IBM XL C/C++ and XL Fortran compilers on Power architectures overview

December 2018

Chapter 1. Introduction

This paper contains a brief history of the XL compiler family. It then provides an overview of the latest IBM® XL C, XL C/C++, and XL Fortran compilers on Power® architectures. It outlines their unique capabilities and cross-platform support, and highlights new features and performance enhancements. This paper also includes a brief discussion of the XL compiler team's strategy to continue exploiting evolving IBM hardware architectures and the latest language standards.

Chapter 2. A brief history of IBM XL compilers

The XL compilers are an advanced technology project that resulted from a joint effort from the IBM Toronto Lab, the Yorktown Heights Research Lab, and other IBM facilities. This collaboration, which began in the 1980s, led to the introduction of the C/370™ compiler in 1988, the XL Fortran for AIX® compiler in 1989, and the XL C and XL C++ compilers for AIX in 1991. IBM announced the Blue Gene program in December 1999 leading to the release of XL compilers for Blue Gene/L™, Blue Gene/P™, and then Blue Gene/Q™. IBM z/OS® XL C/C++ was released in 2001 and is the successor to the OS/390®, MVS™/ESA C/C++ compilers and C/370 compilers. The XL C/C++ and XL Fortran compilers for the Linux operating system running on Power hardware were released in 2003. The XL C/C++ for Linux on z Systems® was released in 2015, which takes advantage of the latest IBM z Systems® servers that run on selected Linux distributions.

Today IBM advances a thirty-plus year investment in the development of XL compilers for the Power series architectures, including the new POWER9™ processor. The quality infrastructure and leading-edge performance capabilities developed by the compiler team are among the finest in the industry. Each compiler executes over a million C, C++, or Fortran test scenarios. IBM's premier customer service ensures fast turnaround time and mandates timely solutions to customer issues. Both the compiler and processor design teams are strongly committed to maximizing performance on all Power processors.

Starting from V13.1.1, IBM XL C/C++ for Linux for little endian distributions combines the Clang front end infrastructure with the advanced optimization technology in the IBM compiler back end. Clang is a component of the LLVM open source compiler and toolchain project and provides the C and C++ language family front end for LLVM. For additional information about Clang, see the LLVM website at http://clang.llvm.org.

Starting from IBM XL C/C++ for Linux, V12.1, IBM XL C/C++ for AIX, V12.1, XL Fortran for Linux, V14.1, and XL Fortran for AIX, V14.1, the compiler provides full support for OpenMP V3.1.

Starting from V13.1.3, XL C/C++ for Linux for little endian distributions is C11 and C++11 compliant.

Starting from IBM XL C/C++ for Linux, V13.1.4 for little endian distributions and XL Fortran for Linux, V15.1.5 for little endian distributions, the Community Edition is available for download and deployment. The Community Edition is a no-charge, unlimited licensed product provided for developers who do not require official IBM support.

Starting from XL Fortran for Linux, V15.1.4 for little endian distributions, the compiler supports the CUDA Fortran programming model, which is a subset of the CUDA constructs, to exploit the NVIDIA GPUs. You can use the commonly used subset of CUDA Fortran that is provided by the compiler to offload computations to the NVIDIA GPUs. Starting from XL C/C++ for Linux, V13.1.5 for little endian distributions and XL Fortran for Linux, V15.1.5 for little endian distributions, the compiler offers OpenMP device constructs to offload compute-intensive parts of an application and associated data to the NVIDIA GPUs.

Starting from V16.1, XL C/C++ for Linux and XL Fortran for Linux are unified to the same version number for better clarification. Further, starting from this release, the compiler stops to ship the big endian distributions.

IBM XL C/C++ for Linux, V16.1.1 and XL Fortran for Linux, V16.1.1 provide the full support for OpenMP V4.0 and OpenMP V4.5.

