Top Features

- Vectorized and threaded for highest performance on all Intel and compatible processors
- Compatible with all C/C++ and Fortran compilers
- Royalty-free, per developer licensing for low cost deployment

Available in the following suites or standalone:

- Intel® Parallel Studio XE
- Intel® C++ Studio XE
- Intel® Fortran Studio XE
- Intel® Cluster Studio XE
- Intel® Composer XE
- Intel® C++ Composer XE
- Intel® Fortran Composer XE

OS Support:

- Windows*
- Linux*
- OS X*

"Intel MKL is indispensable for any high-performance computer user on x86 platforms."

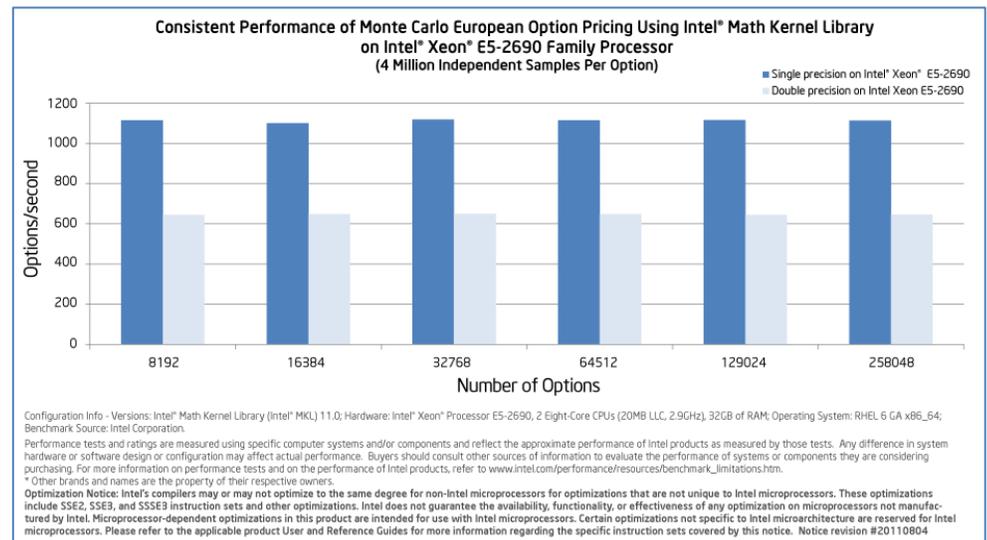Prof. Jack Dongarra, Innovative Computing Lab, University of Tennessee, Knoxville

Performance: Ready to Use

Intel® Math Kernel Library v11.0 (Intel® MKL) includes a wealth of routines to accelerate application performance and reduce development time. Today's processors have increasing core counts, wider vector units and more varied architectures. The easiest way to take advantage of all of that processing power is to use a carefully optimized library designed to harness that potential. Even the best compiler can't compete with the level of performance possible from a hand-optimized library.

Because Intel has done the engineering on these ready-to-use, royalty-free functions, you'll not only have more time to develop new features for your application, but in the long run you'll also save development, debug and maintenance time while knowing that the code you write today will run optimally on future generations of Intel processors.

Intel® MKL includes highly vectorized and threaded Linear Algebra, Fast Fourier Transforms (FFT), Vector Math and Statistics functions. Through a single C or Fortran API call, these functions automatically scale across previous, current and future processor architectures by selecting the best code path for each.

Configuration Info - Versions: Intel® Math Kernel Library (Intel® MKL) 11.0; Hardware: Intel® Xeon® Processor E5-2690, 2 Eight-Core CPUs (20MB LLC, 2.9GHz), 32GB of RAM; Operating System: RHEL 6 GA x86_64; Benchmark Source: Intel Corporation.

Performance tests and ratings are measured using specific computer systems and/or components and reflect the approximate performance of Intel products as measured by those tests. Any difference in system hardware or software design or configuration may affect actual performance. Buyers should consult other sources of information to evaluate the performance of systems or components they are considering purchasing. For more information on performance tests and on the performance of Intel products, refer to www.intel.com/performance/resources/benchmark_limitations.htm.
* Other brands and names are the property of their respective owners.
Optimization Notice: 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. Notice revision #20110804

Intel® MKL delivers industry-leading performance on Monte Carlo and other math-intensive routines

