Developer Experience Now Powered up by Cmake

Building a complicated project on UNIX like system with pure Makefile is always a pain. Same story on Windows if you manually configure all those definitions, libraries, include directories and so on.

Now we are very pleased to announce that, starting from the latest version (v1.0.109), developers can use CMake to easily build your project with [AWS SDK for C++](https://aws.amazon.com/sdk-for-cpp/), as well as uninstall the installed files.

A simple example on Linux using Cmake:

```

cmake\_minimum\_required(VERSION 2.8)

project(myTestS3Encryption)

find\_package(AWSSDK) # here for simplicity we don’t check AWSSDK\_FOUND

set(SERVICE s3-encryption)

AWSSDK\_DETERMINE\_LIBS\_TO\_LINK(SERVICE OUTPUT)

find\_package(PkgConfig) # together with PkgConfig makes life super easier

pkg\_check\_modules(DEPS REQUIRED ${OUTPUT})

link\_directories("${AWSSDK\_LIB\_DIR}")

add\_executable(myTestS3Encryption src/main.cpp)

target\_link\_libraries(myTestS3Encryption ${DEPS\_LIBRARIES})

target\_include\_directories(myTestS3Encryption PUBLIC ${DEPS\_INCLUDE\_DIRS})

target\_compile\_options(myTestS3Encryption PUBLIC ${DEPS\_CFLAGS\_OTHER})

```

Previous to this version, there exist cmake scripts for each individual SDKs, but the functionality is nothing but tell you the existence of a SDK. With the latest version, after running “sudo make install”, a new directory called AWSSDK will be installed. The default install path is:

On UNIX like system: “/usr/local/lib/cmake”,

On Windows: “C:/Program Files/aws-cpp-sdk-all/lib/cmake”

Several cmake scripts are included in this directory, the most important one is AWSSDKConfig.cmake, whose name is complied with the Cmake find\_package() config mode name format.

After calling find\_package(AWSSDK), several useful variables and macros in this script could be called by developers.

Variables: AWSSDK\_LIB\_DIR

AWSSDK\_BIN\_DIR

AWSSDK\_INCLUDE\_DIR