Commands - Directory

```
cmake_minimum_required(
  VERSION <min>[...<max>] [FATAL_ERROR])

cmake_host_system_information(RESULT <var>
  QUERY <NUMBER_OF_LOGICAL_CORES|
    NUMBER_OF_PHYSICAL_CORES|HOSTNAME|FQDN|
    TOTAL_VIRTUAL_MEMORY|AVAILABLE_VIRTUAL_MEMORY|
    TOTAL_PHYSICAL_MEMORY|AVAILABLE_PHYSICAL_MEMORY|
    IS_64BIT|HAS_FPU|HAS_MMX|HAS_MMX_PLUS|HAS_SSE|
    HAS_SSE2|HAS_SSE_FP|HAS_SSE_MMX|HAS_AMD_3DNOW|
    HAS_AMD_3DNOW_PLUS|HAS_IA64|HAS_SERIAL_NUMBER|
    PROCESSOR_SERIAL_NUMBER|PROCESSOR_NAME|
    PROCESSOR_DESCRIPTION|OS_NAME|OS_RELEASE|
    OS_VERSION|OS_PLATFORM> ...)

site_name(var)

message([STATUS|WARNING|AUTHOR_WARNING|SEND_ERROR|
  FATAL_ERROR|DEPRECATION] "message to display" ...)

cmake_policy(VERSION <min>[...<max>])
cmake_policy(SET CMP<NNNN> NEW)
cmake_policy(SET CMP<NNNN> OLD)
cmake_policy(GET CMP<NNNN> <var>)
cmake_policy(PUSH)
cmake_policy(POP)

option(<var> "option description" [initial value])

include(<file|module> [OPTIONAL]
  [RESULT_VARIABLE <var>] [NO_POLICY_SCOPE])

include_external_msproject(projectname location
  [TYPE projectTypeGUID]
  [GUID projectGUID]
  [PLATFORM platformName]
  dep1 dep2 ...)

include_guard([DIRECTORY|GLOBAL])

include_regular_expression(regex_match
  [regex_complain])

add_compile_definitions(<definition> ...)

add_compile_options(<option> ...)

add_definitions(<definition> ...)

remove_definitions(<definition> ...)

add_link_options(<option> ...)

include_directories([AFTER|BEFORE] [SYSTEM]
  dir1 [dir2 ...])

link_directories([AFTER|BEFORE] directory1
  [directory2 ...])

link_libraries([item1 [item2 [...]]]
  [[debug|optimized|general] <item>] ...)

configure_file(<input> <output>
  [COPYONLY] [ESCAPE_QUOTES] [@ONLY]
  [NEWLINE_STYLE [UNIX|DOS|WIN32|LF|CRLF]])

execute_process(COMMAND <cmd1> [args1...]]
  [COMMAND <cmd2> [args2...] [...]]
  [WORKING_DIRECTORY <directory>]
  [TIMEOUT <seconds>]
  [RESULT_VARIABLE <var>]
  [RESULTS_VARIABLE <var>]
  [OUTPUT_VARIABLE <var>]
  [ERROR_VARIABLE <var>]
  [INPUT_FILE <file>]
  [OUTPUT_FILE <file>]
  [ERROR_FILE <file>]
  [OUTPUT_QUIET] [ERROR_QUIET]
  [OUTPUT_STRIP_TRAILING_WHITESPACE]
  [ERROR_STRIP_TRAILING_WHITESPACE]
  [ENCODING <NONE|AUTO|ANSI|OEM|UTF8|UTF-8>])
```

Commands - Project

```
project(<name>
  [VERSION <major>[.<minor>[.<patch>[.<tweak>]]]]
  [DESCRIPTION <description-string>]
  [HOMEPAGE_URL <url-string>]
  [LANGUAGES <language-name>...])

enable_language(<lang> [OPTIONAL])

add_dependencies(<target> [<dependency>]...)

add_library(<name> [STATIC|SHARED|MODULE]
  [EXCLUDE_FROM_ALL] [source1] [source2 ...])
add_library(<name> <SHARED|STATIC|MODULE|
  OBJECT|UNKNOWN> IMPORTED [GLOBAL])
add_library(<name> OBJECT <src>...)
add_library(<name> ALIAS <target>)
add_library(<name> INTERFACE [IMPORTED [GLOBAL]])

add_executable(<name> [WIN32] [MACOSX_BUNDLE]
  [EXCLUDE_FROM_ALL] [source1] [source2 ...])
add_executable(<name> IMPORTED [GLOBAL])
add_executable(<name> ALIAS <target>)

add_custom_target(<name> [ALL] [command1 [args1...]]
  [COMMAND command2 [args2...] ...]
  [DEPENDS depend depend depend ... ]
  [BYPRODUCTS [files...]]
  [WORKING_DIRECTORY dir]
  [COMMENT comment]
  [VERBATIM] [USES_TERMINAL] [COMMAND_EXPAND_LISTS]
  [SOURCES src1 [src2...]])

add_custom_command(OUTPUT output1 [output2 ...]
  COMMAND command1 [ARGS] [args1...]
  [COMMAND command2 [ARGS] [args2...] ...]
  [MAIN_DEPENDENCY depend]
  [DEPENDS [depends...]]
  [BYPRODUCTS [files...]]
  [IMPLICIT_DEPENDS <lang1> dep1 [<lang2> dep2] ...]
  [WORKING_DIRECTORY dir]
  [COMMENT comment]
  [DEPFILE depfile]
  [VERBATIM] [APPEND] [USES_TERMINAL]
  [COMMAND_EXPAND_LISTS])
add_custom_command(TARGET <target>
  PRE_BUILD|PRE_LINK|POST_BUILD
  COMMAND command1 [ARGS] [args1...]
  [COMMAND command2 [ARGS] [args2...] ...]
  [BYPRODUCTS [files...]]
  [WORKING_DIRECTORY dir]
  [COMMENT comment]
  [VERBATIM] [USES_TERMINAL])

add_subdirectory(source_dir [binary_dir]
  [EXCLUDE_FROM_ALL])

find_file(<VAR>
  name | NAMES name1 [name2 ...]
  [HINTS path1 [path2 ... ENV var]]
  [PATHS path1 [path2 ... ENV var]]
  [PATH_SUFFIXES suffix1 [suffix2 ...]]
  [DOC "cache documentation string"]
  [NO_DEFAULT_PATH] [NO_PACKAGE_ROOT_PATH]
  [NO_CMAKE_PATH] [NO_CMAKE_ENVIRONMENT_PATH]
  [NO_SYSTEM_ENVIRONMENT_PATH] [NO_CMAKE_SYSTEM_PATH]
  [CMAKE_FIND_ROOT_PATH_BOTH|
   ONLY_CMAKE_FIND_ROOT_PATH|
   NO_CMAKE_FIND_ROOT_PATH])

find_path(<VAR> name | NAMES name1 [name2 ...]
  [HINTS path1 [path2 ... ENV var]]
  [PATHS path1 [path2 ... ENV var]]
  [PATH_SUFFIXES suffix1 [suffix2 ...]]
  [DOC "cache documentation string"]
  [NO_DEFAULT_PATH] [NO_PACKAGE_ROOT_PATH]
  [NO_CMAKE_PATH] [NO_CMAKE_ENVIRONMENT_PATH]
  [NO_SYSTEM_ENVIRONMENT_PATH] [NO_CMAKE_SYSTEM_PATH]
  [CMAKE_FIND_ROOT_PATH_BOTH|
   ONLY_CMAKE_FIND_ROOT_PATH|
   NO_CMAKE_FIND_ROOT_PATH])
```

```
find_package(<PackageName> [version] [EXACT]
  [QUIET] [MODULE] [REQUIRED]
  [[COMPONENTS] [components...]]
  [OPTIONAL_COMPONENTS components...]
  [NO_POLICY_SCOPE])
find_package(<PackageName> [version] [EXACT] [QUIET]
  [REQUIRED] [[COMPONENTS] [components...]]
  [CONFIG|NO_MODULE] [NO_POLICY_SCOPE]
  [NAMES name1 [name2 ...]] [CONFIGS cfg1 [cfg2 ...]]
  [HINTS path1 [path2 ...]] [PATHS path1 [path2 ...]]
  [PATH_SUFFIXES suffix1 [suffix2 ...]]
  [NO_DEFAULT_PATH] [NO_PACKAGE_ROOT_PATH]
  [NO_CMAKE_PATH] [NO_CMAKE_ENVIRONMENT_PATH]
  [NO_SYSTEM_ENVIRONMENT_PATH]
  [NO_CMAKE_PACKAGE_REGISTRY] [NO_CMAKE_SYSTEM_PATH]
  [NO_CMAKE_SYSTEM_PACKAGE_REGISTRY]
  [CMAKE_FIND_ROOT_PATH_BOTH|
   ONLY_CMAKE_FIND_ROOT_PATH|
   NO_CMAKE_FIND_ROOT_PATH])

find_program(<var>
  name | NAMES name1 [name2 ...] [NAMES_PER_DIR]
  [HINTS path1 [path2 ... ENV var]]
  [PATHS path1 [path2 ... ENV var]]
  [PATH_SUFFIXES suffix1 [suffix2 ...]]
  [DOC "cache documentation string"]
  [NO_DEFAULT_PATH] [NO_PACKAGE_ROOT_PATH]
  [NO_CMAKE_PATH] [NO_CMAKE_ENVIRONMENT_PATH]
  [NO_SYSTEM_ENVIRONMENT_PATH] [NO_CMAKE_SYSTEM_PATH]
  [CMAKE_FIND_ROOT_PATH_BOTH|
   ONLY_CMAKE_FIND_ROOT_PATH|
   NO_CMAKE_FIND_ROOT_PATH])

find_library(<VAR>
  name | NAMES name1 [name2 ...] [NAMES_PER_DIR]
  [HINTS path1 [path2 ... ENV var]]
  [PATHS path1 [path2 ... ENV var]]
  [PATH_SUFFIXES suffix1 [suffix2 ...]]
  [DOC "cache documentation string"]
  [NO_DEFAULT_PATH] [NO_PACKAGE_ROOT_PATH]
  [NO_CMAKE_PATH] [NO_CMAKE_ENVIRONMENT_PATH]
  [NO_SYSTEM_ENVIRONMENT_PATH] [NO_CMAKE_SYSTEM_PATH]
  [CMAKE_FIND_ROOT_PATH_BOTH|
   ONLY_CMAKE_FIND_ROOT_PATH|
   NO_CMAKE_FIND_ROOT_PATH])

export(EXPORT <export-name> [NAMESPACE <namespace>]
  [FILE <filename>])
export(TARGETS [target1 [target2 [...]]]
  [NAMESPACE <namespace>] [APPEND] FILE <filename>
  [EXPORT_LINK_INTERFACE_LIBRARIES])
export(PACKAGE <PackageName>)
export(TARGETS [target1 [target2 [...]]]
  [ANDROID_MK <filename>])

load_cache(pathToCacheFile READ_WITH_PREFIX
  prefix entry1...)
load_cache(pathToCacheFile [EXCLUDE entry1...]
  [INCLUDE_INTERNALS entry1...])

try_compile(RESULT_VAR <bindir> <srcdir>
  <projectName> [<targetName>]
  [CMAKE_FLAGS <flags>...] [OUTPUT_VARIABLE <var>])
try_compile(RESULT_VAR <bindir>
  <srcfile|SOURCES srcfile...>
  [CMAKE_FLAGS <flags>...]
  [COMPILE_DEFINITIONS <defs>...]
  [LINK_LIBRARIES <libs>...] [OUTPUT_VARIABLE <var>]
  [COPY_FILE <fileName> [COPY_FILE_ERROR <var>]]
  [<LANG>_STANDARD <std>]
  [<LANG>_STANDARD_REQUIRED <bool>]
  [<LANG>_EXTENSIONS <bool>])

try_run(RUN_RESULT_VAR COMPILE_RESULT_VAR
  bindir srcfile [CMAKE_FLAGS <flags>...]
  [COMPILE_DEFINITIONS <defs>...]
  [LINK_LIBRARIES <libs>...]
  [COMPILE_OUTPUT_VARIABLE <var>]
  [RUN_OUTPUT_VARIABLE <var>] [OUTPUT_VARIABLE <var>]
  [ARGS <args>...])
```

```
install(TARGETS targets... [EXPORT <export-name>]
  [[ARCHIVE|LIBRARY|RUNTIME|OBJECTS|FRAMEWORK|BUNDLE|
     PRIVATE_HEADER|PUBLIC_HEADER|RESOURCE]
   [DESTINATION <dir>]
   [PERMISSIONS permissions...]
   [CONFIGURATIONS [Debug|Release|...]]
   [COMPONENT <component>]
   [NAMELINK_COMPONENT <component>]
   [OPTIONAL] [EXCLUDE_FROM_ALL]
   [NAMELINK_ONLY|NAMELINK_SKIP] ] [...]
  [INCLUDES DESTINATION [<dir> ...]])
install(<FILES|PROGRAMS> files... DESTINATION <dir>
  [PERMISSIONS permissions...]
  [CONFIGURATIONS [Debug|Release|...]]
  [COMPONENT <component>]
  [RENAME <name>] [OPTIONAL] [EXCLUDE_FROM_ALL])
install(DIRECTORY dirs... DESTINATION <dir>
  [FILE_PERMISSIONS permissions...]
  [DIRECTORY_PERMISSIONS permissions...]
  [USE_SOURCE_PERMISSIONS] [OPTIONAL] [MESSAGE_NEVER]
  [CONFIGURATIONS [Debug|Release|...]]
  [COMPONENT <component>] [EXCLUDE_FROM_ALL]
  [FILES_MATCHING]
  [[PATTERN <pattern> | REGEX <regex>]
   [EXCLUDE] [PERMISSIONS permissions...]] [...])
install([[SCRIPT <file>] [CODE <code>]]
  [COMPONENT <component>] [EXCLUDE_FROM_ALL] [...])
install(EXPORT <export-name> DESTINATION <dir>
  [NAMESPACE <namespace>] [[FILE <name>.cmake]|
  [PERMISSIONS permissions...]
  [CONFIGURATIONS [Debug|Release|...]]
  [EXPORT_LINK_INTERFACE_LIBRARIES]
  [COMPONENT <component>]
  [EXCLUDE_FROM_ALL])
install(EXPORT_ANDROID_MK <export-name>
  DESTINATION <dir> [...])
```

