

# OVERVIEW OF INTEL® VTUNE™ AMPLIFIER, ADVISOR & INSPECTOR



Dr. Sri Doddapaneni  
Senior Director, Developer Products @ Intel

## Optimization Notice

Copyright © 2019, Intel Corporation. All rights reserved.

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



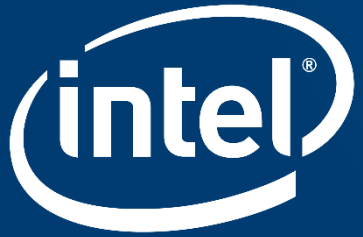
# Understanding Intel Analyses Toolbox

- **Design** of software aided by Advisor
- **Correctness** checked by Inspector
- **Performance** profiled, bottlenecks analyzed, and software/HW context shown by VTune:

Wide-Angle Lens

Telephoto Lens





# **TRIAGE WITH VTUNE PERFORMANCE SNAPSHOTS**

# Better, Faster Application Performance Snapshot

## Intel® VTune™ Amplifier – Performance Profiler

### Better Answers

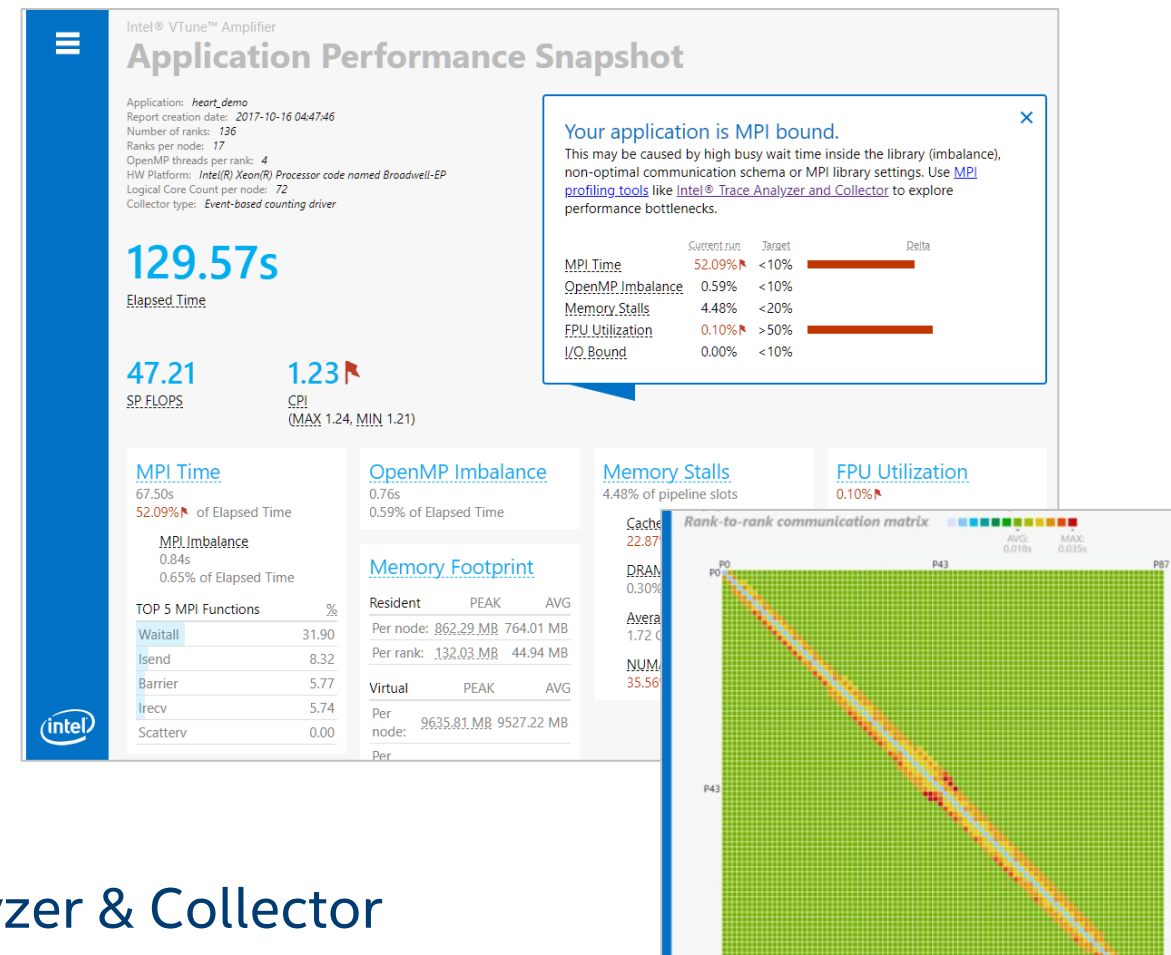
- CPU utilization analysis of physical cores

### Less Overhead

- Lower MPI trace overhead & faster result processing
- New data selection & pause/resume let you focus on useful data

### Easier to Use

- Visualize rank-to-rank & node-to-node MPI communications
- Easily configure profiling for Intel® Trace Analyzer & Collector



Free Download: [intel.com/performance-snapshot](https://intel.com/performance-snapshot)

#### Optimization Notice

Copyright © 2019, Intel Corporation. All rights reserved.

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





# Application Performance Snapshot

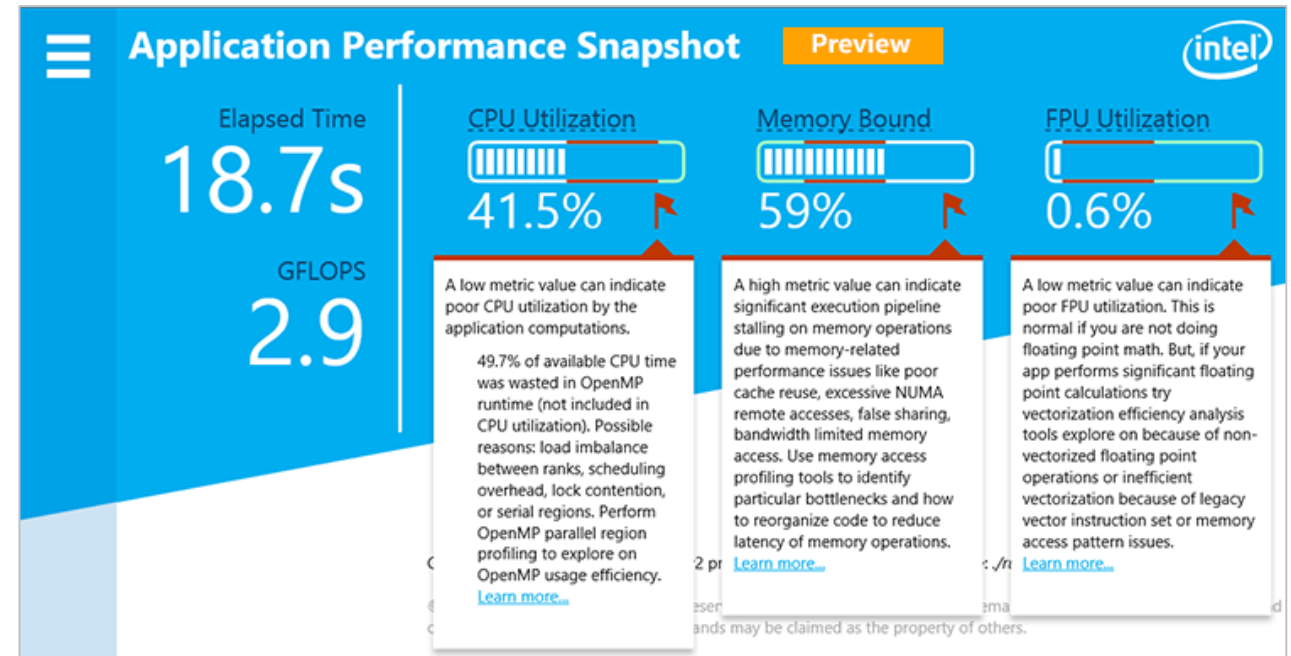
Discover opportunities for better performance with vectorization & threading

## Objectives

- Simple enough to run during a coffee break
- Highlight where code modernization can help

## Users

- Performance teams – fast prioritization of which apps will benefit most
- All Developers – size the potential performance gain from code modernization



## Non-Objectives

- Actionable tuning data – that is another tool. Snapshot is just a fast “health” check.