Top Features

Intel® Math Kernel Library Highlighted Features

Included in Intel® Parallel Studio XE

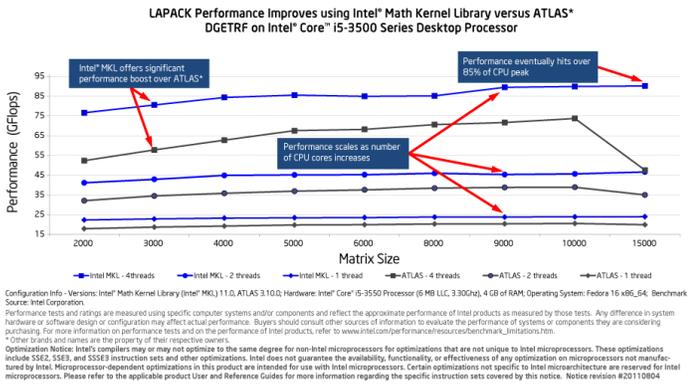| Linear Algebra	Fast Fourier Transforms	Vector Math	Random Number Generators	Summary Statistics	Data Fitting
• BLAS • LAPACK • Sparse Solvers	• Multi-dimensional (up to 7D) • FFTW* Interfaces	• Trigonometric • Hyperbolic • Exponential, Logarithmic • Power / Root • Rounding	• Congruent • Recursive • Wichman-Hill • Mersenne Twister • Sobol • Neiderreiter • RDRAND-based	• Kurtosis • Variation coefficient • Quantiles, order statistics • Min/max • Variance-covariance	• Splines • Interpolation • Cell search
CLUSTER • ScaLAPACK	**CLUSTER** • Cluster FFT				

Optimized Math Operations running on Windows*, Linux* and OS X*
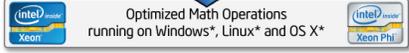
Comprehensive Math Functionality – Covers Range of Application Needs

Intel® MKL contains a wealth of threaded and vectorized complex math functions to accelerate a wide variety of software applications. Why write these functions yourself when Intel has already done the work for you?

Major functional categories include Linear Algebra, Fast Fourier Transforms (FFT), Vector Math and Statistics. Cluster-based versions of LAPACK and FFT are also included to support MPI-based distributed memory computing.
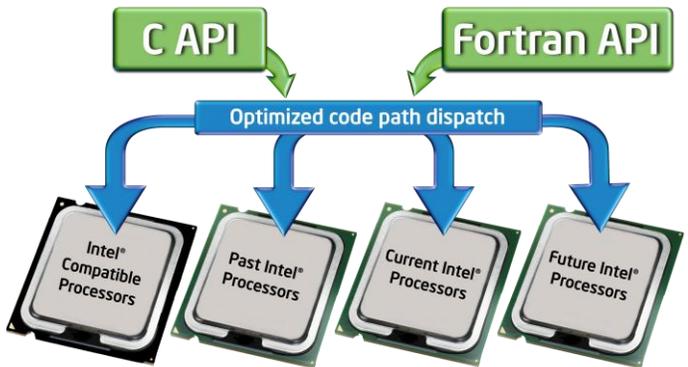

LAPACK Performance Improves using Intel® Math Kernel Library versus ATLAS*
DGETRF on Intel® Core™ i5-3500 Series Desktop Processor

Intel® MKL offers significant performance boost over ATLAS*

Performance eventually hits over 85% of CPU peak

Performance scales as number of CPU cores increases

— Intel MKL - 4 threads — Intel MKL - 2 threads — Intel MKL - 1 thread — ATLAS - 4 threads — ATLAS - 2 threads — ATLAS - 1 thread

Configuration Info - Versions: Intel® Math Kernel Library (Intel® MKL) 11.0, ATLAS 3.10.0; Hardware: Intel® Core™ i5-3550 Processor (6 MB LLC, 3.30Ghz), 4 GB of RAM; Operating System: Fedora 16 x86_64; Benchmark Source: Intel Corporation.
Performance tests and ratings are measured using specific computer systems and/or components and reflect the approximate performance of Intel products as measured by those tests. Any difference in system hardware or software design or configuration may affect actual performance. Buyers should consult other sources of information to evaluate the performance of systems or components they are considering purchasing. For more information on performance tests and on the performance of Intel products, refer to www.intel.com/performance/resources/benchmark_limitations.htm.
* Other brands and names are the property of their respective owners.
Optimization Notice: 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. Notice revision #20110804

Standard APIs – For Immediate Performance Results

Wherever available, Intel® MKL uses de facto industry standard APIs so that minimal code changes are required to switch from another library. This makes it quick and easy to improve your application performance through simple function substitutions or relinking.