Commands - Targets

```
target_sources(<target>
  <INTERFACE|PUBLIC|PRIVATE> [items1...]
  [<INTERFACE|PUBLIC|PRIVATE> [items2...] ...])

target_include_directories(<target> [SYSTEM] [BEFORE]
  <INTERFACE|PUBLIC|PRIVATE> [items1...]
  [<INTERFACE|PUBLIC|PRIVATE> [items2...] ...])

target_compile_definitions(<target>
  <INTERFACE|PUBLIC|PRIVATE> [items1...]
  [<INTERFACE|PUBLIC|PRIVATE> [items2...] ...])

target_compile_options(<target> [BEFORE]
  <INTERFACE|PUBLIC|PRIVATE> [items1...]
  [<INTERFACE|PUBLIC|PRIVATE> [items2...] ...])

target_link_options(<target> [BEFORE]
  <INTERFACE|PUBLIC|PRIVATE> [items1...]
  [<INTERFACE|PUBLIC|PRIVATE> [items2...] ...])

target_link_libraries(<target>
   <PRIVATE|PUBLIC|INTERFACE> <item>...
  [<PRIVATE|PUBLIC|INTERFACE> <item>...]...)
target_link_libraries(<target> <item>...)

target_link_directories(<target> [BEFORE]
  <INTERFACE|PUBLIC|PRIVATE> [items1...]
  [<INTERFACE|PUBLIC|PRIVATE> [items2...] ...])

target_compile_features(<target>
  <PRIVATE|PUBLIC|INTERFACE> <feature> [...])

target_precompile_headers(<target>
  <INTERFACE|PUBLIC|PRIVATE> [header1...]
  [<INTERFACE|PUBLIC|PRIVATE> [header2...] ...])

source_group(<name> [FILES <src>...]
  [REGULAR_EXPRESSION <regex>])
source_group(TREE <root> [PREFIX <prefix>]
  [FILES <src>...])

aux_source_directory(<dir> <var>)
```

Commands - Properties

```
set_property(
  <GLOBAL |
   DIRECTORY [dir] |
   TARGET [target1 [target2 ...]] |
   SOURCE [src1 [src2 ...]] |
   INSTALL [file1 [file2 ...]] |
   TEST [test1 [test2 ...]] |
   CACHE [entry1 [entry2 ...]]>
  [APPEND] [APPEND_STRING]
  PROPERTY <name> [value1 [value2 ...]])

get_property(<var>
  <GLOBAL |
   DIRECTORY [dir] |
   TARGET <target> |
   SOURCE <source> |
   INSTALL <file> |
   TEST <test> |
   CACHE <entry> |
   VARIABLE>
   PROPERTY <name>
   [SET|DEFINED|BRIEF_DOCS|FULL_DOCS])

define_property(<GLOBAL|DIRECTORY|TARGET|SOURCE|
  TEST|VARIABLE|CACHED_VARIABLE>
  PROPERTY <name> [INHERITED]
  BRIEF_DOCS <brief-doc> [docs...]
  FULL_DOCS <full-doc> [docs...])

set_target_properties(target1 target2 ...
  PROPERTIES prop1 value1 prop2 value2 ...)

get_target_property(<var> <target> <property>)

set_source_files_properties([file1 [file2 [...]]]
  PROPERTIES prop1 value1 [prop2 value2 [...]])

get_source_file_property(var file property)

set_tests_properties(test1 [test2...]
  PROPERTIES prop1 value1 prop2 value2)

get_test_property(test property var)

set_directory_properties(PROPERTIES prop1 value1
  prop2 value2)

get_directory_property(<var> [DIRECTORY <dir>]
  <prop-name>)

get_cmake_property(<var> property)
```

Commands - Special

```
fltk_wrap_ui(resultingLibraryName
  source1 source2 ... sourceN)

qt_wrap_cpp(resultingLibraryName DestName
  SourceLists ...)

qt_wrap_ui(resultingLibraryName HeadersDestName
  SourcesDestName SourceLists ...)
```

Commands - Scripting

```
set(<var> <value>... [PARENT_SCOPE])
set(<var> <value>... CACHE
  <BOOL|FILEPATH|PATH|STRING|INTERNAL> <docstring>
  [FORCE])
set(ENV{<var>} <value>...)

cmake_language(CALL <cmd> [<arg>...])
cmake_language(EVAL CODE <code>...)
cmake_language(DEFER <options>... CALL <cmd> [<arg
      ↪ >...])

mark_as_advanced([CLEAR|FORCE] var [var2 ...])

unset(<var> [CACHE | PARENT_SCOPE])
unset(ENV{environment})

variable_watch(<var name> [<command to execute>])

math(EXPR <output-var> <expression>
  [OUTPUT_FORMAT <HEXADECIMAL|DECIMAL>])

separate_arguments(<var>
  <NATIVE|UNIX|WINDOWS>_COMMAND "<args>")

list(LENGTH <list> <out-var>)
list(GET <list> <index> [<index> ...] <out-var>)
list(JOIN <list> <glue> <out-var>)
list(SUBLIST <list> <begin> <length> <out-var>)
list(FIND <list> <value> <out-var>)
list(APPEND <list> [<element>...])
list(FILTER <list> {INCLUDE|EXCLUDE} REGEX <regex>)
list(INSERT <list> <index> [<element>...])
list(REMOVE_ITEM <list> <value>...)
list(REMOVE_AT <list> <index>...)
list(REMOVE_DUPLICATES <list>)
list(TRANSFORM <list> <ACTION> [...])
list(REVERSE <list>)
list(SORT <list> [...])

string(FIND <str> <substr> <out-var> [...])
string(REPLACE <match-str> <replace-str>
  <out-var> <input>...)
string(REGEX MATCH <match-regex>
  <out-var> <input>...)
string(REGEX MATCHALL <match-regex>
  <out-var> <input>...)
string(REGEX REPLACE <match-regex>
  <replace-expr> <out-var> <input>...)
string(APPEND <str-var> [<input>...])
string(PREPEND <str-var> [<input>...])
string(CONCAT <out-var> [<input>...])
string(JOIN <glue> <out-var> [<input>...])
string(TOLOWER <str> <out-var>)
string(TOUPPER <str> <out-var>)
string(LENGTH <str> <out-var>)
string(SUBSTRING <str> <begin> <length> <out-var>)
string(STRIP <str> <out-var>)
string(GENEX_STRIP <str> <out-var>)
string(COMPARE <op> <str1> <str2> <out-var>)
string(<HASH> <out-var> <input>)
string(ASCII <number>... <out-var>)
string(CONFIGURE <str> <out-var> [...])
string(MAKE_C_IDENTIFIER <str> <out-var>)
string(RANDOM [<option>...] <out-var>)
string(TIMESTAMP <out-var> [<format-str>] [UTC])
string(UUID <out-var> ...)

file(READ <filename> <out-var> [...])
file(STRINGS <filename> <out-var> [...])
file(<HASH> <filename> <out-var>)
file(TIMESTAMP <filename> <out-var> [...])
file({WRITE|APPEND} <filename> <content>...)
file({TOUCH|TOUCH_NOCREATE} [<file>...])
file(GENERATE OUTPUT <output-file> [...])
file({GLOB|GLOB_RECURSE} <out-var> [...]
  [<globbing-expr>...])
file(RENAME <oldname> <newname>)
file({REMOVE|REMOVE_RECURSE } [<files>...])
file(MAKE_DIRECTORY [<dir>...])
file({COPY|INSTALL} <file>...
  DESTINATION <dir> [...])
file(RELATIVE_PATH <out-var> <directory> <file>)
file({TO_CMAKE_PATH|TO_NATIVE_PATH} <path> <out-var>)
file(DOWNLOAD <url> <file> [...])
file(UPLOAD <file> <url> [...])
file(LOCK <path> [...])
```

```
cmake_path(GET <path-var> ROOT_NAME <out-var>)
cmake_path(GET <path-var> ROOT_DIRECTORY <out-var>)
cmake_path(GET <path-var> ROOT_PATH <out-var>)
cmake_path(GET <path-var> FILENAME <out-var>)
cmake_path(GET <path-var> EXTENSION [LAST_ONLY] <out-
      ↪ var>)
cmake_path(GET <path-var> STEM [LAST_ONLY] <out-var>)
cmake_path(GET <path-var> RELATIVE_PART <out-var>)
cmake_path(GET <path-var> PARENT_PATH <out-var>)
cmake_path(HAS_ROOT_NAME <path-var> <out-var>)
cmake_path(HAS_ROOT_DIRECTORY <path-var> <out-var>)
cmake_path(HAS_ROOT_PATH <path-var> <out-var>)
cmake_path(HAS_FILENAME <path-var> <out-var>)
cmake_path(HAS_EXTENSION <path-var> <out-var>)
cmake_path(HAS_STEM <path-var> <out-var>)
cmake_path(HAS_RELATIVE_PART <path-var> <out-var>)
cmake_path(HAS_PARENT_PATH <path-var> <out-var>)
cmake_path(IS_ABSOLUTE <path-var> <out-var>)
cmake_path(IS_RELATIVE <path-var> <out-var>)
cmake_path(IS_PREFIX <path-var> <input> [NORMALIZE] <
      ↪ out-var>)
cmake_path(COMPARE <input1> <OP> <input2> <out-var>)
cmake_path(SET <path-var> [NORMALIZE] <input>)
cmake_path(APPEND <path-var> [<input>...] [
      ↪ OUTPUT_VARIABLE <out-var>])
cmake_path(APPEND_STRING <path-var> [<input>...] [
      ↪ OUTPUT_VARIABLE <out-var>])
cmake_path(REMOVE_FILENAME <path-var> [
      ↪ OUTPUT_VARIABLE <out-var>])
cmake_path(REPLACE_FILENAME <path-var> <input> [
      ↪ OUTPUT_VARIABLE <out-var>])
cmake_path(REMOVE_EXTENSION <path-var> [LAST_ONLY] [
      ↪ OUTPUT_VARIABLE <out-var>])
cmake_path(REPLACE_EXTENSION <path-var> [LAST_ONLY] <
      ↪ input> [OUTPUT_VARIABLE <out-var>])
cmake_path(NORMAL_PATH <path-var> [OUTPUT_VARIABLE <
      ↪ out-var>])
cmake_path(RELATIVE_PATH <path-var> [BASE_DIRECTORY <
      ↪ input>] [OUTPUT_VARIABLE <out-var>])
cmake_path(ABSOLUTE_PATH <path-var> [BASE_DIRECTORY <
      ↪ input>] [NORMALIZE] [OUTPUT_VARIABLE <out-
      ↪ var>])
cmake_path(NATIVE_PATH <path-var> [NORMALIZE] <out-
      ↪ var>)
cmake_path(CONVERT <input> TO_CMAKE_PATH_LIST <out-
      ↪ var>)
cmake_path(CONVERT <input> TO_NATIVE_PATH_LIST <out-
      ↪ var>)
cmake_path(HASH <path-var> <out-var>)

get_filename_component(<var> <FileName>
  <DIRECTORY|NAME|EXT|NAME_WE|PATH> [CACHE])
get_filename_component<ABSOLUTE|REALPATH>
  [BASE_DIR <base_dir>] [CACHE])
get_filename_component(<var> <FileName>
  PROGRAM [PROGRAM_ARGS <arg_var>] [CACHE])
```

Commands - Flow Control

```
function(<name> [arg1 [arg2 [arg3 ...]]])
  command(ARGS ...)...
endfunction()

macro(<name> [arg1 [arg2 [arg3 ...]]])
  command(ARGS ...)...
endmacro()

return ()

cmake_parse_arguments(<prefix> <options>
  <one_value_keywords> <multi_value_keywords>
  args...)
cmake_parse_arguments(PARSE_ARGV N <prefix>
  <options> <one_value_keywords>
  <multi_value_keywords>)

foreach(loop_var arg1 arg2 ...)
  command(ARGS ...)...
endforeach()

foreach(loop_var RANGE total)
foreach(loop_var RANGE start stop [step])
foreach(loop_var IN
  [LISTS [list1 [...]]] [ITEMS [item1 [...]]])
while(condition)
  command(ARGS ...)...
endwhile()
```

```
break()

continue()

if(expression)
  command(ARGS ...)...
elseif(expression2)
  command(ARGS ...)...
else(expression3)
  command(ARGS ...)...
endif()

if((expression) AND (expression OR (expression)))
if(<constant>)
if(<var|str>)
if(NOT <expression>)
if(<expr1> AND <expr2>)
if(<expr1> OR <expr2>)
if(COMMAND command-name)
if(POLICY policy-id)
if(TARGET target-name)
if(TEST test-name)
if(EXISTS path-to-file-or-directory)
if(DEFINED <var>)
if(file1 IS_NEWER_THAN file2)
if(IS_DIRECTORY path-to-directory)
if(IS_SYMLINK filename)
if(IS_ABSOLUTE path)
if(<var|str> MATCHES regex)
if(<var|str> LESS <var|str>)
if(<var|str> GREATER <var|str>)
if(<var|str> EQUAL <var|str>)
if(<var|str> LESS_EQUAL <var|str>)
if(<var|str> GREATER_EQUAL <var|str>)
if(<var|str> STRLESS <var|str>)
if(<var|str> STRGREATER <var|str>)
if(<var|str> STREQUAL <var|str>)
if(<var|str> STRLESS_EQUAL <var|str>)
if(<var|str> STRGREATER_EQUAL <var|str>)
if(<var|str> VERSION_LESS <var|str>)
if(<var|str> VERSION_GREATER <var|str>)
if(<var|str> VERSION_EQUAL <var|str>)
if(<var|str> VERSION_LESS_EQUAL <var|str>)
if(<var|str> VERSION_GREATER_EQUAL <var|str>)
if(<var|str> IN_LIST <var>)
```

