Intel® Math Kernel Library 10.2 Update 5 for Linux*
Release Notes

Document number: 321363-004US

Contents:

1 Introduction

Intel® Math Kernel Library (Intel® MKL) is a library of highly optimized, threaded math routines for science, engineering, and financial applications that require maximum performance. It offers optimized BLAS, LAPACK, Sparse Solvers, Fast Fourier Transforms, Vector Math, and more for all the latest Intel® architectures. Intel MKL can also be fully integrated with Eclipse*.

This document provides system requirements, installation instructions, issues and limitations, and legal information.

To learn more about this product's:

- New features, see the "New in Intel® MKL…" section below.
- Documentation, help, and samples, see the Intel® MKL documentation (mkl_documentation.htm) in the $MKLROOT/doc directory.
- Technical support, including answers to questions not addressed in the installed product, visit the technical support forum at: http://www.intel.com/software/products/support/mkl.

Please remember to register your product at https://registrationcenter.intel.com/ by providing your email address. This helps Intel recognize you as a valued customer in the support forum.

New in Intel® MKL 10.2 Update 5

- New Features
    - Incorporated the LAPACK 3.2.1 update primarily consisting of fixes to LAPACK 3.2
- Performance improvements
    - FFTs
        - Improved performance for complex FFTs, 3D and higher on the Intel® 64 architecture
    - VSL
        - Improved performance of the MT19937 and MT2203 basic random number generators (BRNGs) on the 45nm Intel® Core™2 Duo processor and newer processors in 64-bit libraries
- Usability/Interface improvements
    - Added support for Boost version 1.41.0 in the ublas examples
    - Included Fortran 95 interfaces for the diagonally dominant solver functionality (?DTSVB, ?DTTRFB, ?DTTRSB)
    - Extended the Fortran 90 interface for the cluster FFTs to support GNU Fortran
    - Significantly reduced the memory consumption of in-place, multi-dimensional cluster FFTs

For further information on improvements in this and previous releases see the knowledgebase article on new features.

2 System Requirements

Supported Architectures and Terminology

Intel® Math Kernel Library supports the following architectures:

- **IA-32 Architecture** refers to systems based on 32-bit processors generally compatible with the Intel Pentium® processors, (for example, Intel® Pentium® 4 processor or Intel® Xeon® processor), or processors from other manufacturers supporting the same instruction set and running a 32-bit operating system.

- **Intel® 64 Architecture** refers to systems based on IA-32 architecture processors which have 64-bit architectural extensions, for example, Intel® Core™2 processor family, running a 64-bit operating system. If the system is running a 32-bit operating system, then IA-32 architecture applies instead. Systems based on AMD processors running a 64-bit operating system are also supported.
- **IA-64 Architecture** refers to systems based on the Intel® Itanium® processor running a 64-bit operating system.

System Requirements

Hardware

To install and use Intel® MKL you will need a system with a supported processor and 1.3 GB of free hard disk space plus an additional 450 MB during installation for download and temporary files (host system only).

Supported processors:

- Intel® Core™ processor family
- Intel® Xeon® processor family
- Intel® Itanium® processor family
- Intel® Pentium® 4 processor family
- Intel® Pentium® III processor
- Intel® Pentium® processor (300 MHz or faster)
- Intel® Celeron® processor
- AMD Athlon* and Opteron* processors

Software

To use Intel® MKL you will need a supported compiler and MPI implementation.

Following is the list of supported operating systems:

- Red Hat* Enterprise Linux* 3, 4, 5 (IA-32 / Intel® 64 / IA-64)
- SUSE LINUX Enterprise Server* 9, 10, 11 (IA-32 / Intel® 64 / IA-64)
- SGI ProPack* for Linux 4, 5 (Intel® 64 / IA-64)
- Red Hat* Fedora* 10, 11 (IA-32 / Intel® 64)

- Debian* GNU/Linux 4.0 (IA-32 / Intel® 64 / IA-64)
- Ubuntu* 8.10 (IA-32 / Intel® 64)
- Asianux* Server 3 (IA-32 / Intel® 64 / IA-64)
- Turbolinux* 11 (IA-32 / Intel® 64 / IA-64)

  Note: These Linux* distributions are supported, and Intel® MKL should work on many more. If you have trouble with your distribution, do let us know.