In IBM XL C/C++ for AIX V16.1, in addition to the legacy IBM XL-based compiler front end, the compiler provides a Clang-based front end, which is invoked by the **xlclang** or **xlclang++** invocation command. The Clang-based front end supplies the full support of C11, C++11, and C++14 language standards and the enhanced GCC compatibility. In V16.1, XL C/C++ for AIX and XL Fortran for AIX are unified to the same version number for better clarification.

Chapter 3. Multiple platforms and language standard support

IBM XL compilers deliver extensive and industry leading optimization technology on operating systems including AIX, Linux, Linux on z Systems, z/OS, and z/VM®. They support and exploit the full performance capabilities of architectures such as IBM Power Systems™ and z Systems. The modular structure of the XL C/C++ and XL Fortran compilers delivers optimizations and functionality on multiple platforms to multiple languages, and IBM continues to enhance and improve its optimization technology to exploit the advancements in new hardware. The compiler products are derived from a common code base, thus the features and optimizations are tested in multiple languages on multiple platforms. A common code base makes source-level portability of applications between IBM platforms easier.

IBM XL compilers strive to maximize both the performance of scientific, technical, and commercial applications on server platforms, and application portability through multiple platform support and standards compliance. IBM XL compilers on Power architectures conform to the following standards and industry specifications:

- C/C++ standards
  - Information Technology - Programming languages - C, ISO/IEC 9899:1990
  - Information Technology - Programming languages - C, ISO/IEC 9899:1999
  - Information Technology - Programming languages - C++, ISO/IEC 14882:1998
  - Information Technology - Programming languages - C++, ISO/IEC 14882:2003(E)
  - Information Technology - Programming languages - Extensions for the programming language C to support new character data types, ISO/IEC TR 19769
  - Draft Technical Report on C++ Library Extensions, ISO/IEC TR 19768:2007
  - Information Technology - Programming languages - C, ISO/IEC 9899:2011 (XL C/C++ for Linux, starting from V13.1.2 for little endian distributions and XL C/C++ for AIX, V16.1 that is invoked by **xlclang/xlclang++** is C11 compliant.)
  - Information Technology - Programming languages - C++, ISO/IEC 14882:2011 (XL C/C++ for Linux, starting from V13.1.3 for little endian distributions and XL C/C++ for AIX, V16.1 that is invoked by **xlclang/xlclang++** is C++11 compliant.)
  - Information Technology - Programming languages - C++, ISO/IEC 14882:2014 (Starting from V13.1.2, XL C/C++ for Linux for little endian distributions provides partial support for C++14. XL C/C++ for AIX, V16.1 that is invoked by **xlclang/xlclang++** is C++14 compliant.)
- Fortran standards
  - ANSI X3.9-1978 (FORTRAN 77)
  - Information technology - Programming languages - Fortran, ISO/IEC 1539-1:1991(E) and ANSI X3. 198- 1992 (Fortran 90)
  - Information technology - Programming languages - Fortran, ISO/IEC 1539-1:1997 (Fortran 95)
  - Information technology - Programming languages - Fortran, ISO/IEC 1539-1:2004 (Fortran 2003)

  - Information technology - Programming languages - Fortran, ISO/IEC 1539-1:2010 (Starting from V14.1, XL Fortran provides partial support for Fortran 2008.)
  - Information technology - Programming languages - Fortran, ISO/IEC TS 29113 (Starting from V15.1, XL Fortran provides partial support for the Technical Specification for further interoperability with C.)
- Other supported industry specifications
  - Industry specifications that are supported by all XL compilers
    - OpenMP Application Program Interface V3.1
    - IEEE 754-2008 IEEE Standard for Floating-Point Arithmetic
    - IEEE POSIX 1003.2-1992 & X/Open CAE Specification, System Interface Definitions, Issue 4 and Issue 4 Version 2
    - A subset of GNU C and C++ language extensions
    - Common Fortran language extensions defined by other compiler vendors, in addition to those defined by IBM
    - Industry extensions that are found in Fortran products from various compiler vendors
    - Extensions specified in SAA Fortran
  - Industry specifications that are supported by XL C/C++/Fortran for AIX
    - AltiVec Technology Programming Interface Manual
    - Partial support for OpenMP Application Program Interface V4.0
    - Language extensions to support vector programming
  - Industry specifications that are supported by XL C/C++/Fortran for Linux
    - AltiVec Technology Programming Interface Manual
    - OpenMP Application Program Interface V4.0
    - OpenMP Application Program Interface V4.5
    - Language extensions to support vector programming

Standards conformance provides ease of code portability between multiple operating systems and hardware platforms. The IBM XL compilers also focus on showcasing processor performance against key UNIX and SPEC benchmarks.

