

PROTEUS VR TEMPLATE

First release: March 2016 (under the name "SteamVR Template")

- Latest Version 6.2 (under the name "Proteus VR Template")
- Latest release July 30, 2019

What can I do with that, that I cannot do with the "VR Template" in Unreal?

Well, rapidly: It's single or multiplayer (LAN, Oculus Home or Steam); It uses Oculus Avatars (or not); It uses voice over IP (VOIP); full of nice features fully using Oculus functions; compatible Oculus Go/Quest/Rift/S and HTC Vive; minimum use of C++ (but still some, since some methods not possible only in blueprints); hey, we were the first to propose a template back in 2016!

FOR NOOBS: This is super simple. Even if you think Visual Studio is the name of a VFX/Hollywood startup somewhere around LA with people working in open spaces and flip-flops, it requires few manipulations to set up; I could add few instructions how-to on demand.

This template consists of virtual reality compatible pawn and settings, ready to drop in your single or multiplayer app. So what you can do is choose the features you want, and build yourself/modify a pawn based on the functionalities you need.

Discussion on Unreal forum at <https://forums.unrealengine.com/development-discussion/vr-ar-development/106631-single-multiplayer-touch-vive-proteus-blueprint-only-template>

Download it at <https://github.com/ProteusVRpublic/ProteusTemplate>

Compatible with the following Unreal Engine versions:

Unreal Engine 4.22*, latest commit 5f77421 (July 25, 2019), found at <https://github.com/Oculus-VR/UnrealEngine>

* minor changes to the engine possibly / mandatory required



What's new in version 6.2

- Unreal 4.22.3
- “Ghost” function for clients to appear / disappear
- Works with Oculus Quest, but in reduced 72Hz mode due to 1.39 plugin problem
- 1.39 enables FFR+MultiView+Vulkan
- 1.39 enables Avatars 64-bits on the Quest

Compatibility

- Oculus Plugin 1.39
- Oculus Platform SDK 1.32
- Oculus Audio SDK 1.34
- Oculus Avatar SDK 1.36
- Steam Platform 1.39

To install as a template, just unzip into the appropriate templates directory like C:\Program Files\Unreal Engine[Version]\Templates for launcher version or[ForkLocation]\UE4\Templates for source version. Launch a new project, and you'll find it in the blueprint section.

- To open as a project file, open the project with the launcher or directly from the .uproject file.



COMPILING UNREAL ENGINE FROM SOURCE

1. Download the Unreal Engine .zip file from github (or sync through github app)
2. Unzip
3. Setup.bat
4. Replace \Engine\Plugins\Runtime\Oculus\OculusAvatar with the files provided
5. GenerateProjectFiles.bat
6. Open UE4.sln with Visual Studio
7. Make the modifications needed to the engine if needed (SEE NEXT PAGE)
8. Build UE4
9. Build UnrealLightmass

Then

1. Download the latest Proteus Template .zip from github (or sync through github app)
2. Right-mouse click on Proteus.uproject, Switch Unreal Engine Version, for correct engine version
3. 10) Open Proteus.sln with Visual Studio
4. Build Proteus
5. Now ready to open Proteus.uproject



DO I NEED TO MAKE MODIFICATIONS TO THE ENGINE?

1. I want to develop for Oculus Quest and/or Go

change

\Engine\Plugins\Online\OnlineSubsystemOculus\Source\Private\OculusIdentityCallbackProxy.cpp

Line 26

from:

```
DelegateHandle = Online::GetIdentityInterface()->AddOnLoginCompleteDelegate_Handle(
```

to:

```
DelegateHandle = OculusIdentityInterface->AddOnLoginCompleteDelegate_Handle(
```