Commands - Test

```
enable_testing()

add_test(NAME <name> COMMAND <command> [<arg>...]
  [CONFIGURATIONS <config>...]
  [WORKING_DIRECTORY <dir>])

create_test_sourcelist(sourceListName driverName
  test1 test2 test3
  EXTRA_INCLUDE include-file
  FUNCTION function)

build_command(<var> [CONFIGURATION <config>]
  [TARGET <target>])

ctest_build([BUILD <build-dir>] [APPEND]
  [CONFIGURATION <config>] [FLAGS <flags>]
  [PROJECT_NAME <project-name>]
  [TARGET <target-name>]
  [NUMBER_ERRORS <num-err-var>]
  [NUMBER_WARNINGS <num-warn-var>]
  [RETURN_VALUE <result-var>]
  [CAPTURE_CMAKE_ERROR <result-var>])

ctest_configure([BUILD <build-dir>]
  [SOURCE <source-dir>] [APPEND]
  [OPTIONS <options>] [RETURN_VALUE <result-var>]
  [QUIET] [CAPTURE_CMAKE_ERROR <result-var>])

ctest_coverage([BUILD <build-dir>] [APPEND]
  [LABELS <label>...] [RETURN_VALUE <result-var>]
  [CAPTURE_CMAKE_ERROR <result-var] [QUIET])

ctest_empty_binary_directory(directory)
```

```
ctest_memcheck([BUILD <build-dir>] [APPEND]
  [START <start-number>] [END <end-number>]
  [STRIDE <stride-number>]
  [EXCLUDE <regex>] [INCLUDE <regex>]
  [EXCLUDE_LABEL <regex>] [INCLUDE_LABEL <regex>]
  [EXCLUDE_FIXTURE <regex>]
  [EXCLUDE_FIXTURE_SETUP <regex>]
  [EXCLUDE_FIXTURE_CLEANUP <regex>]
  [PARALLEL_LEVEL <level>] [TEST_LOAD <threshold>]
  [SCHEDULE_RANDOM <ON|OFF>]
  [STOP_TIME <time-of-day>]
  [RETURN_VALUE <result-var>]
  [DEFECT_COUNT <defect-count-var>] [QUIET])

ctest_read_custom_files(directory ...)

ctest_run_script([NEW_PROCESS] script_file_name
  script_file_name1 script_file_name2 ...
  [RETURN_VALUE var])

ctest_sleep(<seconds>)
ctest_sleep(<time1> <duration> <time2>)

ctest_start(<model> [<source> [<binary>]]
  [TRACK <track>] [QUIET])
ctest_start([<model> [<source> [<binary>]]]
  [TRACK <track>] APPEND [QUIET])

ctest_submit([PARTS <part>...] [FILES <file>...]
  [HTTPHEADER <header>] [RETRY_COUNT <count>]
  [RETRY_DELAY <delay>] [RETURN_VALUE <result-var>]
  [CAPTURE_CMAKE_ERROR <result-var>] [QUIET])
ctest_submit(CDASH_UPLOAD <file>
  [CDASH_UPLOAD_TYPE <type>] [HTTPHEADER <header>]
  [RETRY_COUNT <count>] [RETRY_DELAY <delay>]
  [RETURN_VALUE <result-var>] [QUIET])

ctest_test([BUILD <build-dir>] [APPEND]
  [START <start-number>] [END <end-number>]
  [STRIDE <stride-number>]
  [EXCLUDE <regex>] [INCLUDE <regex>]
  [EXCLUDE_LABEL <regex>] [INCLUDE_LABEL <regex>]
  [EXCLUDE_FIXTURE <regex>]
  [EXCLUDE_FIXTURE_SETUP <regex>]
  [EXCLUDE_FIXTURE_CLEANUP <regex>]
  [PARALLEL_LEVEL <level>] [TEST_LOAD <threshold>]
  [SCHEDULE_RANDOM <ON|OFF>]
  [STOP_TIME <time-of-day>]
  [RETURN_VALUE <result-var>]
  [CAPTURE_CMAKE_ERROR <result-var>] [QUIET])

ctest_update([SOURCE <source-dir>]
  [RETURN_VALUE <result-var>]
  [CAPTURE_CMAKE_ERROR <result-var>]
  [QUIET])

ctest_upload(FILES <file>... [QUIET]
  [CAPTURE_CMAKE_ERROR <result-var>])
```

Generator Logical Expr

```
$<<condition>:true-string>
$<AND:?[,?]...>
$<BOOL:...>
$<COMPILE_FEATURES:feature[,feature]...>
$<COMPILE_LANGUAGE:lang>
$<COMPILE_LANG_AND_ID:lang,compiler_id>
$<CONFIG:cfg>
$<CUDA_COMPILER_ID:comp>
$<CUDA_COMPILER_VERSION:ver>
$<CXX_COMPILER_ID:comp>
$<CXX_COMPILER_VERSION:ver>
$<C_COMPILER_ID:comp>
$<C_COMPILER_VERSION:ver>
$<DEVICE_LINK:list>
$<EQUAL:a,b>
$<Fortran_COMPILER_ID:comp>
$<Fortran_COMPILER_VERSION:ver>
$<HOST_LINK:list>
$<IF:?,true-value...,false-value...>
$<IN_LIST:a,b>
$<ISPC_COMPILER_ID:comp>
$<ISPC_COMPILER_VERSION:ver>
$<LINK_LANGUAGE:lang>
$<LINK_LANG_AND_ID:lang,compiler_id>
$<NOT:?>
```

Generator Info Expr

```
$<OBJCXX_COMPILER_ID:comp>
$<OBJCXX_COMPILER_VERSION:ver>
$<OBJC_COMPILER_ID:comp>
$<OBJC_COMPILER_VERSION:ver>
$<OR:?[,?]...>
$<PLATFORM_ID:comp>
$<STREQUAL:a,b>
$<TARGET_EXISTS:tgt>
$<TARGET_POLICY:pol>
$<VERSION_EQUAL:v1,v2>
$<VERSION_GREATER:v1,v2>
$<VERSION_GREATER_EQUAL:v1,v2>
$<VERSION_LESS:v1,v2>
$<VERSION_LESS_EQUAL:v1,v2>
```

Generator Info Expr

```
$<COMPILE_LANGUAGE>
$<CONFIG>
$<CUDA_COMPILER_ID>
$<CUDA_COMPILER_VERSION>
$<CXX_COMPILER_ID>
$<CXX_COMPILER_VERSION>
$<C_COMPILER_ID>
$<C_COMPILER_VERSION>
$<Fortran_COMPILER_ID>
$<Fortran_COMPILER_VERSION>
$<INSTALL_PREFIX>
$<INSTALL_PREFIX>
$<ISPC_COMPILER_ID>
$<ISPC_COMPILER_VERSION>
$<LINK_LANGUAGE>
$<OBJCXX_COMPILER_ID>
$<OBJCXX_COMPILER_VERSION>
$<OBJC_COMPILER_ID>
$<OBJC_COMPILER_VERSION>
$<PLATFORM_ID>
$<TARGET_BUNDLE_CONTENT_DIR:tgt>
$<TARGET_BUNDLE_DIR:tgt>
$<TARGET_FILE:tgt>
$<TARGET_FILE_BASE_NAME:tgt>
$<TARGET_FILE_DIR:tgt>
$<TARGET_FILE_NAME:tgt>
$<TARGET_FILE_PREFIX:tgt>
$<TARGET_FILE_SUFFIX:tgt>
$<TARGET_LINKER_FILE:tgt>
$<TARGET_LINKER_FILE_BASE_NAME:tgt>
$<TARGET_LINKER_FILE_DIR:tgt>
$<TARGET_LINKER_FILE_NAME:tgt>
$<TARGET_LINKER_FILE_PREFIX:tgt>
$<TARGET_LINKER_FILE_SUFFIX:tgt>
$<TARGET_PDB_FILE:tgt>
$<TARGET_PDB_FILE_BASE_NAME:tgt>
$<TARGET_PDB_FILE_DIR:tgt>
$<TARGET_PDB_FILE_NAME:tgt>
$<TARGET_PROPERTY:prop>
$<TARGET_PROPERTY:tgt,prop>
$<TARGET_SONAME_FILE:tgt>
$<TARGET_SONAME_FILE_DIR:tgt>
$<TARGET_SONAME_FILE_NAME:tgt>
```

Generator Output Expr

```
$<0:...>
$<1:...>
$<ANGLE-R>
$<BUILD_INTERFACE:...>
$<COMMA>
$<COMMAND_CONFIG:...>
$<FILTER:list,INCLUDE|EXCLUDE,regex>
$<GENEX_EVAL:...>
$<INSTALL_INTERFACE:...>
$<JOIN:list,...>
$<LINK_ONLY:...>
$<LOWER_CASE:...>
$<MAKE_C_IDENTIFIER:...>
$<OUTPUT_CONFIG:...>
$<REMOVE_DUPLICATES:list>
$<SEMICOLON>
$<SHELL_PATH:...>
$<TARGET_GENEX_EVAL:tgt,...>
$<TARGET_NAME:...>
$<TARGET_NAME_IF_EXISTS:...>
$<TARGET_OBJECTS:objLib>
$<UPPER_CASE:...>
```

Properties - Global Scope

ADDITIONAL_CLEAN_FILES
ALLOW_DUPLICATE_CUSTOM_TARGETS
AUTOGEN_SOURCE_GROUP
AUTOGEN_TARGETS_FOLDER
AUTOMOC_SOURCE_GROUP
AUTOMOC_TARGETS_FOLDER
AUTORCC_SOURCE_GROUP
CMAKE_CUDA_KNOWN_FEATURES
CMAKE_CXX_KNOWN_FEATURES
CMAKE_C_KNOWN_FEATURES
CMAKE_ROLE
DEBUG_CONFIGURATIONS
DISABLED_FEATURES
ECLIPSE_EXTRA_CPROJECT_CONTENTS
ECLIPSE_EXTRA_NATURES
ENABLED_FEATURES
ENABLED_LANGUAGES
FIND_LIBRARY_USE_LIB32_PATHS
FIND_LIBRARY_USE_LIB64_PATHS
FIND_LIBRARY_USE_LIBX32_PATHS
FIND_LIBRARY_USE_OPENBSD_VERSIONING
GENERATOR_IS_MULTI_CONFIG
GLOBAL_DEPENDS_DEBUG_MODE
GLOBAL_DEPENDS_NO_CYCLES
IN_TRY_COMPILE
JOB_POOLS
PACKAGES_FOUND
PACKAGES_NOT_FOUND
PREDEFINED_TARGETS_FOLDER
REPORT_UNDEFINED_PROPERTIES
RULE_LAUNCH_COMPILE
RULE_LAUNCH_CUSTOM
RULE_LAUNCH_LINK
RULE_MESSAGES
TARGET_ARCHIVES_MAY_BE_SHARED_LIBS
TARGET_MESSAGES
TARGET_SUPPORTS_SHARED_LIBS
USE_FOLDERS
XCODE_EMIT_EFFECTIVE_PLATFORM_NAME

Properties - Directories

ADDITIONAL_CLEAN_FILES
ADDITIONAL_MAKE_CLEAN_FILES
BINARY_DIR
BUILDSYSTEM_TARGETS
CACHE_VARIABLES
CLEAN_NO_CUSTOM
CMAKE_CONFIGURE_DEPENDS
COMPILE_DEFINITIONS
COMPILE_OPTIONS
DEFINITIONS
EXCLUDE_FROM_ALL
IMPLICIT_DEPENDS_INCLUDE_TRANSFORM
INCLUDE_DIRECTORIES
INCLUDE_REGULAR_EXPRESSION
INTERPROCEDURAL_OPTIMIZATION_<CONFIG>
INTERPROCEDURAL_OPTIMIZATION
LABELS
LINK_DIRECTORIES
LINK_OPTIONS
LISTFILE_STACK
MACROS
PARENT_DIRECTORY
RULE_LAUNCH_COMPILE
RULE_LAUNCH_CUSTOM
RULE_LAUNCH_LINK
SOURCE_DIR
SUBDIRECTORIES
TESTS
TEST_INCLUDE_FILES
VARIABLES
VS_GLOBAL_SECTION_POST_<section>
VS_GLOBAL_SECTION_PRE_<section>
VS_STARTUP_PROJECT

Properties - Source Files

ABSTRACT
AUTORCC_OPTIONS
AUTOUIC_OPTIONS
COMPILE_DEFINITIONS
COMPILE_FLAGS
COMPILE_OPTIONS
EXTERNAL_OBJECT
Fortran_FORMAT
GENERATED
HEADER_FILE_ONLY
INCLUDE_DIRECTORIES
KEEP_EXTENSION
LABELS
LANGUAGE
LOCATION
MACOSX_PACKAGE_LOCATION
OBJECT_DEPENDS
OBJECT_OUTPUTS
SKIP_AUTOGEN
SKIP_AUTOMOC
SKIP_AUTORCC
SKIP_AUTOUIC
SYMBOLIC
VS_COPY_TO_OUT_DIR
VS_CSHARP_<tagname>
VS_DEPLOYMENT_CONTENT
VS_DEPLOYMENT_LOCATION
VS_INCLUDE_IN_VSIX
VS_RESOURCE_GENERATOR
VS_SHADER_DISABLE_OPTIMIZATIONS
VS_SHADER_ENABLE_DEBUG
VS_SHADER_ENTRYPOINT
VS_SHADER_FLAGS
VS_SHADER_MODEL
VS_SHADER_OBJECT_FILE_NAME
VS_SHADER_OUTPUT_HEADER_FILE
VS_SHADER_TYPE
VS_SHADER_VARIABLE_NAME
VS_TOOL_OVERRIDE
VS_XAML_TYPE
WRAP_EXCLUDE
XCODE_EXPLICIT_FILE_TYPE
XCODE_FILE_ATTRIBUTES
XCODE_LAST_KNOWN_FILE_TYPE
ABSTRACT
AUTORCC_OPTIONS
AUTOUIC_OPTIONS
COMPILE_DEFINITIONS
COMPILE_FLAGS
COMPILE_OPTIONS
EXTERNAL_OBJECT
Fortran_FORMAT
Fortran_PREPROCESS
GENERATED
HEADER_FILE_ONLY
INCLUDE_DIRECTORIES
KEEP_EXTENSION
LABELS
LANGUAGE
LOCATION
MACOSX_PACKAGE_LOCATION
OBJECT_DEPENDS
OBJECT_OUTPUTS
SKIP_AUTOGEN
SKIP_AUTOMOC
SKIP_AUTORCC
SKIP_AUTOUIC
SKIP_PRECOMPILE_HEADERS
SKIP_UNITY_BUILD_INCLUSION
Swift_DEPENDENCIES_FILE
Swift_DIAGNOSTICS_FILE
SYMBOLIC
UNITY_GROUP
VS_COPY_TO_OUT_DIR
VS_CSHARP_<tagname>
VS_DEPLOYMENT_CONTENT
VS_DEPLOYMENT_LOCATION
VS_INCLUDE_IN_VSIX
VS_RESOURCE_GENERATOR
VS_SETTINGS
VS_SHADER_DISABLE_OPTIMIZATIONS
VS_SHADER_ENABLE_DEBUG
VS_SHADER_ENTRYPOINT
VS_SHADER_FLAGS
VS_SHADER_MODEL
VS_SHADER_OBJECT_FILE_NAME
VS_SHADER_OUTPUT_HEADER_FILE
VS_SHADER_TYPE
VS_SHADER_VARIABLE_NAME
VS_TOOL_OVERRIDE
VS_XAML_TYPE
WRAP_EXCLUDE
XCODE_EXPLICIT_FILE_TYPE
XCODE_FILE_ATTRIBUTES
XCODE_LAST_KNOWN_FILE_TYPE

