[https://mxe.cc/#tutorial](https://mxe.cc/" \l "tutorial)

Zo weinig mogelijk sudo gebruiken.

(stap 2 moet niet. En stap 3b kan maar liever 3a)

Het is moeilijker om te cross compilen van linux naar windows dan omgekeerd.

Je moet het verschil tussen windows en linux build kunnen zetten op het verslag.

Waarom cross compilation als je een ander os targert? Andere manier om er naar te kijken bv c:\\ is in windows maar niet in linux (vraag bestand project)

1)

**git clone https://github.com/mxe/mxe.git**

2)

Make **qt – “version”** (command om het te kunnn laten cross compilen)

Om te zien hoeveel cores je make laat gebruiken doe je best **make –j “nr of cores” qt-”version”**

Version is het beste om versie 5 te gebruiken.

3)

Gaat er van uit dat stap 2 gelukt is.

**export PATH=/*where MXE is installed*/usr/bin:$PATH**

de path=/where de where is om te zeggen waar de executable staat. Zodat die niet overal moet kijken om het systeem.

Als je bv /home/wim/mxe doet gaat deze enke de mxe executable zien. En je moet dan het 2de deel installed/user/bin zetten om u path er aan toe te voegen aan het standaard path.

Best nooit “$(TARGET)/bin” doen want dan is het enkel voor je native systeem en niet voor een ander systeem.

**unset `env | \**

**grep -vi '^EDITOR=\|^HOME=\|^LANG=\|MXE\|^PATH=' | \**

**grep -vi 'PKG\_CONFIG\|PROXY\|^PS1=\|^TERM=' | \**

**cut -d '=' -f1 | tr '\n' ' '`**

het beste is om in de console waar je het gemaakt hebt om alles eerste te unsetten en dan enkel het nodige te gaan zetten.

Hoe kun je het dan wel doen: (script)

myENV.sh

set path

unset ENV

. myENV.sh ( de spatie is belangrijk. Anders ben je het path kwijt als het script is afgelopen)

./script om het script te laden.

4)

**/<where-MXE-is-installed>/usr/i686-w64-mingw32.static/qt5/bin/qmake**

**make**

je hebt hier 2 commands.

Als je cmake gebruikt moet je de pakage nog exptra exporten.

linux

1. MXE 🡪 Builds toolchain (cross)

Eerst toolchain builden

1. MXE toolchain builds

Your app

Toolchain applicatie builden.

windows

1. Runs your app