Reference: <https://github.com/Oculus-VR/UnrealEngine/pull/26>

2. I want to use Oculus Avatars on the Quest, but I don't have access to Quest Platform API

change

\Engine\Plugins\Runtime\Oculus\OculusVR\Source\OculusHMD\OculusMobile_APL.xml

Line 223

from:

```
com.oculus.svclib.OVREntitlementChecker.doAutomatedCheck(this);
```

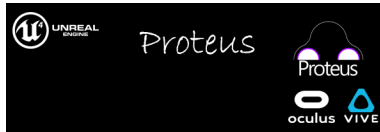
to:

```
<!-- com.oculus.svclib.OVREntitlementChecker.doAutomatedCheck(this); -->
```

3. I want to nativize assets without errors when compiling

change

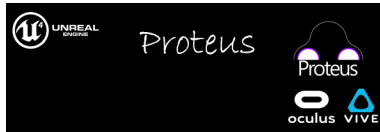
\Engine\Source\Runtime\Engine\Classes\GameFramework\PlayerController.h



line 1053 private to public, because 'PlayDynamicForceFeedback' is a private member of 'APlayerController', for reasons unknown

Reference:

<https://answers.unrealengine.com/questions/831405/playdynamicforcefeedback-node-in-420-fails-to-cook.html>



How can I install it in my project?

In the template

- Open "Proteus.uproject"
- Inside the editor, right-click on the folder "Proteus_Multi" and select Migrate
- Migrate everything inside the folder to your project, inside the folder your project/Content/

In your project

- If needed, adjust Project settings (see "What are the best Project Settings for VR?")
- In ProjectSettings/Maps & Mode (see page 10):
 - Ensure that Game Instance Class/GameInfoInstance is selected
 - Select entryMap as Editor Starting Map and Game Default Map
 - Select MainMenuGM as GameMode
- Don't forget to put a navmesh bound volume to allow teleportation!
- Pawn will spawn at playerstart (placed on the floor) and will teleport on navmesh
 - Put a PlayerStart tag in entryMap level: PS1
 - In the all other maps: PS1, PS2, PS3, PS4
- In the control panel found in MainMenuPC, select your options

Features compatibility:

Oculus Rift / S

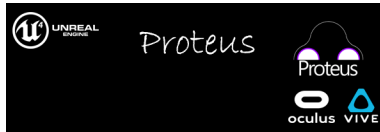
- Avatar: Oculus Avatars or Custom Avatar
- Multiplayer: Oculus Home, or Steam / LAN (Custom Avatars only)
- VOIP: Yes

Oculus Go / Quest

- Avatar: Oculus Avatars or Custom Avatar
- Multiplayer: Oculus Home (Go only, Quest developers need access to Quest API)
- VOIP: No (Oculus Avatars), Yes (Custom Avatars)

HTC Vive

- Avatar: Custom Avatar
- Multiplayer: Steam / LAN
- VOIP: Yes



What is the default input mapping?

L Touch Index Trigger

Not mapped

L Touch Hand Trigger / R Touch Hand Trigger / L Vive Trigger / R Vive Trigger / Go Trigger

Grab

L Touch Y Button / L Vive Grip

Guardian / Chaperone visibility

L Touch X Button / R Touch A Button / L Vive Touchpad / R Vive Touchpad / Go Trackpad

Teleport

R Touch B Button / R Vive Grip / Go Home Button

Menus Selection

L Touch Thumbstick Button / L Vive Menu Button

Set Spectator Screen Mode (Oculus Rift / S) / or Vive Camera (HTC Vive)