Properties - Targets

<CONFIG>_OUTPUT_NAME
<CONFIG>_POSTFIX
<LANG>_CLANG_TIDY
<LANG>_COMPILER_LAUNCHER
<LANG>_CPPCHECK
<LANG>_CPPLINT
<LANG>_INCLUDE_WHAT_YOU_USE
<LANG>_VISIBILITY_PRESET
ADDITIONAL_CLEAN_FILES
AIX_EXPORT_ALL_SYMBOLS
ALIASED_TARGET
ALIAS_GLOBAL
ANDROID_ANT_ADDITIONAL_OPTIONS
ANDROID_API_MIN
ANDROID_API
ANDROID_ARCH
ANDROID_ASSETS_DIRECTORIES
ANDROID_GUI
ANDROID_JAR_DEPENDENCIES
ANDROID_JAR_DIRECTORIES
ANDROID_JAVA_SOURCE_DIR
ANDROID_NATIVE_LIB_DEPENDENCIES
ANDROID_NATIVE_LIB_DIRECTORIES
ANDROID_PROCESS_MAX
ANDROID_PROGUARD_CONFIG_PATH
ANDROID_PROGUARD
ANDROID_SECURE_PROPS_PATH
ANDROID_SKIP_ANT_STEP
ANDROID_STL_TYPE
ARCHIVE_OUTPUT_DIRECTORY_<CONFIG>
ARCHIVE_OUTPUT_DIRECTORY
ARCHIVE_OUTPUT_NAME_<CONFIG>
ARCHIVE_OUTPUT_NAME
AUTOGEN_BUILD_DIR
AUTOGEN_ORIGIN_DEPENDS
AUTOGEN_PARALLEL
AUTOGEN_TARGET_DEPENDS
AUTOMOC_COMPILER_PREDEFINES
AUTOMOC_DEPEND_FILTERS
AUTOMOC_EXECUTABLE
AUTOMOC_MACRO_NAMES
AUTOMOC_MOC_OPTIONS
AUTOMOC_PATH_PREFIX
AUTOMOC
AUTORCC_EXECUTABLE
AUTORCC_OPTIONS
AUTORCC
AUTOUIC_EXECUTABLE
AUTOUIC_OPTIONS
AUTOUIC_SEARCH_PATHS
AUTOUIC
BINARY_DIR
BUILD_RPATH_USE_ORIGIN
BUILD_RPATH
BUILD_WITH_INSTALL_NAME_DIR
BUILD_WITH_INSTALL_RPATH
BUNDLE_EXTENSION
BUNDLE
COMMON_LANGUAGE_RUNTIME
COMPATIBLE_INTERFACE_BOOL
COMPATIBLE_INTERFACE_NUMBER_MAX
COMPATIBLE_INTERFACE_NUMBER_MIN
COMPATIBLE_INTERFACE_STRING
COMPILE_DEFINITIONS
COMPILE_FEATURES
COMPILE_FLAGS
COMPILE_OPTIONS
COMPILE_PDB_NAME_<CONFIG>
COMPILE_PDB_NAME
COMPILE_PDB_OUTPUT_DIRECTORY_<CONFIG>
COMPILE_PDB_OUTPUT_DIRECTORY
CROSSCOMPILING_EMULATOR
CUDA_ARCHITECTURES
CUDA_EXTENSIONS
CUDA_PTX_COMPILATION
CUDA_RESOLVE_DEVICE_SYMBOLS
CUDA_RUNTIME_LIBRARY
CUDA_SEPARABLE_COMPILATION
CUDA_STANDARD_REQUIRED
CUDA_STANDARD
CXX_EXTENSIONS
CXX_STANDARD_REQUIRED
CXX_STANDARD
C_EXTENSIONS
C_STANDARD_REQUIRED
C_STANDARD
DEBUG_POSTFIX
DEFINE_SYMBOL
DEPLOYMENT_ADDITIONAL_FILES
DEPLOYMENT_REMOTE_DIRECTORY
DEPRECATION
DISABLE_PRECOMPILE_HEADERS
DOTNET_TARGET_FRAMEWORK_VERSION
DOTNET_TARGET_FRAMEWORK
ENABLE_EXPORTS
EXCLUDE_FROM_ALL
EXCLUDE_FROM_DEFAULT_BUILD_<CONFIG>
EXCLUDE_FROM_DEFAULT_BUILD
EXPORT_COMPILE_COMMANDS
EXPORT_NAME
EXPORT_PROPERTIES
EchoString
FOLDER
FRAMEWORK_MULTI_CONFIG_POSTFIX_<CONFIG>
FRAMEWORK_VERSION
FRAMEWORK
Fortran_FORMAT
Fortran_MODULE_DIRECTORY
Fortran_PREPROCESS
GENERATOR_FILE_NAME
GHS_INTEGRITY_APP
GHS_NO_SOURCE_GROUP_FILE
GNUtoMS
HAS_CXX
IMPLICIT_DEPENDS_INCLUDE_TRANSFORM
IMPORTED_COMMON_LANGUAGE_RUNTIME
IMPORTED_CONFIGURATIONS
IMPORTED_GLOBAL
IMPORTED_IMPLIB_<CONFIG>
IMPORTED_IMPLIB
IMPORTED_LIBNAME_<CONFIG>
IMPORTED_LIBNAME
IMPORTED_LINK_DEPENDENT_LIBRARIES_<CONFIG>
IMPORTED_LINK_DEPENDENT_LIBRARIES
IMPORTED_LINK_INTERFACE_LANGUAGES_<CONFIG>
IMPORTED_LINK_INTERFACE_LANGUAGES
IMPORTED_LINK_INTERFACE_LIBRARIES_<CONFIG>
IMPORTED_LINK_INTERFACE_LIBRARIES
IMPORTED_LINK_INTERFACE_MULTIPLICITY_<CONFIG>
IMPORTED_LINK_INTERFACE_MULTIPLICITY
IMPORTED_LOCATION_<CONFIG>
IMPORTED_LOCATION
IMPORTED_NO_SONAME_<CONFIG>
IMPORTED_NO_SONAME
IMPORTED_OBJECTS_<CONFIG>
IMPORTED_OBJECTS
IMPORTED_SONAME_<CONFIG>
IMPORTED_SONAME
IMPORTED
IMPORT_PREFIX
IMPORT_SUFFIX
INCLUDE_DIRECTORIES
INSTALL_NAME_DIR
INSTALL_REMOVE_ENVIRONMENT_RPATH
INSTALL_RPATH_USE_LINK_PATH
INSTALL_RPATH
INTERFACE_AUTOUIC_OPTIONS
INTERFACE_COMPILE_DEFINITIONS
INTERFACE_COMPILE_FEATURES
INTERFACE_COMPILE_OPTIONS
INTERFACE_INCLUDE_DIRECTORIES
INTERFACE_LINK_DEPENDS
INTERFACE_LINK_DIRECTORIES
INTERFACE_LINK_LIBRARIES
INTERFACE_LINK_OPTIONS
INTERFACE_POSITION_INDEPENDENT_CODE
INTERFACE_PRECOMPILE_HEADERS
INTERFACE_SOURCES
INTERFACE_SYSTEM_INCLUDE_DIRECTORIES
INTERPROCEDURAL_OPTIMIZATION_<CONFIG>
INTERPROCEDURAL_OPTIMIZATION
IOS_INSTALL_COMBINED
ISPC_HEADER_DIRECTORY
ISPC_HEADER_SUFFIX
ISPC_INSTRUCTION_SETS
JOB_POOL_COMPILE
JOB_POOL_LINK
JOB_POOL_PRECOMPILE_HEADER
LABELS
LIBRARY_OUTPUT_DIRECTORY_<CONFIG>
LIBRARY_OUTPUT_DIRECTORY
LIBRARY_OUTPUT_NAME_<CONFIG>
LIBRARY_OUTPUT_NAME
LINKER_LANGUAGE
LINK_DEPENDS_NO_SHARED
LINK_DEPENDS
LINK_DIRECTORIES
LINK_FLAGS_<CONFIG>
LINK_FLAGS
LINK_INTERFACE_LIBRARIES_<CONFIG>
LINK_INTERFACE_LIBRARIES
LINK_INTERFACE_MULTIPLICITY_<CONFIG>
LINK_INTERFACE_MULTIPLICITY
LINK_LIBRARIES
LINK_OPTIONS
LINK_SEARCH_END_STATIC
LINK_SEARCH_START_STATIC
LINK_WHAT_YOU_USE
LOCATION_<CONFIG>
LOCATION
MACHO_COMPATIBILITY_VERSION
MACHO_CURRENT_VERSION
MACOSX_BUNDLE_INFO_PLIST
MACOSX_BUNDLE
MACOSX_FRAMEWORK_INFO_PLIST
MACOSX_RPATH
MANUALLY_ADDED_DEPENDENCIES
MAP_IMPORTED_CONFIG_<CONFIG>
MSVC_RUNTIME_LIBRARY
NAME
NO_SONAME
NO_SYSTEM_FROM_IMPORTED
OBJCXX_EXTENSIONS
OBJCXX_STANDARD_REQUIRED
OBJCXX_STANDARD
OBJC_EXTENSIONS
OBJC_STANDARD_REQUIRED
OBJC_STANDARD
OPTIMIZE_DEPENDENCIES
OSX_ARCHITECTURES_<CONFIG>
OSX_ARCHITECTURES
OUTPUT_NAME_<CONFIG>
OUTPUT_NAME
PCH_INSTANTIATE_TEMPLATES
PCH_WARN_INVALID
PDB_NAME_<CONFIG>
PDB_NAME
PDB_OUTPUT_DIRECTORY_<CONFIG>
PDB_OUTPUT_DIRECTORY
POSITION_INDEPENDENT_CODE
PRECOMPILE_HEADERS_REUSE_FROM
PRECOMPILE_HEADERS
PREFIX
PRIVATE_HEADER
PROJECT_LABEL
PUBLIC_HEADER
RESOURCE
RULE_LAUNCH_COMPILE
RULE_LAUNCH_CUSTOM
RULE_LAUNCH_LINK
RUNTIME_OUTPUT_DIRECTORY_<CONFIG>
RUNTIME_OUTPUT_DIRECTORY
RUNTIME_OUTPUT_NAME_<CONFIG>
RUNTIME_OUTPUT_NAME
SKIP_BUILD_RPATH
SOURCES
SOURCE_DIR
SOVERSION
STATIC_LIBRARY_FLAGS_<CONFIG>
STATIC_LIBRARY_FLAGS
STATIC_LIBRARY_OPTIONS
SUFFIX
Swift_DEPENDENCIES_FILE
Swift_LANGUAGE_VERSION
Swift_MODULE_DIRECTORY
Swift_MODULE_NAME
TYPE
UNITY_BUILD_BATCH_SIZE
UNITY_BUILD_CODE_AFTER_INCLUDE
UNITY_BUILD_CODE_BEFORE_INCLUDE
UNITY_BUILD_MODE
UNITY_BUILD_UNIQUE_ID
UNITY_BUILD
VERSION
VISIBILITY_INLINES_HIDDEN
VS_CONFIGURATION_TYPE
VS_DEBUGGER_COMMAND_ARGUMENTS
VS_DEBUGGER_COMMAND
VS_DEBUGGER_ENVIRONMENT
VS_DEBUGGER_WORKING_DIRECTORY
VS_DESKTOP_EXTENSIONS_VERSION
VS_DOTNET_DOCUMENTATION_FILE
VS_DOTNET_REFERENCEPROP_<refname>_TAG_<tagname>
VS_DOTNET_REFERENCES_COPY_LOCAL
VS_DOTNET_REFERENCES
VS_DOTNET_REFERENCE_<refname>

VS_DOTNET_TARGET_FRAMEWORK_VERSION
VS_DPI_AWARE
VS_GLOBAL_<variable>
VS_GLOBAL_KEYWORD
VS_GLOBAL_PROJECT_TYPES
VS_GLOBAL_ROOTNAMESPACE
VS_IOT_EXTENSIONS_VERSION
VS_IOT_STARTUP_TASK
VS_JUST_MY_CODE_DEBUGGING
VS_KEYWORD
VS_MOBILE_EXTENSIONS_VERSION
VS_NO_SOLUTION_DEPLOY
VS_PACKAGE_REFERENCES
VS_PLATFORM_TOOLSET
VS_PROJECT_IMPORT
VS_SCC_AUXPATH
VS_SCC_LOCALPATH
VS_SCC_PROJECTNAME
VS_SCC_PROVIDER
VS_SDK_REFERENCES
VS_SOLUTION_DEPLOY
VS_SOURCE_SETTINGS_<tool>
VS_USER_PROPS
VS_WINDOWS_TARGET_PLATFORM_MIN_VERSION
VS_WINRT_COMPONENT
VS_WINRT_EXTENSIONS
VS_WINRT_REFERENCES
WIN32_EXECUTABLE
WINDOWS_EXPORT_ALL_SYMBOLS
XCODE_ATTRIBUTE_<an-attribute>
XCODE_EMBED_<type>_PATH
XCODE_EMBED_<type>
XCODE_EMBED_FRAMEWORKS_CODE_SIGN_ON_COPY
XCODE_EMBED_FRAMEWORKS_REMOVE_HEADERS_ON_COPY
XCODE_EXPLICIT_FILE_TYPE
XCODE_GENERATE_SCHEME
XCODE_LINK_BUILD_PHASE_MODE
XCODE_PRODUCT_TYPE
XCODE_SCHEME_ADDRESS_SANITIZER_USE_AFTER_RETURN
XCODE_SCHEME_ADDRESS_SANITIZER
XCODE_SCHEME_ARGUMENTS
XCODE_SCHEME_DEBUG_AS_ROOT
XCODE_SCHEME_DEBUG_DOCUMENT_VERSIONING
XCODE_SCHEME_DISABLE_MAIN_THREAD_CHECKER
XCODE_SCHEME_DYNAMIC_LIBRARY_LOADS
XCODE_SCHEME_DYNAMIC_LINKER_API_USAGE
XCODE_SCHEME_ENVIRONMENT
XCODE_SCHEME_EXECUTABLE
XCODE_SCHEME_GUARD_MALLOC
XCODE_SCHEME_MAIN_THREAD_CHECKER_STOP
XCODE_SCHEME_MALLOC_GUARD_EDGES
XCODE_SCHEME_MALLOC_SCRIBBLE
XCODE_SCHEME_MALLOC_STACK
XCODE_SCHEME_THREAD_SANITIZER_STOP
XCODE_SCHEME_THREAD_SANITIZER
XCODE_SCHEME_UNDEFINED_BEHAVIOUR_SANITIZER_STOP
XCODE_SCHEME_UNDEFINED_BEHAVIOUR_SANITIZER
XCODE_SCHEME_WORKING_DIRECTORY
XCODE_SCHEME_ZOMBIE_OBJECTS
XCTEST

