1. Introduction

Intel® Parallel Studio XE Composer Edition Fortran includes the Intel® Fortran compiler, Intel® Math Kernel Library, and GNU GDB® 7.10.

Intel® Parallel Studio XE Composer Edition C++ includes the Intel® C++ compiler, Intel® Math Kernel Library, Intel® Performance Primitives, Intel® Threading Building Blocks, Intel® Data Analytics Acceleration Library (Intel® DAAL), and GNU GDB® 7.10.
This document describes how to install the product, provides a summary of new and changed features and includes notes about features and problems not described in the product documentation.

On completing the Intel® Parallel Studio XE installation process, locate the getstart*.htm file in the documentation_2017 folder under the target installation path. This file is a documentation map to navigate to various information resources of the Intel® Parallel Studio XE.

1.1 What Every User Should Know About This Release
- This update includes mitigations for speculative execution side-channel issues. See https://software.intel.com/en-us/articles/using-intel-compilers-to-mitigate-speculative-execution-side-channel-issues for more details.

1.2 What’s New
This section highlights important changes from the previous product version. For more information on what is new in each component, please read the individual component release notes. The latest documentation for all components can be found at https://software.intel.com/en-us/intel-parallel-studio-xe-support/documentation.

Changes in Update 8
- Bug fixes in Intel® C/C++ Compiler and Intel® Fortran Compiler.

Changes in Update 7
- Compilers updated to the latest versions.
- Intel® C++ Compiler and Intel® Fortran Compiler provide new options to mitigate branch target injection.
- This update can be installed even if your support service has expired using your existing Serial Number.

Changes in Update 4
- Support Xcode* 8.3
- Intel® Math Kernel Library (Intel® MKL), Intel® Integrated Performance Primitives (Intel® IPP) are distributed under Intel Simplified Software License. Intel® Threading Building Blocks (Intel® TBB) for Linux*, and macOS*, Intel® Data Analytics Acceleration Library (Intel® DAAL) for Linux*, and macOS* are distributed under Apache License, Version 2.0. See compilers_and_libraries_2017/licensing/ folder under the target installation path for reference

Changes in Update 2
- Supporting macOS* 10.12.1 and Xcode* 8.2
- Fixes for reported problems
Changes in Update 1

- Supporting macOS 10.12 and Xcode 8
- Fixes for reported problems

Changes since Intel® Parallel Studio XE 2016

- All components updated to current versions
- Intel® Threading Building Blocks: see [install-dir]/
  - static_partitioner class is now a fully supported feature.
  - async_node class is now a fully supported feature.
  - Improved dynamic memory allocation replacement on Windows* OS to skip
    DLLs for which replacement cannot be done, instead of aborting.
  - For 64-bit platforms, quadrupled the worst-case limit on the amount
    of memory the Intel TBB allocator can handle.
  - Added TBB_USE_GLIBCXX_VERSION macro to specify the version of GNU
    libstdc++ when it cannot be properly recognized, e.g. when used with Clang
    on Linux* OS. Inspired by a contribution from David A.
  - Added graph/stereo example to demonstrate tbb::flow::async_msg.
  - Removed a few cases of excessive user data copying in the flow graph.
  - Reworked split_node to eliminate unnecessary overheads.
  - Added support for C++11 move semantics to the argument of
    tbb::parallel_do_feeder::add() method.
  - Added C++11 move constructor and assignment operator to
    tbb::combinable template class.
  - Added tbb::this_task_arena::max_concurrency() function and
    max_concurrency() method of class task_arena returning the maximal
    number of threads that can work inside an arena.
  - Deprecated tbb::task_arena::current_thread_index() static method;
    use tbb::this_task_arena::current_thread_index() function
    instead.
  - All examples for commercial version of library moved online:
    https://software.intel.com/en-us/p...code-samples. Examples are
    available as a standalone package or as a part of Intel® Parallel Studio XE or
    Intel® System Studio Online Samples packages.

