google / **googletest**

Branch: **master** ▾ **googletest** / googletest / docs / **Pkgconfig.md** Find file Copy path

SoapGentoo Add documentation for pkg-config 9cacce4 on 14 Aug

1 contributor

147 lines (101 sloc) 3.9 KB

Using GoogleTest from various build systems

GoogleTest comes with pkg-config files that can be used to determine all necessary flags for compiling and linking to GoogleTest (and GoogleMock). Pkg-config is a standardised plain-text format containing

- the includedir (-I) path
- necessary macro (-D) definitions
- further required flags (-pthread)
- the library (-L) path
- the library (-l) to link to

All current build systems support pkg-config in one way or another. For all examples here we assume you want to compile the sample `samples/sample3_unittest.cc`.

CMake

Using `pkg-config` in CMake is fairly easy:

```
cmake_minimum_required(VERSION 3.0)

cmake_policy(SET CMP0048 NEW)
project(my_gtest_pkgconfig VERSION 0.0.1 LANGUAGES CXX)

find_package(PkgConfig)
pkg_search_module(GTEST REQUIRED gtest_main)

add_executable(testapp samples/sample3_unittest.cc)
target_link_libraries(testapp ${GTEST_LDFLAGS})
target_compile_options(testapp PUBLIC ${GTEST_CFLAGS})

include(CTest)
add_test(first_and_only_test testapp)
```

It is generally recommended that you use `target_compile_options + _CFLAGS` over `target_include_directories + _INCLUDE_DIRS` as the former includes not just -I flags (GoogleTest might require a macro indicating to internal headers that all libraries have been compiled with threading enabled. In addition, GoogleTest might also require `-pthread` in the compiling step, and as such splitting the pkg-config `Cflags` variable into include dirs and macros for `target_compile_definitions()` might still miss this). The same recommendation goes for using `_LDFLAGS` over the more commonplace `_LIBRARIES`, which happens to discard `-L` flags and `-pthread`.

Autotools

Finding GoogleTest in Autoconf and using it from Automake is also fairly easy:

In your `configure.ac`:

```
AC_PREREQ([2.69])
AC_INIT([my_gtest_pkgconfig], [0.0.1])
AC_CONFIG_SRCDIR([samples/sample3_unittest.cc])
AC_PROG_CXX

PKG_CHECK_MODULES([GTEST], [gtest_main])

AM_INIT_AUTOMAKE([foreign subdir-objects])
AC_CONFIG_FILES([Makefile])
AC_OUTPUT
```

and in your `Makefile.am` :

```
check_PROGRAMS = testapp
TESTS = $(check_PROGRAMS)

testapp_SOURCES = samples/sample3_unittest.cc
testapp_CXXFLAGS = $(GTEST_CFLAGS)
testapp_LDADD = $(GTEST_LIBS)
```

Meson

Meson natively uses pkgconfig to query dependencies:

```
project('my_gtest_pkgconfig', 'cpp', version : '0.0.1')

gtest_dep = dependency('gtest_main')

testapp = executable(
  'testapp',
  files(['samples/sample3_unittest.cc']),
  dependencies : gtest_dep,
  install : false)

test('first_and_only_test', testapp)
```

Plain Makefiles

Since `pkg-config` is a small Unix command-line utility, it can be used in handwritten `Makefile` s too:

```
GTEST_CFLAGS = `pkg-config --cflags gtest_main`
GTEST_LIBS = `pkg-config --libs gtest_main`

.PHONY: tests all

tests: all
	./testapp

all: testapp

testapp: testapp.o
	$(CXX) $(CXXFLAGS) $(LDFLAGS) $< -o $@ $(GTEST_LIBS)

testapp.o: samples/sample3_unittest.cc
	$(CXX) $(CPPFLAGS) $(CXXFLAGS) $< -c -o $@ $(GTEST_CFLAGS)
```

Help! pkg-config can't find GoogleTest!

Let's say you have a `CMakeLists.txt` along the lines of the one in this tutorial and you try to run `cmake` . It is very possible that you get a failure along the lines of:

```
-- Checking for one of the modules 'gtest_main'
CMake Error at /usr/share/cmake/Modules/FindPkgConfig.cmake:640 (message):
  None of the required 'gtest_main' found
```

These failures are common if you installed GoogleTest yourself and have not sourced it from a distro or other package manager. If so, you need to tell pkg-config where it can find the `.pc` files containing the information. Say you installed GoogleTest to `/usr/local`, then it might be that the `.pc` files are installed under `/usr/local/lib64/pkgconfig`. If you set

```
export PKG_CONFIG_PATH=/usr/local/lib64/pkgconfig
```

pkg-config will also try to look in `PKG_CONFIG_PATH` to find `gtest_main.pc`.