R Touch Thumbstick Button / R Vive Menu Button

Open / Close Menu

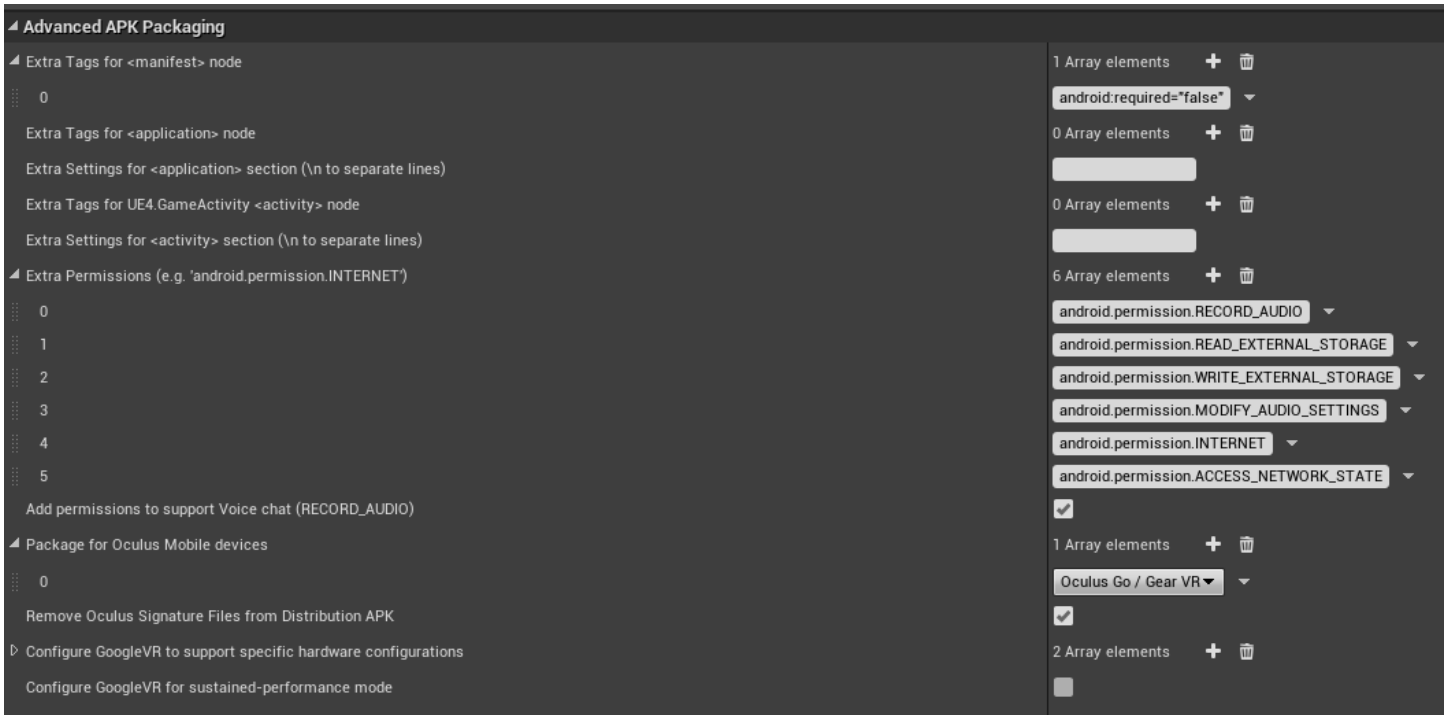
What is the default input mapping for the Oculus Go?

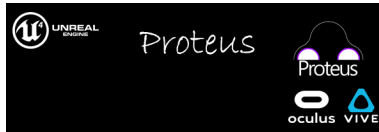
Oculus Go Buttons correspond to:

- Oculus Go Home Button: Same as (Right) Touch A Button
- Oculus Go Trackpad (x-y axis): Same as Right Touch Thumbstick x-y axis
- Oculus Go Trackpad Button: Same as Right Touch Thumbstick Button
- Oculus Go Trigger: Same as Right Touch Trigger
- Oculus Go Back Button: Reserved

