you know how the function is being called. For functions with multiple calling sequences, this lets you see if one of the call stacks is hotter than the others.



Source view: See the results on your source

Source view shows you the exact location on your source. Just double-click on the function names in any of the analysis views to see the source.



Tune core utilization

Green = Ideal Red = Poor

Length of bar is time, color is utilization while the function is running

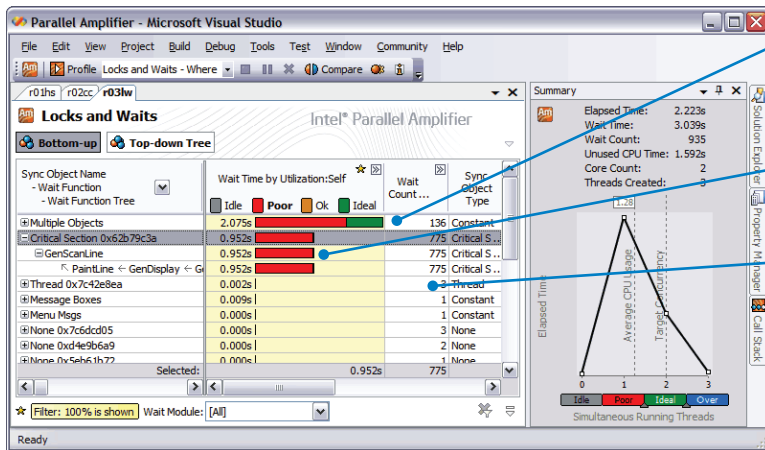
Average CPU utilization

Most of the time this app only uses one core

Call stack information (not shown) is available

Concurrency analysis: When are cores idle?

Like hotspot analysis, concurrency analysis finds the functions where you are spending the most time. But it also shows you how well you are utilizing multiple cores. Color indicates the core utilization while the function is running. A green bar means all the cores are working. A red bar means cores are underutilized. When red, add threading and get all the cores working for you. This helps you ensure application performance scales as more cores are added.



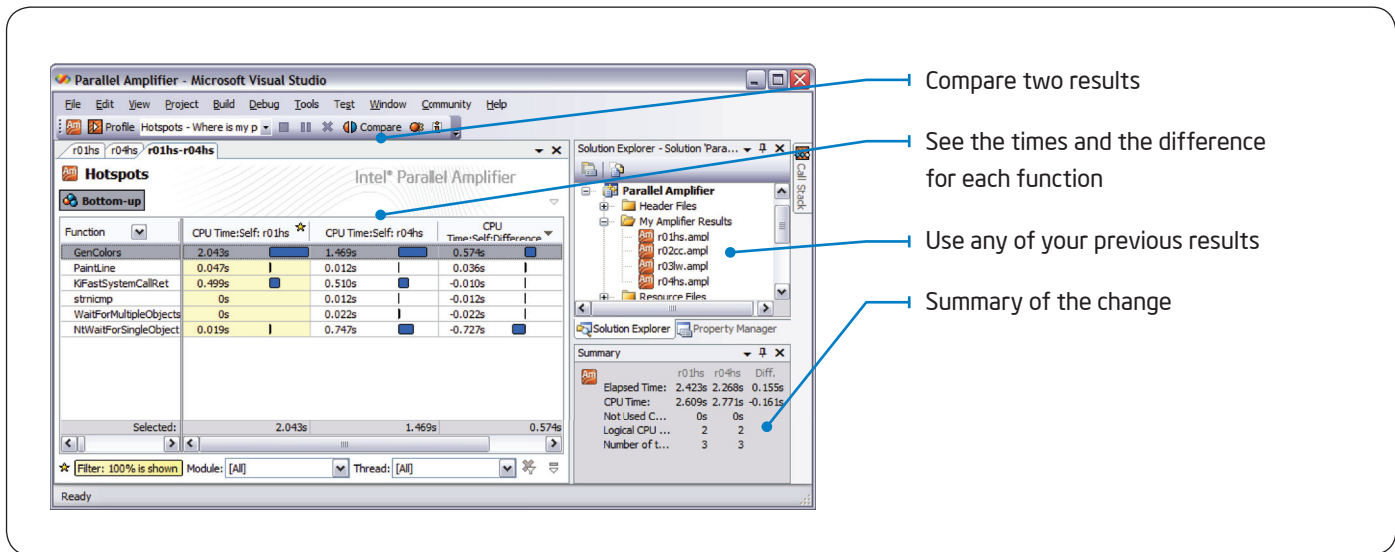
Length of bar is wait time, color is number of cores utilized during the wait

Waiting with underutilized cores hurts performance

Wait count helps identify interesting waits

Locks and waits analysis: Where are the bad waits?