Simply substituting Intel® MKL's LAPACK (Linear Algebra PACKage), for example, can yield 500% or higher performance improvement (benchmark left.)

In addition to the industry-standard BLAS and LAPACK linear algebra APIs, Intel® MKL also supports MIT's FFTW C interface for Fast Fourier Transforms.
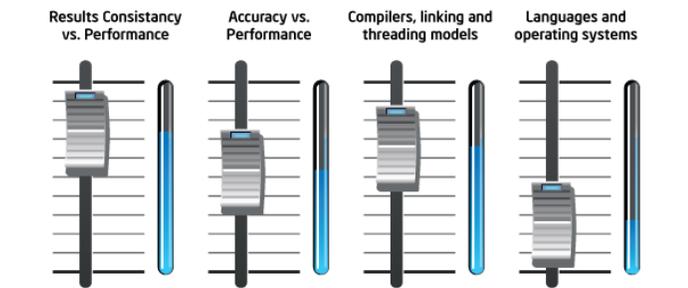

C API

Fortran API

Optimized code path dispatch

Intel® Compatible Processors

Past Intel® Processors

Current Intel® Processors

Future Intel® Processors

Highest Performance and Scalability across Past, Present & Future Processors – Easily and Automatically

Behind a single C or Fortran API, Intel® MKL includes multiple code paths -- each optimized for specific generations of Intel and compatible processors. With no code-branching required by application developers, Intel® MKL utilizes the best code path for maximum performance.

Even before future processors are released, new code paths are added under these same APIs. Developers just link to the newest version of Intel® MKL and their applications are ready to take full advantage of the newest processor architectures.

In the case of the new Intel® Xeon Phi™ coprocessors, in addition to full native optimization support, Intel® MKL can also automatically determine the best load balancing between the host CPU and the Intel Xeon Phi coprocessors.

Results Consistency vs. Performance

Accuracy vs. Performance

Compilers, linking and threading models

Languages and operating systems

Flexibility to Meet Developer Requirements

Developers have many requirements to meet. Sometimes these requirements conflict and need to be balanced. Need consistent floating point results with the best application performance possible? Want faster vector math performance and don't need maximum accuracy? Intel® MKL gives you control over the necessary tradeoffs.

Intel® MKL is also compatible with your choice of compilers, languages, operating systems, linking and threading models. One library solution across multiple environments means only one library to learn and manage.

Details

Linear Algebra

Intel® MKL BLAS provides optimized vector-vector (Level 1), matrix-vector (Level 2) and matrix-matrix (Level 3) operations for single and double precision real and complex types. Level 1 BLAS routines operate on individual vectors, e.g., compute scalar product, norm, or the sum of vectors. Level 2 BLAS routines provide matrix-vector products, rank 1 and 2 updates of a matrix, and triangular system solvers. Level 3 BLAS level 3 routines include matrix-matrix products, rank k matrix updates, and triangular solvers with multiple right-hand sides.

Intel® MKL LAPACK provides extremely well-tuned LU, Cholesky, and QR factorization and driver routines that can be used to solve linear systems of equations. Eigenvalue and least-squares solvers are also included, as are the latest LAPACK 3.3.1 interfaces and enhancements.

If your application already relies on the BLAS or LAPACK functionality, simply re-link with Intel® MKL to get better performance on Intel and compatible architectures.

Fast Fourier Transforms

Intel® MKL FFTs include many optimizations and should provide significant performance gains over other libraries for medium and large transform sizes. The library supports a broad variety of FFTs, from single and double precision 1D to multi-dimensional, complex-to-complex, real-to-complex, and real-to-real transforms of arbitrary length. Support for both FFTW* interfaces simplifies the porting of your FFTW-based applications.

Vector Math

Intel® MKL provides optimized vector implementations of computationally intensive core mathematical operations and functions for single and double precision real and complex types. The basic vector arithmetic operations include element-by-element summation, subtraction, multiplication, division, and conjugation as well as rounding operations such as floor, ceil, and round to the nearest integer. Additional functions include power, square root, inverse, logarithm, trigonometric, hyperbolic, (inverse) error and cumulative normal distribution, and pack/unpack. Enhanced capabilities include accuracy, denormalized number handling, and error mode controls, allowing users to customize the behavior to meet their individual needs.

Statistics

Intel® MKL includes random number generators and probability distributions that can deliver significant application performance. The functions provide the user the ability to pair Random-Number Generators such as Mersenne Twister and, Niederreiter with a variety of Probability Distributions including Uniform, Gaussian and Exponential.

