.. option:: -S <path-to-source>

Path to root directory of the CMake project to build.

.. option:: -B <path-to-build>

Path to directory which CMake will use as the root of build directory.

If the directory doesn't already exist CMake will make it.

.. option:: -C <initial-cache>

Pre-load a script to populate the cache.

When CMake is first run in an empty build tree, it creates a

``CMakeCache.txt`` file and populates it with customizable settings for

the project. This option may be used to specify a file from which

to load cache entries before the first pass through the project's

CMake listfiles. The loaded entries take priority over the

project's default values. The given file should be a CMake script

containing :command:`set` commands that use the ``CACHE`` option, not a

cache-format file.

References to :variable:`CMAKE\_SOURCE\_DIR` and :variable:`CMAKE\_BINARY\_DIR`

within the script evaluate to the top-level source and build tree.

.. option:: -D <var>:<type>=<value>, -D <var>=<value>

Create or update a CMake ``CACHE`` entry.

When CMake is first run in an empty build tree, it creates a

``CMakeCache.txt`` file and populates it with customizable settings for

the project. This option may be used to specify a setting that

takes priority over the project's default value. The option may be

repeated for as many ``CACHE`` entries as desired.

If the ``:<type>`` portion is given it must be one of the types

specified by the :command:`set` command documentation for its

``CACHE`` signature.

If the ``:<type>`` portion is omitted the entry will be created

with no type if it does not exist with a type already. If a

command in the project sets the type to ``PATH`` or ``FILEPATH``

then the ``<value>`` will be converted to an absolute path.

This option may also be given as a single argument:

``-D<var>:<type>=<value>`` or ``-D<var>=<value>``.

It's important to note that the order of ``-C`` and ``-D`` arguments is

significant. They will be carried out in the order they are listed, with the

last argument taking precedence over the previous ones. For example, if you

specify ``-DCMAKE\_BUILD\_TYPE=Debug``, followed by a ``-C`` argument with a

file that calls:

.. code-block:: cmake

set(CMAKE\_BUILD\_TYPE "Release" CACHE STRING "" FORCE)

then the ``-C`` argument will take precedence, and ``CMAKE\_BUILD\_TYPE`` will

be set to ``Release``. However, if the ``-D`` argument comes after the ``-C``

argument, it will be set to ``Debug``.

If a ``set(... CACHE ...)`` call in the ``-C`` file does not use ``FORCE``,

and a ``-D`` argument sets the same variable, the ``-D`` argument will take

precedence regardless of order because of the nature of non-``FORCE``

``set(... CACHE ...)`` calls.

.. option:: -U <globbing\_expr>

Remove matching entries from CMake ``CACHE``.

This option may be used to remove one or more variables from the

``CMakeCache.txt`` file, globbing expressions using ``\*`` and ``?`` are

supported. The option may be repeated for as many ``CACHE`` entries as

desired.

Use with care, you can make your ``CMakeCache.txt`` non-working.

.. option:: -G <generator-name>

Specify a build system generator.

CMake may support multiple native build systems on certain

platforms. A generator is responsible for generating a particular

build system. Possible generator names are specified in the

:manual:`cmake-generators(7)` manual.

If not specified, CMake checks the :envvar:`CMAKE\_GENERATOR` environment

variable and otherwise falls back to a builtin default selection.

.. option:: -T <toolset-spec>

Toolset specification for the generator, if supported.

Some CMake generators support a toolset specification to tell

the native build system how to choose a compiler. See the

:variable:`CMAKE\_GENERATOR\_TOOLSET` variable for details.

.. option:: -A <platform-name>

Specify platform name if supported by generator.

Some CMake generators support a platform name to be given to the

native build system to choose a compiler or SDK. See the

:variable:`CMAKE\_GENERATOR\_PLATFORM` variable for details.

.. option:: --toolchain <path-to-file>

.. versionadded:: 3.21

Specify the cross compiling toolchain file, equivalent to setting

:variable:`CMAKE\_TOOLCHAIN\_FILE` variable. Relative paths are interpreted as

relative to the build directory, and if not found, relative to the source

directory.

.. option:: --install-prefix <directory>

.. versionadded:: 3.21

Specify the installation directory, used by the

:variable:`CMAKE\_INSTALL\_PREFIX` variable. Must be an absolute path.

.. option:: -Wno-dev

Suppress developer warnings.

Suppress warnings that are meant for the author of the

``CMakeLists.txt`` files. By default this will also turn off

deprecation warnings.

.. option:: -Wdev

Enable developer warnings.

Enable warnings that are meant for the author of the ``CMakeLists.txt``

files. By default this will also turn on deprecation warnings.

.. option:: -Wdeprecated

Enable deprecated functionality warnings.

Enable warnings for usage of deprecated functionality, that are meant

for the author of the ``CMakeLists.txt`` files.

.. option:: -Wno-deprecated

Suppress deprecated functionality warnings.

Suppress warnings for usage of deprecated functionality, that are meant

for the author of the ``CMakeLists.txt`` files.

.. option:: -Werror=<what>

Treat CMake warnings as errors. ``<what>`` must be one of the following:

``dev``

Make developer warnings errors.

Make warnings that are meant for the author of the ``CMakeLists.txt`` files

errors. By default this will also turn on deprecated warnings as errors.

``deprecated``

Make deprecated macro and function warnings errors.

Make warnings for usage of deprecated macros and functions, that are meant

for the author of the ``CMakeLists.txt`` files, errors.

.. option:: -Wno-error=<what>

Do not treat CMake warnings as errors. ``<what>`` must be one of the following:

``dev``

Make warnings that are meant for the author of the ``CMakeLists.txt`` files not

errors. By default this will also turn off deprecated warnings as errors.

``deprecated``

Make warnings for usage of deprecated macros and functions, that are meant

for the author of the ``CMakeLists.txt`` files, not errors.