Properties - Tests

ATTACHED_FILES_ON_FAIL
ATTACHED_FILES
COST
DEPENDS
DISABLED
ENVIRONMENT
FAIL_REGULAR_EXPRESSION
FIXTURES_CLEANUP
FIXTURES_REQUIRED
FIXTURES_SETUP
LABELS
MEASUREMENT
PASS_REGULAR_EXPRESSION
PROCESSORS
PROCESSOR_AFFINITY
REQUIRED_FILES
RESOURCE_GROUPS
RESOURCE_LOCK
RUN_SERIAL
SKIP_REGULAR_EXPRESSION
SKIP_RETURN_CODE
TIMEOUT_AFTER_MATCH
TIMEOUT
WILL_FAIL
WORKING_DIRECTORY

Properties - Cache Entries

ADVANCED
HELPSTRING
MODIFIED
STRINGS
TYPE
VALUE

Properties - Installed Files

CPACK_DESKTOP_SHORTCUTS
CPACK_NEVER_OVERWRITE
CPACK_PERMANENT
CPACK_START_MENU_SHORTCUTS
CPACK_STARTUP_SHORTCUTS
CPACK_WIX_ACL

Variables - Information

CMAKE_AR
CMAKE_ARGC
CMAKE_ARGV0
CMAKE_AUTOGEN_ORIGIN_DEPENDS
CMAKE_BINARY_DIR
CMAKE_BUILD_TOOL
CMAKE_CACHEFILE_DIR
CMAKE_CACHE_MAJOR_VERSION
CMAKE_CACHE_MINOR_VERSION
CMAKE_CACHE_PATCH_VERSION
CMAKE_CFG_INTDIR
CMAKE_CLANG_VFS_OVERLAY
CMAKE_COMMAND
CMAKE_CPACK_COMMAND
CMAKE_CROSSCOMPILING
CMAKE_CROSSCOMPILING_EMULATOR
CMAKE_CROSS_CONFIGS
CMAKE_CTEST_ARGUMENTS
CMAKE_CTEST_COMMAND
CMAKE_CUDA_ARCHITECTURES
CMAKE_CUDA_COMPILE_FEATURES
CMAKE_CURRENT_BINARY_DIR
CMAKE_CURRENT_FUNCTION
CMAKE_CURRENT_FUNCTION_LIST_DIR
CMAKE_CURRENT_FUNCTION_LIST_FILE
CMAKE_CURRENT_FUNCTION_LIST_LINE
CMAKE_CURRENT_LIST_DIR
CMAKE_CURRENT_LIST_FILE
CMAKE_CURRENT_LIST_LINE
CMAKE_CURRENT_SOURCE_DIR
CMAKE_DIRECTORY_LABELS
CMAKE_DL_LIBS
CMAKE_DOTNET_TARGET_FRAMEWORK
CMAKE_DOTNET_TARGET_FRAMEWORK_VERSION
CMAKE_EDIT_COMMAND
CMAKE_EXECUTABLE_SUFFIX
CMAKE_EXECUTE_PROCESS_COMMAND_ECHO
CMAKE_EXTRA_GENERATOR
CMAKE_EXTRA_SHARED_LIBRARY_SUFFIXES
CMAKE_FIND_PACKAGE_NAME
CMAKE_FIND_PACKAGE_SORT_DIRECTION
CMAKE_FIND_PACKAGE_SORT_ORDER
CMAKE_GENERATOR
CMAKE_GENERATOR_INSTANCE
CMAKE_GENERATOR_PLATFORM
CMAKE_GENERATOR_TOOLSET
CMAKE_GHS_NO_SOURCE_GROUP_FILE
CMAKE_GLOBAL_AUTOGEN_TARGET
CMAKE_GLOBAL_AUTOGEN_TARGET_NAME
CMAKE_GLOBAL_AUTORCC_TARGET
CMAKE_GLOBAL_AUTORCC_TARGET_NAME
CMAKE_HOME_DIRECTORY
CMAKE_IMPORT_LIBRARY_PREFIX
CMAKE_IMPORT_LIBRARY_SUFFIX
CMAKE_JOB_POOL_COMPILE
CMAKE_JOB_POOL_LINK
CMAKE_JOB_POOL_PRECOMPILE_HEADER
CMAKE_JOB_POOLS
CMAKE_<LANG>_COMPILER_AR
CMAKE_<LANG>_COMPILER_RANLIB
CMAKE_LINK_LIBRARY_SUFFIX
CMAKE_LINK_SEARCH_END_STATIC
CMAKE_LINK_SEARCH_START_STATIC
CMAKE_MAJOR_VERSION
CMAKE_MAKE_PROGRAM
CMAKE_MATCH_COUNT
CMAKE_MAXIMUM_RECURSION_DEPTH
CMAKE_MESSAGE_CONTEXT
CMAKE_MESSAGE_CONTEXT_SHOW
CMAKE_MESSAGE_INDENT
CMAKE_MESSAGE_LOG_LEVEL
CMAKE_MATCH_<n>
CMAKE_MINIMUM_REQUIRED_VERSION
CMAKE_MINOR_VERSION
CMAKE_NETRC
CMAKE_NETRC_FILE
CMAKE_PARENT_LIST_FILE
CMAKE_PATCH_VERSION
CMAKE_PCH_INSTANTIATE_TEMPLATES
CMAKE_PCH_WARN_INVALID
CMAKE_PROJECT_DESCRIPTION
CMAKE_PROJECT_HOMEPAGE_URL
CMAKE_PROJECT_INCLUDE
CMAKE_PROJECT_INCLUDE_BEFORE
CMAKE_PROJECT_NAME
CMAKE_PROJECT_VERSION
CMAKE_PROJECT_VERSION_MAJOR
CMAKE_PROJECT_VERSION_MINOR
CMAKE_PROJECT_VERSION_PATCH
CMAKE_PROJECT_VERSION_TWEAK
CMAKE_RANLIB
CMAKE_ROOT
CMAKE_RULE_MESSAGES
CMAKE_SCRIPT_MODE_FILE
CMAKE_SHARED_LIBRARY_PREFIX
CMAKE_SHARED_LIBRARY_SUFFIX
CMAKE_SHARED_MODULE_PREFIX
CMAKE_SHARED_MODULE_SUFFIX
CMAKE_SIZEOF_VOID_P
CMAKE_SKIP_INSTALL_RULES
CMAKE_SKIP_RPATH
CMAKE_SOURCE_DIR
CMAKE_STATIC_LIBRARY_PREFIX
CMAKE_STATIC_LIBRARY_SUFFIX
CMAKE_TOOLCHAIN_FILE
CMAKE_TWEAK_VERSION
CMAKE_UNITY_BUILD
CMAKE_UNITY_BUILD_BATCH_SIZE
CMAKE_UNITY_BUILD_UNIQUE_ID
CMAKE_VERBOSE_MAKEFILE
CMAKE_VERSION
CMAKE_VS_DEVENV_COMMAND
CMAKE_VS_INTEL_Fortran_PROJECT_VERSION
CMAKE_VS_JUST_MY_CODE_DEBUGGING
CMAKE_VS_MSBUILD_COMMAND
CMAKE_VS_NsightTegra_VERSION
CMAKE_VS_PLATFORM_NAME
CMAKE_VS_PLATFORM_NAME_DEFAULT
CMAKE_VS_PLATFORM_TOOLSET
CMAKE_VS_PLATFORM_TOOLSET_CUDA
CMAKE_VS_PLATFORM_TOOLSET_CUDA_CUSTOM_DIR
CMAKE_VS_PLATFORM_TOOLSET_HOST_ARCHITECTURE
CMAKE_VS_PLATFORM_TOOLSET_VERSION
CMAKE_VS_WINDOWS_TARGET_PLATFORM_VERSION
CMAKE_VS_WINDOWS_TARGET_PLATFORM_VERSION_MAXIMUM
CMAKE_XCODE_GENERATE_SCHEME
CMAKE_XCODE_PLATFORM_TOOLSET
<PROJECT-NAME>_BINARY_DIR
<PROJECT-NAME>_DESCRIPTION
<PROJECT-NAME>_HOMEPAGE_URL
<PROJECT-NAME>_SOURCE_DIR
<PROJECT-NAME>_VERSION
<PROJECT-NAME>_VERSION_MAJOR
<PROJECT-NAME>_VERSION_MINOR
<PROJECT-NAME>_VERSION_PATCH
<PROJECT-NAME>_VERSION_TWEAK
PROJECT_BINARY_DIR
PROJECT_DESCRIPTION
PROJECT_HOMEPAGE_URL
PROJECT_NAME
PROJECT_SOURCE_DIR
PROJECT_VERSION
PROJECT_VERSION_MAJOR
PROJECT_VERSION_MINOR
PROJECT_VERSION_PATCH
PROJECT_VERSION_TWEAK

Variables - Change Behavior

BUILD_SHARED_LIBS
CMAKE_ABSOLUTE_DESTINATION_FILES
CMAKE_AIX_EXPORT_ALL_SYMBOLS
CMAKE_APPBUNDLE_PATH
CMAKE_APPLE_SILICON_PROCESSOR
CMAKE_AUTOMOC_RELAXED_MODE
CMAKE_BACKWARDS_COMPATIBILITY
CMAKE_BUILD_TYPE
CMAKE_CODEBLOCKS_COMPILER_ID
CMAKE_CODEBLOCKS_EXCLUDE_EXTERNAL_FILES
CMAKE_CODELITE_USE_TARGETS
CMAKE_COLOR_MAKEFILE
CMAKE_CONFIGURATION_TYPES
CMAKE_DEBUG_TARGET_PROPERTIES
CMAKE_DEFAULT_BUILD_TYPE
CMAKE_DEFAULT_CONFIGS
CMAKE_DEPENDS_IN_PROJECT_ONLY
CMAKE_DEPENDS_USE_COMPILER
CMAKE_DISABLE_FIND_PACKAGE_<PackageName>
CMAKE_DISABLE_PRECOMPILE_HEADERS
CMAKE_ECLIPSE_GENERATE_LINKED_RESOURCES
CMAKE_ECLIPSE_GENERATE_SOURCE_PROJECT
CMAKE_ECLIPSE_MAKE_ARGUMENTS
CMAKE_ECLIPSE_RESOURCE_ENCODING
CMAKE_ECLIPSE_VERSION
CMAKE_ERROR_DEPRECATED
CMAKE_ERROR_ON_ABSOLUTE_INSTALL_DESTINATION
CMAKE_EXPORT_COMPILE_COMMANDS
CMAKE_EXPORT_NO_PACKAGE_REGISTRY
CMAKE_EXPORT_PACKAGE_REGISTRY
CMAKE_FIND_APPBUNDLE
CMAKE_FIND_DEBUG_MODE
CMAKE_FIND_FRAMEWORK
CMAKE_FIND_LIBRARY_CUSTOM_LIB_SUFFIX
CMAKE_FIND_LIBRARY_PREFIXES
CMAKE_FIND_LIBRARY_SUFFIXES
CMAKE_FIND_NO_INSTALL_PREFIX
CMAKE_FIND_PACKAGE_NO_PACKAGE_REGISTRY
CMAKE_FIND_PACKAGE_NO_SYSTEM_PACKAGE_REGISTRY
CMAKE_FIND_PACKAGE_PREFER_CONFIG
CMAKE_FIND_PACKAGE_RESOLVE_SYMLINKS
CMAKE_FIND_PACKAGE_WARN_NO_MODULE
CMAKE_FIND_ROOT_PATH
CMAKE_FIND_ROOT_PATH_MODE_INCLUDE
CMAKE_FIND_ROOT_PATH_MODE_LIBRARY
CMAKE_FIND_ROOT_PATH_MODE_PACKAGE
CMAKE_FIND_ROOT_PATH_MODE_PROGRAM
CMAKE_FIND_USE_CMAKE_ENVIRONMENT_PATH
CMAKE_FIND_USE_CMAKE_PATH
CMAKE_FIND_USE_CMAKE_SYSTEM_PATH
CMAKE_FIND_USE_PACKAGE_REGISTRY
CMAKE_FIND_USE_PACKAGE_ROOT_PATH
CMAKE_FIND_USE_SYSTEM_ENVIRONMENT_PATH
CMAKE_FIND_USE_SYSTEM_PACKAGE_REGISTRY
CMAKE_FRAMEWORK_PATH
CMAKE_IGNORE_PATH
CMAKE_INCLUDE_DIRECTORIES_BEFORE
CMAKE_INCLUDE_DIRECTORIES_PROJECT_BEFORE
CMAKE_INCLUDE_PATH
CMAKE_INSTALL_DEFAULT_COMPONENT_NAME
CMAKE_INSTALL_DEFAULT_DIRECTORY_PERMISSIONS
CMAKE_INSTALL_MESSAGE
CMAKE_INSTALL_PREFIX
CMAKE_INSTALL_PREFIX_INITIALIZED_TO_DEFAULT
CMAKE_INSTALL_REMOVE_ENVIRONMENT_RPATH
CMAKE_LIBRARY_PATH
CMAKE_LINK_DIRECTORIES_BEFORE
CMAKE_MFC_FLAG
CMAKE_MODULE_PATH
CMAKE_NOT_USING_CONFIG_FLAGS
CMAKE_POLICY_DEFAULT_CMP<NNNN>
CMAKE_POLICY_WARNING_CMP<NNNN>
CMAKE_PREFIX_PATH
CMAKE_PROGRAM_PATH
CMAKE_PROJECT_<PROJECT-NAME>_INCLUDE
CMAKE_PROJECT_<PROJECT-NAME>_INCLUDE_BEFORE
CMAKE_SKIP_INSTALL_ALL_DEPENDENCY
CMAKE_STAGING_PREFIX
CMAKE_SUBLIME_TEXT_2_ENV_SETTINGS
CMAKE_SUBLIME_TEXT_2_EXCLUDE_BUILD_TREE
CMAKE_SUPPRESS_REGENERATION
CMAKE_SYSROOT
CMAKE_SYSROOT_COMPILE
CMAKE_SYSROOT_LINK
CMAKE_SYSTEM_APPBUNDLE_PATH
CMAKE_SYSTEM_FRAMEWORK_PATH
CMAKE_SYSTEM_IGNORE_PATH
CMAKE_SYSTEM_INCLUDE_PATH
CMAKE_SYSTEM_LIBRARY_PATH
CMAKE_SYSTEM_PREFIX_PATH
CMAKE_SYSTEM_PROGRAM_PATH
CMAKE_USER_MAKE_RULES_OVERRIDE
CMAKE_WARN_DEPRECATED
CMAKE_WARN_ON_ABSOLUTE_INSTALL_DESTINATION
CMAKE_XCODE_GENERATE_TOP_LEVEL_PROJECT_ONLY
CMAKE_XCODE_LINK_BUILD_PHASE_MODE
CMAKE_XCODE_SCHEME_ADDRESS_SANITIZER
CMAKE_XCODE_SCHEME_ADDRESS_SANITIZER_USE_AFTER_RETURN
CMAKE_XCODE_SCHEME_DEBUG_DOCUMENT_VERSIONING
CMAKE_XCODE_SCHEME_THREAD_SANITIZER
CMAKE_XCODE_SCHEME_THREAD_SANITIZER_STOP
CMAKE_XCODE_SCHEME_UNDEFINED_BEHAVIOUR_SANITIZER
CMAKE_XCODE_SCHEME_UNDEFINED_BEHAVIOUR_SANITIZER_STOP
CMAKE_XCODE_SCHEME_WORKING_DIRECTORY
CMAKE_XCODE_SCHEME_DISABLE_MAIN_THREAD_CHECKER
CMAKE_XCODE_SCHEME_MAIN_THREAD_CHECKER_STOP
CMAKE_XCODE_SCHEME_MALLOC_SCRIBBLE
CMAKE_XCODE_SCHEME_MALLOC_GUARD_EDGES
CMAKE_XCODE_SCHEME_GUARD_MALLOC
CMAKE_XCODE_SCHEME_ZOMBIE_OBJECTS
CMAKE_XCODE_SCHEME_MALLOC_STACK
CMAKE_XCODE_SCHEME_DYNAMIC_LINKER_API_USAGE
CMAKE_XCODE_SCHEME_ENVIRONMENT
CMAKE_XCODE_SCHEME_DYNAMIC_LIBRARY_LOADS
<PackageName>_ROOT