**Free download:** <http://www.intel.com/performance-snapshot>

Also included with Intel® Parallel Studio and Intel® VTune™ Amplifier products.

### Optimization Notice

Copyright © 2019, Intel Corporation. All rights reserved.

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



# Storage Performance Snapshot

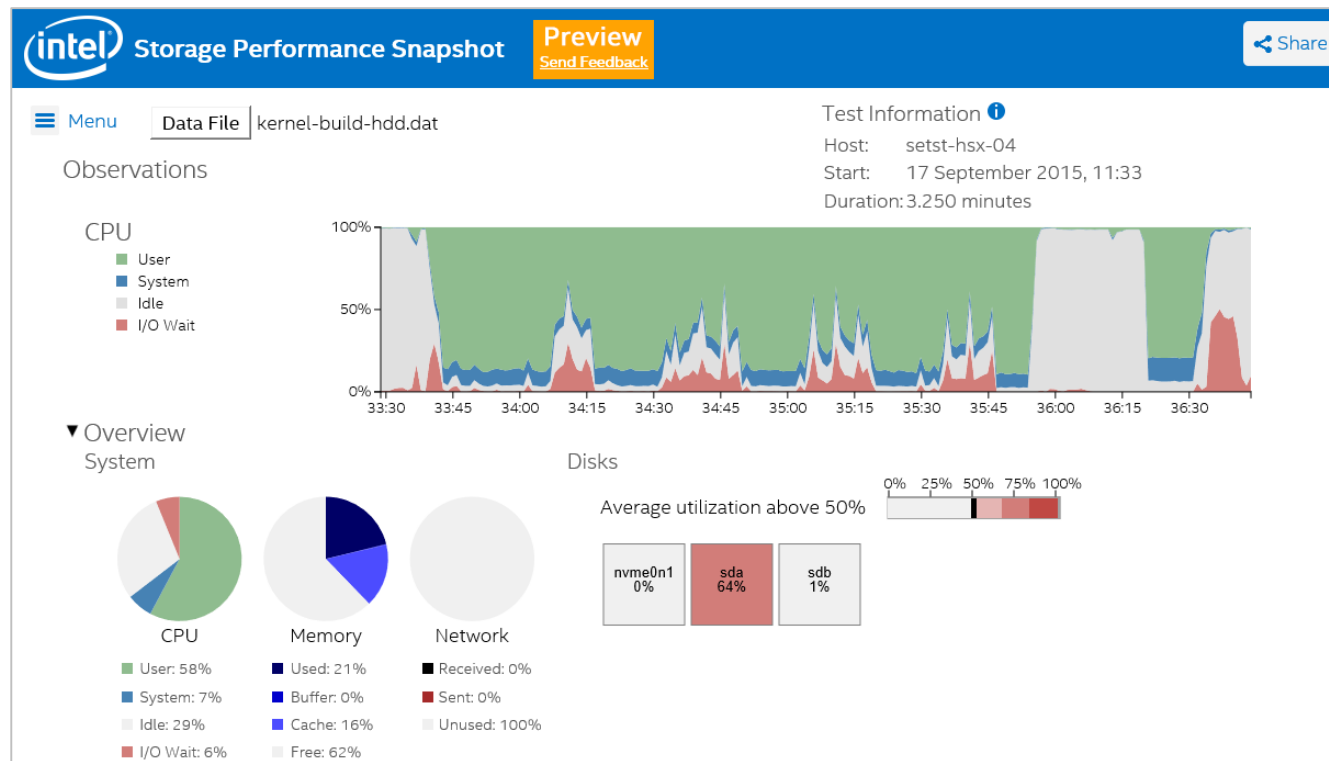
Discover if faster storage can improve server/workstation performance

## Learn It On One Coffee Break

- Easy setup
- Quickly see meaningful data
- System view of workload
- Any architecture

## Targeted Systems

- Servers & workstations with directly attached storage
- Not scale out storage clusters
- Linux kernel 2.6 or newer  
dstat 0.7 or newer
- Windows Server 2012, Windows 8  
or newer Windows OS



**Free download:** <http://www.intel.com/performance-snapshot>

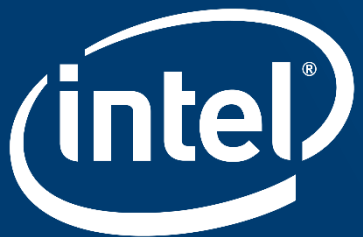
Also included with Intel® Parallel Studio and Intel® VTune™ Amplifier products.

### Optimization Notice

Copyright © 2019, Intel Corporation. All rights reserved.

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





# **VTUNE™ AMPLIFIER – PLATFORM PROFILER**

# Intel® VTune Amplifier – Platform Profiler

## Tune Workloads & System Configuration

### Finds

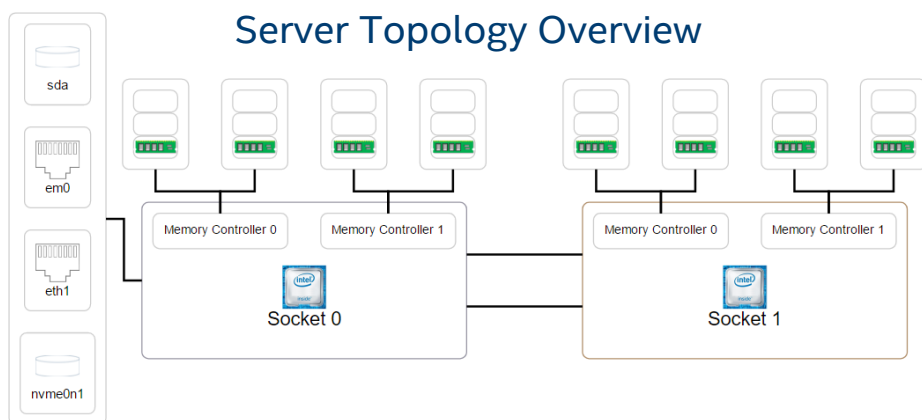
- Configuration issues
- Poorly tuned software

### Target Users

- Infrastructure Architects
- Software Architects & QA

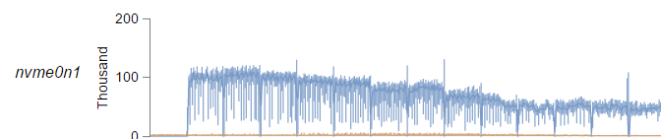
### Performance Metrics

- Extended capture (minutes to hours)
- Low overhead – coarse grain metrics
- Sampling OS & hardware performance counters
- RESTful API for easy analysis by scripts

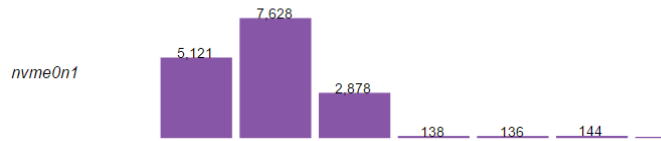


### Timelines & Histograms

IOPS

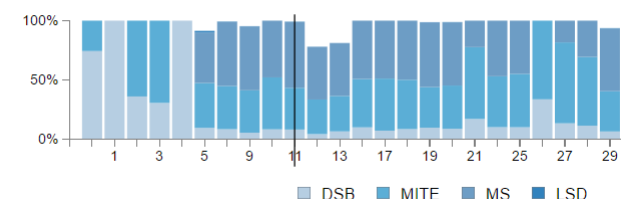


Queue Depth Distribution

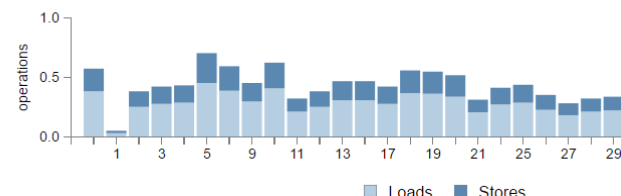


### Core to Core Comparisons

uOPS Delivered (average/core)



Memory Ops Per Instruction (average/core)



### Optimization Notice

Copyright © 2019, Intel Corporation. All rights reserved.

