

Table of Contents

Platforms and compilers supported by LHCb production software.....	1
Supported compilers and platforms (binary distribution available for x86_64 architecture, optimised and debug builds).....	1
Production platforms (binary distribution available in optimised and debug modes).....	2
Special development platforms (builds for master branch in the nightlies).....	2
Other ports.....	2
Old production platforms.....	2

Platforms and compilers supported by LHCb production software

Source code and binaries for supported platforms are available in CVMFS (directory /cvmfs/lhcb.cern.ch/lib/lhcb - \$LHCBRELEASES). The following platform+compiler combinations are supported.

See also:

- Installing the LHCb Software on Linux Platforms not Officially Supported
- Using the CernVM virtual machine
- CodeAnalysisTools
- Supported platforms for distcc at CERN
- Installation and distribution of LHCb software is also tested via docker containers. Instructions can be found here.

On lxplus, you can check which compiler/platform combination is supported for a given application version.
e.g.: `lb-run --list-platforms DaVinci/v44r11p3`

Supported compilers and platforms (binary distribution available for x86_64 architecture, optimised and debug builds)

- On master branch (Run3 software development, production and analysis)
 - ◆ gcc 9 (libstdc++ reference) with C++17, on centos7
 - ◆ clang 8.0 (configuration x86_64-centos7-clang8-opt). Not fully supported (some failing compilations) but distribution available on cvmfs
- On run2-patches branch (Run1+Run2 software development and analysis)
 - ◆ gcc 9 (libstdc++ reference) with C++17, on centos7
 - ◆ clang 8.0 (configuration x86_64-centos7-clang8-opt). Not fully supported (some failing compilations) but distribution available on cvmfs
- On 2018-patches branch (2018 Moore, Reco18, Stripping34 (pp), Stripping35 (IFT) maintenance and incremental restriping)
 - ◆ gcc 6.2 with C++14 compilation enabled and new gcc ABI, on slc6 and centos7
 - ◆ gcc 7 with C++14 compilation enabled and new gcc ABI, on centos7
- On 2017-patches branch (2017 Moore, Reco17, Stripping29 (pp), Stripping33 (IFT) maintenance and incremental restriping)
 - ◆ gcc 6.2 with C++14 compilation enabled and new gcc ABI, on slc6 and centos7
 - ◆ gcc 4.9 with C++14 compilation enabled (subset supported by gcc49), on slc6
- On 2016-patches branch (2016 Reco16, Stripping28 (pp), Stripping30 (IFT) maintenance and incremental restriping)
 - ◆ gcc 4.9 with C++14 compilation enabled (subset supported by gcc49), on slc6
- On hlt-2016-patches branch (2016 Moore maintenance)
 - ◆ gcc 4.9 with C++14 compilation enabled (subset supported by gcc49), on slc6
- On reco15-patches branch (2015 Moore and Reco15 maintenance)
 - ◆ gcc 4.9 with C++11 only, on slc6 * On stripping24-patches branch (2015 Stripping24 (pp) maintenance and incremental restriping)
 - ◆ gcc 4.9 with C++14 compilation enabled (subset supported by gcc49), on slc6
- On stripping21-patches branch (2011 and 2012 Stripping21 (pp) incremental restriping)
 - ◆ gcc 4.9 with C++14 compilation enabled (subset supported by gcc49), on slc6
- On digi14-patches branch (digi14 digitisation maintenance)

- ◆ gcc 4.9 with C++11 only, on slc6
- ◆ gcc 4.8 with C++11, on slc6
- On sim09-patches branch (sim09 simulation maintenance)
 - ◆ gcc 4.9 with C++11 only, on slc6
 - ◆ gcc 4.8 with C++11, on slc6
- On stripping21-firstpass-patches branch (2011 and 2012 Stripping21 (pp) first pass maintenance)
 - ◆ gcc 4.8 with C++11, on slc6
- On reco14-patches branch (2011 and 2012 Reco14 maintenance)
 - ◆ gcc 4.6 (C++11 compilation not supported) on slc5
- On hlt-2012-patches branch (2012 Moore maintenance)
 - ◆ gcc 4.6 (C++11 compilation not supported) on slc5
- On hlt-2011-patches branch (2011 Moore maintenance)
 - ◆ gcc 4.3 (C++11 compilation not supported) on slc5