IBM XL compiler availability

Table 1. IBM XL C/C++ compilers currently available (December 2018)

IBM XL C/C++ Compilers	Product Information
XL C for AIX	https://www.ibm.com/us-en/marketplace/xl-c-aix-compiler-power
XL C/C++ for AIX	https://www.ibm.com/us-en/marketplace/xl-cpp-aix-compiler-power
XL C/C++ for Linux	https://www.ibm.com/us-en/marketplace/xl-cpp-linux-compiler-power
XL C/C++ for z/VM	https://www.ibm.com/us-en/marketplace/xl-cpp-compiler-zvm
z/OS XL C/C++	https://www.ibm.com/us-en/marketplace/xl-cpp-compiler-zos

Table 2. IBM XL Fortran compilers currently available (December 2018)

IBM XL Fortran Compilers	Product Information
XL Fortran for AIX	https://www.ibm.com/us-en/marketplace/xl-fortran-aix-compiler-power
XL Fortran for Linux	https://www.ibm.com/us-en/marketplace/xl-fortran-linux-compiler-power

The XL C, XL C/C++, and XL Fortran compilers for AIX are also supported on PASE for i. For more information about PASE for i, see:

http://www.ibm.com/support/knowledgecenter/ssw_ibm_i_71/rzalf/rzalfintro.htm

Key features

IBM XL compilers for AIX and Linux for big endian distributions support both 32-bit and 64-bit modes and support many common features across languages and platforms wherever possible. The compilers help you exploit new POWER® hardware, and provide a wide range of tunable optimizations, such as:

- Instruction path length reduction
- Whole program analysis across compilation units
- Loop optimization for parallelism, locality and instruction scheduling
- Profile-directed feedback to guide optimization based on past runs
- Optimized OpenMP

Note: XL C/C++ for AIX, V16.1 that is invoked by **xlclang** or **xlclang++** does not support the OpenMP programming.

Starting from XL C/C++ for Linux, V13.1.6, XL Fortran for Linux, V15.1.6, XL C/C++ for AIX, V16.1, and XL Fortran for AIX, V16.1, the compilers utilize POWER9 technology. Built-in functions unlock POWER9 technology to enable you to have direct access to POWER9 features at the application level.

For debugging capability on AIX, you have the choice of any symbolic debugger that supports the AIX XCOFF executable format including dbx, TotalView, DDT, and IBM Debugger for AIX. IBM Debugger for AIX is included only with IBM XL C for AIX and IBM XL C/C++ for AIX. Additionally, the compilers offer directives and options to assist you in debugging optimized code.

Note: In V16.1, XL C/C++ for AIX stops to ship IBM Debugger for AIX.

On Linux, you can use debuggers including DDT, gdb, or TotalView. TotalView supports debugging OpenMP applications.

Starting from XL C/C++ for Linux, V13.1.5 for little endian distributions and XL Fortran for Linux, V15.1.5 for little endian distributions, you can offload compute-intensive parts of an application and associated data to the NVIDIA GPUs by using the OpenMP device constructs that are supported by the compiler. Starting from IBM XL C/C++ for Linux, V12.1, IBM XL C/C++ for AIX, V12.1, XL Fortran for Linux, V14.1, and XL Fortran for AIX, V14.1, the compiler provides full support for OpenMP V3.1. XL C/C++ for Linux, V16.1.1 and XL Fortran for Linux, V16.1.1 provide full support for OpenMP 4.0 and OpenMP 4.5.

