
UnrealEngineWiki

Release 0.0.1

JeffCube

Jan 11, 2022

CONTENTS:

1	Contents	3
1.1	Version Control	3
1.2	Multiplayer	3
1.3	UI	4
1.4	Pawns And Controllers	6

UnrealEngineWiki is a wiki containing information on various topics about the Unreal game engine.

Note: This project is under active development.

CONTENTS

1.1 Version Control

1.1.1 Setup

Links

- [UE4 Git Tutorial](#)
- [UE4 Multiplayer Tutorial](#)
- [UE4 Hosting and Joining Games](#)

1.2 Multiplayer

1.2.1 Multiplayer Quick Reference

Links

- [UE4 Multiplayer Tutorial](#)
- [UE4 Hosting and Joining Games](#)
- [UE4 Online Session Nodes](#)
- [Local Multiplayer Spawning System](#)

Note: When dealing with server client multiplayer the local player index is always 0

- [UE4 Multiplayer Blueprint Series](#)
- [UE4 Network Compendium](#)

Replication

- Setting Replication to reliable should be for essential functions, not things like particle effects. Your network can get bogged down if too many things are set to be reliable
- Player Controllers Exist both on the client and the server. There are 2 versions of them. One that exists on the server and the one that exists on each client. Player controllers on the client are not aware of other player controllers so we are going to do some things to help get information from other player controllers into our lobby menu.

Useful Nodes

- **Is server:** Lets you know if you are the server or not. Useful in control flow logic.
- **Has Authority + Branch (Shortcut: Switch Has Authority):** Allows you to run code depending if you are the authority (server) or remote (client).
- **Execute Console Command:** Useful for server travel and joining games
 - **servertravel:** If you are the host take the entire server with you to a map:

```
servertravel /JeffSandbox/Maps/Arena_0
```

Note: To when `servertravel`, use it within the game mode (done in tutorial). Also make sure seamless travel is set in class defaults for the game mode.

Also `servertravel` should be done in standalone mode to avoid bugs

- **open:** Use to join a server by its ip address:

```
open <ip_address>
```

- If you are kicking someone, make sure that kick function is running on the owning client
- If a bug is running you ragged. it may be because a reference to self is missing somewhere.

Spawning

- For designated spawn points, on the player start there is a player start tag you can define. You can filter out based on tag and assign 2 different locations of where to spawn people.

1.3 UI

1.3.1 UI Quick Reference

Tables

- [UE4 Tables and Widgets Reordering Tutorial](#)
- [UE4 Online Subsystem](#)
- [UE4 Online Subsystem Steam](#)

- [More Comprehensive Multiplayer Tutorial](#)

Materials

- [Striped Materials](#)
- [Varying Materials With Time](#)
- [Vertex Painting](#)
- [Sprite Materials](#)
- [Ripple Effect](#)
- [Two Sided Material](#)

Notes

- Solution for future generations: Sprites must be marked as 'Movable' if you want to change their color. [link](#)

Multiplayer

- [Local Multiplayer Spawning System](#)

Note: When dealing with server client multiplayer the local player index is always 0

- [UE4 Multitplayer Blueprint Series](#)
- [UE4 Network Compendium](#)

Replication

- Setting Replication to reliable should be for essential functions, not things like particle effects. Your network can get bogged down if too many things are set to be reliable
- Player Controllers Exist both on the client and the server. There are 2 versions of them. One that exists on the server and the one that exists on each client. Player controllers on the client are not aware of other player controllers so we are going to do some things to help get information from other player controllers into our lobby menu.

Useful Nodes

- [Is server](#): Lets you know if you are the server or not. Useful in control flow logic.
- [Has Authority](#) + Branch (Shortcut: [Switch Has Authority](#)): Allows you to run code depending if you are the authority (server) or remote (client).
- [Execute Console Command](#): Useful for server travel and joining games
 - `servertravel`: If you are the host take the entire server with you to a map:

`servertravel /JeffSandbox/Maps/Arena_0`

Note: To when `servertravel`, use it within the game mode (done in tutorial). Also make sure seamless travel is set in class defaults for the game mode.

Also `servertravel` should be done in standalone mode to avoid bugs

- **open:** Use to join a server by its ip address:

`open <ip_address>`

- If you are kicking someone, make sure that kick function is running on the owning client
- If a bug is running you ragged. it may be because a reference to self is missing somewhere.

Spawning

- For designated spawn points, on the player start there is a player start tag you can define. You can filter out based on tag and assign 2 different locations of where to spawn people.

UI

- If your widget appears over other widgets you don't need a canvas panel

1.4 Pawns And Controllers

1.4.1 Links

- [Setting Up Character Movement](#)
- [Possessing Pawns](#)
- [Input](#)
- [Get / Set Control Rotation](#)
- [Static Camera: When you want your pawn to look out of a specific camera](#)

1.4.2 Default Camera

Looking at a default player controller you will see the viewport contains a camera. When a Player Controller possesses a pawn, if no camera on that pawn exists the player controller uses the location / rotation of the pawn as the default camera location / rotation. In the case of a Character Pawn the camera location is determined by the pawn location and the Base Eye Height Settings under the Camera settings in the Character details panel.

1.4.3 Rotating Your Character

In UE4 tutorials, it is common to see that within the Character Pawn, the XAxis input of the mouse is connected to the Add Controller Yaw Input Function and the YAxis input to the Add Controller Pitch Input. When using these functions the user will find that they can now rotate the player along the yaw direction (Scanning horizontally) and the pitch direction (up and down). However if they take a look at the pawn itself they will find that the pawn only rotates along the Yaw and not the Pitch. This is because by default the pawn has Use Controller Rotation Yaw set to true and the other rotations are set to False.