IBM XL C/C++ for Linux on z Systems, V1.2

IBM

High-performance XL C/C++ compiler for Linux on z Systems

IBM® XL C/C++ for Linux on z Systems™, V1.2 is a C/C++ compiler for application development that takes advantage of the latest IBM z Systems servers that run on selected Linux distributions. This product strengthens the platform, exploits the z Systems environment, and provides superior performance by generating highly optimized code for execution on IBM z Systems. It is built on the performance gains from many years of IBM compiler optimization experience with existing XL C/C++ compilers that are available for IBM z/OS®, IBM z/VM®, IBM AIX® and Linux on IBM Power Systems™. With XL C/C++ for Linux on z Systems, you can create and port applications for execution on the next generation of IBM z Systems supporting selected Linux distributions while maximizing hardware utilization with fast performance.

Highlights

XL C/C++ for Linux on z Systems, V1.2 features the following:

- Vector single instruction, multiple data (SIMD) and auto-SIMD support for the new vector extension facility (SIMD) instructions available on IBM z13™ servers
- Generation of highly optimized code exploiting z Systems servers
- Support of programming language standards, including C11 compliance and additional features of the latest C++11 and C++14 standards
- High level of source compatibility with GNU Compiler Collection (GCC) while providing binary coexistence
- Enhancements to the ATLAS and IBM MASS libraries to take advantage of the vector facility in z13 for potential performance improvements

Vector SIMD and auto-SIMD support

XL C/C++ for Linux on z Systems, V1.2 supports the vector single instruction, multiple data (SIMD) and auto-SIMD instructions that are provided with the z13 vector extension facility and corresponding vector programming language extensions. This exploitation of the vector facility in z13 helps improve the performance of your XL C/C++ applications.

Generation of highly optimized code

As the newest addition to the IBM compiler family, IBM XL C/C++ for Linux on z Systems, V1.2 brings mature IBM compiler technology to Linux distributions running on IBM z Systems servers. XL C/C++ supports generation of highly optimized code exploiting z Systems servers, adheres to programming standards, and provides GNU Compiler Collection (GCC) compatibility that allows you to easily port your applications to Linux distributions running on z Systems servers.

The Clang infrastructure

XL C/C++ for Linux on z Systems leverages the Clang infrastructure from the open source community for a portion of its compiler front 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. XL C/C++ for Linux on z Systems combines the Clang front end infrastructure with the advanced optimization technology in the IBM compiler back end.

New architecture and tune compiler options for the z Systems technology

XL C/C++ for Linux on z Systems supports the generation of z Systems hardware running SUSE Linux Enterprise Server 11 (SLES 11), SLES 12, Red Hat Enterprise Linux 6 (RHEL 6), and RHEL 7.

The -qarch compiler option specifies the processor architecture for which code is generated. The -qtune compiler option tunes instruction selection, scheduling, and other architecture-dependent performance enhancements to run best on a specific hardware architecture. With XL C/C++ for Linux on z Systems, architecture and tune compiler suboptions are available to specify code generation for the new IBM z13.

In the XL C/C++ for Linux on z Systems compiler, -march=z13 instructs the compiler to produce code that can exploit the new instructions in the z13 processors and -mtune=z13 enables optimizations specifically for the z13 processors.

Optimization capabilities

One of the key strengths of the XL C/C++ for Linux on z Systems is optimization. This compiler offers the benefit of optimization technology, which evolved at IBM since the late 1980s combining extensive knowledge of the hardware with a comprehensive understanding of compiler technology and what users look for in a compiler when building user applications, especially for new workloads, such as big data, cloud, mobile, and social. The optimizations can decrease execution time and make your applications run faster, producing code that is highly tuned for execution on z Systems. The optimizer includes three base optimization levels, which allow you to choose from minimal optimization to intense program analysis that range from local basic block to subprogram unit scopes, file-level, and whole-program analysis. The higher the optimization level, the more intense the program analysis becomes. Sophisticated optimization techniques such as interprocedural analysis can also be applied to your code.

Levels of optimization include:

-O0　Provides minimal optimization, which is best for debugging.

-O2　Provides strong low-level optimization and benefits most programs.

-O3　Provides intense low-level optimization analysis and base-level loop analysis.

In addition, -qipa can be used for aggressive optimization of the whole program, including aggressive data flow analysis and loop transformations.