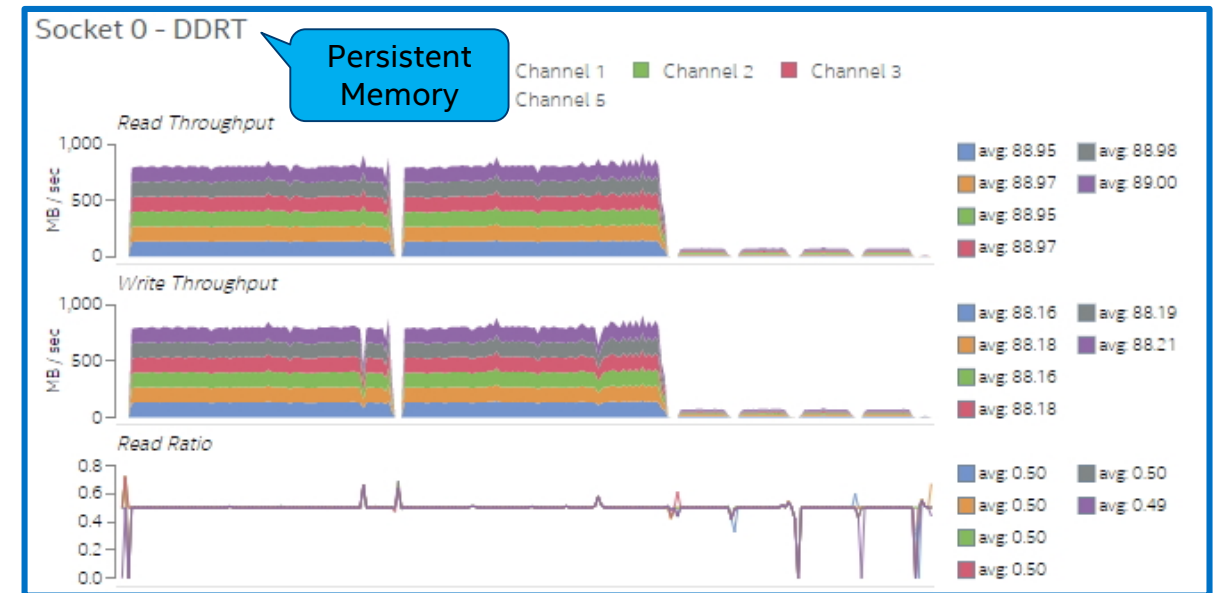
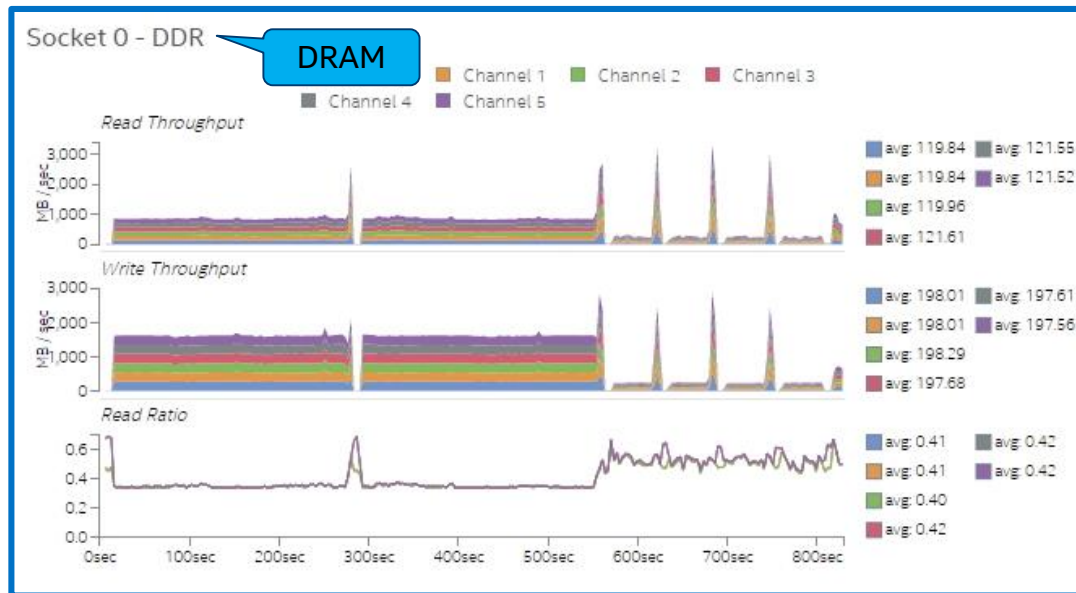
\*Other names and brands may be claimed as the property of others.



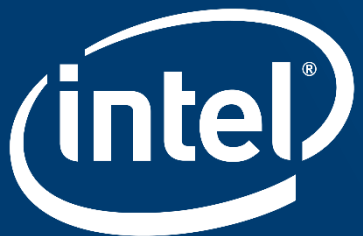


# Tune Persistent Memory Systems

Memory mode systems: Ensure DRAM bandwidth > persistent memory bandwidth



This ratio should be small



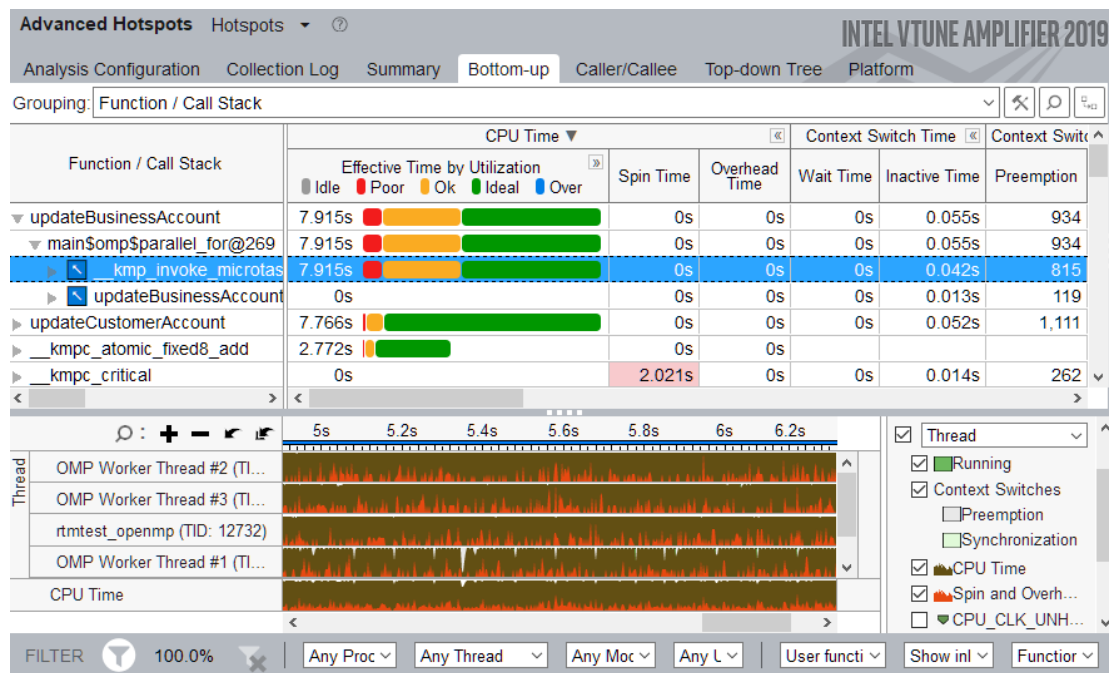
# VTUNE™ AMPLIFIER – CODE ANALYSES

# Analyze & Tune Application Performance

## Intel® VTune™ Amplifier—Performance Profiler

### Save Time Optimizing Code

- Accurately profile C, C++, Fortran\*, Python\*, Go\*, Java\*, or any mix
- Optimize CPU, threading, memory, cache, storage
- Save time: rich analysis leads to insight
- Take advantage of [Priority Support](#)
  - Connects customers to Intel engineers for confidential inquiries (paid versions)



### What's New in 2019 Release (partial list)

- New Platform Profiler! - Longer Data Collection
- A more accessible user interface provides a simplified profiling workflow
- Smarter, faster Application Performance Snapshot: Analyze CPU utilization of physical cores, pause/resume, more... (Linux\*)
- Improved JIT profiling for server-side/cloud applications
- SPDK and DPDK I/O analysis - measure "empty" polling cycles
- New Platform Profiler! - Longer Data Collection
- A more accessible user interface provides a simplified profiling workflow
- Additional embedded OSs and environments

Learn More: [software.intel.com/intel-vtune-amplifier-xe](https://software.intel.com/intel-vtune-amplifier-xe)

#### Optimization Notice

Copyright © 2019, Intel Corporation. All rights reserved.