Production platforms (binary distribution available in optimised and debug modes)

- CentOS7 with gcc 9.* compiler in 64-bit mode (configurations x86_64-centos7-gcc9-opt, x86_64-centos7-gcc9-dbg (using gcc -Og option))
 - ◆ Built with -m sse4.2
 - ◆ Available for Run 3 software (master branch) and Run 1 and Run 2 software (run2-patches branch)

Special development platforms (builds for master branch in the nightlies)

- avx2+fma (used for throughput tests)
- do0 (debug without optimizations)
- skylake_avx512+vecwid256 (testing)

Other ports

- ARM
 - ◆ A port to ARM is ongoing. Main difficulty is vectorized code (vectorclass and VC library specifically), reverted to scalar on ARM
 - ◆ Brunel was successfully ported in 2013 to the ARM processor. See here for R&D for the ARM done at that time
- AMD
 - ◆ Plan to integrate AMD machines in the nightlies
- There is no plan for a port to Mac OSX 10.*. Some old instructions on building from source can be found here.

Old production platforms

- CentOS7 with gcc 8.* compiler in 64-bit mode (configurations x86_64-centos7-gcc8-opt, x86_64-centos7-gcc8-dbg (using gcc -Og option))
 - ◆ Built with -m sse4.2
 - ◆ Available for Run 3 software (master branch) until v50r6 stack and Run 1 and Run 2 software (run2-patches branch) until v45r2 stack
 - ◆ Default platform on lxplus since 11th June 2019.