XL compilers include support for the Basic Linear Algebra Subprograms (BLAS) routines and IBM Mathematical Acceleration Subsystem (MASS) libraries. On both AIX and Linux, BLAS tuned for POWER8® and MASS libraries tuned for POWER9 are provided. The compilers can also take advantage of IBM's highly tuned Engineering and Scientific Subroutine Library (ESSL) mathematical library on supported platforms.

XL compilers support vector data types and vector instruction generation on selected POWER architectures. The compilers can also automatically exploit vector processor features from source code loops that iterate through arrays, an optimization known as automatic SIMDization (for Single Instruction, Multiple Data). Specific vector types supported include VMX and VSX on AIX and Linux.

Chapter 4. IBM XL compiler architecture

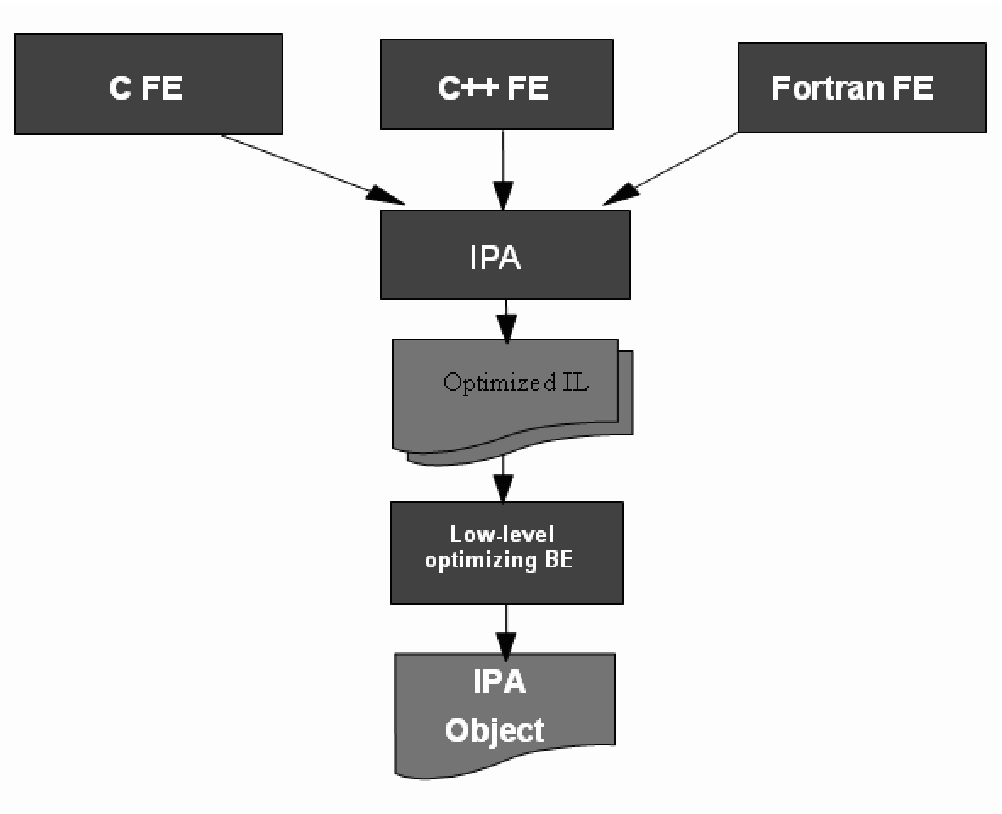
IBM XL compilers use a modular design, beginning with multiple language-parsing front-ends (FE) that generate a unified intermediate language (IL). The low-level optimizing back end (BE) processes the intermediate language, enabling common optimizations that apply to all languages.