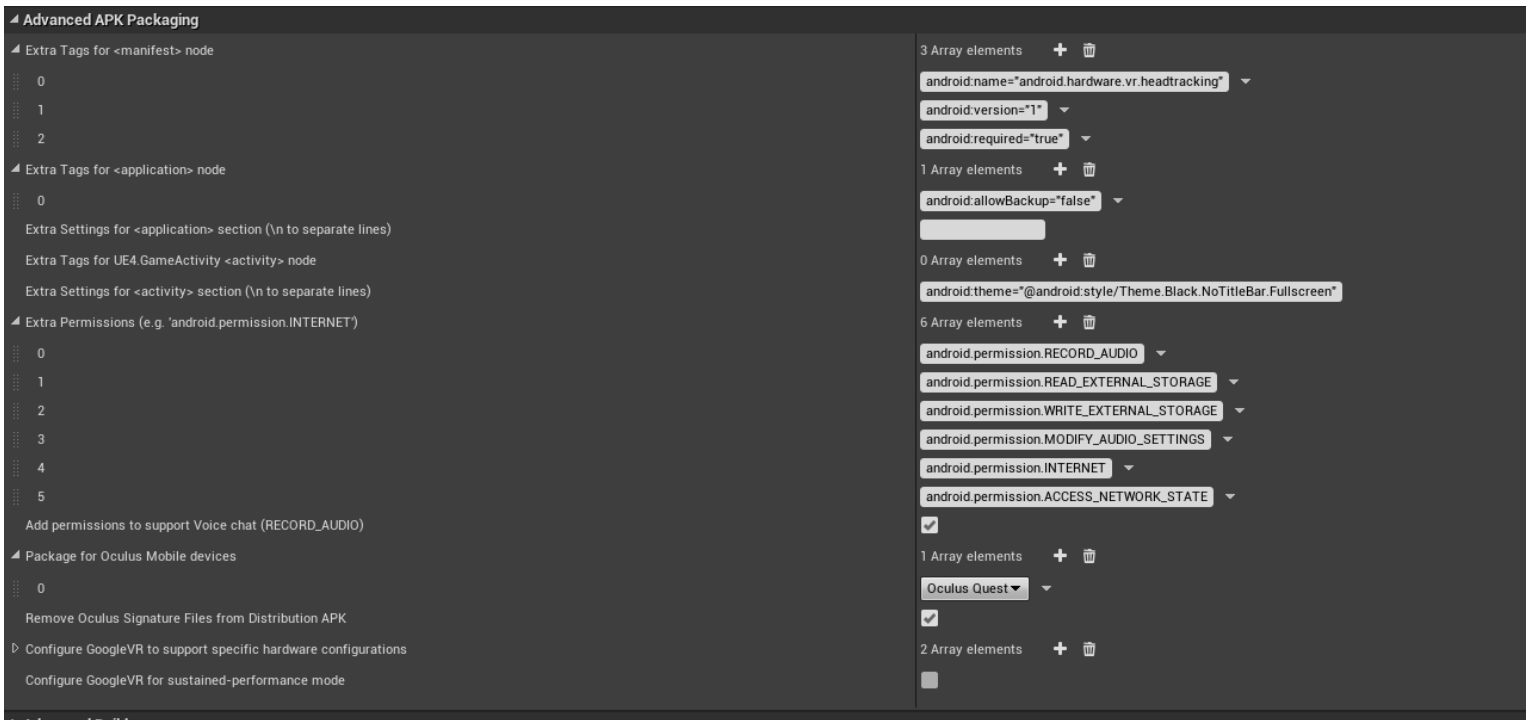
What are the best Project Settings for VR?

- You will find the best settings already set in ProjectSettings
- Oculus Go / Quest: 4x Mobile MSAA is hardcoded, no matter what
- Oculus Go works in OpenGL ES2 and ES 3.1
 - Use Android SDK 21, 32-bits Android libraries (armeabi-v7a)
 - Advanced APK settings:





- Oculus Quest works in OpenGL ES2 and ES 3.1, and Vulkan
 - Quest supports arm64, but Mobile Audio plugin is 32-bits
 - Select Android SDK 23
 - Select armeabi-v7a if using Mobile Audio Plugin
 - Select armeabi-v7a or arm64-v8a otherwise
 - Advanced APK settings for Oculus Quest:





Don't forget the settings in Project Settings:

Plugins - OculusVR

Configure the OculusVR plugin

🔒 These settings are saved in DefaultEngine.ini, which is currently writable.

