

Build Instructions

Project structure

The following sub-projects should be present:

- /src/cpplocate
- /src/cppassist
- /src/cppfs
- /src/cppexpose
- /src/qmltoolbox
- /src/glbinding
- /src/globjects
- /src/glkernel
- /src/gloperate
- /src/gtx3

Additionally, two external dependencies are required:

- /src/3rdparty/glm
- /src/3rdparty/proj

Build external dependencies

There is a build script to build all 3rd-party dependencies at once:

```
cd <GTX_DIR>/src/3rdparty
./build-externals.sh
```

To do it manually, follow the instructions below.

Build glm

```
cd <GTX_DIR>/src/3rdparty/glm
cmake .
make
```

(to rebuild, remove CMakeCache.txt before building again)

Build proj

```
cd <GTX_DIR>/src/3rdparty/proj
mkdir build
cd build
cmake ..
make
```

```
cp -r lib ..
cd ..

(to rebuild, rm -rf build)
```

Setup build scripts and configuration

Each sub-project has a build script *configure*, which helps setup the necessary configuration for building the project and build the project itself. The build script uses configuration files in *.localconfig*, which contain the necessary environment variables and cmake parameters to configure and build the project.

To setup the build scripts when *.localconfig* does not yet exist, call *configure*:

```
cd cpplocate
./configure
```

This has generated the folder *.localconfig* and the file *.localconfig/default*. Adjust *.localconfig/default* to reflect for example the project options you want to use, the search path for external libraries, and the installation path for the project.

.localconfig/default:

```
# CMake generator
CMAKE_GENERATOR="Unix Makefiles"

# Build directory and build type
BUILD_DIR="build"
BUILD_TYPE="Release"

# Include default options
source <GTX_DIR>/src/dependencies
```

For simplicity, we source another file *dependencies* here, which contains the paths for all of the projects and dependencies in a single file, so we don't have to duplicate that for all of the projects.

/src/dependencies:

```
# Installation directory
export CMAKE_OPTIONS="${CMAKE_OPTIONS} -DCMAKE_INSTALL_PREFIX=<GTX_DIR>/install"

# Dependencies
export CMAKE_PREFIX_PATH="${CMAKE_PREFIX_PATH}:<GTX_DIR>/src/3rdparty/glm"
export CMAKE_PREFIX_PATH="${CMAKE_PREFIX_PATH}:<GTX_DIR>/src/3rdparty/proj"
export CMAKE_PREFIX_PATH="${CMAKE_PREFIX_PATH}:<GTX_DIR>/src/cpplocate"
export CMAKE_PREFIX_PATH="${CMAKE_PREFIX_PATH}:<GTX_DIR>/src/cppassist"
export CMAKE_PREFIX_PATH="${CMAKE_PREFIX_PATH}:<GTX_DIR>/src/cppfs"
export CMAKE_PREFIX_PATH="${CMAKE_PREFIX_PATH}:<GTX_DIR>/src/cppexpose"
```

```
export CMAKE_PREFIX_PATH="${CMAKE_PREFIX_PATH}:<GTX_DIR>/src/qmltoolbox"
export CMAKE_PREFIX_PATH="${CMAKE_PREFIX_PATH}:<GTX_DIR>/src/glbinding"
export CMAKE_PREFIX_PATH="${CMAKE_PREFIX_PATH}:<GTX_DIR>/src/globjects"
export CMAKE_PREFIX_PATH="${CMAKE_PREFIX_PATH}:<GTX_DIR>/src/glkernel"
export CMAKE_PREFIX_PATH="${CMAKE_PREFIX_PATH}:<GTX_DIR>/src/gloperate"
```

Adjust the options for each project in its *.localconfig/default* file, or adjust the options for all projects together in *dependencies*.

Available options and configurations

The following options are available and should be set in the configuration files:

- **CMAKE_GENERATOR**: The type of project files that shall be generated (e.g., “Unix Makefiles”)
- **BUILD_DIR**: Name of the subdirectory for the build (e.g., “build”)
- **BUILD_TYPE**: The build configuration (e.g., “Release”, or “Debug”)
- **CMAKE_OPTIONS**: List of additional cmake options, such as **CMAKE_INSTALL_PREFIX** to define the install location.
- **CMAKE_PREFIX_PATH**: List of search paths for dependencies.

Lists should always be appended and separated by ‘:’, e.g.:

```
export CMAKE_OPTIONS="${CMAKE_OPTIONS} -DCMAKE_INSTALL_PREFIX=<GTX_DIR>/install"
export CMAKE_OPTIONS="${CMAKE_OPTIONS} -DOPTION_BUILD_EXAMPLES:BOOL=OFF"
...

export CMAKE_PREFIX_PATH="${CMAKE_PREFIX_PATH}:<GTX_DIR>/src/cpplocate"
export CMAKE_PREFIX_PATH="${CMAKE_PREFIX_PATH}:<GTX_DIR>/src/cppassist"
...
```

Configure and build

After setting up the configuration, a project can be configured and built.

To setup the project, call the *./configure* script again:

```
cd cpplocate
./configure
```

This will create the build directory, in which the project can be built:

```
cd build
make -j4
```

To install the resulting package into the system (not necessary for development):

```
make install
```

Project dependencies and order

Due to the dependencies between the project, they should be built in the following order:

- cpplocate
- cppassist
- cppfs
- cppexpose
- qmltoolbox
- glbinding
- globjects
- glkernel
- gloperate
- gtx3