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



# What's New for 2019?

## Intel® VTune Amplifier Performance Profiler

New, Simplified Setup and More Intelligible Results

New Platform Profiler - Longer Data Collection

- Find hardware configuration issues
- Identify poorly tuned applications

Smarter, Faster Application Performance Snapshot

- Smarter: CPU utilization analysis of physical cores
- Faster: Lower overhead, data selection, pause/resume

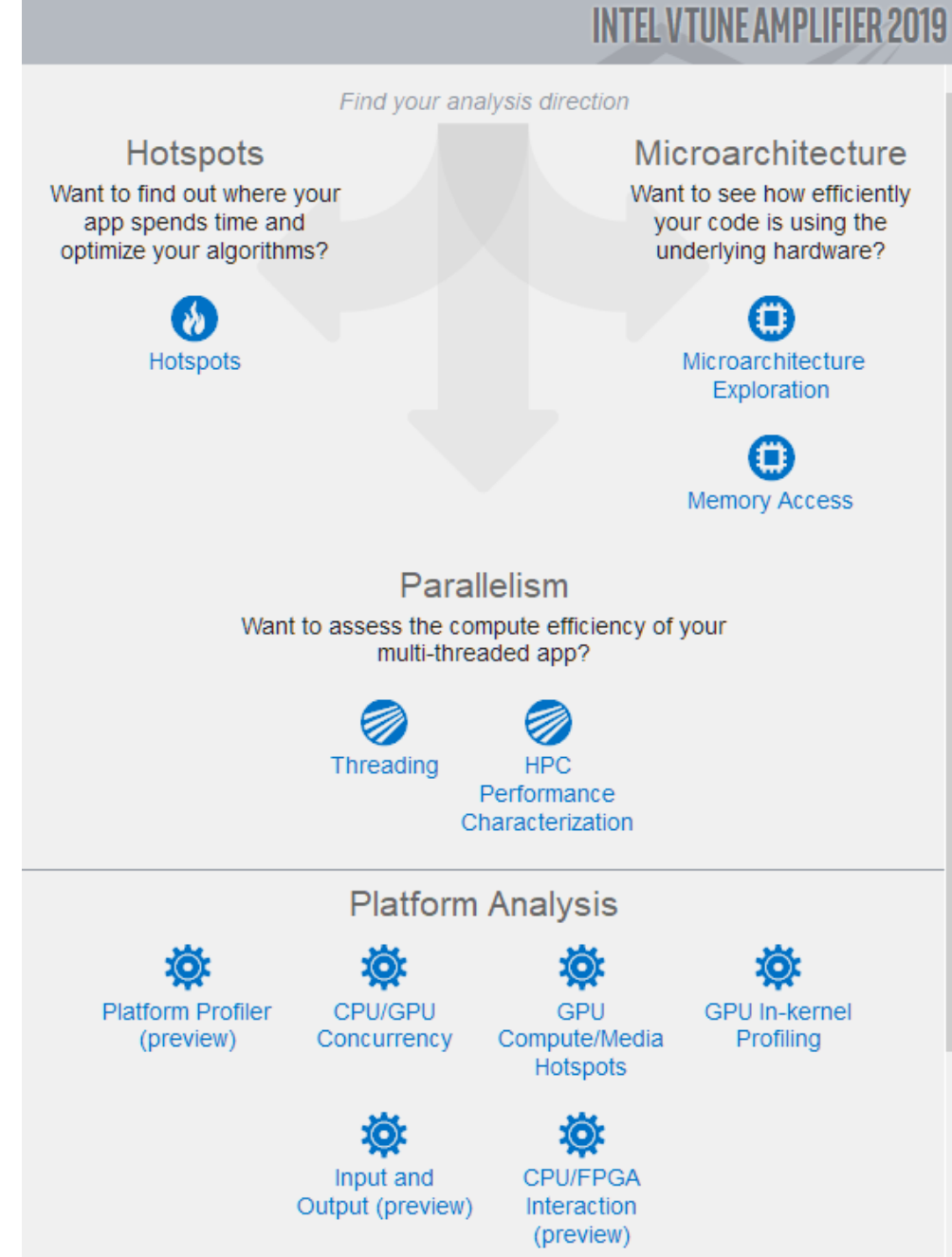
Added Cloud, Container & Linux .NET Support

- JIT profiling on LLVM\* or HHVM PHP servers
- Java\* analysis on OpenJDK 9 and Oracle\* JDK 9
- .NET support on Linux\* plus Hyper-V\* support

SPDK and DPDK I/O Analysis - Measure "Empty" Polling Cycles

Balance CPU/FPGA Loading

Additional Embedded OSs & Environments



### Optimization Notice

Copyright © 2019, Intel Corporation. All rights reserved.

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

# Easier Setup, More Intelligible Results

## Intel® VTune™ Amplifier – Performance Profiler

### Fresh, Accessible Analysis Setup

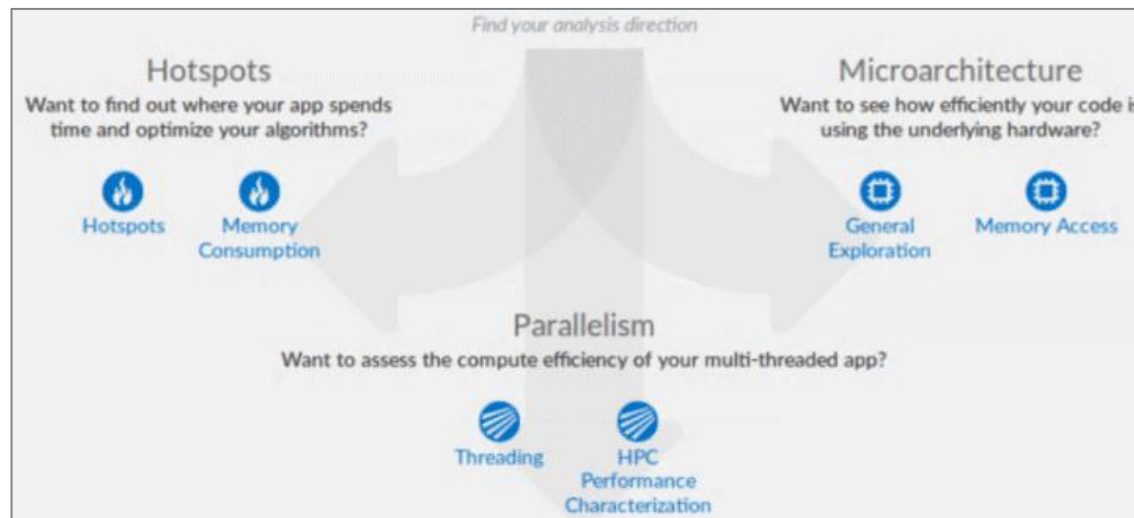
- Simplified workflow
- More familiar terminology
- More logical groupings

### Performance Insights

- Suggestions for further analysis

### Improved Displays

- New hardware pipeline display



#### Hotspots Insights

If you see significant hotspots in the Top Hotspots list, switch to the [Bottom-up](#) view for in-depth analysis per function. Otherwise, use the [Caller/Callee](#) view to track critical paths for these hotspots.

