



Overview

The Intel® MPI Library for Windows* OS is a multi-fabric message passing library based on ANL* MPICH3* and OSU* MVAPICH2* that implements the Message Passing Interface, version 3 (MPI-3) specification.

To receive technical support and updates, you need to register your Intel® Software Development Product. See section [Technical Support](#).

Product Contents

The Intel® MPI Library Runtime Environment (RTO) contains the tools you need to run programs including SMPD services and supporting utilities, dynamic libraries, and documentation.

The Intel® MPI Library Development Kit (SDK) includes all of the Runtime Environment components plus include files and modules, interface libraries, debug libraries and test codes.

Related Products and Services

Information on Intel® Software Development Products is available at <http://www.intel.com/software/products>.

What's New

See <https://software.intel.com/en-us/articles/intel-cluster-tools-deprecation-information> for a current list of deprecated features

Intel® MPI Library 5.1 Update 3

- Deprecation list updated. New deprecations:
 - SSHM
 - SMPD
 - Epoll
 - JMI
 - PVFS2
- New algorithms and selection mechanism for nonblocking collectives.
- Added I_MPI_BCAST_ADJUST_SEGMENT variable to control MPI_Bcast.
- Fixed long count support for some collective messages.
- Binding kit reworked with support for Intel(R) Many Integrated Core

Contents

- [Overview](#)
- [What's New](#)
- [Key Features](#)
- [System Requirements](#)
- [Installation Notes](#)
- [Special Features and Known Issues](#)
- [Documentation](#)
- [Technical Support](#)
- [Copyright and Licenses](#)
- [Disclaimer and Legal Information](#)

Architecture and support for ILP64 on third party compilers.

Intel® MPI Library 5.1 Update 2

- ILP64 support enhancements, support for MPI modules in Fortran 90.

Intel® MPI Library 5.1 Update 1

- Change the named-user licensing scheme. See more details in the Installation Instructions section of *Intel® MPI Library Installation Guide*.
- Bug fixes

Intel® MPI Library 5.1

- Added the Troubleshooting chapter to the *Intel® MPI Library User's Guide*.
- Added the `MPI_Pcontrol` feature for internal statistics.
- Increased the possible space for `MPI_TAG`.
- Changed the default installation directory to `C:\Program Files (x86)\IntelSWTools`. See the *README* document for details.
- Bug fixes

NOTE: Intel® MPI Benchmarks is delivered as part of Intel® MPI Library. For the new features of Intel® MPI Benchmarks, see the What's New section in *Intel® MPI Benchmarks README*.

Intel® MPI Library 5.0 Update 3

- Support for the rename mechanism for the file, stats.txt, to avoid overwriting
- Native statistics collection can be controlled with `MPI_Pcontrol`
- Bug fixes

Intel® MPI Library 5.0 Update 2

- Enhancements to statistics gathering mode
- Bug fixes

Intel® MPI Library 5.0 Update 1

- Directory structure update. New shortcuts have been added to always point to the most recently installed version of the Intel® MPI Library
- Bug fixes, including:
 - Resolving problem where Hydra with `-localroot` causes `pmi_proxy` to only spawn on local host.
- Collective performance improvements
- Documentation update
- Man pages copyright updated
- Added support for `-fopenmp` in `mpiicc`, `mpiicpc`, and `mpiifort`
- Improved pinning under job schedulers

Intel® MPI Library 5.0

- Support for Hydra* process manager on Windows* OS by default
- Added option `I_MPI_JOB_RESPECT_PROCESS_PLACEMENT` to honor process placement from job schedulers
- All IA-32 architecture support has been removed
- Added debug information without private symbols to optimized libraries. Added `.pdb` files to get call stack when an application crashes.
- Implement the MPI-3 standard including but not limited to:
 - Non-blocking collective operations
 - Fast one-sided operations
 - Large counts for messages greater than 2GB
- Allow permuted entries in machine file when running a single instance of pmi-proxy
- Support for mixed operating systems in the Hydra* process manager
- Make the following changes to documentation:
 - Changed the Intel® MPI Library Getting Started Guide to Intel® MPI Library User's Guide
 - Add the Intel® MPI Library Getting Started page
 - Add the tutorial: MPI Tuner for Intel® MPI Library
- Bug fixes
- Deprecate MPD and SMPD process managers

