

Intel® Threading Building Blocks 3.0

In-Depth

Contents

Intel® Threading Building Blocks 3.0	3
Features	4
What's New in Intel TBB 3.0	5
Technical Support	6
Which Intel TBB License Is Right for Your Needs?	6

Intel® Threading Building Blocks (Intel® TBB) 3.0

Everything You Expect in Parallel Application Design—Portability, Scalability, Reliability, Simplicity

Intel® Threading Building Blocks 3.0 (Intel® TBB) is an award-winning C++ template library that abstracts threads to tasks to create reliable, portable, and scalable parallel applications. Use Intel TBB for a simple and rapid way of developing robust task-based parallel applications that scale to available processor cores, are compatible with multiple environments, and easier to maintain code. Intel TBB is the most proficient way to design future-proof parallel applications that tap into the power and performance of multicore and manycore hardware platforms.

Enhanced Productivity—Intel TBB uses task-based abstractions that make it easier to get scalable and reliable parallel applications with fewer lines of code. Task-based algorithms, containers, and synchronization primitives simplify parallel application development.

Portability—Intel TBB is validated and commercially supported on Windows*, Linux*, and Mac OS* platforms. It is also available on FreeBSD*, IA Solaris*, Xbox* 360, and PowerPC-based systems via the open source community. Organizations can expand their customer base by using a production-ready, open solution for parallelism that is available on a broad range of platforms.

Comprehensive—Intel TBB provides a suite of components that help simplify and speed development of scalable, high performance parallel applications. It provides parallel algorithms, concurrent containers, task scheduling, scalable memory allocation, and synchronization primitives. In addition, the runtime library provides an optimal-size thread pool; task granularity and performance-oriented scheduling; automatic load balancing through task stealing; and cache efficiency and memory reuse.

Future-proof Applications—Application performance automatically improves as processor core count increases by using abstract tasks. Sophisticated task scheduler dynamically maps tasks to threads to balance the load among available cores, preserve cache locality, and maximize parallel performance.

Performance Advantage—Optimized for scalable multicore architectures including Non-Uniform Memory Access (NUMA), Intel TBB delivers higher-performing and reliable code with less effort than hand threading.

Intel® Threading Building Blocks v3.0		
Generic Parallel Algorithms <ul style="list-style-type: none"> ▪ parallel_for(range) ▪ parallel_reduce ▪ parallel_for_each(begin, end) ▪ parallel_do ▪ parallel_invoke ▪ pipeline ▪ parallel_pipeline ▪ parallel_sort ▪ parallel_scan 	Concurrent Containers <ul style="list-style-type: none"> ▪ concurrent_hash_map ▪ concurrent_queue ▪ concurrent_bounded_queue ▪ concurrent_vector ▪ concurrent_unordered_map 	Task Scheduler <ul style="list-style-type: none"> ▪ task_group ▪ structured_task_group ▪ task_scheduler_init ▪ task_scheduler_observer
Synchronization Primitives <ul style="list-style-type: none"> ▪ atomic ▪ mutex ▪ recursive_mutex ▪ spin_mutex ▪ spin_rw_mutex ▪ queuing_mutex 	<ul style="list-style-type: none"> ▪ queuing_rw_mutex ▪ reader_writer_lock ▪ critical_section ▪ condition_variable ▪ null_mutex ▪ null_rw_mutex 	Memory Allocation <ul style="list-style-type: none"> ▪ tbb_allocator ▪ cache_aligned_allocator ▪ scalable_allocator ▪ zero_allocator
Thread Local Storage <ul style="list-style-type: none"> ▪ enumerable_thread_specific ▪ combinable 	Threads <ul style="list-style-type: none"> ▪ thread 	Miscellaneous <ul style="list-style-type: none"> ▪ tick_count ▪ captured_exception ▪ moveable_exception

Figure 1. Intel TBB offers comprehensive, abstracted templates, containers, and classes for parallelism. Version 3.0 expands usage models and improves performance and usability. Italic text in table means “new function”.