Variables - System Description

ANDROID
APPLE
BORLAND
CACHE
CMAKE_CL_64
CMAKE_COMPILER_2005
CMAKE_HOST_APPLE
CMAKE_HOST_SOLARIS
CMAKE_HOST_SYSTEM
CMAKE_HOST_SYSTEM_NAME
CMAKE_HOST_SYSTEM_PROCESSOR
CMAKE_HOST_SYSTEM_VERSION
CMAKE_HOST_UNIX
CMAKE_HOST_WIN32
CMAKE_LIBRARY_ARCHITECTURE
CMAKE_LIBRARY_ARCHITECTURE_REGEX
CMAKE_OBJECT_PATH_MAX
CMAKE_OPTIMIZE_DEPENDENCIES
CMAKE_SYSTEM
CMAKE_SYSTEM_NAME
CMAKE_SYSTEM_PROCESSOR
CMAKE_SYSTEM_VERSION
CYGWIN
ENV
GHS-MULTI
IOS
MINGW
MSVC
MSVC10
MSVC11
MSVC12
MSVC14
MSVC60
MSVC70
MSVC71
MSVC80
MSVC90
MSVC_IDE
MSVC_TOOLSET_VERSION
MSVC_VERSION
MSYS
UNIX
WIN32
WINCE
WINDOWS_PHONE
WINDOWS_STORE
XCODE
XCODE_VERSION

Variables - Build Control

CMAKE_ANDROID_ANT_ADDITIONAL_OPTIONS
CMAKE_ANDROID_API
CMAKE_ANDROID_API_MIN
CMAKE_ANDROID_ARCH
CMAKE_ANDROID_ARCH_ABI
CMAKE_ANDROID_ARM_MODE
CMAKE_ANDROID_ARM_NEON
CMAKE_ANDROID_ASSETS_DIRECTORIES
CMAKE_ANDROID_EXCEPTIONS
CMAKE_ANDROID_GUI
CMAKE_ANDROID_JAR_DEPENDENCIES
CMAKE_ANDROID_JAR_DIRECTORIES
CMAKE_ANDROID_JAVA_SOURCE_DIR
CMAKE_ANDROID_NATIVE_LIB_DEPENDENCIES
CMAKE_ANDROID_NATIVE_LIB_DIRECTORIES
CMAKE_ANDROID_NDK
CMAKE_ANDROID_NDK_DEPRECATED_HEADERS
CMAKE_ANDROID_NDK_TOOLCHAIN_HOST_TAG
CMAKE_ANDROID_NDK_TOOLCHAIN_VERSION
CMAKE_ANDROID_NDK_VERSION
CMAKE_ANDROID_PROCESS_MAX
CMAKE_ANDROID_PROGUARD
CMAKE_ANDROID_PROGUARD_CONFIG_PATH
CMAKE_ANDROID_RTTI

CMAKE_ANDROID_SECURE_PROPS_PATH
CMAKE_ANDROID_SKIP_ANT_STEP
CMAKE_ANDROID_STANDALONE_TOOLCHAIN
CMAKE_ANDROID_STL_TYPE
CMAKE_ARCHIVE_OUTPUT_DIRECTORY
CMAKE_ARCHIVE_OUTPUT_DIRECTORY_<CONFIG>
CMAKE_AUTOGEN_PARALLEL
CMAKE_AUTOGEN_VERBOSE
CMAKE_AUTOMOC
CMAKE_AUTOMOC_COMPILER_PREDEFINES
CMAKE_AUTOMOC_DEPEND_FILTERS
CMAKE_AUTOMOC_MACRO_NAMES
CMAKE_AUTOMOC_MOC_OPTIONS
CMAKE_AUTOMOC_PATH_PREFIX
CMAKE_AUTORCC
CMAKE_AUTORCC_OPTIONS
CMAKE_AUTOUIC
CMAKE_AUTOUIC_OPTIONS
CMAKE_AUTOUIC_SEARCH_PATHS
CMAKE_BUILD_RPATH
CMAKE_BUILD_RPATH_USE_ORIGIN
CMAKE_BUILD_WITH_INSTALL_NAME_DIR
CMAKE_BUILD_WITH_INSTALL_RPATH
CMAKE_COMPILE_PDB_OUTPUT_DIRECTORY
CMAKE_COMPILE_PDB_OUTPUT_DIRECTORY_<CONFIG>
CMAKE_<CONFIG>_POSTFIX
CMAKE_CUDA_SEPARABLE_COMPILATION
CMAKE_DEBUG_POSTFIX
CMAKE_ENABLE_EXPORTS
CMAKE_EXE_LINKER_FLAGS
CMAKE_EXE_LINKER_FLAGS_<CONFIG>
CMAKE_EXE_LINKER_FLAGS_<CONFIG>_INIT
CMAKE_EXE_LINKER_FLAGS_INIT
CMAKE_FOLDER
CMAKE_FRAMEWORK
CMAKE_FRAMEWORK_MULTI_CONFIG_POSTFIX_<CONFIG>
CMAKE_Fortran_FORMAT
CMAKE_Fortran_MODULE_DIRECTORY
CMAKE_Fortran_PREPROCESS
CMAKE_GNUtoMS
CMAKE_INCLUDE_CURRENT_DIR
CMAKE_INCLUDE_CURRENT_DIR_IN_INTERFACE
CMAKE_INSTALL_NAME_DIR
CMAKE_INSTALL_RPATH
CMAKE_INSTALL_RPATH_USE_LINK_PATH
CMAKE_INTERPROCEDURAL_OPTIMIZATION
CMAKE_INTERPROCEDURAL_OPTIMIZATION_<CONFIG>
CMAKE_IOS_INSTALL_COMBINED
CMAKE_ISPC_HEADER_DIRECTORY
CMAKE_ISPC_HEADER_SUFFIX
CMAKE_ISPC_INSTRUCTION_SETS
CMAKE_<LANG>_CLANG_TIDY
CMAKE_<LANG>_COMPILER_LAUNCHER
CMAKE_<LANG>_CPPCHECK
CMAKE_<LANG>_CPPLINT
CMAKE_<LANG>_INCLUDE_WHAT_YOU_USE
CMAKE_<LANG>_VISIBILITY_PRESET
CMAKE_LIBRARY_OUTPUT_DIRECTORY
CMAKE_LIBRARY_OUTPUT_DIRECTORY_<CONFIG>
CMAKE_LIBRARY_PATH_FLAG
CMAKE_LINK_DEF_FILE_FLAG
CMAKE_LINK_DEPENDS_NO_SHARED
CMAKE_LINK_INTERFACE_LIBRARIES
CMAKE_LINK_LIBRARY_FILE_FLAG
CMAKE_LINK_LIBRARY_FLAG
CMAKE_LINK_WHAT_YOU_USE
CMAKE_MACOSX_BUNDLE
CMAKE_MACOSX_RPATH
CMAKE_MAP_IMPORTED_CONFIG_<CONFIG>
CMAKE_MODULE_LINKER_FLAGS
CMAKE_MODULE_LINKER_FLAGS_<CONFIG>
CMAKE_MODULE_LINKER_FLAGS_<CONFIG>_INIT
CMAKE_MODULE_LINKER_FLAGS_INIT
CMAKE_MSVCIDE_RUN_PATH
CMAKE_MSVC_RUNTIME_LIBRARY
CMAKE_NINJA_OUTPUT_PATH_PREFIX
CMAKE_NO_BUILTIN_CHRPATH
CMAKE_NO_SYSTEM_FROM_IMPORTED
CMAKE_OBJCXX_EXTENSIONS
CMAKE_OBJCXX_STANDARD
CMAKE_OBJCXX_STANDARD_REQUIRED
CMAKE_OBJC_EXTENSIONS
CMAKE_OBJC_STANDARD
CMAKE_OBJC_STANDARD_REQUIRED
CMAKE_OSX_ARCHITECTURES
CMAKE_OSX_DEPLOYMENT_TARGET
CMAKE_OSX_SYSROOT
CMAKE_PDB_OUTPUT_DIRECTORY
CMAKE_PDB_OUTPUT_DIRECTORY_<CONFIG>
CMAKE_POSITION_INDEPENDENT_CODE
CMAKE_RUNTIME_OUTPUT_DIRECTORY
CMAKE_RUNTIME_OUTPUT_DIRECTORY_<CONFIG>
CMAKE_SHARED_LINKER_FLAGS
CMAKE_SHARED_LINKER_FLAGS_<CONFIG>
CMAKE_SHARED_LINKER_FLAGS_<CONFIG>_INIT
CMAKE_SHARED_LINKER_FLAGS_INIT
CMAKE_SKIP_BUILD_RPATH
CMAKE_SKIP_INSTALL_RPATH
CMAKE_STATIC_LINKER_FLAGS
CMAKE_STATIC_LINKER_FLAGS_<CONFIG>
CMAKE_STATIC_LINKER_FLAGS_<CONFIG>_INIT
CMAKE_STATIC_LINKER_FLAGS_INIT
CMAKE_TRY_COMPILE_CONFIGURATION
CMAKE_TRY_COMPILE_PLATFORM_VARIABLES
CMAKE_TRY_COMPILE_TARGET_TYPE
CMAKE_USE_RELATIVE_PATHS
CMAKE_VISIBILITY_INLINES_HIDDEN
CMAKE_VS_GLOBALS
CMAKE_VS_INCLUDE_INSTALL_TO_DEFAULT_BUILD
CMAKE_VS_INCLUDE_PACKAGE_TO_DEFAULT_BUILD
CMAKE_VS_SDK_EXCLUDE_DIRECTORIES
CMAKE_VS_SDK_EXECUTABLE_DIRECTORIES
CMAKE_VS_SDK_INCLUDE_DIRECTORIES
CMAKE_VS_SDK_LIBRARY_DIRECTORIES
CMAKE_VS_SDK_LIBRARY_WINRT_DIRECTORIES
CMAKE_VS_SDK_REFERENCE_DIRECTORIES
CMAKE_VS_SDK_SOURCE_DIRECTORIES
CMAKE_VS_WINRT_BY_DEFAULT
CMAKE_WIN32_EXECUTABLE
CMAKE_WINDOWS_EXPORT_ALL_SYMBOLS
CMAKE_XCODE_ATTRIBUTE_<an-attribute>
CMAKE_XCODE_BUILD_SYSTEM
EXECUTABLE_OUTPUT_PATH
LIBRARY_OUTPUT_PATH