General Oculus

Launch Oculus Performance Window

Launch Oculus Platform Window

Splash Screen

Auto Enabled ☐

Splash Descs 0 Array elements + 🗑

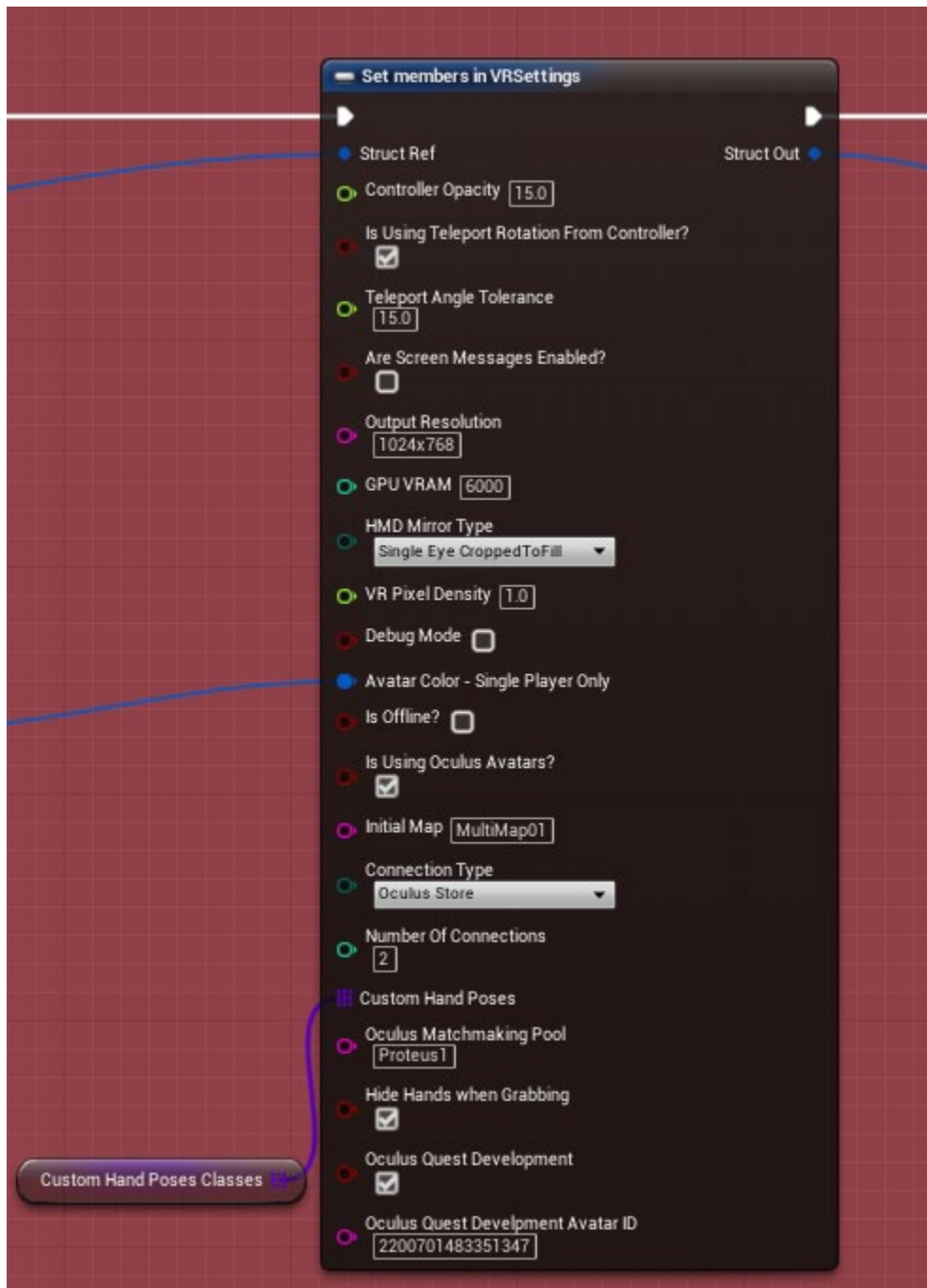
PC

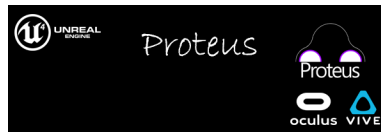
Supports Dash	<input checked="" type="checkbox"/>
Composites Depth	<input checked="" type="checkbox"/>
HQDistortion	<input checked="" type="checkbox"/>
Pixel Density Min	0.5 ▾
Pixel Density Max	1.0 ▾

Mobile

CPULevel	4 ▾
GPULevel	4 ▾
FFRLevel	FFR Off ▾
Chroma Correction	<input checked="" type="checkbox"/>
Recenter HMDWith Controller	<input checked="" type="checkbox"/>

The Control Panel in MainMenuPC

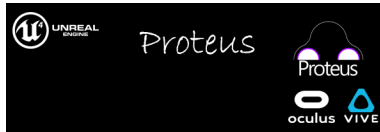




Setting	
Controller Opacity	0 up to whatever, 1 is not entirely opaque, just 15 should be fine
Teleport Rotation from Controller	If unselected, rotation is from thumbstick / trackpad
Teleport Angle Tolerance	Safety feature to teleport only on surfaces under a certain inclination. Between 0 and 90 degrees. Put 90 degrees to disable it.
Enable Screen Messages	Yes / No (in development and editor mode)
Output Resolution	The resolution on your monitor.
GPU VRAM	GPU Video Memory, can be useful to tweak to stream large textures, by default 4000 (MB). You should put it around 1 GB under your GPU VRAM
HMD Mirror Mode	See below
VR Pixel Density	Use a higher number for better quality, and a lower one for better performance
Debug Mode	Enable debug mode (no VR – see below)