Waiting too long on a lock is a common source of performance problems. It's not bad to wait while all the cores are busy (green), but it is bad to wait when there are unused cores available (red).



Compare results: Quickly see what changed

This gives you a fast way to check progress when tuning and also makes for handy regression analysis.

Features

- It is fully integrated with Microsoft* Visual Studio*.
- It supports Microsoft and Intel® compilers.
- It works with all models for threading offered by Intel® Parallel Studio and Microsoft Visual Studio, Intel® Threading Building Blocks (Intel® TBB), Intel® Cilk Plus, OpenMP*, and WinAPI.
- It is intuitive and was specifically designed for threaded applications.

Use Intel Parallel Amplifier 2011 to:

- Find application hotspots and view them on the source.
- Tune parallel applications for scalable performance using concurrency analysis.
- Use locks and waits analysis to find critical waits that limit parallel performance.
- Compare results to quickly see what changed or find regressions.
- Build applications that automatically scale for manycore.

“The performance profiling capabilities couldn’t be easier to understand and use. Once the analysis is done, it takes 2 seconds to find the problem and identify what can be improved or fixed in my source code.”

David Mercier
Senior Independent Software Developer

System Requirements

- Microsoft Visual Studio 2005*, 2008*, or 2010* (except the Express Edition)
- For the latest system requirements, go to: www.intel.com/software/products/systemrequirements/

Support

Purchase of Intel® Parallel Studio products include Premium Support service which allows you to submit questions, access to product updates, and technical documentation.

For more information, go to <http://software.intel.com/sites/support/>

Download a Trial Version Today

Evaluation copy available at: www.intel.com/software/products/ParallelStudio/

Optimization Notice

Intel® compilers, associated libraries and associated development tools may include or utilize options that optimize for instruction sets that are available in both Intel® and non-Intel microprocessors (for example SIMD instruction sets), but do not optimize equally for non-Intel microprocessors. In addition, certain compiler options for Intel compilers, including some that are not specific to Intel micro-architecture, are reserved for Intel microprocessors. For a detailed description of Intel compiler options, including the instruction sets and specific microprocessors they implicate, please refer to the “Intel® Compiler User and Reference Guides” under “Compiler Options.” Many library routines that are part of Intel® compiler products are more highly optimized for Intel microprocessors than for other microprocessors. While the compilers and libraries in Intel® compiler products offer optimizations for both Intel and Intel-compatible microprocessors, depending on the options you select, your code and other factors, you likely will get extra performance on Intel microprocessors.

Intel® compilers, associated libraries and associated development tools 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 Intel® Streaming SIMD Extensions 2 (Intel® SSE2), Intel® Streaming SIMD Extensions 3 (Intel® SSE3), and Supplemental Streaming SIMD Extensions 3 (Intel® 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.

While Intel believes our compilers and libraries are excellent choices to assist in obtaining the best performance on Intel® and non-Intel microprocessors, Intel recommends that you evaluate other compilers and libraries to determine which best meet your requirements. We hope to win your business by striving to offer the best performance of any compiler or library; please let us know if you find we do not.

Notice revision #20101101

The Ultimate All-in-One Performance Toolkit—Intel® Parallel Studio 2011

Designed for today's serial applications and tomorrow's software innovators

Intel brings simplified threading to Microsoft Visual Studio* C++ developers with a complete productivity solution designed to optimize serial and new threaded applications for multicore and scale for manycore.

INNOVATIVE THREADING ASSISTANT

Intel® Parallel Advisor 2011: Demystify and speed threaded application design.

COMPILER AND THREADED LIBRARIES

Intel® Parallel Composer 2011: Develop effective applications with a C/C++ compiler and advanced threaded libraries.

MEMORY AND THREADING ERROR CHECKER

Intel® Parallel Inspector 2011: Ensure application reliability with proactive parallel memory and threading error checking.

THREADING AND PERFORMANCE PROFILER

Intel® Parallel Amplifier 2011: Quickly find bottlenecks and tune threaded applications for scalable multicore performance.