Variables - Languages

CMAKE_COMPILER_IS_GNUCC
CMAKE_COMPILER_IS_GNUCXX
CMAKE_COMPILER_IS_GNUG77
CMAKE_CUDA_HOST_COMPILER
CMAKE_CUDA_RESOLVE_DEVICE_SYMBOLS
CMAKE_CUDA_RUNTIME_LIBRARY
CMAKE_CUDA_EXTENSIONS
CMAKE_CUDA_STANDARD
CMAKE_CUDA_STANDARD_REQUIRED
CMAKE_CUDA_TOOLKIT_INCLUDE_DIRECTORIES
CMAKE_CXX_COMPILE_FEATURES
CMAKE_CXX_EXTENSIONS
CMAKE_CXX_STANDARD
CMAKE_CXX_STANDARD_REQUIRED
CMAKE_C_COMPILE_FEATURES
CMAKE_C_EXTENSIONS
CMAKE_C_STANDARD
CMAKE_C_STANDARD_REQUIRED
CMAKE_Fortran_MODDIR_DEFAULT
CMAKE_Fortran_MODDIR_FLAG
CMAKE_Fortran_MODOUT_FLAG
CMAKE_INTERNAL_PLATFORM_ABI
CMAKE_<LANG>_ANDROID_TOOLCHAIN_MACHINE
CMAKE_<LANG>_ANDROID_TOOLCHAIN_PREFIX
CMAKE_<LANG>_ANDROID_TOOLCHAIN_SUFFIX
CMAKE_<LANG>_ARCHIVE_APPEND
CMAKE_<LANG>_ARCHIVE_CREATE
CMAKE_<LANG>_ARCHIVE_FINISH
CMAKE_<LANG>_BYTE_ORDER
CMAKE_<LANG>_COMPILER
CMAKE_<LANG>_COMPILER_ABI
CMAKE_<LANG>_COMPILER_ARCHITECTURE_ID
CMAKE_<LANG>_COMPILER_EXTERNAL_TOOLCHAIN
CMAKE_<LANG>_COMPILER_ID
CMAKE_<LANG>_COMPILER_LOADED
CMAKE_<LANG>_COMPILER_PREDEFINES_COMMAND
CMAKE_<LANG>_COMPILER_TARGET
CMAKE_<LANG>_COMPILER_VERSION
CMAKE_<LANG>_COMPILER_VERSION_INTERNAL
CMAKE_<LANG>_COMPILE_OBJECT
CMAKE_<LANG>_CREATE_SHARED_LIBRARY
CMAKE_<LANG>_CREATE_SHARED_MODULE
CMAKE_<LANG>_CREATE_STATIC_LIBRARY
CMAKE_<LANG>_FLAGS
CMAKE_<LANG>_FLAGS_<CONFIG>
CMAKE_<LANG>_FLAGS_<CONFIG>_INIT
CMAKE_<LANG>_FLAGS_DEBUG
CMAKE_<LANG>_FLAGS_DEBUG_INIT
CMAKE_<LANG>_FLAGS_INIT
CMAKE_<LANG>_FLAGS_MINSIZEREL
CMAKE_<LANG>_FLAGS_MINSIZEREL_INIT
CMAKE_<LANG>_FLAGS_RELEASE
CMAKE_<LANG>_FLAGS_RELEASE_INIT
CMAKE_<LANG>_FLAGS_RELWITHDEBINFO
CMAKE_<LANG>_FLAGS_RELWITHDEBINFO_INIT
CMAKE_<LANG>_IGNORE_EXTENSIONS
CMAKE_<LANG>_IMPLICIT_INCLUDE_DIRECTORIES
CMAKE_<LANG>_IMPLICIT_LINK_DIRECTORIES
CMAKE_<LANG>_IMPLICIT_LINK_FRAMEWORK_DIRECTORIES
CMAKE_<LANG>_IMPLICIT_LINK_LIBRARIES
CMAKE_<LANG>_LIBRARY_ARCHITECTURE
CMAKE_<LANG>_LINKER_PREFERENCE
CMAKE_<LANG>_LINKER_PREFERENCE_PROPAGATES
CMAKE_<LANG>_LINKER_WRAPPER_FLAG
CMAKE_<LANG>_LINKER_WRAPPER_FLAG_SEP
CMAKE_<LANG>_LINK_EXECUTABLE
CMAKE_<LANG>_LINK_LIBRARY_FILE_FLAG
CMAKE_<LANG>_LINK_LIBRARY_FLAG
CMAKE_<LANG>_LINK_LIBRARY_SUFFIX
CMAKE_<LANG>_OUTPUT_EXTENSION
CMAKE_<LANG>_PLATFORM_ID
CMAKE_<LANG>_SIMULATE_ID
CMAKE_<LANG>_SIMULATE_VERSION
CMAKE_<LANG>_SIZEOF_DATA_PTR
CMAKE_<LANG>_SOURCE_FILE_EXTENSIONS
CMAKE_<LANG>_STANDARD_INCLUDE_DIRECTORIES
CMAKE_<LANG>_STANDARD_LIBRARIES
CMAKE_Swift_LANGUAGE_VERSION
CMAKE_Swift_MODULE_DIRECTORY
CMAKE_Swift_NUM_THREADS
CMAKE_USER_MAKE_RULES_OVERRIDE_<LANG>

Variables - CTest

CTEST_BINARY_DIRECTORY
CTEST_BUILD_COMMAND
CTEST_BUILD_NAME
CTEST_BZR_COMMAND
CTEST_BZR_UPDATE_OPTIONS
CTEST_CHANGE_ID
CTEST_CHECKOUT_COMMAND
CTEST_CONFIGURATION_TYPE
CTEST_CONFIGURE_COMMAND
CTEST_COVERAGE_COMMAND
CTEST_COVERAGE_EXTRA_FLAGS
CTEST_CURL_OPTIONS
CTEST_CUSTOM_COVERAGE_EXCLUDE
CTEST_CUSTOM_ERROR_EXCEPTION
CTEST_CUSTOM_ERROR_MATCH
CTEST_CUSTOM_ERROR_POST_CONTEXT
CTEST_CUSTOM_ERROR_PRE_CONTEXT
CTEST_CUSTOM_MAXIMUM_FAILED_TEST_OUTPUT_SIZE
CTEST_CUSTOM_MAXIMUM_NUMBER_OF_ERRORS
CTEST_CUSTOM_MAXIMUM_NUMBER_OF_WARNINGS
CTEST_CUSTOM_MAXIMUM_PASSED_TEST_OUTPUT_SIZE
CTEST_CUSTOM_MEMCHECK_IGNORE
CTEST_CUSTOM_POST_MEMCHECK
CTEST_CUSTOM_POST_TEST
CTEST_CUSTOM_PRE_MEMCHECK
CTEST_CUSTOM_PRE_TEST
CTEST_CUSTOM_TESTS_IGNORE
CTEST_CUSTOM_WARNING_EXCEPTION
CTEST_CUSTOM_WARNING_MATCH
CTEST_CVS_CHECKOUT
CTEST_CVS_COMMAND
CTEST_CVS_UPDATE_OPTIONS
CTEST_DROP_LOCATION
CTEST_DROP_METHOD
CTEST_DROP_SITE
CTEST_DROP_SITE_CDASH
CTEST_DROP_SITE_PASSWORD
CTEST_DROP_SITE_USER
CTEST_EXTRA_COVERAGE_GLOB
CTEST_GIT_COMMAND
CTEST_GIT_INIT_SUBMODULES
CTEST_GIT_UPDATE_CUSTOM
CTEST_GIT_UPDATE_OPTIONS
CTEST_HG_COMMAND
CTEST_HG_UPDATE_OPTIONS
CTEST_LABELS_FOR_SUBPROJECTS
CTEST_MEMORYCHECK_COMMAND
CTEST_MEMORYCHECK_COMMAND_OPTIONS
CTEST_MEMORYCHECK_SANITIZER_OPTIONS
CTEST_MEMORYCHECK_SUPPRESSIONS_FILE
CTEST_MEMORYCHECK_TYPE
CTEST_NIGHTLY_START_TIME
CTEST_P4_CLIENT
CTEST_P4_COMMAND
CTEST_P4_OPTIONS
CTEST_P4_UPDATE_OPTIONS
CTEST_RESOURCE_SPEC_FILE
CTEST_RUN_CURRENT_SCRIPT
CTEST_SCP_COMMAND
CTEST_SITE
CTEST_SOURCE_DIRECTORY
CTEST_SUBMIT_URL
CTEST_SVN_COMMAND
CTEST_SVN_OPTIONS
CTEST_SVN_UPDATE_OPTIONS
CTEST_TEST_LOAD
CTEST_TEST_TIMEOUT
CTEST_TRIGGER_SITE
CTEST_UPDATE_COMMAND
CTEST_UPDATE_OPTIONS
CTEST_UPDATE_VERSION_ONLY
CTEST_UPDATE_VERSION_OVERRIDE
CTEST_USE_LAUNCHERS

Variables - CPack

CPACK_ABSOLUTE_DESTINATION_FILES
CPACK_COMPONENT_INCLUDE_TOPLEVEL_DIRECTORY
CPACK_ERROR_ON_ABSOLUTE_INSTALL_DESTINATION
CPACK_INCLUDE_TOPLEVEL_DIRECTORY
CPACK_INSTALL_DEFAULT_DIRECTORY_PERMISSIONS
CPACK_PACKAGING_INSTALL_PREFIX
CPACK_SET_DESTDIR
CPACK_WARN_ON_ABSOLUTE_INSTALL_DESTINATION

C++ Features

cxx_std_98
cxx_std_11
cxx_std_14
cxx_std_17
cxx_std_20
cxx_aggregate_default_initializers
cxx_alias_templates
cxx_alignas
cxx_alignof
cxx_attributes
cxx_attribute_deprecated
cxx_auto_type
cxx_binary_literals
cxx_constexpr
cxx_contextual_conversions
cxx_decltype_incomplete_return_types
cxx_decltype
cxx_decltype_auto
cxx_default_function_template_args
cxx_defaulted_functions
cxx_defaulted_move_initializers
cxx_delegating_constructors
cxx_deleted_functions
cxx_digit_separators
cxx_enum_forward_declarations
cxx_explicit_conversions
cxx_extended_friend_declarations
cxx_extern_templates
cxx_final
cxx_func_identifier
cxx_generalized_initializers
cxx_generic_lambdas
cxx_inheriting_constructors
cxx_inline_namespaces
cxx_lambdas
cxx_lambda_init_captures
cxx_local_type_template_args
cxx_long_long_type
cxx_noexcept
cxx_nonstatic_member_init
cxx_nullptr
cxx_override
cxx_range_for
cxx_raw_string_literals
cxx_reference_qualified_functions
cxx_relaxed_constexpr
cxx_return_type_deduction
cxx_right_angle_brackets
cxx_rvalue_references
cxx_sizeof_member
cxx_static_assert
cxx_strong_enums
cxx_thread_local
cxx_trailing_return_types
cxx_unicode_literals
cxx_uniform_initialization
cxx_unrestricted_unions
cxx_user_literals
cxx_variable_templates
cxx_variadic_macros
cxx_variadic_templates
cxx_template_template_parameters

Modules

```
AddFileDependencies
AndroidTestUtilities
BundleUtilities
CheckCCompilerFlag
CheckCompilerFlag
CheckCSourceCompiles
CheckCSourceRuns
CheckCXXCompilerFlag
CheckCXXSourceCompiles
CheckCXXSourceRuns
CheckCXXSymbolExists
CheckFortranCompilerFlag
CheckFortranFunctionExists
CheckFortranSourceCompiles
CheckFortranSourceRuns
CheckFunctionExists
CheckIPOSupported
CheckIncludeFileCXX
CheckIncludeFile
CheckIncludeFiles
CheckLanguage
CheckLibraryExists
CheckLinkerFlag
CheckOBJCCompilerFlag
CheckOBJCSourceCompiles
CheckOBJCSourceRuns
CheckOBJCXXCompilerFlag
CheckOBJCXXSourceCompiles
CheckOBJCXXSourceRuns
CheckPIESupported
CheckPrototypeDefinition
CheckSourceCompiles
CheckSourceRuns
CheckStructHasMember
CheckSymbolExists
CheckTypeSize
CheckVariableExists
CMakeAddFortranSubdirectory
CMakeBackwardCompatibilityCXX
CMakeDependentOption
CMakeDetermineVSServicePack
CMakeExpandImportedTargets
CMakeFindDependencyMacro
CMakeFindFrameworks
CMakeFindPackageMode
CMakeForceCompiler
CMakeGraphVizOptions
CMakePackageConfigHelpers
CMakeParseArguments
CMakePrintHelpers
CMakePrintSystemInformation
CMakePushCheckState
CMakeVerifyManifest
CPackComponent
CPackIFW
CPackIFWConfigureFile
CPack
CSharpUtilities
CTest
CTestCoverageCollectGCOV
CTestScriptMode
CTestUseLaunchers
Dart
DeployQt4
Documentation
ExternalData
ExternalProject
FeatureSummary
FetchContent
FindALSA
FindArmadillo
FindASPELL
FindAVIFile
FindBISON
FindBLAS
FindBacktrace
FindBoost
FindBullet
FindBZip2
FindCABLE
FindCoin3D
FindCUDA
FindCUDAToolkit
FindCups
FindCURL
FindCurses
FindCVS
FindCxxTest
FindCygwin
FindDart
FindDCMTK
FindDevIL
FindDoxygen
FindEnvModules
FindEXPAT
FindFLEX
FindFLTK2
FindFontconfig
FindFLTK
FindFreetype
FindGCCXML
FindGDAL
FindGettext
FindGIF
FindGit
FindGLEW
FindGLUT
FindGnuplot
FindGnuTLS
FindGSL
FindGTest
FindGTK2
FindGTK
FindHDF5
FindHg
FindHSPELL
FindHTMLHelp
FindIce
FindIcotool
FindICU
FindImageMagick
FindIconv
FindIntl
FindITK
FindJasper
FindJava
FindJNI
FindJPEG
FindKDE3
FindKDE4
FindLAPACK
FindLATEX
FindLibArchive
FindLibinput
FindLibLZMA
FindLibXml2
FindLibXslt
FindLTTngUST
FindLua50
FindLua51
FindLua
FindMatlab
FindMFC
FindMotif
FindMPEG2
FindMPEG
FindMPI
FindODBC
FindOpenACC
FindOpenAL
FindOpenCL
FindOpenGL
FindOpenMP
FindOpenSceneGraph
FindOpenSSL
FindOpenThreads
FindosgAnimation
FindosgDB
Findosg_functions
FindosgFX
FindosgGA
FindosgIntrospection
FindosgManipulator
FindosgParticle
FindosgPresentation
FindosgProducer
FindosgQt
Findosg
FindosgShadow
FindosgSim
FindosgTerrain
FindosgText
FindosgUtil
FindosgViewer
FindosgVolume
FindosgWidget
FindPackageHandleStandardArgs
FindPackageMessage
FindPatch
FindPerlLibs
FindPerl
FindPHP4
FindPhysFS
FindPike
FindPkgConfig
FindPNG
FindPostgreSQL
FindProducer
FindProtobuf
FindPython
FindPython2
FindPython3
FindPythonInterp
FindPythonLibs
FindQt3
FindQt4
FindQt
FindQuickTime
FindRTI
FindRuby
FindSDL_image
FindSDL_mixer
FindSDL_net
FindSDL
FindSDL_sound
FindSDL_ttf
FindSelfPackers
FindSQLite3
FindSquish
FindSubversion
FindSWIG
FindTCL
FindTclsh
FindTclStub
FindThreads
FindTIFF
FindUnixCommands
FindVTK
FindVulkan
FindWget
FindWish
FindwxWidgets
FindwxWindows
FindXCTest
FindXalanC
FindXercesC
FindX11
FindXMLRPC
FindZLIB
FortranCInterface
GenerateExportHeader
GetPrerequisites
GNUInstallDirs
GoogleTest
InstallRequiredSystemLibraries
MacroAddFileDependencies
ProcessorCount
SelectLibraryConfigurations
SquishTestScript
TestBigEndian
TestCXXAcceptsFlag
TestForANSIForScope
TestForANSIStreamHeaders
TestForSSTREAM
TestForSTDNamespace
UseEcos
UseJavaClassFilelist
UseJava
UseJavaSymlinks
UsePkgConfig
UseSWIG
UsewxWidgets
Use_wxWindows
WriteBasicConfigVersionFile
WriteCompilerDetectionHeader
```