Avatar Color	Avatar color when not using the Oculus Avatars
Is Offline?	You will be logged in but will not accept any connections
Is Using Oculus Avatars?	Use Custom or Oculus Avatars
Initial Map	The map you'll load into following entryMap
ConnectionType	LAN / Oculus Home / Steam. You need to be connected to Oculus Home to use the Oculus Avatars
Oculus Matchmaking Pool	Oculus Matchmaking Pool, as set in Developer Dashboard
Custom Hand Poses	See Custom Hand Poses section
Hide Hands when Grabbing	More intuitive
Oculus Quest Development	Use Oculus Avatars with Quest without Quest API
Oculus Quest Dev Avatar ID	When above option is checked, enter Avatar ID

HMD Mirror Modes

Mode	Oculus Rift	SteamVR	PS VR	Notes
Disabled	✓	✓		For the best performance on HMDs, this mode disables Spectator Screen output.
SingleEyeLetterboxed	✓	✓	✓	This mode is primarily intended for debugging purposes, showing only one letterboxed eye on the screen.
Undistorted	✓	✓	✓	This is a debug mode, showing the entire rendered area for both eyes.
Distorted	✓			This mode is only supported by Oculus. Specifically, this is an Oculus specific debug mode, showing chromatic aberrations, etc.
SingleEye	✓	✓	✓	Much like the Undistorted mode, this a debug mode for only one eye. Because this mode stretches the scene, it may be useful for identifying small artifacts in the scene.
SingleEyeCroppedToFit	✓	✓	✓	This mode crops the eye to fill the entire screen.



What is the "Debug Mode" ?

