Table of Contents

Platforms and compilers supported by LHCb production software.............................................................1
  Supported compilers........................................................................................................................................1
  Production platforms (binary distribution available in optimised and debug modes)...............................1
  Development platforms (binary builds available in the nightlies).........................................................2
  Future platforms.....................................................................................................................................2
  Old production platforms.......................................................................................................................2
  Other ports...........................................................................................................................................3
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 - SLHCRELEASES). 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-sdb-query listPlatforms DaVinci v36r1p3`

Supported compilers

- On master branch (Run3 software)
  - gcc 8 (libstdc++ reference) with C++17
  - gcc 7 will be dropped after Gaudi v31r0 release
- On run2-patches branch (Run1+Run2 software)
  - gcc 6.2 with C++14 compilation enabled and new gcc ABI.
  - gcc 7 with C++14 compilation enabled and new gcc ABI.
- Older compilers in legacy analysis preservation branches, see nightlies summary page

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

- 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 since 19th February 2018.
  - Available only for Run1 and Run2 software
- CERN Scientific Linux 6 (SLC6) with gcc 7.* compiler in 64-bit mode (configurations x86_64-slc6-gcc7-opt, x86_64-slc6-gcc7-dbg (using gcc -Og option))
  - Built with -m sse4.2
- 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
- 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 only for Upgrade software (master branch)
- 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 only for Upgrade software (master branch)
Development platforms (binary builds available in the nightlies)

- clang 6.0 (configuration x86_64-centos7-clang60-opt). See nightly slot: lhcb-lcg-dev3. To set up the environment (requires LbScripts >= v9r2p6):
  - export CMTCONFIG=x86_64-centos7-clang60-opt
  - then use it from cvmfs, e.g.: lb-dev --nightly lhcb-lcg-dev3 Brunel/HEAD

Future platforms

- icc18
  - Would be useful in order to use the Intel profiling tools, but currently has problems with range v3, that Intel are investigating
  - C++ standard support:
- ARM
  - A port to ARM is ongoing
- AVX512
  - A build for AVX512 is foreseen. This will be introduced together with a new platform ID convention, see proposal here

Old production platforms

- 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)
  - 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 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

● 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

Other ports

● Brunel was successfully ported in 2013 to the ARM processor. See here for R&D for the ARM done at that time
● There is no plan for a port to Mac OSX 10.*. Some old instructions on building from source can be found here.