Changes affecting backward compatibility:

- Renamed following methods and types in async_node class:
  Old                          New
  async_gateway_type => gateway_type
  async_gateway()    => gateway()
  async_try_put()    => try_put()
  async_reserve()    => reserve_wait()
  async_commit()     => release_wait()

- Internal layout of some flow graph nodes has changed; recompilation is
  recommended for all binaries that use the flow graph.

- Intel® Data Analytics Acceleration Library:
- Added support of a new Neural Network layer “softmax with cross-entropy loss”.
- Added quality metrics for linear regression.
- Improved interfaces (methods in classes for support of Neural Network based computations).

- Documentation updates
- Tutorials and sample codes have been removed from installation packages and are now available online at Intel® Software Product Samples and Tutorials
- Intel® Data Analytics Acceleration Library removed from Fortran language only editions.
- All updates in one major version of the compiler are now supported in Xcode* IDE integration
- Intel® compiler support for additional features in OpenMP* 4.0 and 4.5 Specifications
- Intel® C++ compiler support for features in C++14 and support for C11 feature _Atomic
- Intel® Fortran compiler support for more features in Fortran 2008 and Draft Fortran 2015
- A single high level compiler switch -fp-model consistent to generate code that will give consistent, reproducible floating-point results between different runs, optimization levels and processors or microarchitectures, for single-threaded code.
- Compiler options to annotate source files with compiler optimization reports
- Code alignment attribute (C++) or directive (Fortran) for functions; code alignment pragma or directive for a specified loop; compiler options to align all loops (or not).

1.3 Product Contents

*Intel® Parallel Studio XE 2017 Composer Edition for macOS* includes the following components:

The table below lists the product components and related documentation.

<table>
<thead>
<tr>
<th>Component</th>
<th>Version</th>
<th>On-Disk Documentation</th>
<th>Release Notes</th>
</tr>
</thead>
</table>

### 1.4 Additional Information for Intel-provided Debug Solutions


### 1.5 System Requirements

For an explanation of architecture names, see [http://intel.ly/q9JVjE](http://intel.ly/q9JVjE)

- A 64-bit Intel®-based Apple® Mac® system host (development for 32-bit is still supported)
- 2GB RAM minimum, 4GB RAM recommended
- 7GB free disk space
- One of the following combinations of OS X*, Xcode* and the Xcode SDK:
  - Xcode* 7.x and 8.x
  - OS X* 10.11 and macOS* 10.12
- If doing command line development, the Command Line Tools component of Xcode* is required

Note: Advanced optimization options or very large programs may require additional resources such as memory or disk space.
1.6 Documentation
Product documentation can be found in the documentation_2017 folder as shown under Installation Folders.

1.7 Samples
Tutorials and sample codes have been removed from installation packages and are now available online at Intel® Software Product Samples and Tutorials.

1.8 Technical Support
Your feedback is very important to us. To receive technical support for the tools provided in this product and technical information including FAQ’s and product updates, you are encouraged to register your product at the Intel® Software Development Products Registration Center.

NOTE: Registering for support varies for release product or pre-release products (alpha, beta, etc.) – only released software products have support web pages at http://software.intel.com/sites/support/.

To register for an account, please visit the Intel® Software Development Products Registration Center website at http://www.intel.com/software/products/registrationcenter/index.htm. If you have forgotten your password, please follow the instructions on the login page for forgotten password.

2 Installation Notes

The installation of the product requires a valid license file or serial number.

If you will be using Xcode*, please make sure that a supported version of Xcode is installed. If you install a new version of Xcode in the future, you must reinstall the Intel Parallel Studio XE 2017 afterwards.

The Command Line Tools component, required for command-line development, is not installed by default. It can be installed using the Components tab of the Downloads preferences panel.

You will need to have administrative or “sudo” privileges to install, change or uninstall the product.

Follow the prompts to complete installation.

Note that there are several different downloadable files available, each providing different combinations of components. Please read the download web page carefully to determine which file is appropriate for you.