32 Bit Support of Intel® MPI Library & Intel® Trace Analyzer and Collector

Inclusion of 32-bit binaries in the Intel® MPI Library and Intel® Trace Analyzer and Collector products is being deprecated. If 32-bit support is required, we advise that you remain on Intel® MPI Library version 4.1 Updates and Intel® Trace Analyzer and Collector version 8.1 Updates, which continue to include 32-bit binaries. The Intel® MPI Library 5.0 and Intel® Trace Analyzer and Collector 9.0 releases will not include 32-bit binaries. Many developers have already migrated to 64-bit implementations of both their applications and of Intel libraries and tools. If you have concerns about this deprecation, please send us feedback by submitting an issue at the Intel® Premier Customer Support site (<http://premier.intel.com>) as soon as possible with your contact information.

Intel® MPI Library 4.1 Update 3

- Intel® Xeon™ E5 V2 and Intel® Xeon™ E7 V2 family processors additional performance tuning
- New online documentation format
- Bug fixes

Intel® MPI Library 4.1 Update 2

- Intel® Xeon™ E5 V2 and Intel® Xeon™ E7 V2 family processors performance tuning
- Allow permuted entries in machine file when running a single instance of pmi-proxy
- Bug fixes

Intel® MPI Library 4.1 Update 1

- Intel® Xeon Phi™ offload model support
- Hydra* (Scalable process manager) support on Windows* OS (experimental)
- Microsoft* Network Direct support
- Bug fixes

Intel® MPI Library 4.1

- Support for the MPI-2.2 standard
- Backward compatibility with Intel MPI Library 4.0.x based applications
- New documentation in the HTML format
- Support for Intel® Composer XE 2013
- Support for clusters with different Intel® Architecture Processors
- Bug Fixes

Key features

This release of the Intel® MPI Library supports the following major features:

- MPI-1, MPI-2.2 and MPI-3 specification conformance with some limitations. See [Special Features and Known Issues](#)
- - Support for any combination of the following interconnection fabrics:
 - Shared memory
 - RDMA-capable network fabrics through DAPL*, such as InfiniBand* and Myrinet*
 - Sockets, for example, TCP/IP over Ethernet*, Gigabit Ethernet*, and other interconnects
- (SDK only) Support for Intel® 64 architecture clusters using:
 - Intel® C++ Compiler version 14.0 through 16.0 and higher
 - Intel® Fortran Compiler version 14.0 through 16.0 and higher
 - Microsoft* Visual C++* Compilers
- (SDK only) C, C++, Fortran* 77 and Fortran 90 language bindings
- (SDK only) Dynamic linking

System Requirements

The following sections describe supported hardware and software

Supported Hardware

Systems based on the Intel® 64 architecture, in particular:

- Intel® Core™ 2 processor family or higher
- Intel® Xeon® E5 v3 processor families recommended
- Intel® Xeon® E7 v2 processor families recommended
- 1 GB of RAM per core
- 2 GB of RAM per core recommended
- 1 GB of free hard disk space

Supported Software

Operating Systems:

- Systems based on the Intel® 64 architecture:
 - Microsoft* Windows* HPC Server 2012

- o Microsoft* Windows* HPC Pack 2012
- o Microsoft* Windows 7*
- o Microsoft* Windows 8*
- o Microsoft* Windows 8.1*

(SDK only) Compilers:

- Intel® C++ Compiler 14.0 through 16.0 for Windows* OS
- Intel® Fortran Compiler 14.0 through 16.0 for Windows* OS
- Microsoft* Visual Studio 2010*
- Microsoft* Visual Studio 2012*
- Microsoft* Visual Studio 2013*
- Microsoft* Visual Studio 2015*
- Microsoft* Visual C++* Compilers

Batch Systems:

- Microsoft* job scheduler
- Altair* PBS Pro* 9.2 and higher

Recommended InfiniBand Software:

- Windows* OpenFabrics* (WinOF*) 2.0 or higher
- Windows* OpenFabrics* Enterprise Distribution (winOFED*) 3.2 RC1 or higher for Microsoft* Network Direct support
- Mellanox* WinOF* Rev 4.40 or higher

Supported Languages

- For Intel® Professional Edition Compilers: C, C++, Fortran 77, Fortran 90

Installation Notes

Launch the installer and follow the instructions. See *Intel® MPI Library for Windows* OS Installation Guide* for details.

Using Intel® Software License Manager

If you have purchased a “floating” license, see [Intel® Software License Manager Getting Started Tutorial](#) for information on how to install using a license file or license manager. This article also provides a source for the Intel® Software License Manager that can be installed on any of a wide variety of systems.

Special Features and Known Issues

NOTE: The following list includes the information until Intel® MPI Library 5.1 Update 1 is released. For the most up-to-date list of known issues, as well as latest tips and tricks on using the library, visit the Intel(R) MPI Library for Windows* Knowledge Base at <http://software.intel.com/en-us/articles/intel-mpi-library-for-windows-kb/all/>.

- Support for Fortran 2008 is not yet implemented in Intel® MPI Library for Windows*.
- Switching on statistics gathering could result in increased time in MPI_Finalize.
- In order to run a mixed operating system job (Linux* and Windows*), all binaries must link to the same single or multithreaded MPI library. The single and multithreaded libraries are incompatible with each other and should not be mixed. Note that the pre-compiled binaries for the Intel® MPI Benchmarks are inconsistent (Linux* version links to multithreaded, Windows* version links

to single threaded) and as such, at least one must be rebuilt to match the other.

- Intel® MPI Library 5.0 for Windows* OS is binary compatible with the majority of Intel MPI Library 4.1.x-based applications. Recompile your application only if you use:
 - `MPI_Dist_graph_create`, `MPI_Dist_graph_create_adjacent`, `MPI_Dist_graph_neighbors`, `MPI_Dist_graph_neighbors_count`, (C, C++, Fortran)
 - `MPI::Get_address` (C++ only)
- Intel® MPI Library 4.1 for Windows* OS is binary compatible with the majority of Intel MPI Library 4.0.x-based applications. Recompile your application only if you use:
 - MPI C++ binding
- Intel® MPI Library 4.1 for Windows* OS implements the MPI-2.2 standard. On top of this, the aliasing of the send and receive buffers in the following collective routines will be rejected:
 - `MPI_Gather`, `MPI_Gatherv`
 - `MPI_Scatter`, `MPI_Scatterv`
 - `MPI_Allgather`, `MPI_Allgatherv`
 - `MPI_Alltoall`, `MPI_Alltoallv`, `MPI_Alltoallw`

If your application depends on the pre-MPI-2.2 behavior, set the environment variable `I_MPI_COMPATIBILITY` to 4. If your application depends on the pre-MPI-2.1 behavior, set the environment variable `I_MPI_COMPATIBILITY` to 3.