#### Explore Additional Insights

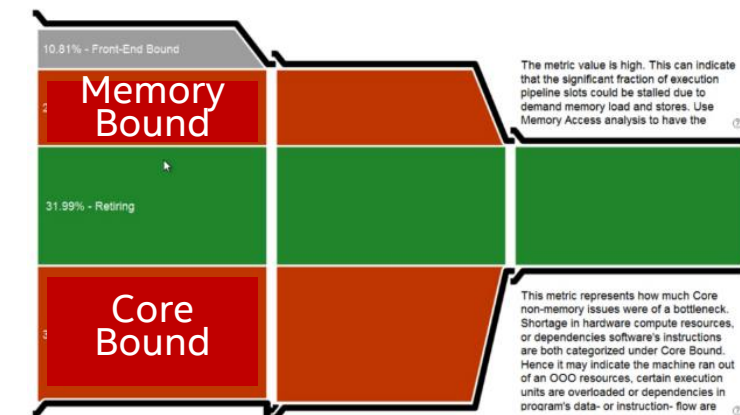
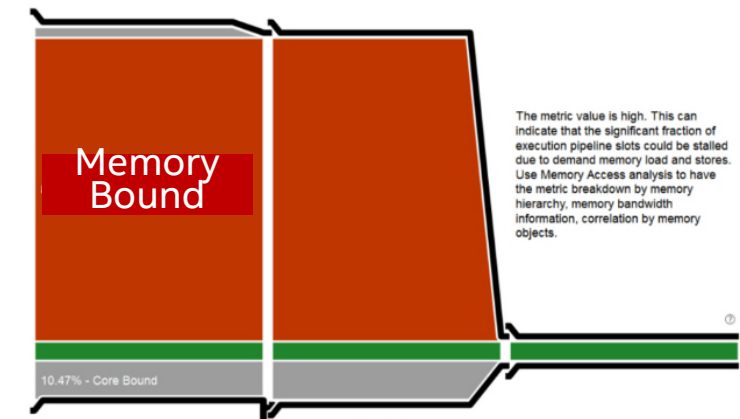
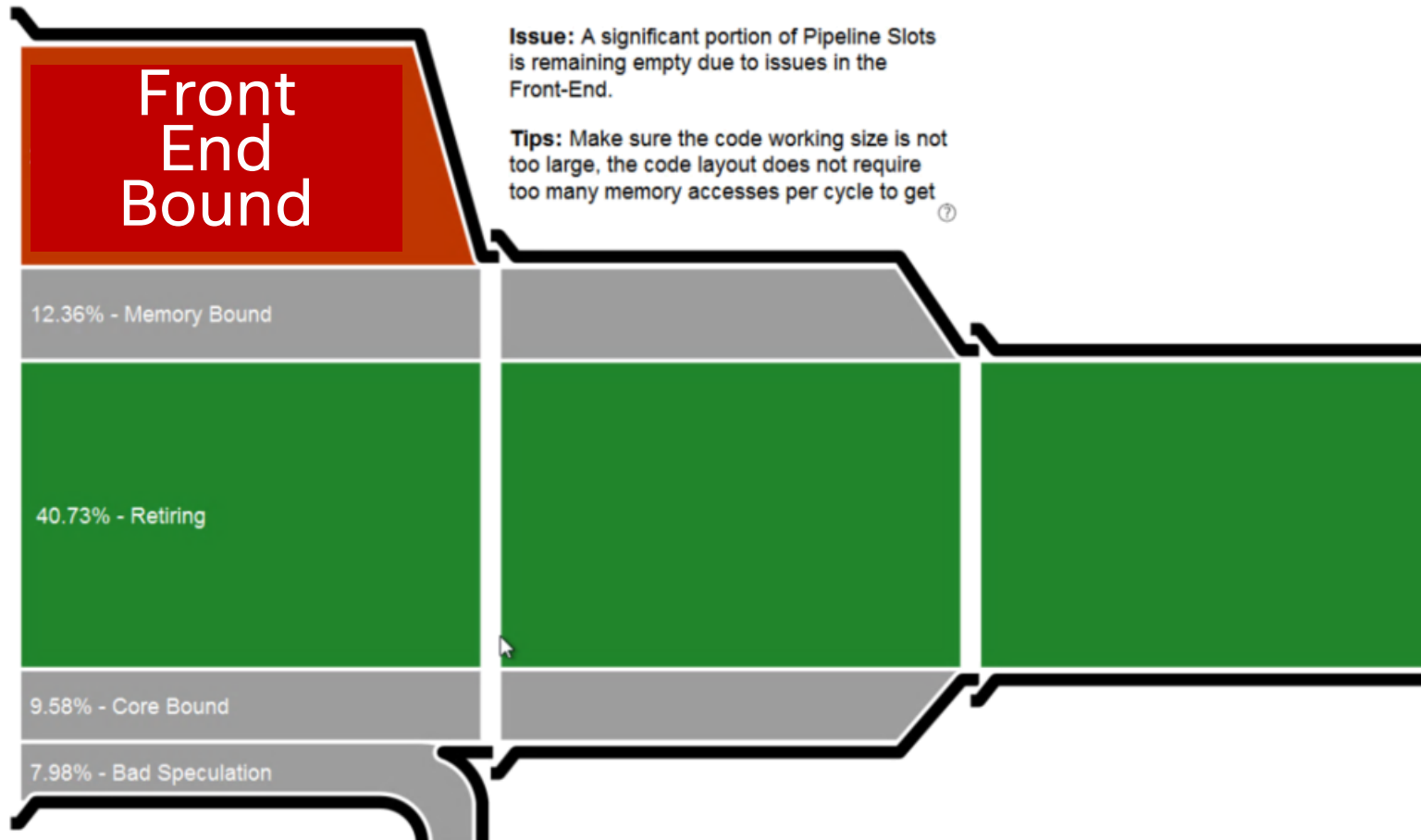
Parallelism ⓘ : 17.8% (15.622 out of 88 logical CPUs) 📈

Use ⓘ [Concurrency](#) to explore more opportunities to increase parallelism in your application.



# Visualize the Micro-Architectural Bottleneck

## Intel® VTune™ Amplifier – Performance Profiler



### Optimization Notice

Copyright © 2019, Intel Corporation. All rights reserved.

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

# More HPC Performance Data

## Intel® VTune™ Amplifier – Performance Profiler

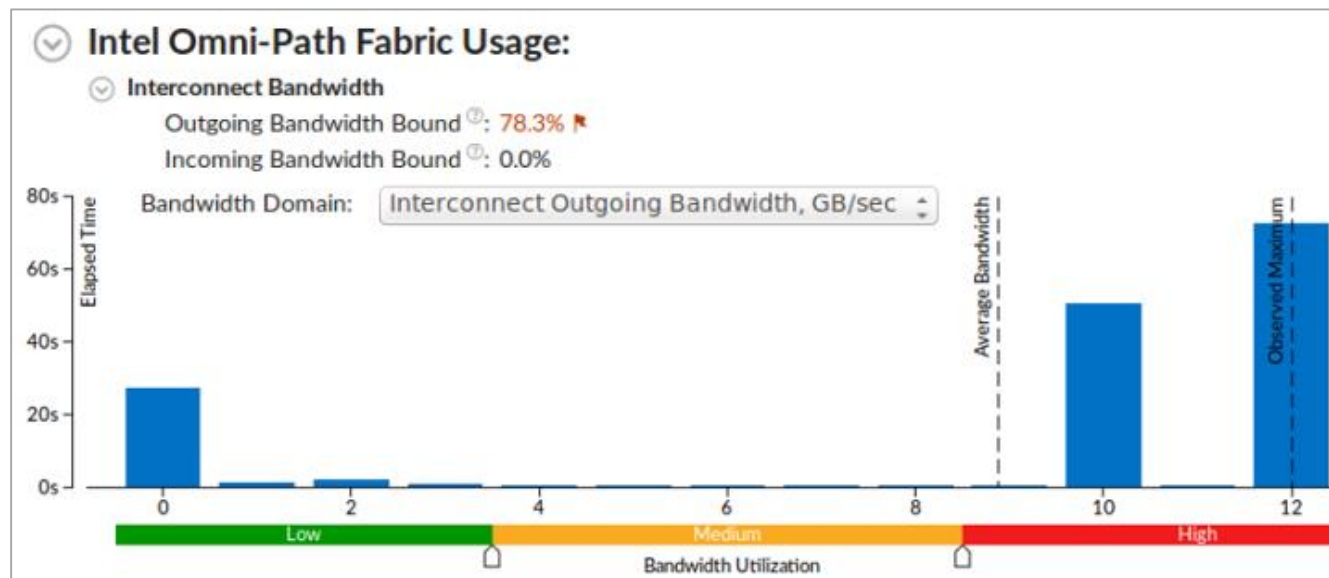
### Thread Affinity Report

- Thread pinning per socket, physical and logical core
- Compliment to CPU utilization & NUMA analysis