You do not need to uninstall previous versions or updates before installing a newer version – the new version will coexist with the older versions.

2.1 Installation on macOS* 10.11

macOS* 10.11 introduces a new security policy called System Integrity Protection. This significantly impacts certain installation scenarios. Please see Intel® Parallel Studio XE support for OS X* 10.11 for important information before attempting to install on macOS* 10.11.

You will need to have administrative or “sudo” privileges to install, change, or uninstall the product.

2.2 Online Installation now available

The electronic installation package for Intel® Parallel Studio XE offers as an alternative a smaller installation package that dynamically downloads and then installs packages selected to be installed. This requires a working internet connection and potentially a proxy setting if you are behind an internet proxy. Full packages are provided alongside where you download this online install package if a working internet connection is not available. The online installer may be downloaded and saved as an executable file which can then be launched from the command line.
2.2.1 Storing Online Installer Download Content

The online installer stores the downloaded content in the form-factor of the standard install package which can then be copied and reused offline on other systems. The default download location is `/var/[login]/Downloads`. This location may be changed with the `INTEL_SWTOOLS_DOWNLOAD_DIR` environment variable. The online installer also supports a download only mode which allows the user to create a package without installation. This mode is enabled with the `INTEL_SWTOOLS_DOWNLOAD_DIR` environment variable.

2.3 Intel® Software Manager

The installation now provides an Intel® Software Manager to provide a simplified delivery mechanism for product updates and provide current license status and news on all installed Intel® software products.

You can also volunteer to provide Intel anonymous usage information about these products to help guide future product design. This option, the Intel® Software Improvement Program, is not enabled by default – you can opt-in during installation or at a later time, and may opt-out at any time. For more information please see http://intel.ly/SoftwareImprovementProgram.

2.4 Using a License Server

If you have purchased a "floating" license, see http://intel.ly/pjGfwC for information on how to install using a license file or license server. This article also provides a source for the Intel® License Server that can be installed on any of a wide variety of systems.

2.5 Support of Non-Interactive Custom Installation

Intel Parallel Studio XE supports the saving of user install choices during an 'interactive' install in a configuration file that can then be used for silent installs. This configuration file is created when the following option is used from the command line install:

- `export INTEL_SWTOOLS_DUPLICATE_MODE=config_file_name`: it specifies the configuration file name. If full path file name is specified, the `INTEL_SWTOOLS_DOWNLOAD_DIR` environment variable is ignored and the installable package will be created under the directory where the configuration file is.
- `export INTEL_SWTOOLS_DOWNLOAD_DIR=dir_name`: optional, it specifies where the configuration file will be created. If this option is omitted, the installation package and the configuration file will be created under the default download directory:

  `/tmp/intel/downloads/<package_id>`

2.6 Installation Folders

In an effort to improve and more tightly unify the user experience when using multiple compilers and libraries from multiple Intel® Software Development Tools, the directory layout has changed in this release of Intel® Parallel Studio XE. This directory structure should remain stable for the next future major release. If you have questions, please see this explained in more detail at http://intel.ly/1Nn2GjV.
The compiler installs, by default, under /opt/intel – this is referenced as <install-dir> in the remainder of this document. You are able to specify a different location.

Under <install-dir> are the following directories (not all may be present in a given installation):

