

# Building projects with CMake

Victor Eijkhout

Fall 2023



CMake is a portable build system that is becoming a *de facto* standard for C++ package management.  
Also usable with C and Fortran.



1 Help! This software uses CMake!

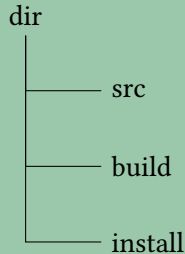
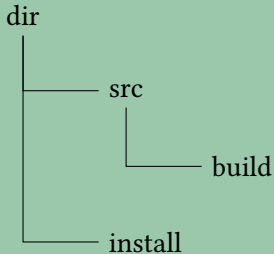
2 Help! I want to write CMake myself!

3 Help! I want people to use my CMake package!



# Help! This software uses CMake!





- In-source build: pretty common
- Out-of-source build: cleaner because never touches the source tree
- Some people skip the install step, and use everything from the build directory.



You have just installed a CMake-based library.  
Now you need it in your own code, or in another library.  
How easy can we make that?



- Many packages come with a `package.pc` file
- Add that location to `PKG_CONFIG_PATH`
- The package can now be found by other CMake-based packages.



# Help! I want to write CMake myself!





Build an executable from a single source file:

```
cmake_minimum_required( VERSION 3.12 )  
project( singleprogram VERSION 1.0 )  
  
add_executable( program program.cxx )  
install( TARGETS program DESTINATION . )
```



## Header-only:

```
cmake_minimum_required( VERSION 3.12 )
project( eigentest )

find_package( PkgConfig REQUIRED )
pkg_check_modules( EIGEN REQUIRED eigen3 )

add_executable( eigentest eigentest.cxx )
target_include_directories(
    myprogram PUBLIC
    ${EIGEN_INCLUDE_DIRS})
```



# Help! I want people to use my CMake package!

