Bug Report

Release 1.0.0

mccakit

May 22, 2025

CONTENTS

I wanted to cross compile a simple GLFW + GLAD app from a Windows X86_64 Host to a Linux X86_64 Target by using a LLVM toolchain, both custom and builtin.To that end I acquired GLFW by cloning the repository and GLAD by using the generator. I used a docker container to obtain a sysroot containing libraries to link against. But xmake failed to build the GLFW package with cmake.

CHAPTER ONE

STEPS TO REPRODUCE

1. Acquire llvm toolchain by compiling from source

```
scoop install make cmake python
git clone --recursive git@github.com:tcottin/llvm-project.git
cd llvm-project
git switch clangd-doxygen-parser
cmake -S llvm -B build -G Ninja ^
    -DLLVM_ENABLE_EH=ON ^
    -DLLVM_ENABLE_RTTI=ON ^
    -DLLVM_ENABLE_ASSERTIONS=ON ^
    -DLLVM_TARGETS_TO_BUILD="X86;ARM;AArch64;RISCV;WebAssembly" ^
    -DCMAKE_BUILD_TYPE=Release ^
    -DLLVM_ENABLE_PROJECTS="clang;lld;clang-tools-extra;lldb" ^
    -DLLVM_ENABLE_RUNTIMES="libcxx;compiler-rt;libcxxabi"
ninja -S build -j
cmake --install build --prefix C:/dev/llvm
```

2. Acquire the dev branch of xmake by downloading `xmake-dev.win64.exe` from the `Releases` in the github repository and running it

3. Acquire the sysroot by running a debian:bookworm image then installing dependencies with `apt`, or just use the one I provided at the repo

```
docker pull debian:bookworm
docker run -it --name=deb debian:bookworm
apt update; apt install apt install libc6-dev libstdc++-12-dev libc++-dev libc++abi-dev libwayland-dev
exit; docker export deb -o deb.tar
```

4. Export the sysroot by using `7z` with admin rights

5. Try out examples I provided

CHAPTER

TWO

WHAT DO I EXPECT

1. You should be able to compile the examples and see them working in a Linux OS Machines