Intel® MKL also provides computationally intensive core/building blocks for statistical analysis both in and out-of-core. This enables users to compute basic statistics, estimation of dependencies, data outlier detection, and missing value replacements. These features can be used to speed-up applications in computational finance, life sciences, engineering/simulations, databases, and other areas.

Data Fitting

Intel® MKL includes a rich set of splines functions for 1-dimensional interpolation. These are useful in a variety of application domains including data analytics (e.g. histograms), geometric modeling and surface approximation. Splines included are linear, quadratic, cubic, look-up, stepwise constant and user-defined.

What's New

Feature	Benefit
Conditional Numerical Reproducibility	Overcome the inherently non-associativity characteristics of floating-point arithmetic results with new support in the Intel® Math Kernel Library.
Optimized for Haswell, Ivy Bridge and Intel® Xeon Phi™ coprocessors	Intel® MKL is optimized for the latest and upcoming processor architectures to deliver the best performance in the industry. Support for the new digital random number generator provides truly random seeding of statistical calculations.
Automatic offload and compute load balancing between Intel® Xeon® processors and Intel Xeon Phi coprocessors	For Linear Algebra functionality, Intel® MKL can automatically determine the best way to utilize a system containing one or more Intel Xeon Phi coprocessors. The developer simply calls an MKL function and doesn't have to worry about the details.
Data Fitting functions	A rich set of splines are now included to optimize 1-dimensional interpolation calculations used in a variety of application domains.

Purchase Options: Language Specific Suites

Several suites are available combining the tools to build, verify and tune your application. The product covered in this product brief is highlighted in blue. Single or multi-user licenses along with volume, academic, and student discounts are available.

Components	Suites >>	Intel® Cluster Studio XE	Intel® Parallel Studio XE	Intel® C++ Studio XE	Intel® Fortran Studio XE	Intel® Composer XE	Intel® C++ Composer XE	Intel® Fortran Composer XE
	Intel® C / C++ Compiler	●	●	●		●	●	
	Intel® Fortran Compiler	●	●		●	●		●
	Intel® Integrated Performance Primitives³	●	●	●		●	●	
	Intel® Math Kernel Library³	●	●	●	●	●	●	●
	Intel® Cilk™ Plus	●	●	●		●	●	
	Intel® Threading Building Blocks	●	●	●		●	●	
	Intel® Inspector XE	●	●	●	●			
	Intel® VTune™ Amplifier XE	●	●	●	●			
	Intel® Advisor XE	●	●	●	●			
	Static Analysis	●	●	●	●			
	Intel® MPI Library	●						
	Intel® Trace Analyzer & Collector	●						
	Rogue Wave IMSL* Library²							●
	Operating System¹	W, L	W, L	W, L	W, L	W, L	W, L, O	W, L, O

Note: ¹ Operating System: W=Windows, L= Linux, O= OS X*.   ² Available in Intel® Visual Fortran Composer XE for Windows with IMSL*
   ³ Not available individually on OS X, it is included in Intel® C++ & Fortran Composer XE suites for OS X

Technical Specifications

Specs at a Glance	
Processor Support	Validated for use with multiple generations of Intel and compatible processors including but not limited to: Intel® Xeon™ Processor, Intel® Core™ processor family and Intel® Atom™ processor family.
Operating Systems	Use the same API for application development on multiple operating systems: Windows*, Linux* and OS X*.
Development Tools and Environments	Compatible with compilers from vendors that follow platform standards (e.g., Microsoft*, GCC, Intel). Can be integrated with Microsoft Visual Studio* 2008, 2010 and 2012.
Programming Languages	Natively supports C++ and Fortran development. Cross-language compatible with Java*, C#, Python* and other languages.
System Requirements	Refer to www.intel.com/software/products/systemrequirements/ for details on hardware and software requirements.
Support	All product updates, Intel® Premier Support services and Intel® Support Forums are included for one year. Intel Premier Support gives you secure, web-based, engineer-to-engineer support.
Community	Share experiences with other users of Intel® TBB and other parallel programming tools at the Intel moderated forum: http://software.intel.com/en-us/forums/

Learn more about Intel Math Kernel Library
- Click or enter the link below: http://intel.ly/intel-mkl
- Or scan the QR code on the left

Download a free 30-day evaluation
- Click or enter the link below: http://intel.ly/sw-tools-eval
- Click on 'Compilers and Libraries' link