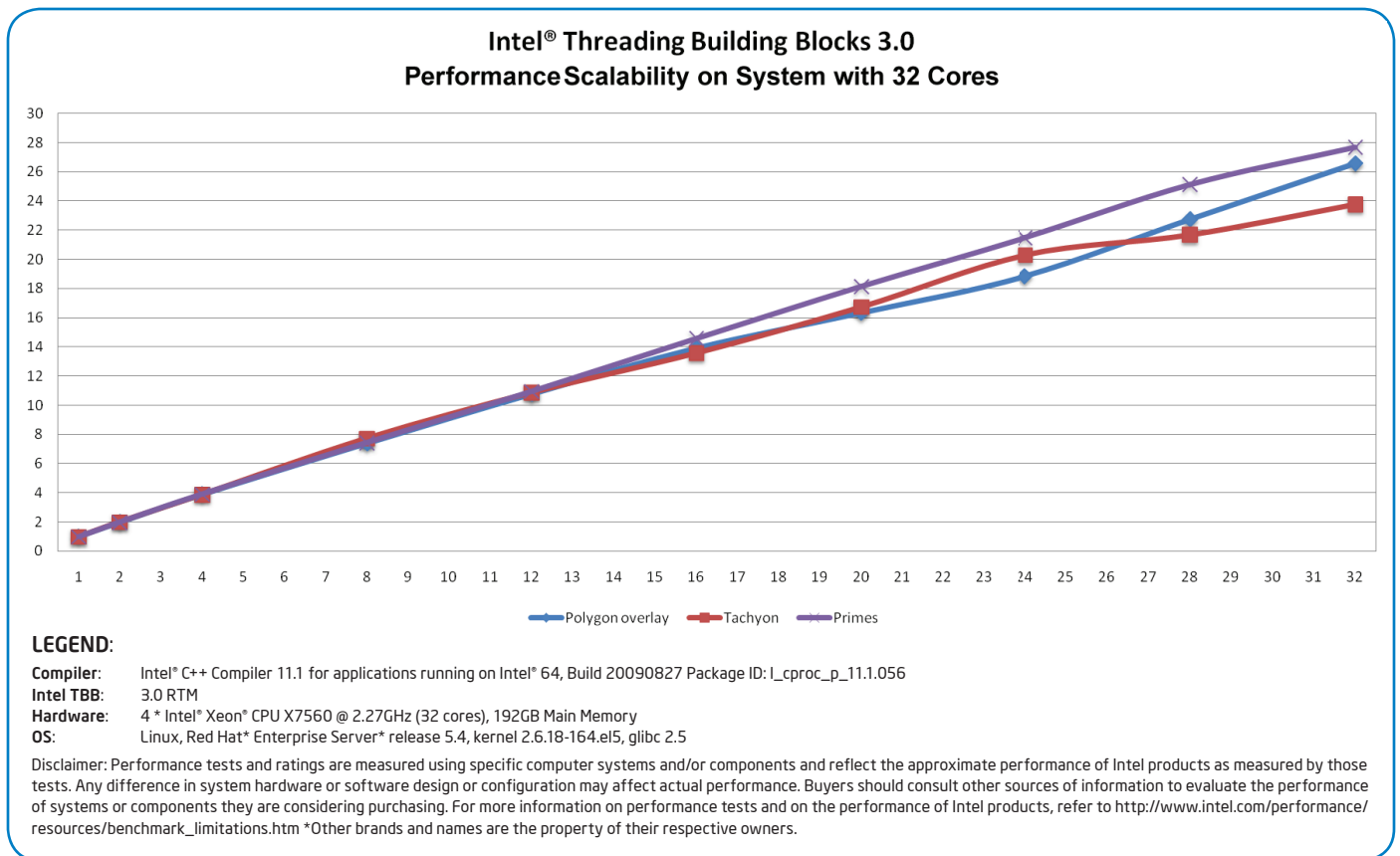


Figure 3. Excellent scalability and improved performance using Intel TBB versus a serial implementation. Windows and Mac OS X developers can expect similar results.

What's New in Intel TBB 3.0—More Compatibility, Performance, Tasks, and Classes

Extended Compatibility—More Choices, Interoperability, and Compatibility Support:

- Microsoft Visual Studio 2010* Parallel Patterns Library (PPL) and Concurrency Runtime (ConcRT) compatibility and interoperability support
- Microsoft Windows 7* support
- Apple Mac OS X* Snow Leopard* support

Enhanced Task Scheduler Features—More Predictable and Expanded Task Functions

- Starvation-proof scheduling tasks for queue-like work
- Master Thread Isolation
- Task group context can now be created and destroyed in different threads

Enhanced Parallel Application Performance on Multicore Processors

- Memory Allocator includes additional optimizations for large blocks (>8KB)

Expanded Use Cases, Classes, and Functions—Added Interoperability and Capabilities:

- Expanded lambdas support—improved pipeline, added function parallel_pipeline a strongly typed lambda-friendly interface for building and running pipelines
- New C++ 0x Condition Variable
- New concurrent_unordered_map

Improved Documentation—More Examples

- Added new Design Patterns documentation that articulates usage models and common patterns

"The Maya team has successfully used Intel's TBB technology to internally parallelize Maya for several releases. Now thanks to Intel, TBB lets Maya plug-in developers access the same advanced parallelism features that we've used at no additional charge."

Gordon Bradley,
Maya Performance Team Lead

"Bringing Intel into the Integrated Partners Program and integrating TBB into Unreal Engine 3* ensures our licensees are within reach of the best development tools available," said **Dr. Michael Capps, president of Epic Games**. "This partnership will help minimize the work required to author multithreaded applications for Unreal-powered games and, in the long term, provide developers with greater access to Intel® Tools."

"The Intel® TBB malloc was an important tool in achieving good parallel speedups for our threaded applications, and a drop-in replacement for the memory allocator in the C standard library."

Ron Henderson,
R&D FX Manager
DreamWorks Animation

Technical Support

Intel® Software Development Product purchases may include a year of support services, which provide access to Intel® Premier Support and all product updates during that time. Intel Premier Support gives you online access to technical notes, application notes, and documentation.

Which Intel TBB License Is Right for Your Needs?

Intel TBB is available commercially as a binary distribution, and in open source in both source and binary forms. If you need commercial support services you should purchase either a standalone commercial license or take advantage of the considerable value in purchasing the Intel® Parallel Studio or Intel® Compiler Professional Edition. If your legal counsel is comfortable with your use of software under the Intel TBB open source license and you do not require commercial support services, please download the latest version of open source Intel TBB (threadingbuildingblocks.org). Finally, if you require the ability to modify or distribute the commercial source code of TBB, contact your Intel representative for more information.

When built from source, Intel TBB is intended to be highly portable, and so it supports a wide variety of operating systems and platforms. Binary distributions, including commercial distributions, are validated and officially supported for a variety of select hardware, software, operating systems, and compilers.