Process/Thread	Affinity
-----	-----
./heart_demo (rank 0)	
OMP Master Thread #0 (TID: 109656	S0:C0:L0
OMP Worker Thread #1 (TID: 109672	S0:C0:L44
OMP Worker Thread #2 (TID: 109674	S0:C1:L1
OMP Worker Thread #3 (TID: 109676	S0:C1:L45
./heart_demo (rank 1)	
OMP Master Thread #0 (TID: 109502	S1:C0:L22
OMP Worker Thread #1 (TID: 109664	S1:C0:L66
OMP Worker Thread #2 (TID: 109666	S1:C1:L23
OMP Worker Thread #3 (TID: 109668	S1:C1:L67

### Intel® Omni-Path Fabric Metrics

- Bandwidth & Packet Rate histogram & overtime display to identify performance bottlenecks caused by interconnect limits



# Tune SPDK & DPDK I/O

Measure “Empty” Polling Cycles

Gather Key I/O Metrics – See:

- How each device performs
- I/O imbalance among SSDs
- Throughput distribution per device
- Thread activity colored by throughput
- PCIe traffic breakdown per device
- What causes I/O communication to drop
- Socket interconnect traffic

SPDK & DPDK Use Polling, Not Interrupts

- CPU always 100%, even if unloaded
- Intel® VTune™ Amplifier identifies “empty” spinning

## I/O Summary

### SPDK Info

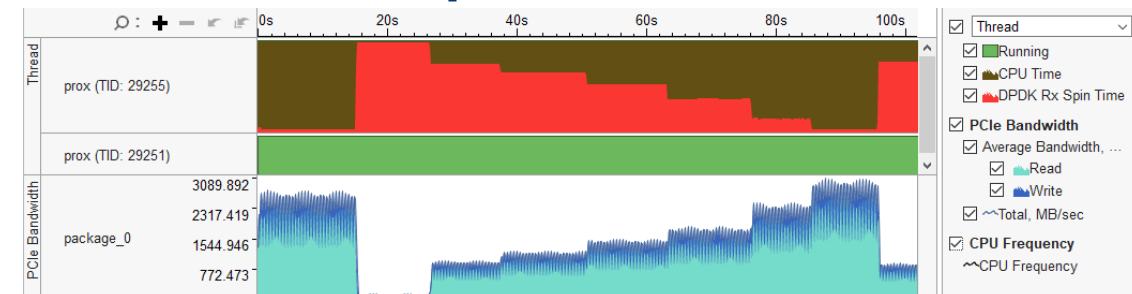
Reads:	3,595,310
Read Bytes:	28095.9 MB
Writes:	3,595,563
Written Bytes:	28090.3 MB
SPDK Effective Time <sup>②</sup> :	66.403s

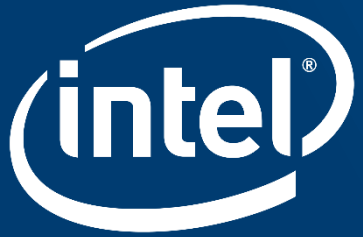
## See Device Imbalance

### SPDK Info

Reads:	3,595,310
bdev_Nvme0n1_0x55b977f9fe50:	2,012,685
bdev_Nvme1n1_0x55b977f9aa40:	1,582,481
bdev_Nvme2n1_0x55b9780a50d0:	73
bdev_Nvme3n1_0x55b9780a52d0:	71
Read Bytes:	28095.9 MB
bdev_Nvme0n1_0x55b977f9fe50:	15727.2 MB
bdev_Nvme1n1_0x55b977f9aa40:	12365.6 MB
bdev_Nvme2n1_0x55b9780a50d0:	1.60547 MB
bdev_Nvme3n1_0x55b9780a52d0:	1.47656 MB

## See spin time (red)





# ADVISOR

THREADING AND VECTORIZATION ASSISTANT

# “Automatic” Vectorization Often Not Enough

A good compiler can still benefit greatly from vectorization optimization

## Compiler will not always vectorize

- Check for Loop Carried Dependencies using [Intel® Advisor](#)
- All clear? Force vectorization.  
C++ use: `pragma simd`, Fortran use: `SIMD` directive

## Not all vectorization is efficient vectorization

- Stride of 1 is more cache efficient than stride of 2 and greater. Analyze with [Intel® Advisor](#).
- Consider data layout changes  
[Intel® SIMD Data Layout Templates](#) can help

Benchmarks on prior slides did not all “auto vectorize.” Compiler directives were used to force vectorization and get more performance.

Arrays of structures are great for intuitively organizing data, but are much less efficient than structures of arrays. Use the [Intel® SIMD Data Layout Templates](#) (Intel® SDLT) to map data into a more efficient layout for vectorization.



# Get Breakthrough Vectorization Performance

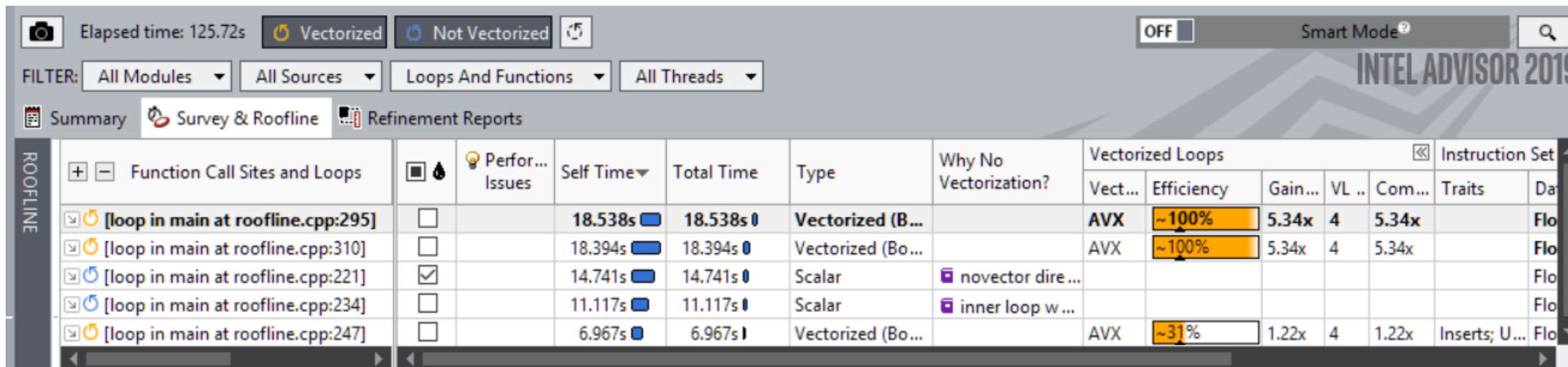
Intel® Advisor—Vectorization Advisor

## Faster Vectorization Optimization

- Vectorize where it will pay off most
- Quickly ID what is blocking vectorization
- Tips for effective vectorization
- Safely force compiler vectorization
- Optimize memory stride

## Data & Guidance You Need

- Compiler diagnostics + Performance Data + SIMD efficiency
- Detect problems & recommend fixes
- Loop-Carried Dependency Analysis
- Memory Access Patterns Analysis



The screenshot shows the Intel Advisor 2019 Vectorization Advisor interface. At the top, there are filters for 'All Modules', 'All Sources', 'Loops And Functions', and 'All Threads'. Below these are tabs for 'Summary', 'Survey & Roofline', and 'Refinement Reports'. The main table displays a list of function call sites and loops, including their self time, total time, type, and why no vectorization was applied. The table also shows vectorized loops with their efficiency, gain, and instruction set.

