IBM XL C/C++ for Linux, V13.1.5

Improve application performance and developer productivity using the latest IBM XL C/C++ compiler for Linux

Highlights

IBM® XL C/C++ for Linux, V13.1.5:

- Includes a compiler to support Linux for little endian distributions.
- Offers an alternative no-charge, fully functional community edition.
- Leverages the capabilities of the latest POWER8® architecture.
- Maximizes application performance through industry leading optimization technology.
- Eases application migration to Power Systems™.
- Supports a number of OpenMP 4.5 features including the constructs to allow applications offloading computation and data to the NVIDIA GPU (available with select POWER8 server models).

Delivers an XL C/C++ compiler to support Linux for little endian distributions

IBM XL C/C++ for Linux contains two C/C++ compilers that support:

- Linux for little endian distributions on IBM Power® servers configured for little endian mode, supporting Ubuntu Server 14.04, 14.10, and 16.04, SLES 12, SLES 12 SP1, RHEL 7.1, RHEL 7.2, RHEL 7.3, and CentOS 7
- Linux for big endian distributions on IBM Power servers configured for big endian mode, supporting RHEL 6, RHEL 7, and SLES 11

XL C/C++ for Linux for the little endian architecture leverages the Clang infrastructure from the open source community for a portion of its compiler frontend. 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. The XL C/C++ compiler combines the Clang front-end infrastructure with IBM advanced optimization technology and code generation.

XL C/C++ for Linux for little endian distributions also provides a great level of GNU compatibility.

Offers a no-charge, fully functional community edition

XL C/C++ for Linux Community Edition for little endian distributions is a no-charge, fully functional C/C++ compiler. The release of the community edition allows for convenient availability of the XL C/C++ compiler so that developers can experience the advantages of IBM compiler technology on the POWER8 platform. This product is not warranted and does not provide for any subscription, service, or support.

The full XL C/C++ for Linux compiler priced per user continues to be available for order. To obtain full warranty and world-class IBM support, consider licensing the full XL C/C++ for Linux compiler.

Leverages the capabilities of the latest POWER8 architecture

XL C/C++ for Linux, V13.1.5 generates code that leverages the capabilities of the latest POWER8 architecture. Compiler suboptions for architecture and tuning specify code generation on the POWER8 processor architecture. `-qarch=pwr8` instructs the compiler to produce code that can fully exploit the POWER8 architecture. `-qtune=pwr8` enables optimizations, such as instruction scheduling, that maximize performance on POWER8 systems, while allowing for binary compatibility with previous POWER® processors.

XL C/C++ provides built-in functions for direct programmer access to the POWER architecture. While most programmers will rely on the compiler to exploit processor features automatically, built-in functions give you an easy way to access specific instructions or processor features using C or C++ function call syntax and C or C++ variables. XL C/C++ for Linux, V13.1.5 provides built-in functions supporting POWER8 features such as vector processing, cryptography, cache management, and transactional memory.

The Mathematical Acceleration Subsystem (MASS) libraries contain frequently used elementary and special mathematical functions that enable improved performance over the corresponding standard system library functions. These highly tuned MASS libraries are enhanced to support the POWER8 architecture. In addition to the scalar library, MASS for POWER8 includes both a vector version (for arbitrary length arrays) and a SIMD or Single-Instruction Multiple Data version (for the vector datatypes). All the vector and SIMD POWER8 MASS libraries exploit the POWER8 vector instruction set, as does the scalar library for Linux for little endian distributions.

Maximizes application performance through industry-leading compiler optimization technology

The optimization and hardware exploitation features in IBM XL C/C++ help improve programming productivity. The XL C/C++ compiler generates code that delivers leading-edge performance from existing and new hardware, often with no source code changes.

XL C/C++ for Linux implements the GCC symbol visibility attributes, pragmas, and options that describe whether and how an entity that is defined in one module can be referenced or used in other modules. Using the visibility attributes for entities, you can get the following benefits:

- Decreasing the size of shared libraries
- Reducing the chance of symbol collision
- Allowing more optimization for the compile and link phases
- Improving the efficiency of dynamic linking

Profile directed feedback (PDF) optimization collects information about an application run with typical input data and then applies transformations to the program based on that information. PDF can ensure that the performance of the application is optimized for its important inputs. Application profile monitoring and profile directed feedback capabilities minimize the need for manual tuning to achieve desirable performance on large and complex applications.

Eases application migration to IBM Power Systems

Make your applications portable with the XL compilers, which offer industry compliant programming languages and extensions. XL compilers help programmers easily maintain and run their applications on IBM systems.

IBM XL C/C++ conforms to the following programming language specifications for C/C++: C89, C99, C++ 98, C++03, C11, and C++11 standards.

Augmenting the standardized language levels, the XL C/C++ compiler has implemented C and C++ language extensions to support vector programming and a subset of GNU C and C++ language extensions. In addition, the XL C++ compiler maintains close support of Boost C++ library releases.

Supports a number of OpenMP 4.5 features

OpenMP is a portable and scalable programming model that gives programmers a simple, flexible, and standard interface for developing parallel applications for platforms ranging from the desktop to the supercomputer. XL C/C++ for Linux provides full support for OpenMP 3.1 and partial support for OpenMP 4.5.

Starting from V13.1.5, the OpenMP 4.5 features such as device constructs and data mapping are added so that applications can accelerate by offloading compute-intensive parts of an application and associated data to the NVIDIA GPU that comes with a number of POWER8 servers. These device constructs include target, teams, distribute, and target data. The runtime functions omp_get_default_device, omp_get_num_teams, omp_get_team_num, and omp_is_initial_device are supported to query the target environment. When combined with the POWER8 CPU, the NVIDIA GPU provides a unique platform for heterogeneous and high-performance computing, which has proven to be highly efficient for running several technical computing workloads. This computational capability is built on top of massively parallel and multi-threaded cores within the NVIDIA GPUs and the IBM POWER8 processors. Parallel operations within applications, such as data analysis or HPC (High performance computing) workloads, can be offloaded to GPUs.

XL C/C++ for Linux, V13.1.5 for little endian distributions also provides high performance computing with support for Power System servers such as the S822LC (Minsky), which contain two POWER8 CPUs, up to four NVIDIA Tesla P100 GPUs, and NVIDIA's accelerated NVLink interconnect.

Summary

IBM compilers allow applications to take advantage of virtually all the hardware exploitation features provided by IBM processors including POWER8. By utilizing leading-edge optimization technologies in IBM compilers, organizations can improve their return on investment in hardware assets, while increasing programmer productivity.

Organizations often wait until they upgrade their hardware to upgrade their compilers. However, given that the compilers can deliver significant improvements in application performance and programmer productivity, compilers offer a cost-effective way to get more out of existing technology. By periodically upgrading compilers, programmers can take advantage of new language, usability and optimization features, and stay ahead of competitors on the technology curve.

For more information

To learn more about the IBM XL C/C++ for Linux compiler, contact your IBM representative, IBM Business Partner, or visit XL C/C++ for Linux at http://ibm.biz/xlcpp-linux.

Get started today by downloading a Community Edition of XL C/C++ for Linux at www.ibm.com/developerworks/downloads/r/xlcpluslinux/.

November 2016