

CMake Plugin

Plugin Information

View CMake [on the plugin site](#) for more information.

This plugin can be used to build `cmake` based projects within Jenkins. It provides

- a build-step that generates the build scripts from a `CMakeLists.txt` file and allows to run these with the appropriate build tool,
- a build-step to invoke some tools of the CMake suite (CMake/CPack/CTest) with arbitrary arguments plus
- automatic installation of the CMake tool suite.

Screenshots (for the impatient)

Screenshots showing the configuration can be viewed here [CMake Build Configuration](#), here [Tool Configuration](#) and here [Global Configuration](#).

Global configuration

To ease the pain of provisioning Jenkins worker nodes, the plugin can download a recent version of `cmake` on demand from [cmake.org](#) and use that exclusively for a build. (Thanks to `cmake.org` for constantly providing portable/relocatable binaries for public download.) Jenkins admins may configure the available CMake versions on the [global configuration page](#).

Build-script generator build step

This build step generates the build scripts from a `CMakeLists.txt` file and is able to run the scripts. It accepts the following parameters:

1. CMake version to use — downloaded on demand, selectable versions get configured on the [global configuration page](#)
2. CMake Buildsript Generator
3. Source directory
4. Build directory
5. CMake Build Type — Debug/Release/...
6. Cache file — to pre-populate `cmake` cache variables
7. Clean/Incremental build — to clean the build directory prior to buildsript generation
8. Pass arbitrary command-line arguments to `cmake`.

To perform the actual build, this build step tries to detect the actual build tool corresponding to the chosen CMake Buildsript Generator and

- Can run the actual build tool as a sub-build step
 - with arbitrary (but tool dependend) arguments
 - with extra system environment variables.
- Supports to run the build tool by an extra `execute-shell` or `execute-batch` build step (in case auto detection fails).

CMake/CPack/CTest execution step

This build step allows to invoke the corresponding tool of the CMake suite with arbitrary command-line arguments. It accepts the following configuration parameters:

1. CMake version to use
2. Working directory
3. Command-line arguments.

Known Issues ([go to Tracker](#))

Key	Summary	T	Created	Updated	Due	Assignee	Reporter	P	Status	Resolution
JENKINS-55215	Log from <code>cmake/ctest</code> steps isn't fully captured on OsX		Dec 16, 2018	Jan 15, 2019		Unassigned	Marian Klymov		Open	Unresolved
JENKINS-47188	CmakeTool fails with no message when binaries are not available		Sep 28, 2017	Mar 27, 2018		Martin Weber	Emil Stykke		Open	Unresolved
JENKINS-31215	<code>cmake</code> installation on osx: missing files & strange layout		Oct 28, 2015	Mar 27, 2018		Martin Weber	mistafunk		Open	Unresolved

3 issues

Change Log

- Version 2.6.0 (21 Jun 2018)
- Version 2.5.2 (05 May 2018)
- Version 2.5.1 (31 Mar 2018)
- Version 2.5.0 (27 Mar 2018)
- Version 2.4.6 (11 Feb 2018)
- Version 2.4.5 (04 Apr 2017)
- Version 2.4.4 (20 Sep 2016)
- Version 2.4.3 (26 Jun 2016)
- Version 2.4.2 (09 May 2016)
- Version 2.4.1 (24 Feb 2016)
- Version 2.4.0 (6 Feb 2016) DO NOT INSTALL!
- Version 2.3.3 (20 Nov 2015)
- Version 2.3.2 (9 Oct 2015)
- Version 2.3.1 (4 Oct 2015)
- Version 2.3 (11 Sep 2015)
- Version 2.2 (28 Aug 2015)
- Version 2.1 (11 Jul 2015)
- Version 2.0 (5 Jul 2015)
- Version 1.10 (14 Jun 2015)
- Version 1.9 (17 Apr 2011)

Version 2.6.0 (21 Jun 2018)

- FIXED JENKINS-51904: cmake install from archive paths are wrong.
Compatibility note: To be compatible with the *install from archive*-installer, it is no longer possible to specify a non-default tool name (e.g. *mingw32-cmake* instead of *cmake*) as the *Path to cmake* on the global tool configuration page. Consider using a cmake toolchain file instead.

This change **affects only** cmake installations that do **not install automatically**. Any default *InSearchPath* installation with a tool name of just *cmake* is auto-migrated. Anyway, You will find messages in the Jenkins log if the cmake installation cannot be auto-migrated.

- FIXED JENKINS-52104: Plugin always fills arguments field with "all" for build tool.

Version 2.5.2 (05 May 2018)

- FIXED JENKINS-51060: Environment variables are not passed to cmake steps when running in pipeline with Env.