Function Call Sites and Loops	Perfor... Issues	Self Time	Total Time	Type	Why No Vectorization?	Vectorized Loops	Instruction Set
						Vect... Efficiency Gain... VL .. Com... Traits Da	
[loop in main at roofline.cpp:295]	<input type="checkbox"/>	18.538s	18.538s	Vectorized (B...		AVX ~100% 5.34x 4 5.34x	Flo
[loop in main at roofline.cpp:310]	<input type="checkbox"/>	18.394s	18.394s	Vectorized (Bo...		AVX ~100% 5.34x 4 5.34x	Flo
[loop in main at roofline.cpp:221]	<input checked="" type="checkbox"/>	14.741s	14.741s	Scalar	novector dire...		Flo
[loop in main at roofline.cpp:234]	<input type="checkbox"/>	11.117s	11.117s	Scalar	inner loop w...		Flo
[loop in main at roofline.cpp:247]	<input type="checkbox"/>	6.967s	6.967s	Vectorized (Bo...		AVX ~31% 1.22x 4 1.22x Inserts; U...	Flo

Optimize for Intel® AVX-512 with or without access to AVX-512 hardware

<http://intel.ly/advisor-xe>

### Optimization Notice

Copyright © 2019, Intel Corporation. All rights reserved.

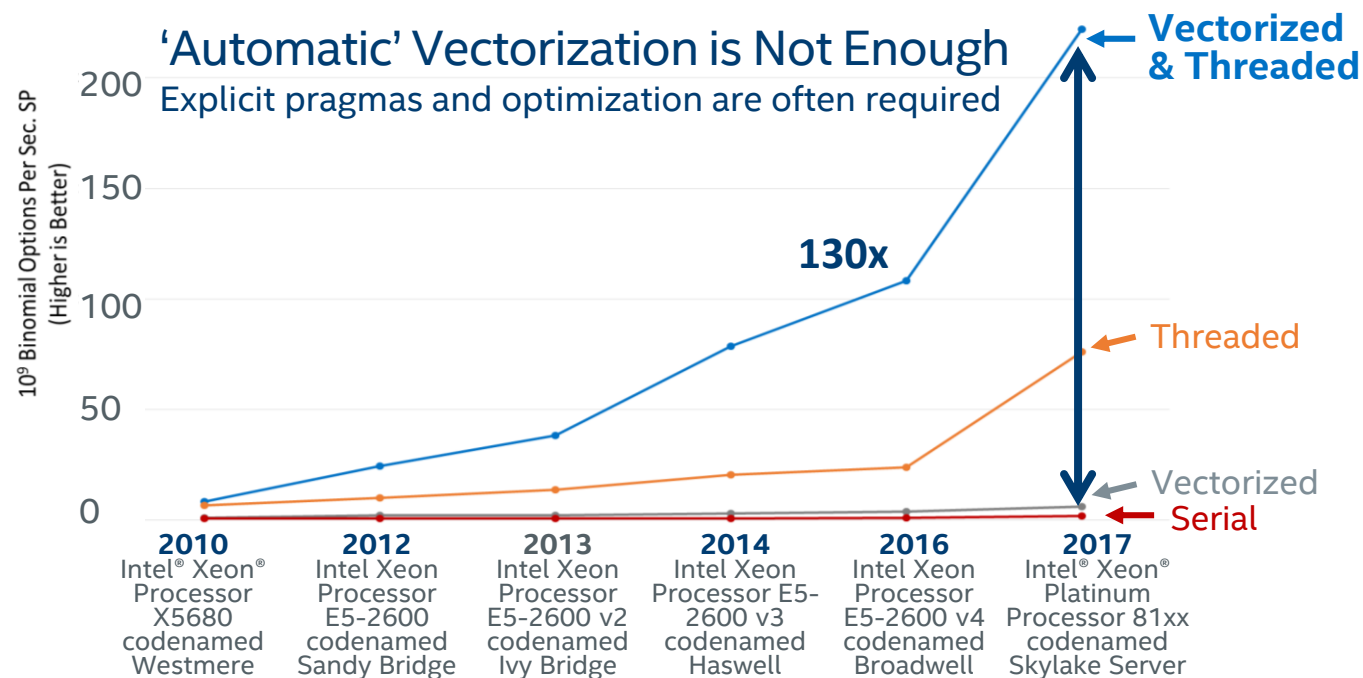
\*Other names and brands may be claimed as the property of others.



# Modernize Your Code with Intel® Advisor

Optimize Vectorization, Prototype Threading, Create and Analyze Flow Graphs

The Difference is Growing with Each New Hardware Generation



## Modern Performant Code

- Vectorized (uses Intel® AVX-512/AVX2)
- Efficient memory access
- Threaded

## Intel® Advisor

- Adds & optimizes vectorization
- Analyzes memory patterns
- Quickly prototypes threading

## New for 2019 Release (partial list)

- Enhanced hierarchical roofline analysis
- Shareable HTML roofline
- Flow graph analysis

Benchmark: Binomial Options Pricing Model <https://software.intel.com/en-us/articles/binomial-options-pricing-model-code-for-intel-xeon-phi-coprocessor>

Benchmark results were obtained prior to implementation of recent software patches and firmware updates intended to address exploits referred to as "Spectre" and "Meltdown". Implementation of these updates may make these results inapplicable to your device or system. For more complete information about performance and benchmark results, visit [www.intel.com/benchmarks](http://www.intel.com/benchmarks). See [Vectorize & Thread](#) or [Performance Dies Configurations for 2010-2017 Benchmarks](#) in Backup. Benchmark source: Intel Corporation.

Learn More: <http://intel.ly/advisor-xe>

### Optimization Notice

Copyright © 2019, Intel Corporation. All rights reserved.

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

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, & SSSE3 instruction sets & 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. Notice Revision #20110804