CMake CLI - General *(incomplete list)*

```
--help
--version
--help-full
--help-manual <man> [<f>]
--help-command <cmd> [<f>]
--help-module <mod> [<f>]
--help-policy <cmp> [<f>]
--help-property <prop> [<f>]
--help-variable <var> [<f>]
--system-information [file]
--log-level <level>
--log-context
--debug-trycompile
--debug-output
--debug-find
--trace
--trace-expand
--trace-format=<fmt>
--trace-source=<file>
--warn-unitialized
--graphviz=<file>
--find-package <options>...
--profiling-output=<path>
--preset <preset>
-S <path-to-source>
-B <path-to-build>
-D <var>[=<value>]
-D <var>:<type>=<value>
-U <glob_expr>
-G <generator-name>
-T <toolset-spec>
-A <platform-name>
-L[A|H]
```

CMake CLI - Build Tool *(incomplete list)*

```
--build <dir>
--config <cfg>
--target <tgt>
--preset <preset>
--use-stderr
--clean-first
-j [<jobs>] | --parallel [<jobs>]
```

CMake CLI - Scripting Tool

```
-P <file>
```

CMake CLI - Command Tool

```
-E <command> [<options>...]

capabilities
cat <files>...
chdir <dir> <cmd> [<arg>...]
compare_files <file1> <file2>
copy <file>... <destination>
copy_directory <dir>... <dest>
copy_if_different <file>... <dest>
create_symlink <old> <new>
create_hardlink <old> <new>
echo [<string>...]
echo_append [<string>...]
env [--unset=NAME]... [NAME=VALUE]... COMMAND [ARG]...
environment
false
make_directory <dir>...
md5sum <file>...
sha1sum <file>...
sha224sum <file>...
sha256sum <file>...
sha384sum <file>...
sha512sum <file>...
remove [-f] <file>...
remove_directory <dir>
rename <oldname> <newname>
rm [-rRf] <file> <dir>...
server
sleep
tar [cxt][vf][zjJ] file.tar [<options>...] [--] [<file>...]
  --files-from=<file>
  --mtime=<date>
  --format=<7zip|gnutar|pax|paxr|zip>
time <command> [<args>...]
touch <file>
touch_nocreate <file>
true
```

CMake CLI - Windows Tools

```
delete_regv <key>
env_vs8_wince <sdkname>
env_vs9_wince <sdkname>
write_regv <key> <value>
```

Modern CMake - CMake 3.20

https://cmake.org/cmake/help/v3.20/

Simple Executable

```
cmake_minimum_required(VERSION 3.20)
project(simple_executable)
add_executable(${PROJECT_NAME})
target_sources(${PROJECT_NAME} PRIVATE main.cpp)
```

Static Library

```
cmake_minimum_required(VERSION 3.20)
project(static_library)
add_library(${PROJECT_NAME} STATIC)
target_sources(${PROJECT_NAME}
  PRIVATE
    src/echo.cpp
    include/simple/echo.hpp)
target_include_directories(${PROJECT_NAME}
  PUBLIC
  $<BUILD_INTERFACE:${CMAKE_CURRENT_SOURCE_DIR}/include>)
```

Shared Library

```
cmake_minimum_required(VERSION 3.20)
project(shared_library)
add_library(${PROJECT_NAME} SHARED)
target_sources(${PROJECT_NAME}
  PUBLIC include/simple/echo.hpp
  PRIVATE src/echo.cpp src/echo_private.hpp)
target_include_directories(${PROJECT_NAME}
  PUBLIC
  $<BUILD_INTERFACE:${CMAKE_CURRENT_SOURCE_DIR}/include>)
```

Object Library

```
cmake_minimum_required(VERSION 3.20)
project(object_library)

add_library(${PROJECT_NAME}-obj OBJECT)
target_sources(${PROJECT_NAME}-obj
  PRIVATE src/echo.cpp include/simple/echo.hpp)
target_include_directories(${PROJECT_NAME}-obj
  PUBLIC
  $<BUILD_INTERFACE:${CMAKE_CURRENT_SOURCE_DIR}/include>)

set_property(TARGET ${PROJECT_NAME}-obj
  PROPERTY POSITION_INDEPENDENT_CODE 1)

add_library(${PROJECT_NAME}-static STATIC
  $<TARGET_OBJECTS:${PROJECT_NAME}-obj>)
add_library(${PROJECT_NAME}-shared SHARED
  $<TARGET_OBJECTS:${PROJECT_NAME}-obj>)
```

Header Only Library

```
cmake_minimum_required(VERSION 3.20)
project(headeronly_library)

add_library(${PROJECT_NAME} INTERFACE)
target_include_directories(${PROJECT_NAME}
  INTERFACE
  $<BUILD_INTERFACE:${CMAKE_CURRENT_SOURCE_DIR}/library/
        ↪ include>)

add_executable(example)
target_sources(example
  PRIVATE src/main.cpp)
target_link_libraries(example
  PRIVATE ${PROJECT_NAME})
```

Find Packages

```
cmake_minimum_required(VERSION 3.20)
project(executable_with_libs)
find_package(ZLIB REQUIRED)
find_package(Boost 1.58 COMPONENTS program_options REQUIRED
       ↪ )
add_executable(${PROJECT_NAME})
target_sources(${PROJECT_NAME}
  PRIVATE src/main.cpp)
target_link_libraries(${PROJECT_NAME}
  PRIVATE ZLIB::ZLIB Boost::program_options)
```

Add CTest

```
cmake_minimum_required(VERSION 3.20)
project(library_with_test)

add_library(${PROJECT_NAME} STATIC)
target_sources(${PROJECT_NAME}
  PUBLIC library/include/library/echo.hpp
  PRIVATE library/src/echo.cpp)
target_include_directories(${PROJECT_NAME}
  PUBLIC
  $<BUILD_INTERFACE:${CMAKE_CURRENT_SOURCE_DIR}/library/include>)

add_executable(testrunner)
target_sources(testrunner
  PRIVATE testrunner/src/main.cpp)
target_link_libraries(testrunner
  PRIVATE ${PROJECT_NAME})

enable_testing()
add_test(
  NAME ${PROJECT_NAME}-test-1
  COMMAND testrunner "Hello World")
add_test(
  NAME ${PROJECT_NAME}-test-2
  COMMAND testrunner "Hallo Welt")
```

Import Library

```
cmake_minimum_required(VERSION 3.20)
project(import_library)

find_path(HO_LIBRARY_PATH
  NAMES library
  PATHS ${CMAKE_CURRENT_SOURCE_DIR}/../headeronly_library/library/include
  NO_DEFAULT_PATH
)

if(HO_LIBRARY_PATH)
  add_library(headeronly::headeronly INTERFACE IMPORTED)
  set_target_properties(headeronly::headeronly
    PROPERTIES
    INTERFACE_INCLUDE_DIRECTORIES ${HO_LIBRARY_PATH})
endif()

add_executable(example)
target_sources(example
  PRIVATE src/main.cpp)
target_link_libraries(example
  PRIVATE headeronly::headeronly)
```

Module: ExternalProject

```
cmake_minimum_required(VERSION 3.20)
project(externalproject)

find_package(Boost 1.58 COMPONENTS program_options REQUIRED)

file(MAKE_DIRECTORY ${CMAKE_CURRENT_BINARY_DIR}/local/include)

include(ExternalProject)
ExternalProject_Add(extern-zlib
  PREFIX ${CMAKE_CURRENT_BINARY_DIR}/zlib
  URL
    file://${CMAKE_CURRENT_SOURCE_DIR}/extern/zlib-1.2.8.tar.gz
  URL_HASH
    MD5=44d667c142d7cda120332623eab69f40
  CMAKE_ARGS
    -DCMAKE_C_COMPILER=${CMAKE_C_COMPILER}
    -DCMAKE_BUILD_TYPE=Release
    -DCMAKE_INSTALL_PREFIX=${CMAKE_CURRENT_BINARY_DIR}/local
  INSTALL_DIR ${CMAKE_CURRENT_BINARY_DIR}/local)

add_library(zlib::zlib STATIC IMPORTED)
add_dependencies(zlib::zlib extern-zlib)
set_target_properties(zlib::zlib PROPERTIES
  IMPORTED_LOCATION
    ${CMAKE_CURRENT_BINARY_DIR}/local/lib/${CMAKE_STATIC_LIBRARY_PREFIX}z${
          ↪ CMAKE_STATIC_LIBRARY_SUFFIX}
  INTERFACE_INCLUDE_DIRECTORIES
    ${CMAKE_CURRENT_BINARY_DIR}/local/include)

add_executable(${PROJECT_NAME})
target_sources(${PROJECT_NAME}
  PRIVATE src/main.cpp)
target_link_libraries(${PROJECT_NAME}
  PRIVATE zlib::zlib Boost::program_options)
```

Qt Support

```
cmake_minimum_required(VERSION 3.20)
project(app_qt5)

if(NOT EXISTS $ENV{QTHOME})
  message(FATAL_ERROR "directory QTHOME does not exist")
endif()

set(CMAKE_AUTOMOC ON)
set(CMAKE_AUTOUIC ON)
set(CMAKE_AUTORCC ON)

find_package(Qt5Core REQUIRED)
find_package(Qt5Widgets REQUIRED)

add_executable(${PROJECT_NAME})
target_sources(${PROJECT_NAME}
  PRIVATE
    src/main.cpp
    src/demo_application.cpp)
target_link_libraries(${PROJECT_NAME}
  PRIVATE Qt5::Core Qt5::Widgets)
```

CPack

```
cmake_minimum_required(VERSION 3.20)
project(package-demo VERSION 1.2.3)

add_library(${PROJECT_NAME} STATIC)
target_sources(${PROJECT_NAME}
  PRIVATE
    src/echo.cpp
    include/simple/echo.hpp)
target_include_directories(${PROJECT_NAME}
  PUBLIC
  $<BUILD_INTERFACE:${CMAKE_CURRENT_SOURCE_DIR}/include>
  $<INSTALL_INTERFACE:include>)

install(DIRECTORY ${CMAKE_CURRENT_SOURCE_DIR}/include/
  DESTINATION include)

install(TARGETS ${PROJECT_NAME}
  RUNTIME DESTINATION bin
  LIBRARY DESTINATION lib
  ARCHIVE DESTINATION lib
  INCLUDES DESTINATION include)

set(CPACK_PACKAGE_VERSION_MAJOR ${PROJECT_VERSION_MAJOR})
set(CPACK_PACKAGE_VERSION_MINOR ${PROJECT_VERSION_MINOR})
set(CPACK_PACKAGE_VERSION_PATCH ${PROJECT_VERSION_PATCH})
set(CPACK_PACKAGE_VERSION ${PROJECT_VERSION})
set(CPACK_PACKAGE_NAME "${PROJECT_NAME}")
set(CPACK_PACKAGE_VENDOR "Mario Konrad")
set(CPACK_PACKAGE_CONTACT
  "Mario Konrad <mario.konrad@gmx.net>")
set(CPACK_PACAKGE_DESCRIPTION "Package demo.")
set(CPACK_PACKAGE_DESCRIPTION_SUMMARY "Library")
set(CPACK_MONOLITHIC_INSTALL 1)
set(CPACK_STRIP_FILES true)

include(CPack)
```

Build:

```
cmake -B builddir source-dir
cmake --build builddir
cd builddir
cpack -G ZIP
```

Libray Installation

```
cmake_minimum_required(VERSION 3.20)
project(lib_install VERSION 1.2.3)

add_library(${PROJECT_NAME} SHARED)
target_sources(${PROJECT_NAME}
  PRIVATE
    include/simple/echo.hpp
    src/echo.cpp
    src/echo_private.hpp)
target_include_directories(${PROJECT_NAME}
  PUBLIC
    $<BUILD_INTERFACE:${CMAKE_CURRENT_SOURCE_DIR}/include>
    $<INSTALL_INTERFACE:include>)

set_target_properties(${PROJECT_NAME}
  PROPERTIES
    VERSION ${PROJECT_VERSION}
    SOVERSION ${PROJECT_VERSION})

set(targets_export_name ${PROJECT_NAME}-targets)
set(include_install_dir include)

install(DIRECTORY ${CMAKE_CURRENT_SOURCE_DIR}/include/
  DESTINATION ${include_install_dir})
install(TARGETS ${PROJECT_NAME}
  EXPORT ${targets_export_name}
  RUNTIME DESTINATION bin
  LIBRARY DESTINATION lib
  ARCHIVE DESTINATION lib
  INCLUDES DESTINATION ${include_install_dir})
install(EXPORT ${targets_export_name}
  DESTINATION lib/cmake/${PROJECT_NAME})
file(WRITE ${CMAKE_CURRENT_BINARY_DIR}/config.cmake.in [==[
  include("${CMAKE_CURRENT_LIST_DIR}/@targets_export_name@.cmake")
]==])
include(CMakePackageConfigHelpers)
configure_package_config_file(
  ${CMAKE_CURRENT_BINARY_DIR}/config.cmake.in
  ${CMAKE_CURRENT_BINARY_DIR}/${PROJECT_NAME}-config.cmake
  INSTALL_DESTINATION lib/cmake/${PROJECT_NAME}
  PATH_VARS include_install_dir)
write_basic_package_version_file(
  ${CMAKE_CURRENT_BINARY_DIR}/${PROJECT_NAME}-config-version.cmake
  VERSION "${PROJECT_VERSION}" COMPATIBILITY SameMajorVersion)
install(
  FILES
    ${CMAKE_CURRENT_BINARY_DIR}/${PROJECT_NAME}-config.cmake
    ${CMAKE_CURRENT_BINARY_DIR}/${PROJECT_NAME}-config-version.cmake
  DESTINATION
    lib/cmake/${PROJECT_NAME})
```

Build:

```
cmake -B builddir -DCMAKE_INSTALL_PREFIX=install-dir source-dir
cmake --build builddir
cmake --install builddir
```

https://github.com/mariokonrad/cmake-cheatsheet