- The Intel® MPI Library 4.0 Update 2 for Windows* OS provides enhanced security options. Use the `HKEY_LOCAL_MACHINE\Software\Intel\MPI` registry key to define the following registry entries:
 - The `SecureDynamicLibraryLoading` enables the dynamic library loading enhanced security mode. Set the value to `enable|yes|on|1`. This option is disabled by default.
 - The `I_MPI_DAT_LIBRARY` specifies the DAT library to be used in the DLL enhanced security mode.
NOTE: the `I_MPI_DAT_LIBRARY` environment variable has no effect in the dynamic library loading the enhanced security mode. See the *Intel® MPI Library for Windows* OS Reference Manual* for more details on this variable.
- Intel® MPI Library 4.0 for Windows* OS is binary compatible with the majority of Intel MPI Library 3.x-based applications. Recompile your application only if you use:
 - MPI one-sided routines in Fortran (`mpi_accumulate()`, `mpi_alloc_mem()`, `mpi_get()`, `mpi_put()`, `mpi_win_create()`)
 - MPI C++ binding
- Intel® MPI Library 4.0 for Windows* OS implements the MPI-2.1 standard. The functions of the following MPI routines have changed:
 - `MPI_Cart_create()`
 - `MPI_Cart_map()`
 - `MPI_Cart_sub()`
 - `MPI_Graph_create()`

If your application depends on the strict pre-MPI-2.1 behavior, set the environment variable `I_MPI_COMPATIBILITY` to 3.

- The Intel® MPI Library supports the MPI-2 process model for all fabric combinations except the following case:
`I_MPI_FABRICS` is set to `<fabric1>:<fabric2>`, where `<fabric1>` is not `shm`, and `<fabric2>` is not equal to `<fabric1>` (for example, `dapl:tcp`).
- If communication between two existing MPI applications is established using the process attachment mechanism, the library does not control whether the same fabric has been selected for each application. This situation may cause unexpected applications behavior. Set the same `I_MPI_FABRICS` variable for each application to avoid this issue.
- The following restriction exists for the DAPL-capable network fabrics because it relates to support of the MPI-2 process model: if the size of the information about the host used to establish the communication exceeds a certain DAPL provider value, the application fails with an error message similar to:

```
[0:host1][../../dapl_module_util.c:397] error(0x80060028):...:
could not connect DAPL endpoints:
DAT_INVALID_PARAMETER(DAT_INVALID_ARG5)
```
- The Intel® MPI Library uses dynamic connection establishment by default for 64 and more processes. To always establish all connections upfront, set the `I_MPI_DYNAMIC_CONNECTION` environment variable to "disable". This option may improve stability of certain DAPL* providers.
- (SDK only) Provide the `msvcr71.dll` library to the end user if your product redistributes the `mpitune` utility.
- The Intel® MPI Library pins processes automatically. Use the `I_MPI_PIN` and related environment variables to control process pinning. See the *Intel® MPI Library for Windows* OS Reference Manual* for details.
- The Intel® MPI Library enhances message-passing performance on DAPL*-based interconnects by maintaining a cache of virtual-to-physical address translations in the MPI DAPL* data transfer path. To disable the translation cache completely, set the environment variable `I_MPI_RDMA_TRANSLATION_CACHE` to "disable".
- (SDK only) The `nmake` utility does not work correctly if the path to the Intel® MPI Library compiler drivers contains spaces. For instance, `C:\Program Files (x86)\IntelSWTools\MPI\<version>\bin\`. Copy the Intel® MPI Library compiler drivers to another location to avoid this issue.
- An MPI application inherits the current working directory of the `mpiexec` command at the time of its invocation. Use the `-gwdir` or `-wdir` options to override this functionality.
- Intel® MKL 10.0 may create multiple threads depending on various conditions. Follow these rules to correctly use Intel® MKL:
 - Use the thread-safe version of the Intel® MPI Library in conjunction with Intel® MKL
 - Set the `OMP_NUM_THREADS` environment variable to "1" to run the application if linked against the non-thread-safe version of the Intel® MPI Library
- Follow these rules to launch the Intel MPI Library application using the MPICH2* environment:
 - Use the `-port mpiexec` option to specify the MPICH2* SMPD* port number
 - Set the `I_MPI_SMPD_VERSION_CHECK` environment variable to "disable"
- Some operating systems have a limitation on the number of concurrent connections. This limitation may cause job failure with the following error

message:

"no more connections can be made to this remote computer at this time because there are already as many connections as the computer can accept". To reduce the probability of this issue, decrease the auto disconnect time. From the command line do the following:

```
net config server /autodisconnect:time_before_autodisconnect
```

The default value of the auto disconnect time is usually set to 15 minutes.