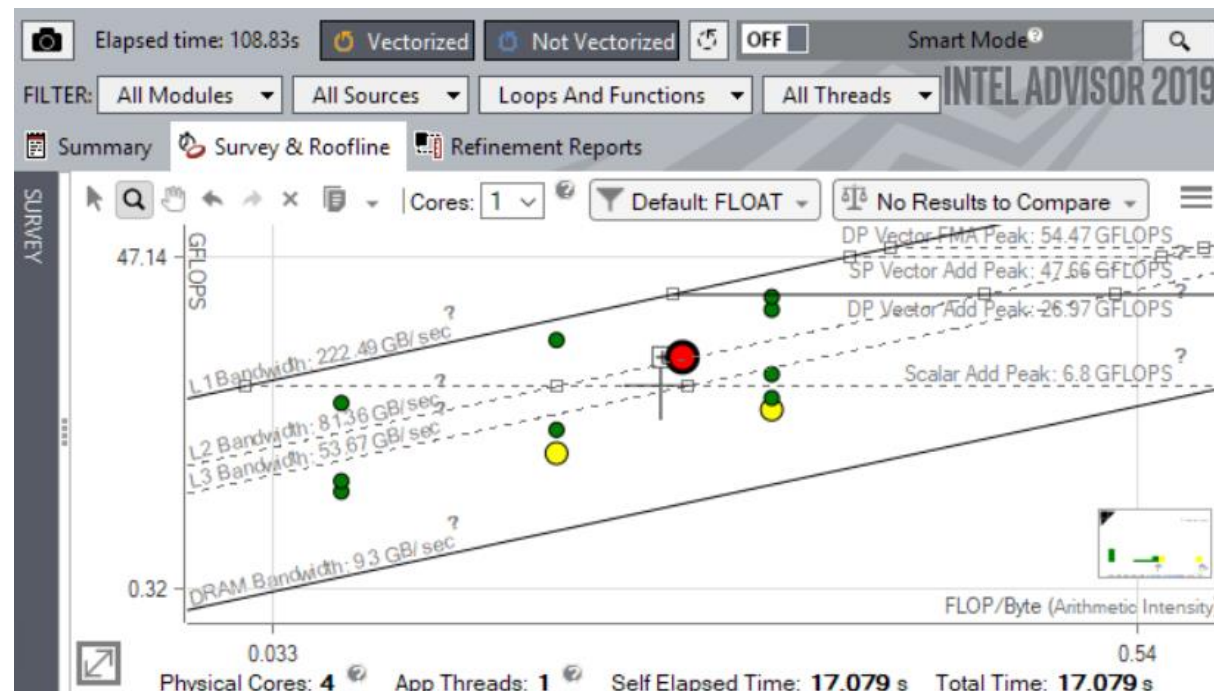


# Find Effective Optimization Strategies

Intel® Advisor—Cache-aware Roofline Analysis

## Roofline Performance Insights

- Highlights poor performing loops
- Shows performance 'headroom' for each loop
  - Which can be improved
  - Which are worth improving
- Shows likely causes of bottlenecks
- Suggests next optimization steps



*"I am enthusiastic about the new "integrated roofline" in Intel® Advisor. It is now possible to proceed with a step-by-step approach with the difficult question of memory transfers optimization & vectorization which is of major importance."*

Nicolas Alferez, Software Architect  
Onera – The French Aerospace Lab

# Design It, Tune, Debug, Then Implement

Intel® Advisor Thread Prototyping—Design without Disrupting Development

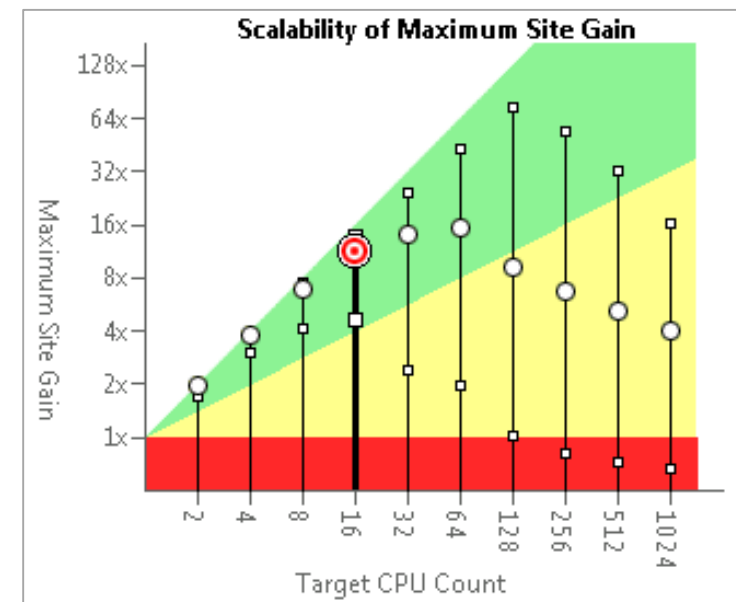
## Have You

- Threaded an app, but seen little benefit?
- Hit a “scalability barrier?”
- Delayed release due to synchronization errors?

## Data Driven Threading Design

- Quickly prototype multiple options
- Project scaling on larger systems
- Find synchronization errors before implementing threading
- Design without disrupting development

**Add Parallelism with Less Effort, Less Risk & More Impact**



*“Intel® Advisor allowed us to quickly prototype ideas for parallelism, saving developer time & effort”*

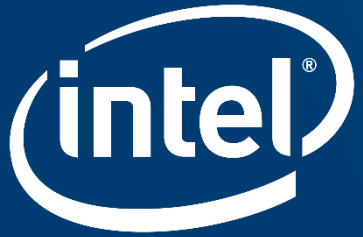
*Simon Hammond  
Senior Technical Staff  
Sandia National Laboratories*

### Optimization Notice

Copyright © 2019, Intel Corporation. All rights reserved.

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





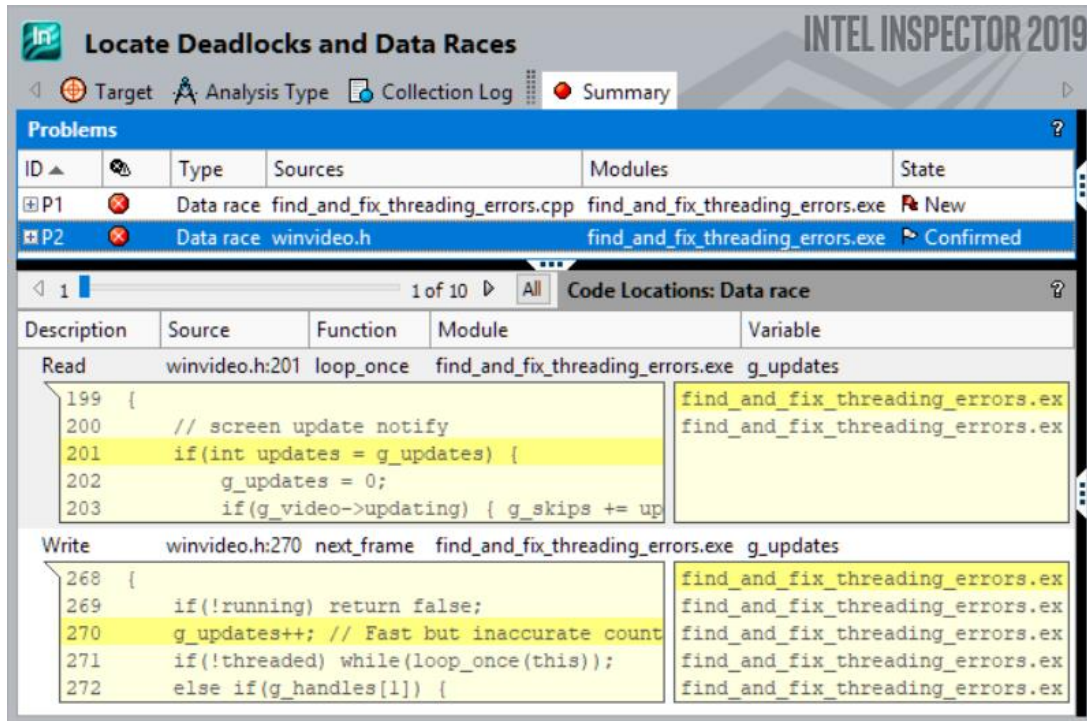
# INSPECTOR

MEMORY AND THREADING ERROR CHECKING



# Debug Memory & Threading with Intel® Inspector

Find & Debug Memory Leaks, Corruption, Data Races, Deadlocks



## Correctness Tools Increase ROI by 12%-21%<sup>1</sup>

- Errors found earlier are less expensive to fix
- Races & deadlocks not easily reproduced
- Memory errors are hard to find without a tool

## Debugger Integration Speeds Diagnosis

- Breakpoint set just before the problem
- Examine variables and threads with the debugger

## What's New in 2019 Release

### Find Persistent Memory Errors

- Missing / redundant cache flushes
- Missing store fences
- Out-of-order persistent memory stores
- PMDK transaction redo logging errors

Learn More: [bit.ly/intel-inspector](https://bit.ly/intel-inspector)

Source: Intel Security, Intel Security Readiness Team, and Carnegie Mellon CyLab NIST: National Institute of Standards & Technology: Square Project Results

### Optimization Notice

Copyright © 2017, Intel Corporation. All rights reserved.

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



# Key Takeaways

- #1 Performance is many things/roles/workflows
- #2 Require a toolbox filled with many specialized tools
- #3 Intel offers such a toolbox and we continue to add to it



- #4 We like to hear about your performance practices



THANK YOU  
谢谢

**Optimization Notice**

Copyright © 2019, Intel Corporation. All rights reserved.

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