Wide range of optimization levels

IBM XL compilers on Power architectures support the default optimization level **-O0** and optimization levels **-O2** to **-O5**. The highest optimization level, **-O5**, performs aggressive optimizations and advanced Interprocedural Analysis (IPA). The optimization process begins with the front-end emitting an intermediate language, which IPA analyzes and optimizes. The optimized intermediate language is then processed by the low-level optimizing back-end for further optimization and object code creation.

The following diagram illustrates the **-O5** compile-step process:

Figure 1: Compile path with IPA used on the compile step



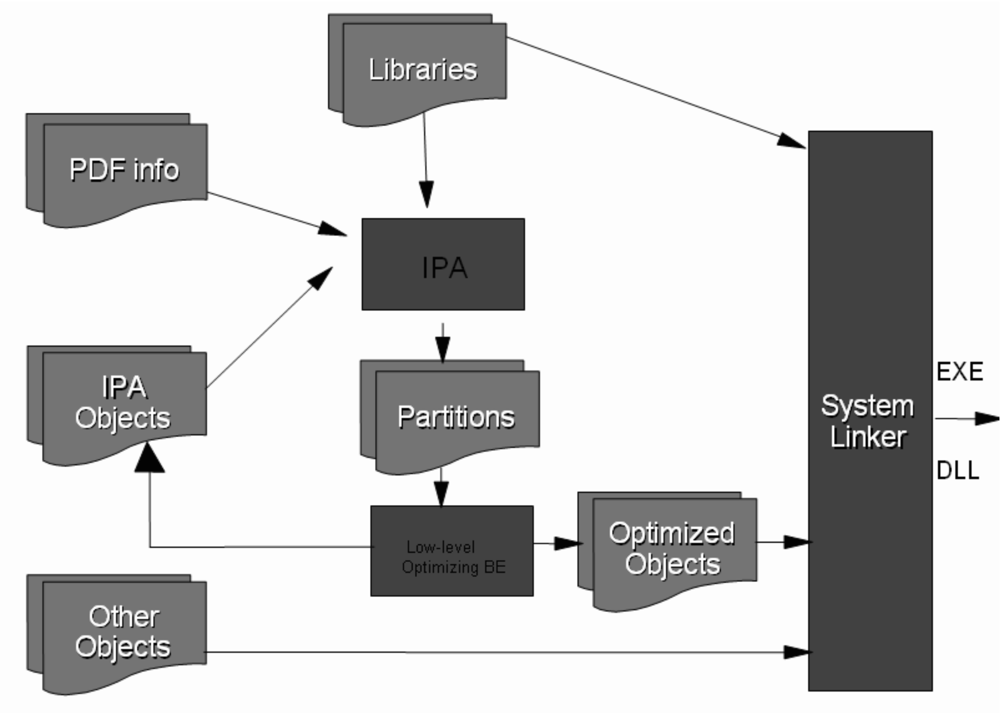
The IPA link-step optimization process combines many sources of information to achieve a view of the entire application. IPA performs the following tasks on the application link step:

- Examines libraries for all external symbols
- Considers any profile-directed feedback (PDF) optimization information

- Processes IPA objects from the IPA compile step and normal non-IPA objects

The IPA optimizer separates the result into partitions, which are subsequently processed by the low-level optimizing back end. The output of this process is an optimized set of objects that the system linker can use to create the final executable or shared library.

Figure 2: Link path when IPA is invoked



Chapter 5. Performance and IBM chip design

The key strength of the IBM XL compiler family is performance. The IBM XL compilers for AIX and Linux are unmatched in their ability to optimize and tune code for execution on Power platforms. The performance gain from years of compiler optimization experience can be seen in the release-to-release compiler improvements that benefit both older chips and the very latest Power processors. The newest of these processors is the POWER9 processor.

Announced in 2016, IBM's new POWER9 architecture is built for the cognitive era in the following aspects:

- The enhanced core and chip architecture is optimized for emerging workloads.
- As a processor family with scale-out and scale-up optimized silicon, it enables a range of platform optimizations and has extreme virtualization capabilities for the cloud.
- It is ideal for acceleration with the extreme CPU/accelerator bandwidth, seamless CPU/accelerator interaction, and broader application of heterogeneous computing.

