ForkENGINE Installation Guides

Lukas Hermanns

March 2014

1 Compiler

The ForkENGINE has been written in modern C++11. Currently the following compilers are supported:

- Microsoft VisualC++ 2013 (MSVC12)
- MinGW 4.7.1 (GNU C++)

2 Setup

2.1 Include Directory

Set the include directory to "ForkENGINE/include". In your sources you can then include the engine modules like in the following example:

```
#include <fengine/core.h>
#include <fengine/video.h>
#include <fengine/scene.h>
```

Or if you always want to include everything, use this header file:

```
#include <fengine/import.h>
```

2.2 Library Directory

There are several library directories.

- Use "ForkENGINE/lib/MSVC12/Win32" when you are creating a 32-bit project with Microsoft Visual Studio 2013.
- Use "ForkENGINE/lib/MSVC12/Win64" when you are creating a 64-bit project with Microsoft Visual Studio 2013.
- Use "ForkENGINE/lib/MinGW" when you are creating a 32-bit project with MinGW on MS/Windows.
- Use "ForkENGINE/lib/Posix" when you are creating a 32-bit project with GNU/C++ on a posix system (e.g. GNU/Linux).

Only link to the libraries you need in your projects. You'll find a list of all libraries and a brief description in the "Libraries" section.

2.3 Binary Directory

2.3.1 MS/Windows

On MS/Windows, the engine consists of the following binary files:

- ForkAnimation.dll
- ForkAudio.dll

- ForkAudioAL.dll (*OpenAL* audio system)
- ForkAudioXA2.dll (*XAudio 2* audio system)
- ForkCore.dll (*IO*, *Math* and *Platform* core systems)
- ForkENGINE.dll
- ForkNetwork.dll
- ForkPhysics.dll
- ForkPhysicsNw.dll (*Newton Game Dynamics* physics system)
- ForkPhysicsPx.dll (*NVIDIA PhysX* physics system)
- ForkRenderer.dll
- ForkRendererD3D11.dll (*Direct3D 11.0* render system)
- ForkRendererGL.dll (*OpenGL* render system)
- ForkScript.dll
- ForkScene.dll
- ForkUtility.dll
- OpenAL32.dll (OpenAL low level audio library)

The best way to use the binary files is to add the directory (respective to the used library directory, (e.g. "ForkENGINE/bin/MSVC12/Win32") to the PATH variable on MS/Windows.

2.3.2 Posix

On posix systems (e.g. GNU/Linux), the engine consists of the following binary files:

- ForkAnimation.so
- ForkAudio.so
- ForkAudioAL.so (*OpenAL* audio system)
- ForkCore.so
- ForkENGINE.so
- ForkNetwork.so
- ForkPhysics.so
- ForkPhysicsNw.so (*Newton Game Dynamics* physics system)
- ForkRenderer.so
- ForkRendererGL.so (*OpenGL* render system)
- ForkScript.so
- ForkScene.so
- ForkUtility.so

2.4 Dependencies

The ForkENGINE has been written very modular, i.e. some libraries are loaded dynamically during run-time. The render system implementations for instance are not statically linked to the engine and therefore should not be linked statically to your projects.

The advantage is that if the host computer does not provide some libraries, your application can handle this dynamically and choose another render system.

Consider the following situation: your application wants to create a Direct3D 11 render system but the host computer does not have the d3d11.dll on MS/Windows. A common result is, that the user gets an error message which states, that a DLL is missing and the program should be re-installed.

In the case of the ForkENGINE your application can catch an exception and create another render system, OpenGL for instance (which runs almost on all platforms). This will reduce errors and libraries can be added or removed in a dynamic manner.

3 Libraries

The ForkENGINE consists of the following libraries:

3.1 ForkCore

Contains core functionalities: platform dependent code (such as creating a frame - also called "window"), input devices (mouse and keyboard), extended file access and math functions.

3.2 ForkScene

Scene management library.

3.3 ForkAnimation

Animation library.

3.4 ForkRenderer

Render system library. There are further libraries which will be loaded dynamically during run-time. These are "ForkRendererGL" (OpenGL) and "ForkRendererD3D11" (Direct3D 11.0).

3.5 ForkNetwork

Network library.

3.6 ForkUtility

Utility library.

3.7 ForkAudio

Audio library. There are further libraries which will be loaded dynamically during run-time. These are "ForkAudioAL" (OpenAL) and "ForkAudioXA2" (XAudio 2).

3.8 ForkPhysics

Physics and collision library. There are further libraries which will be loaded dynamically during run-time. These are "ForkPhysicsNw" (Newton Game Dynamics) and "ForkPhysicsPx" (NVIDIA PhysX).

3.9 ForkScript

Scripting library. This includes the **Mono scripting engine**.

3.10 ForkENGINE

Engine Device etc., connects all sub-libraries.