Version 2.5.1 (31 Mar 2018)

- FIXED The Pipeline snippet generator duplicates argument 'cmakeArgs' as 'arguments and 'buildDir as 'workingDir'.

Version 2.5.0 (27 Mar 2018)

- FIXED JENKINS-34998: Make CMake plugin compatible with pipeline.

DO NOT USE WITH PIPELINE: The Pipeline snippet generator duplicates argument 'cmakeArgs' as 'arguments and 'buildDir as 'workingDir'.

Version 2.4.6 (11 Feb 2018)

- FIXED JENKINS-48102: CMake plugin leaks file handle on CMakeCache.txt.

Version 2.4.5 (04 Apr 2017)

- FIXED JENKINS-43175: NPE at hudson.plugins.cmake.CmakeBuilder.perform()

Version 2.4.4 (20 Sep 2016)

- FIXED JENKINS-38227: Allow letting CMake choose the (default) generator.

Version 2.4.3 (26 Jun 2016)

- FIXED JENKINS-35911: Allow to ignore failure exit codes from CMake/CPack/CTest build step.

Version 2.4.2 (09 May 2016)

- Upgraded to new parent pom.
- Require Java 7 to run the plugin now.

- Integrated Findbugs and fixed potential errors discovered by it.
- CLOSED JENKINS--29142 (cannot be fixed, but mentioning workaround in online help)
- FIXED JENKINS-34613 : Set default value of Generator in Java instead of jelly.

Version 2.4.1 (24 Feb 2016)

- FIXED JENKINS-32657, reopened: Loss of cmake-argument and working dir settings in existing CMake build steps.

Version 2.4.0 (6 Feb 2016) DO NOT INSTALL!

DO NOT INSTALL: Existing build steps will loose their cmake-argument and working dir settings.
For those who have it installed: Do not run *Manage Old Data* from the Jenkins Management page.

- Added build step which allows to invoke some tools of the CMake suite (CMake/CPack/CTest) with arbitrary arguments.
 - FIXED JENKINS-32657, FIXED JENKINS-30695

Version 2.3.3 (20 Nov 2015)

- Build tool step no longer discards the environment set up with EnvInject. This allows to apply a workaround for JENKINS-30114. (Thanks to Armin Novak for reporting and testing on Windows.)

Version 2.3.2 (9 Oct 2015)

- FIXED: JENKINS-30070: Fix path to cmake binary for Mac OS X (Thanks to Guillaume Egles for testing under OS X).

Version 2.3.1 (4 Oct 2015)

- FIXED: JENKINS-30070: Fix 64 bit arch for Mac OS X.

Version 2.3 (11 Sep 2015)

- FIXED: JENKINS-30070: Automatic download and installation of cmake from cmake.org. Can download and install cmake 2.6 and above for Linux, Windows, OS X, SunOS, AIX, HPUX, Irix and FreeBSD (Tested on linux and windows).
- Improved help texts.
- Minor tweaks in error reporting.

Version 2.2 (28 Aug 2015)

- FIXED: JENKINS-29329: Visual Studio/MSBuild projects can not be built implicitly anymore JENKINS-29267 (Thanks to Dominik Aciri for testing under windows).
- Improved online help.
- Changed build message wording from ERROR to WARNING if CMAKE_BUILD_TOOL cannot be determined to avoid end-user confusion.

Version 2.1 (11 Jul 2015)

- FIXED: JENKINS-29267 CMAKE_BUILD_TOOL is deprecated in cmake version 3.0

Version 2.0 (5 Jul 2015)

This version is almost a complete re-write. Unfortunately, due to its new set of features, it will be incompatible with older versions.
Changes include

- Require Jenkins 1.580.3
- Use the standard ToolInstallation mechanism to select the cmake version per-build-step (similar to Ant and Maven).
- Eliminated mis-use of CMAKE_INSTALL_PREFIX.
- Gets the actual build tool name and path from CMakeCache.txt and exposes it as env var `CMAKE_BUILD_TOOL` for subsequent build steps. Users no longer have to care for the name of the build tool.
- Allow to run the build tool multiple times with different targets.
- Allow to pass extra environment variables to the build tool (e.g. DESTDIR=some/dir).

Version 1.10 (14 Jun 2015)

- New maintainer.
- FIXED: Expand environment variables in definition of CMake generator (JENKINS-13049)
- FIXED: Matrix build labels not expanded by CMake plugin (JENKINS-8538)
- ENHANCEMENT: Removed restriction to be applicable to only free-style or matrix project.

Version 1.9 (17 Apr 2011)