Targeting the POWER9 technology, the latest IBM XL compiler products exploit these technology initiatives. The new instructions are fully supported by the compilers, requiring no programmer intervention. The compiler has always supported vector programming through programmer inserted built-in functions. This latest compiler version can now automatically generate vector instructions where the compiler determines the instructions that will provide benefits. In addition, there are significant improvements in parallelization technology to facilitate programming in the multicore environment.

IBM is a member of the Standard Performance Evaluation Corporation (SPEC). SPEC's mission is to identify and maintain standardized benchmarks that will drive high performance computing for many years. SPEC released SPEC CPU2006 in 2006. CPU2006 is a benchmark focused on a system's processor, memory subsystem, and compiler. IBM also continues to participate in the SPEC OMP suite that measures the performance of parallel benchmarks using OpenMP.

For more information regarding SPEC CPU and SPEC OMP, see:

www.spec.org

For more information about the Power Architecture®, see:

- www.ibm.com/systems/power/advantages/power.html
- www.ibm.com/power
- www.power.org

For more information about performance benchmarks of the Power processors, see:

- www.spec.org
- www.tpc.org/tpcc/

You can also subscribe to the very informative and wide ranging IBM developerWorks® newsletters at:

www.ibm.com/developerworks/newsletter/

Chapter 6. Language standards support

The IBM XL compiler team maintains membership in the ISO C, C++, and Fortran committees, representing IBM. This allows IBM to help drive the strategic direction of the languages as standards evolve, and deliver compiler enhancements in support of new language standards as ratification occurs. The C/C++ and Fortran teams are also members of the OpenMP consortium.

XL C and XL C/C++

IBM XL C for AIX, V13.1.3 fully supports C99.

IBM XL C/C++ for AIX, V13.1.3 and XL C/C++ for Linux, V13.1 for big endian distributions support C99 and C++03 and selected features of C11 and C++11.

XL C/C++ offers support of Boost, an open source set of libraries that provides a wide range of capabilities to accelerate your application development efforts. Boost makes C++ programming more elegant, robust, and productive. XL C++ has attained a high degree of compatibility with Boost since Version 7.0 and continues to support Boost as new releases appear. Each version of the compiler is fully tested on one version of Boost, usually the latest. Both the C and C++ compilers plan to continue to track close compatibility with the GNU compilers shipped by the latest Linux operating systems, allowing you to build many open source applications. XL compilers include enhancements to leverage automatic parallelization. The interprocedural optimizer includes enhancements to support many specialized C++ optimizations for templates, exception handling, and virtual functions.

Starting from XL C for AIX, V13.1, XL C/C++ for AIX, V13.1, and XL C/C++ for Linux, V13.1, the following C11 and C++11 features are supported:

C11 features
- `_Noreturn` function specifier
- Anonymous structures (C only)
- Anonymous unions (C only)
- Complex type initializations
- Generic selection (C only)
- Static assertions
- `typedef` redeclarations

C++11 features (C++ only)
- Auto type deduction
- C99 `long long`
- C99 preprocessor features adopted in C++11
- `decltype`
- Defaulted and deleted functions
- Delegating constructors
- Explicit conversion operators
- Explicit instantiation declarations

- Extended friend declarations
- Forward declaration of enumerations
- Generalized constant expressions
- Inline namespace definitions
- `nullptr`
- Reference collapsing
- Right angle brackets
- Rvalue references
- Scoped enumerations
- `static_assert`
- Trailing comma allowed in `enum` declarations
- Trailing return type
- Variadic templates

Starting from V13.1.2, IBM XL C/C++ for Linux for little endian distributions starts to support the C++14 standard, and the following features are added into this release:

- Polymorphic lambda expressions
- Variable templates

Starting from V13.1.3, IBM XL C/C++ for Linux for little endian distributions is C11 and C++11 compliant.

