Tutorial 1: Setup Visual Studio 2005

[*This is ripped from the README and edited slightly for (hopefully) clarity*]

Orx's core is basically platform-independent. All the platform/OS-dependent features are implemented via plugins. These plugins can be loaded at runtime (hotplug) or they can be embedded at link-time.

If you use the non-embedded versions, you'll have to specify which plugins to use. This is more flexible but also requires additional files (the plugins themselves). The embedded version will store everything in Orx's library, meaning you won't be able to choose which plugin to use at runtime, but in exchange, Orx will be more compact and will also run considerably faster.

From the download page you'll find pre-compiled dynamic embedded binaries for Win32, Linux (x86) and MacOS X (ppc/x86). If you want to use the non-embedded versions (to use your own plugins) or the static ones, you will need to compile Orx yourself from the source. Everything compiles out-of-the-box for the hardware platforms cited above.

The embedded versions currently use:
- GLFW-based plugins for display, joystick, keyboard and mouse.
- OpenAL-based plugin for sound.
- Box2D-based plugin for physics.
- homemade plugin for 2D rendering.

Visual Studio 2005

You will require this file: orx-dev-msvs2005-*.zip from the download page. This will give you a copy of the pre-compiled Orx binary, as a dynamic library, with everything you need to get started, compiled in.

When extracted, you will be presented with a number of folders: *bin*, *include* and *lib*.

For now I'm going to create the project in a specific location, feel free to substitute your own in as you see fit.

My project will be in “C:\MyProject” move the three folders previously mentioned, into this directory.

Now we will make our project in Visual Studio:

Note, whatever you call your project, will have a folder created by that name (yes, even with the tick turned off!) So in my case, I now have “C:\MyProject\project\project.sln”. From this point on, all of our directory structures will use this place as the “base” level.

Next, add a new file to the project. "Main.cpp":

Then, it's time to change the project properties:



We set up the working directory, and include directory:



Remember, our 'base' level is here: “C:\MyProject\project\project.sln”, so these will point to “C:\MyProject\bin\” and “C:\MyProject\include”

Next activate debugging on the debug build:



And deactivate debugging on the release build:



Deactivate optimization on the debug build:



Then the preprocessor directives for Debug: (Release doesn't need any, yay!)



The linker needs to be configured too:



And the linker input:



And finally, activate the debugging info in the linker for Debug:

From:
https://www.orx-project.org/wiki/ - **Orx Learning**

Permanent link:
https://www.orx-project.org/wiki/tutorials/community/grey/setup_msvs05?rev=1449432654

Last update: **2025/09/30 17:26 (14 months ago)**