XL C/C++ for Linux on z Systems, V1.2 also supports profile-driven optimization; monitoring code is produced to generate a runtime profile, which helps tune application performance.

Programming language standards

XL C/C++ for Linux on z Systems, V1.2 strives to maximize performance and functionality of your scientific, technical, and commercial applications through standards compliance. With representation from the IBM compiler development team on the ISO/ANSI C and C++ committees, IBM is in a position to understand and influence the latest updates, clarifications, and recommendations to the C and C++ standards. XL C/C++ for Linux on z Systems, V1.2 conforms with the following language standards:

* Information Technology - Programming Languages - C, ISO/IEC 9899:1990 (also known as C90)
* Information Technology - Programming Languages - C, ISO/IEC 9899:1999 (also known as C99)
* Information Technology - Programming Languages - C++, ISO/IEC 14882:1998 (also known as C++98)
* Information Technology - Programming Languages - C++, ISO/IEC 14882:2003(E) (also known as C++2003)
* Information Technology - Programming Languages - Extensions for the programming language C to support new character types, ISO/IEC RT 19769
* Draft Technical Report on C++ Library Extensions, ISO/IEC TR 19768:2007
* Information Technology - Programming Languages - C, ISO/IEC 9899:2011 (also known as C11)
* Information Technology - Programming Languages - C++, ISO/IEC 14882:2011 (also known as C++11) (partial support for C++11)
* Information Technology - Programming Languages - C++, ISO/IEC 14882:2014 (also known as C++14) (partial support for C++14).

XL C/C++ for Linux on z Systems, V1.2 supports the following C++14 features:

* Polymorphic lambda expressions
* Variable templates

XL C/C++ for Linux on z Systems, V1.2 supports the following C++11 features:

- Alignment support
- constexpr
- Explicit overrides and final
- Generalized attributes
- Inheriting constructors
- Local and unnamed types as template arguments
- Monomorphic lambdas expressions
- New character types
- New definitions of POD types
- noexcept Non-static data member initializers
- Range-based for
- Raw string literals
- ref-qualifiers
- Template aliases
- Unicode names (UCN) and unicode literals
- Uniform initialization
- Unrestricted unions
- User-defined literals

GNU compatibility

XL C/C++ for Linux on z Systems provides a greater level of GNU source compatibility. Porting of applications, originally developed with gcc and g++ compilers, is easier than ever before. There is also binary compatibility with GNU-built objects, archives, and shared objects. You now have the versatility to use the IBM compiler to build parts of your application that will benefit from the higher performance offered and still bind the IBM and GNU compiled parts together in a single application.

IBM Mathematical Acceleration Subsystem (MASS) and Automatically Tuned Linear Algebra Software (ATLAS)

Enhancements to the ATLAS and IBM MASS libraries are also delivered to take advantage of the vector facility in z13 for potential performance improvements. The IBM MASS library is intended for accelerated execution of elementary mathematical functions that serves as a higher performance alternative to the standard math library. The ATLAS library is designed to provide linear algebra function support for BLAS (Basic Linear Algebra Subprograms) and LAPACK (Linear Algebra PACKage) functions that are routinely used in Business Analytics and Optimization solutions.

XL C/C++ for Linux on z Systems, V1.2 delivers both single-threaded and multithreaded versions of the ATLAS libraries that are tuned and optimized for z13 processors. These two libraries provide a powerful framework for development of new Business Analytics workloads, porting math-intensive workloads from other platforms, and can help accelerate Business Analytics workloads on z Systems.

System requirements

The following table presents the system requirements for XL C/C++ for Linux on z Systems, V1.2:

Operating system	Hardware
XL C/C++ for Linux on z Systems, V1.2 supports the following operating systems: • Red Hat Enterprise Linux for IBM System z® 6.3 • Red Hat Enterprise Linux for IBM System z 7.0 • Red Hat Enterprise Linux for IBM System z 7.1 • Red Hat Enterprise Linux for IBM System z 7.2 • SUSE Linux Enterprise Server for System z 11 SP3 • SUSE Linux Enterprise Server for System z 12	XL C/C++ for Linux on z Systems, V1.2 runs on the following servers: • IBM z13 (z13) • IBM zEnterprise® EC12 (zEC12) or IBM zEnterprise BC12 (zBC12) • IBM zEnterprise 196 (z196) or IBM zEnterprise 114 (z114) • IBM System z10™ Enterprise Class (z10 EC) or IBM System z10 Business Class (z10 BC)

For more information