- Use the `mpiexec -mapall` option if your application is located or accesses any network drive mapped to your local system.
- Disable User Account Control (UAC) to use domain-based authorization with the delegation ability.
- The following MPI-2.1 feature is not supported by the Intel® MPI Library:
 - Passive target one-sided communication when the target process does not call any MPI functions
- There is a limitation on printing small messages from an MPI application if SMPD used. Immediate print of one byte message could make the application to hang in receiving further data. Keep your messages two and more bytes long to work around this restriction.
- Experimental Hydra (Scalable Process Manager) on Windows* OS has some known limitations such as:
 - `stdin` redirection is not supported for the `-bootstrap` service option.
 - Signal handling support is restricted. It could result in hanging processes in memory in case of incorrect MPI job termination.
 - MPI job termination by specifying `I_MPI_JOB_TIMEOUT` and `I_MPI_JOB_TIMEOUT_SIGNAL` environment variables are not supported
 - Cleaning up the environment after an abnormal MPI job termination by means of `mpicleanup` utility is not supported
- For details on how to use Hydra (Scalable Process Manager) on Windows* OS, see Intel® MPI Library for Linux* OS Reference Manual.
- For recent details on how to use Intel® Xeon Phi™ offload model on Windows* OS please see <http://software.intel.com/mic-developer>.
- Coarray Fortran (CAF) with Intel® Fortran Compiler 14 is incompatible with Intel® MPI Library 5.0. If using CAF, ensure that either Intel® Fortran Compiler 15 or higher is used, or use a 4.x version of Intel® MPI Library.
- The default cluster installation of the Intel® MPI Library Version 5.0 will not work on a default Windows* HPC Server 2012 installation due to Microsoft .NET Framework 3.5 not being installed. To work around this issue install Microsoft* .NET Framework Version 3.5.
- ILP64 is not supported by MPI modules for Fortran* 2008.

NOTE: Many routines in the `libirc.lib` library (linked with the Intel® MPI Library) are more optimized for Intel microprocessors than for non-Intel microprocessors.

Documentation

Intel® MPI Library for Windows* OS Getting Started page contains information on the following subject:

- Compiling and running your MPI program

Intel® MPI Library for Windows* OS User's Guide contains information on the following subjects:

- First steps using the Intel® MPI Library
- First-aid troubleshooting actions

Intel® MPI Library for Windows* OS Reference Manual contains information on the following subjects:

- Command Reference describes commands, options, and environment variables
- Tuning Reference describes environment variables that influence library behavior and performance

Intel® MPI Library for Windows* OS Installation Guide contains information on the following subjects:

- Obtaining, installing, and uninstalling the Intel® MPI Library
- Getting technical support

Tutorial: MPI Tuner for Intel® MPI Library contains information on the following subjects:

- How to use the MPI Tuner for Intel® MPI Library to get optimized configuration files for the runtime library automatically
- How to troubleshoot common issues with the MPI tuner

Notation Conventions

Release Notes and user guide documentation use the notation conventions listed in the following table:

Style	Definition
<i>This type style</i>	indicates an element of syntax, a reserved word, a keyword, a file name, or part of a program example (text appears in lowercase unless UPPERCASE is required)
This type style	indicates what you type as input
<i>This type style</i>	indicates an argument on a command line or an option's argument
[<i>items</i>]	indicates that the items enclosed in brackets are optional
{ <i>item</i> <i>item</i> }	indicates a set of choices from which you must select one
... (ellipses)	indicates that an argument can be repeated several times

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, please register your product at the [Intel® Registration Center](#). If your license includes access to [Intel® Premier Support](#), registration will grant that access.

To receive support for this product, you can submit an issue by logging in to [Intel®](#)

