

Directory Structure

Overview of the directories that make up the engine and game projects.



At the top level, there is the Engine directory as well as any game projects you have. The Engine directory contains the engine itself and all of the tools that come with it. Each game folder contains all of the files pertaining to that game.

Root Directory

- **Engine** - Contains all source code, content, and so on that makes up the engine.
- **Templates** - Collection of project templates available when creating new projects.
- **GenerateProjectFiles.bat** - Used to create the Unreal Engine solution and project files necessary for working with the engine and your game or games in Visual Studio. See [Project Files for IDEs](#) for details.
- **Default.uprojectdirs** - Helper file that allows the engine to discover projects in subdirectories.

Common Directories

Some subdirectories are common amongst both the Engine and game project directories:

- **Binaries** - Contains executable files or other files created during compiling.
- **Build** - Holds files needed for building the engine or game, including files necessary for creating platform-specific builds.
- **Config** - Configuration files for setting values that control engine behavior. Values set in the game project Config files override the values set in the `Engine\Config` directory.
- **Content** - Holds content for the engine or game, including asset packages and maps.
- **DerivedDataCache** - Contains derived data files generated on-load for referenced content. Not having cache files present for referenced content can increase load times dramatically.

- **Intermediate** - Contains temporary files generated during building the engine or game. In game directories, Shaders are stored in the Intermediate directory.
- **Plugins** - Contains plugins used in the engine.
- **Saved** - Contains autosaves, configuration (`.ini`) files, and log files. Additionally, the **Engine > Saved** directory contains crash logs, hardware information, and Swarm options and data.
- **Source** - Contains all of the source files for engine or game, including the engine source code, tools, gameplay classes, etc.
 - **Engine** - Source files in the Engine directory are categorized into the following:
 - **Developer** - Files used by both the editor and engine.
 - **Editor** - Files used by just the editor.
 - **Programs** - External tools used by the engine or editor.
 - **Runtime** - Files used by just the engine.
 - **Game** - Source files in a game project directory are organized by module; one directory per module. Each module contains the following:
 - **Classes** - Contains all gameplay class header (`.h`) files. This directory should not be added to in the future.
 - **Internal** - Contains cross module includes within the same plugin or module without exposing the include more widely.
 - **Private** - Contains all source (`.cpp`) files including gameplay class implementation files and the module implementation file.
 - **Public** - Contains the module header file.



Some directories exist both in the engine's common directories and specific game directories. For example, the **Content** directory might exist in both the engine directory (`../Engine/Content`) and in your game directory (`../GAME_DIR/Content`). Files in your game's content directory are only accessible within that particular game whereas files contained in the engine's content directory is accessible by any project that uses that particular engine distribution.

Modules and Plugins

Unreal Engine functionality is organized into many modules and plugins. One of the primary differences between modules and plugins is that modules only contain code. As an example, when you create a project in Unreal Engine, your project source code is organized into a module with a `*.Build.cs` file. Plugins contain their own source files, binaries, and a `.uplugin` file. Plugins can also contain assets whereas modules cannot. As a result of this, you can redistribute your plugin to use it in other UE projects.



For more information about modules, plugins, and the distinctions between the two, see the [Modules](#) and [Plugins](#) documentation pages.

Engine-specific Directories

Some subdirectories are specific to the Engine directory:

- **Documentation** - Contains the engine documentation, both source and published files.
 - **HTML** - Published HTML documentation files.
 - **Source** - Source markdown documentation files.
- **Extras** - Additional helper and utility files.
- **Programs** - Contains configuration files and log files for projects stored in the Unreal Engine root directory and other Unreal programs such as Unreal Frontend and Unreal Header Tool.
- **Shaders** - Holds the shader source files (`.usf`) for the engine.

Game Project Directories

Directory	Description
Binaries	Contains executable files or other files created during compiling.
Config	Default project settings for your game.
Content	Contains content for the engine or game, including asset packages and maps.
External dependencies	Displays the public engine header files (only visible in Visual Studio).
Intermediate	Contains files that are generated by Unreal Build Tool such as Visual Studio project files. These files can be deleted and rebuilt.
Saved	Contains files that are generated by the engine such as config files and logs. These files can be deleted and rebuilt.
Source	Contains game module object class files.

Solution Directories

Directory	Description
Classes	Contains game object class definitions (<code>.h</code> files).
Config	Default project settings for your game.
External dependencies	Displays the public engine header files (only visible in Visual Studio).

Directory

Description

Private	Contains private game object class implementation files (<code>.cpp</code> files).
Public	Contains public game object class implementation files (<code>.cpp</code> files).