- **bin** – contains symbolic links to executables for the latest installed version
- **lib** – symbolic link to the lib directory for the latest installed version
- **include** – symbolic link to the include directory for the latest installed version
- **man** – symbolic link to the directory containing man pages for the latest installed version
- **ipp** – symbolic link to the directory for the latest installed version of Intel® Integrated Performance Primitives
- **mkl** – symbolic link to the directory for the latest installed version of Intel® Math Kernel Library
- **tbb** – symbolic link to the directory for the latest installed version of Intel® Threading Building Blocks
- **daal** – symbolic link to the directory for the latest installed version of Intel® Data Analytics Acceleration Library (Intel® DAAL).
- **ism** – contains files for Intel® Software Manager
- **compilers_and_libraries** – symbolic link to the compilers_and_libraries_2017 directory
- **compilers_and_libraries_2017** – directory containing symbolic links to subdirectories for the latest installed Intel® C++ Compiler and Libraries 2017 release
- **compilers_and_libraries_2017.<n>.<pkg>** - physical directory containing files for a specific compiler and libraries version. `<n>` is the update number, and `<pkg>` is a package build identifier.
- **documentation_2017** – directory containing documentation for Intel® Parallel Studio XE 2017
- **ide_support_2017** – directory containing IDE integration files for Intel® Parallel Studio XE 2017
- **parallel_studio_xe_2017.<n>.<pkg>** - directory containing license and support information for Intel® Parallel Studio XE 2017, uninstall application, and symbolic links to Intel Parallel Studio XE 2017 components
- **debugger_2017** – directory containing subdirectories with debugger environment scripts, libraries, and binaries for various debugging scenarios

Each **compilers_and_libraries_2017** directory contains a mac subdirectory that contains the following directories that reference the latest installed compilers and libraries for Intel® Parallel Studio XE 2017:

- **bin** – directory containing scripts to establish the compiler and libraries environment and symbolic links to compiler executables for the host platform
- **pkg_bin** – symbolic link to the compiler bin directory
**2.7 Relocating Product After Install**

The Xcode integration is relocatable simply by dragging and dropping the Xcode directory tree to another location.

**2.8 Removal/Uninstall**

It is not possible to remove the compiler while leaving any of the performance library components installed.

1) Open the file `<install-dir>/parallel_studio_xe_2017.<n>.<pkg>/uninstall.app`

2) Follow the prompts

If you are not currently logged in as root you will be asked for the root password.
3 Intel® IPP Cryptography Libraries are Available as a Separate Download
The Intel® IPP cryptography libraries are available as a separate download. For download and installation instructions, please read http://intel.ly/ndrGnR

4 Intel® Math Kernel Library 2017 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'Aszvedo, J. Demmel, I. Dhillon, J. Dongarra, S. Hammarling, G. Henry, A. Petitet, K. Stanley, D. Walker, and R. C. Whaley.

The Intel® MKL Extended Eigensolver functionality is based on the Feast Eigenvalue Solver 2.0 http://www.ecs.umass.edu/~polizzi/feast/

PARDISO in Intel® MKL is compliant with the 3.2 release of PARDISO that is freely distributed by the University of Basel. It can be obtained at http://www.pardiso-project.org.

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.

5 Disclaimer and Legal Information
INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL(R) PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO
LIABILITY WHATSOEVER, AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT. UNLESS OTHERWISE AGREED IN WRITING BY INTEL, THE INTEL PRODUCTS ARE NOT DESIGNED NOR INTENDED FOR ANY APPLICATION IN WHICH THE FAILURE OF THE INTEL PRODUCT COULD CREATE A SITUATION WHERE PERSONAL INJURY OR DEATH MAY OCCUR.

Intel may make changes to specifications and product descriptions at any time, without notice. Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The information here is subject to change without notice. Do not finalize a design with this information.

The products described in this document may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Contact your local Intel sales office or your distributor to obtain the latest specifications and before placing your product order.

Copies of documents which have an order number and are referenced in this document, or other Intel literature, may be obtained by calling 1-800-548-4725, or go to: http://www.intel.com/design/literature.htm


Intel processor numbers are not a measure of performance. Processor numbers differentiate features within each processor family, not across different processor families. Go to: http://www.intel.com/products/processor%5Fnumber/


The GNU* Project Debugger, GDB, is provided under the General GNU Public License GPL V3.

Please consult the licenses included in the distribution for details.
Celeron, Centrino, Intel, Intel logo, Intel386, Intel486, Atom, Core, Itanium, MMX, Pentium, VTune, Cilk, Xeon Phi, and Xeon are trademarks of Intel Corporation in the U.S. and other countries.

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

Copyright © 2018 Intel Corporation. All Rights Reserved.