SupportedPlatforms < LHCb < TWiki

- CERN Scientific Linux 6 (SLC6) with gcc 8.* compiler in 64-bit mode (configurations x86_64-slc6-gcc8-opt, x86_64-slc6-gcc8-dbg (using gcc -Og option))
 - ◆ Built with -m sse4.2
 - ◆ Available for Run 3 software (master branch) until v50r4 stack and Run 1 and Run 2 software (run2-patches branch) until v45r0 stack
 - ◆ Default platform on lxplus6 since 11th June 2019
- CERN Scientific Linux 6 (SLC6) with gcc 6.2.* compiler in 64-bit mode (configurations x86_64-slc6-gcc62-opt, x86_64-slc6-gcc62-dbg (using gcc -Og option))
 - ◆ As of Gaudi v28r1, built with -m sse4.2
 - ◆ Default platform on lxplus from 19th February 2018 to 10th June 2019.
 - ◆ Available only on 2017-patches and 2018-patches branches
- CentOS7 with gcc 7.* compiler in 64-bit mode (configurations x86_64-centos7-gcc7-opt, x86_64-centos7-gcc7-dbg (using gcc -Og option))
 - ◆ Built with -m sse4.2
 - ◆ Available only on 2018-patches branch
- CERN Scientific Linux 6 (SLC6) with gcc 4.9.* compiler in 64-bit mode (configurations x86_64-slc6-gcc49-opt, x86_64-slc6-gcc49-dbg (using gcc -Og option))
 - ◆ C++14 features (subset supported by gcc49) were allowed as of LHCb v41r* software stack (compatibility with gcc48 platforms dropped) but were enabled also for v40r*
 - ◆ Supported up to 2017-patches stack, discontinued as from LHCb v43r* stack
- CERN Scientific Linux 6 (SLC6) with gcc 4.8.* compiler in 64-bit mode (configurations x86_64-slc6-gcc48-opt, x86_64-slc6-gcc48-dbg (using gcc -Og option) and x86_64-slc6-gcc48-do0 (using -O0, only works with cmake).
 - ◆ gcc 4.8 (libstdc++ reference) with C++11 compilation enabled.
 - ◆ C++11 features are supported as of Gaudi v25r0 + LHCb v37r0 (compatibility with gcc46 platforms dropped)
 - ◇ See here for recipes to fix common C++11 compilation errors.
 - ◆ Discontinued as from LHCb v41r* software stack.
- CERN Scientific Linux 6 (SLC6) with gcc 4.6.* compiler in 64-bit mode (CMT configurations x86_64-slc5-gcc46-opt, x86_64-slc5-gcc46-dbg)
 - ◆ Not available for projects based on Gaudi v25r0 or greater
 - ◆ C++11 compilation is not supported for this platform
 - ◆ See also here and here for porting and migration issues
- CERN Scientific Linux 5 (SLC5) with gcc 4.6.* compiler in 64-bit mode (CMT configurations x86_64-slc5-gcc46-opt, x86_64-slc5-gcc46-dbg)
 - ◆ Not available for projects based on Gaudi v25r0 or greater
 - ◆ C++11 compilation is not supported for this platform
 - ◆ Help and suggestions for porting the code to gcc 4.6 can be found on Gcc46PortIssues.
- CERN Scientific Linux 5 (SLC5) with gcc 4.3.* compiler in 64-bit mode (CMT configurations x86_64-slc5-gcc43-opt, x86_64-slc5-gcc43-dbg)
 - ◆ Not available for projects based on Gaudi v23r8 or greater
 - ◆ See here for issues with porting to gcc 4.3 from gcc 3.4.
- CERN Scientific Linux 5 (SLC5) in 64 bit mode with icc 11.1 compiler (CMT configurations x86_64-slc5-icc11-opt, x86_64-slc5-icc11-dbg)
 - ◆ Not available for projects based on Gaudi v23r7 or greater
 - ◆ See here for usage instructions and recipes to fix common warnings.
- CERN Scientific Linux 5 (SLC5) with gcc 4.3.* compiler in 32-bit mode (CMT configurations i686-slc5-gcc43-opt, i686-slc5-gcc43-dbg)
 - ◆ Not available for projects based on Gaudi v23r6 or greater
 - ◆ This is the last available platform with gcc 32-bit support
- CERN Scientific Linux 4 (SLC4) with gcc 3.4.* compiler in 32-bit mode (CMT configurations slc4_ia32_gcc34, slc4_ia32_gcc34_dbg)
 - ◆ Using 32-bit compatibility libraries on systems booted with 64-bit SLC4
 - ◆ Not available for projects based on Gaudi v22r0 or greater

SupportedPlatforms < LHCb < TWiki

- CERN Scientific Linux 4 (SLC4) with gcc 3.4.* compiler in 64-bit mode (CMT configurations slc4_amd64_gcc34, slc4_amd64_gcc34_dbg)
 - ◆ Not available for projects based on Gaudi v22r0 or greater
- CERN Scientific Linux 3 (SLC3) with gcc 3.2.3 compiler (CMT configurations slc3_ia32_gcc323, slc3_ia32_gcc323_dbg)
 - ◆ Binary distribution available in optimised mode for projects older than December 2007
 - ◆ Please note that, for releases in October and November 2006 (LHCb v21r8 to v21r11) binaries were built with the flags `-msse2 -mfpmath=sse` so they will not work on older machines that do not support the sse2 instruction set (e.g. Pentium III, AMD Sempron)
 - ◆ Please note that access to SLC3 machines at CERN is no longer available
 - ◆ Not available for projects based on LHCb v23r0 or greater.
- Windows 32 bit with Visual Studio 9 compiler (CMT configuration i686-winxp-vc9-dbg) (see also Windows Development Environment)
 - ◆ Visual C++ 9.0
 - ◆ Was dropped in October 2011

-- MarcoCattaneo - 2020-02-04

This topic: LHCb > SupportedPlatforms

Topic revision: r76 - 2021-11-29 - MaikBecker



Copyright &© 2008-2022 by the contributing authors. All material on this collaboration platform is the property of the contributing authors.

or Ideas, requests, problems regarding TWiki? use Discourse or Send feedback