CMake

Cross-platform, open-source build system.
CMake Overview

- Build system (meta-make)
- Cross-platform build description
- Platform-specific build files
- Multiple build trees

Source Files

CMake project

Native build environments:
- UNIX makefiles
- VS project
- Xcode project
Build Process

build
src

module1

CMakeLists.txt
File1.c

module2

CMakeLists.txt
File2.c
...

CMakeLists.txt

build

cmake
-cmake
-gui

Cache

User

*.exe, *.dll, *.lib, *.a, *.so

*.vcproj
Makefile
...

*.obj
*.o
CMake Process

**Configure:**
- Environment vars – CMakeCache.txt
- Root CMakeLists.txt parsed and executed
- Additional CMakeLists.txt – add_subdirectory
- In-memory representation – makefiles, targets
- CMakeCache.txt saved

**Generate:**
- CMake targets converted
- As generic as possible
Development Environment

- CMake
- CTest
- CPack
- CDash
**Directory structure**

```
/HelloWorld
 /build
 /src
   program.c
   CMakeLists.txt
```

**CMakeLists.txt**

```cmake
PROJECT(HelloWorld)
ADD_EXECUTABLE(hello program.cpp)

# best practices
SET(hello_src program.cpp ...)
ADD_EXECUTABLE(hello ${hello_src})
# ADD_LIBRARY(lib STATIC ${lib_src})
```

**SHELL**

```
$ cd HelloWorld/build
$ cmake .. /src
$ make
```
Demo – Hello World

- How to use (cmake-gui)
- Libraries
- Dependencies
- Tests
CMake Features

COMMAND_name(args ...)

# Basic building block - VARIABLE
SET(src_files ${src_files} header.hpp code.cpp)

# Flow control
IF(CYGWIN)
    add_definitions(-DUSE_MY_FAVOURITE_COMPILER)
ENDIF()

# External dependency (use FindBoost.cmake file)
FIND_PACKAGE(Boost REQUIRED)
IF (Boost_FOUND)
    include_directories(${Boost_INCLUDE_DIRS})
ENDIF()

# Configuration files
CONFIGURE_FILE(conf.in conf.out)
Demo – Real project – SWP LPR

- License plate recognition – Linux, Windows
- Output: LPR service, Training tool, Dataset tool
- LIBS: OpenCV, Bootstrap, CURL
- Platform/Compiler specific stuff – NOT, AND ...
- Version header – customer specific build
- Debug environment
- Nested directories – include all
- Each build environment has its specifics
- Generate documentation
- Unit testing
Summary

Open source

CMakeLists.txt – difficult to edit

FIND_PACKAGE – not so smart as expected (at least on Windows)

Custom platform/environment generator

Separate x64 projects

Only for multi-platform projects
Further information

Start here: http://www.cmake.org/Wiki/CMake

Tutorial: http://www.cmake.org/cmake-tutorial

Presentation based on: http://www.elpauer.org/stuff/learning_cmake.pdf, ...

VS plugin: https://cmaketoools.codeplex.com/