Building the Orbit Game Engine

Treelab

June 10, 2021

Orbit depends on NVidia PhysX, Eigen and optionally also on Dear ImGui and https://www.opengl.org/resources/libraries/glut/ if you want to use OpenGL as a renderer. It also depends on CMake to build your project.

1 Building NVidia PhysX

1.1 Customization

Go into `physx/buildtools/presets/public` and open your build preset (i.e. for MSVC x64 and VS 2019 `vc16win64.xml`). Set the following values:

```
PX_GENERATE_STATIC_LIBRARIES = True
NV_USE_STATIC_WINCRT         = True
```

If your compiling for Unix, the setting `NV_USE_STATIC_WINCRT` is not important.

1.2 Optional Customization

You may also want to change the following settings to reduce compilation and build time.

```
PX_BUILDSNIPPETS      = False
PX_BUILDPUBLICSAMPLES = False
```

1.3 Building NVidia PhysX

Open a command prompt and change into `physx/` and run `generate_projects` (either the `.sh` or the `.bat`) and select the generator you want to use (the one you changed the settings for in the previous steps).

1.4 Building NVidia PhysX

You can find your build files in a folder in `physx/compiler/` with the name of your chosen generator. Now you want to build the Debug and Release versions of PhysX (i.e. with Visual Studio or using the generated Makefiles).

2 Eigen

Download Eigen into a folder on your system (decompress the downloaded file).

3 ImGui (optional)

Clone ImGui into a folder on your system (or download and decompress it).

4 d3dx12.h

This step is only important if you selected DX12 as your renderer.

The `d3dx12.h` file that comes with Orbit (this is the DirectX 12 helper library by Microsoft that can be found here) must be placed somewhere your build system can find it. Under Visual Studio, you could create a folder on your system, put the file in that folder and set that folder as an additional include directory.

5 Glut

This step is only important if you selected OpenGL as your renderer. Or you want to support OpenGL at some point.

Install Glut (i.e. with `apt-get install freeglut3-dev`).

6 CMake

Open a command prompt and change into the Orbit root directory. Run the cmake command

```
cmake
  -S .
  -B build
  -DPHYSX_ROOT_PATH:PATH=/your/path/to/physx_root
  -DPHYSX_LIBRARY_PATH:PATH=/your/path/to/physx_libs
  -DEIGEN_ROOT_PATH:PATH=/your/path/to/eigen_root
  [-DIMGUI_ROOT_PATH:PATH=/your/path/to/imgui_root]
```

If you do not wish to use ImGui, replace the last line of the command with `-DWITH_DEAR_IMGUI:BOOL=False`. If you do not wish to build the samples, you can also add `-DBUILD_SAMPLES:BOOL=False`.

The PhysX root path is the path to the folder that contains the `physx` and `physxshared` directories. The PhysX Library folder is the folder `physx_root/physx/bin/<configuration>` where `<configuration>` is the configuration that you used to build PhysX. Be careful to select the `.md` folder and not the `.mt` folder. The latter won't work with Orbit.

7 Building Orbit

Now you only have to build Orbit for Debug and Release. Project files have been generated in the `build` folder.