Following is the list of supported C/C++ and Fortran compilers:

- Intel® Fortran Compiler 11.1 for Linux*
- Intel® Fortran Compiler 11.0 for Linux*
- Intel® Fortran Compiler 10.1 for Linux*
- Intel® C++ Compiler 11.1 for Linux*
- Intel® C++ Compiler 11.0 for Linux*
- Intel® C++ Compiler 10.1 for Linux*
- GNU Compiler Collection (gcc, g77, GNU Fortran 4.2.0 and later)
- Absoft* Pro Fortran v10.1 for Linux*
- PGI* Workstation Complete version 10.x.x

Following is the list of MPI implementations that Intel® MKL has been validated against:

- Intel® MPI Library Version 3.0, 3.1, 3.2.x, and 4.0 (http://www.intel.com/go/mpi)
- MPICH2 version 1.2.1 (http://www-unix.mcs.anl.gov/mpi/mpich)
- MPICH version 1.2.x (http://www-unix.mcs.anl.gov/mpi/mpich)
- Open MPI 1.2.x (http://www.open-mpi.org)
- SGI* MPT on Intel® 64 and IA-64 (http://www.sgi.com/products/software/mpt/)

  Note: Usage of MPI linking instructions can be found in the User's Guide in the doc directory.

Following is a list of tools supported with example sources:

- uBLAS examples: Boost C++ library, version 1.x.x

- JAVA examples: J2SE* SDK 1.4.2, JDK 5.0 and 6.0 from Sun Microsystems, Inc.

Note:

- Parts of Intel® MKL have Fortran interfaces, and data structures, while other parts have C interfaces and C data structures. The User Guide in the doc directory contains advice on how to link to Intel® MKL with different compilers and from different programming languages.

3 Installation Notes

Guidance on the installation of Intel® MKL is provided at install time. Links will be provided to a file with step-by-step instructions (filename: Install.txt). This file can also be found in the doc directory.

4 Issues and Limitations

A full list of the known limitations in this release and the issues it resolves can be found on the Intel MKL knowledgebase.

5 Notices

The following change is planned for future versions of Intel MKL. Please contact customer support if you have concerns:

- Content in the libraries containing ‘solver’ in the filenames will be moved to the core library in a future version of Intel MKL. These ‘solver’ libraries will be removed in a future version.

6 Attributions

As referenced in the End User License Agreement, attribution requires, at a minimum, prominently displaying the full Intel product name (e.g. "Intel® Math Kernel Library") and providing a link/URL to the Intel® MKL homepage (http://www.intel.com/software/products/mkl) in both the product documentation and website.

The original versions of the BLAS from which that part of Intel® MKL was derived can be obtained from http://www.netlib.org/blas/index.html.

The original versions of LAPACK from which that part of Intel® MKL was derived can be obtained from http://www.netlib.org/lapack/index.html. The authors of LAPACK are E. Anderson, Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, and D. Sorensen. Our FORTRAN 90/95 interfaces to LAPACK are

similar to those in the LAPACK95 package at http://www.netlib.org/lapack95/index.html. All interfaces are provided for pure procedures.

The original versions of ScaLAPACK from which that part of Intel® MKL was derived can be obtained from http://www.netlib.org/scalapack/index.html. The authors of ScaLAPACK are L. S. Blackford, J. Choi, A. Cleary, E. D'Azevedo, J. Demmel, I. Dhillon, J. Dongarra, S. Hammarling, G. Henry, A. Petitet, K. Stanley, D. Walker, and R. C. Whaley.

PARDISO in Intel® MKL is compliant with the 3.2 release of PARDISO that is freely distributed by the University of Basel.

Some FFT functions in this release of Intel® MKL have been generated by the SPIRAL software generation system (http://www.spiral.net/) under license from Carnegie Mellon University. The Authors of SPIRAL are Markus Puschel, Jose Moura, Jeremy Johnson, David Padua, Manuela Veloso, Bryan Singer, Jianxin Xiong, Franz Franchetti, Aca Gacic, Yevgen Voronenko, Kang Chen, Robert W. Johnson, and Nick Rizzolo.

7 Disclaimer and Legal Information

Intel processor numbers are not a measure of performance. Processor numbers differentiate features within each processor family, not across different processor families. See http://www.intel.com/products/processor_number for details.

This document contains information on products in the design phase of development.