Starting from V16.1, IBM XL C/C++ for Linux, more C++14 features are supported. The full feature list is as follows:

- Binary integer literals
- Digit separators
- Polymorphic lambda expressions
- Relaxing constraints on `constexpr` functions
- Return type deduction for normal functions
- The `deprecated` attribute
- Variable templates

IBM XL C/C++ for AIX, V16.1 that is invoked by **xlclang** or **xlclang++**, which is the Clang-based front end of the compiler, provides the full support for the C11, C++11, and C++14 standards.

XL Fortran

IBM XL Fortran compilers for AIX and Linux support the following Fortran standards:

- The full American National Standard Fortran 77 language (Fortran 77) as defined in document:
  - American National Standard Programming Language Fortran 77, ANSI X3.9-1978
- The full American National Standard Fortran 90 language (Fortran 90) as defined in documents:
  - American National Standard Programming Language Fortran 90, ANSI X3.198-1992

  - Information technology - Programming languages - Fortran, ISO/IEC 1539-1:1991(E)
- The full ISO Fortran 95 language standard (Fortran 95) as defined in document:
  - Information technology - Programming languages - Fortran - Part 1: Base language, ISO/IEC 1539-1:1997
- The full ISO Fortran 2003 language standard (Fortran 2003) as defined in document:
  - Information technology - Programming languages - Fortran - Part 1: Base language, ISO/IEC 1539-1:2004
- Partial support for the ISO Fortran 2008 language standard (Fortran 2008), such as the submodules, separate module subprograms, the CONTIGUOUS attribute, implied-shape array, as defined in document:
  - Information technology - Programming languages - Fortran - Part 1: Base language, ISO/IEC 1539-1:2010

  The compliance status to the Fortran 2008 language standard features is available at http://ibm.biz/Fortran2008Status.
- Partial support for Technical Specification for further interoperability with C (TS 29113), ISO/IEC TS 29113:2012, such as:
  - Interoperability of assumed-rank arguments
  - Interoperability of assumed-type arguments
  - Interoperability of allocatable and pointer arguments
  - Interoperability of optional arguments
  - Interoperable variables in asynchronous communication
  - The ISO_Fortran_binding.h header file

  The compliance status to TS 29113 is available at http://ibm.biz/FortranTS29113Status.

Chapter 7. Industry specifications

The IBM XL compiler family is deeply involved in parallel computing and high performance computing. IBM XL compilers implement the OpenMP specification for shared memory programming model and the VMX/VSX programming interface for AIX and Linux.

Starting from the following releases, full support for the OpenMP API Version 3.1 is introduced:

- XL C for AIX, V12.1
- XL C/C++ for AIX, V12.1
- XL C/C++ for Linux, V12.1 for big endian distributions
- XL C/C++ for Linux, V13.1.2 for little endian distributions
- XL Fortran for AIX, V14.1
- XL Fortran for Linux, V14.1 for big endian distributions
- XL Fortran for Linux, V15.1.2 for little endian distributions

Starting from the following releases, partial support for the OpenMP API Version 4.0 is introduced:

- XL C for AIX, V13.1
- XL C/C++ for AIX, V13.1
- XL C/C++ for Linux, V13.1 for big endian distributions
- XL C/C++ for Linux, V13.1.2 for little endian distributions
- XL Fortran for AIX, V15.1
- XL Fortran for Linux, V15.1 for big endian distributions
- XL Fortran for Linux, V15.1.2 for little endian distributions

Starting from the following releases, partial support for the OpenMP API Version 4.5 is introduced:

- XL C/C++ for Linux, V13.1.3 for little endian distributions
- XL Fortran for Linux, V15.1.3 for little endian distributions

The full support for OpenMP API Version 4.0 and OpenMP API Version 4.5 is introduced in the following releases:

