Building for Arcade and Vanilla

The ThICC Engine is created with portability in mind, and for that reason an “arcade” build is available alongside the standard PC build. Runtime libraries are also statically linked in the engine’s executable for maximum compatibility.

Building for vanilla

As default, Visual Studio configures C++ projects to use multi-threaded DLL runtime libraries (/MD, /MDd). This means that any features used inside of a C++ project from Microsoft’s “Visual C++” library can only be executed if the end user has the correct Visual C++ redistributables installed, in the case of the ThICC Engine, this is the 2015-2019 Visual C++ Redistributable Package. In the past DirectX redistributables would also have been required with this linking method, however in the last few years Microsoft have embedded them within the core Windows framework.

As the ThICC Engine is designed for maximum portability, an alternate Visual Studio project configuration was opted for, being static multi-threaded runtime libraries (/MT, /MTd). This build option has a negative impact on the resulting file size of the compiled executable as runtime libraries are embedded within the application, but makes it available to run standalone just about everywhere – as tested on a “vanilla” machine with a clean Windows 10 install.

Creating the arcade build

To highlight the ThICC Engine’s mission of ultimate portability, it supports an optional build target for “arcade” builds. This is a highly optimised build without a number of potentially performance-intensive systems such as high-detail cubemaps, skyboxes, AI, and high-poly maps. Within the engine’s toolkit maps can be selected for use in arcade or PC builds, and doing so will limit them to these builds, forming a method of exclusivity.

The ThICC Engine’s keybind manager automatically remaps inputs for the arcade build to a standard arcade mapping system, which is:

- JOYSTICK UP
- JOYSTICK DOWN
- JOYSTICK LEFT
- JOYSTICK RGHT
- BLUE
- YELLOW
- GREY 1/2
- GREEN
- RED
- SIDE BUTTON

This can be configured through the engine’s toolkit in a cross-platform input mapping system (see the *ThICC Pipeline Documentation* for information on how to utilise this).

Resolution is also capped in the arcade build at 320x240. This is forced within the engine’s core systems in order to achieve maximum performance. The output render can still be overridden to fullscreen if required.