- XL C/C++ for Linux, V16.1.1
- XL Fortran for Linux, V16.1.1

Chapter 8. IBM and GNU

While ensuring that the IBM XL compiler family has always offered diverse functionality and excellent performance to their users, IBM also has a long standing interest in the development of the GNU compiler. IBM contributes to GCC development, including large parts of the original GCC code for RS/6000®, and the 64-bit retarget for Linux on Power. Further technical contributions include an instruction scheduler, as well as the rights to the graph coloring register allocation patent for use in GCC. Today IBM still seeks to make contributions to the GNU effort in the areas of single static assignment (SSA) optimizations, auto-vectorization, and loop optimization. With these additions, IBM and many others help the GNU compiler grow. IBM XL compilers will also continue to adopt the most useful GNU extensions. In addition to the GNU extensions, with IBM XL compilers, users can achieve the best performance on Power processors coupled with premium customer support.

The intent of the GNU compatibility project is to provide support in the following stages:

- Enable compilation of POSIX compliant source on Linux using the compiler and the GNU headers
- Add the most common GNU language extensions
- Implement full support for enough of the extensions to compile the Linux kernel and most popular Linux applications

The latest XL C/C++ compilers include most of the common GNU language extensions.

Starting from XL C/C++ for Linux, V13.1.1, an additional compiler for Linux for little endian distributions is added, providing a greater level of GNU source and binary compatibility. In addition to creating applications using object files produced by XL compilers or GNU compilers alone, you can also create applications using the object files that are produced by both compilers. The C/C++ compiler for the Linux little endian architecture supports many GCC options as it leverages the Clang infrastructure from the open source community for a portion of its compiler front end. You can find a complete list of those options in "Supported GCC options" in the *XL C/C++ for Linux Compiler Reference*.

IBM XL C/C++ for AIX, V16.1 provides a Clang-based front end that is invoked by the **xlclang** or **xlclang++** invocation command. Many GCC options and GCC pragmas are supported by the Clang-based front end, which means fewer migration efforts are needed when migrating your GCC programs to IBM XL C/C++ for AIX. You are encouraged to use GCC options instead of legacy XL compile options with the Clang-based front end wherever possible. However, no binary compatibility between code generated by IBM XL C/C++ for AIX and GCC is provided. You cannot mix and match binaries generated by **gcc** or **g++** and those by **xlclang** or **xlclang++**.

The IBM XL compilers also have a close affinity with third party tool chains, and seek to adapt to GNU tools such as gdb. There are also specific AIX and Linux performance tools, which can help identify performance bottlenecks.

Chapter 9. Summary

At a basic level, compilers are a bridge between your applications and the hardware architectures on which you run your business. IBM compilers are designed to unleash the full power of IBM processors, including those for the different architectures shipped in the popular IBM Power Systems and z Systems.

IBM XL compilers are designed to improve programmer productivity. The state-of-the-art compilation technology enables programmers to exploit leading-edge performance of the new hardware without source code changes. Developers only need to focus on the logic of the applications and let the compiler figure out the best way to transform and optimize the code generation for the system the application will run on.

Chapter 10. Trial versions, Community Editions, and purchasing

In addition to the full versions, IBM provides the following product editions:

- Community Editions for IBM XL C/C++ for Linux, V16.1.1 and XL Fortran for Linux, V16.1.1, which are no-charge and fully functional editions for unlimited production use.
- Trial versions for IBM XL C/C++ for AIX, V16.1 and XL Fortran for AIX, V16.1, which are no-charge and fully functional versions for an evaluation period of 60 days.

The following link contains the XL C/C++ compiler product web pages where you can find download information about full versions, the Community Editions, and the trial versions if available:

https://www.ibm.com/us-en/marketplace/ibm-c-and-c-plus-plus-compiler-family

Information on how to buy IBM XL C/C++ is also available at this website.

The following link contains the XL Fortran compiler product web pages where you can find download information about full versions, the Community Editions, and the trial versions if available:

https://www.ibm.com/us-en/marketplace/ibm-fortran-compiler-family

Information on how to buy IBM XL Fortran is also available at this website.

Chapter 11. Contacting IBM

IBM welcomes your comments. You can send them to compinfo@cn.ibm.com.

Printed in USA