In debug mode, you can control the pawn with the keyboard, without HMD. You can also test multiplayer with a single computer (see below). Controls are

Left – Right Arrows	Turn left / Turn Right (simulate HMD rotation)
Up – Down Arrows	Go forward / Go back (simulate moving in room-scale)
W,A,S,D:	Move along 2D X,Y axis (simulate HMD location)
Left Shift	Simulate Right Index Trigger / Trigger
Left Control	Simulate Right Hand Trigger / Grip
Left ALT	Simulate B Button / TrackPad Right
Spacebar	Simulate A Button / TrackPad Up
Z	Simulate R Thumbstick Pressed / R Trackpad Pressed

To simulate multiplayer on a single computer, use debug mode with LAN, launch it in New Editor Window mode with:

Play in Standalone Game

Command Line Options
Additional Launch Parameters

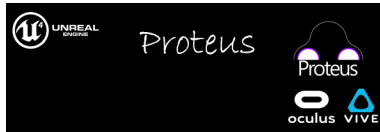
☐ Disable Sound (-nosound)

Multiplayer Options

Number of Players: 2
Server Port: 17777
Server Game Options:
Run Dedicated Server: ☐
Use Single Process: ☐
Editor Multiplayer Mode: Play As Listen Server
Command Line Arguments:
Common Window Sizes:
Multiplayer Window Size (in pixels):
Window Width: 640 Window Height: 480
Create Audio Device for Every Player: ☐

Play on Device

Build Game Before Launch: If project has code, or running a locally built editor
Launch Configuration: Same as Editor
Auto Compile Blueprints on Launch: ☒



MULTIPLAYER MODE

IMPORTANT

- Select ProjectSettings/NavigationSystem/Allow Client Side Navigation
- Don't forget to put

[/Script/Engine.GameSession]

bRequiresPushToTalk=false

in DefaultGame.ini

CHANGE IN DEFAULTENGINE.INI:

Oculus Network

[/Script/Engine.GameEngine]

!NetDriverDefinitions=ClearArray

+NetDriverDefinitions=(DefName="GameNetDriver",DriverClassName="/Script/OnlineSubsystemOculus.OculusNetDriver",DriverClassNameFallback="/Script/OnlineSubsystemUtils.IpNetDriver")

[OnlineSubsystem]

DefaultPlatformService=Oculus

bHasVoiceEnabled=true

PollingIntervalInMs=20

VoiceNotificationDelta=0.2

AllowPeerConnections=True

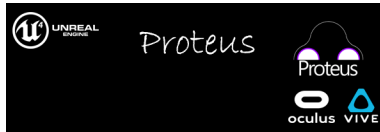
AllowPeerVoice=True

[OnlineSubsystemOculus]

bEnabled=true

RiftAppId=YOUR RIFT APP ID

GearVRAppId=YOUR GO/QUEST APP ID



[/Script/OnlineSubsystemOculus.OculusNetDriver]

NetConnectionClassName=OnlineSubsystemOculus.OculusNetConnection

Steam Network

[/Script/Engine.GameEngine]

!NetDriverDefinitions=ClearArray

+NetDriverDefinitions=(DefName="GameNetDriver",DriverClassName="/Script/OnlineSubsystemSteam.SteamNetDriver",DriverClassNameFallback="/Script/OnlineSubsystemUtils.IpNetDriver")

[OnlineSubsystem]

DefaultPlatformService=Steam

bHasVoiceEnabled=true

PollingIntervalInMs=20

VoiceNotificationDelta=0.2

AllowPeerConnections=True

AllowPeerVoice=True

[OnlineSubsystemSteam]

bEnabled=true

SteamDevAppId=480

bVACEnabled=0

GameServerQueryPort=27015

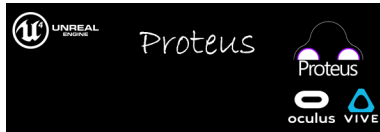
bRelaunchInSteam=false

GameVersion=1.0.0.0

bVACEnabled=1

bAllowP2PPacketRelay=true

P2PConnectionTimeout=90



[/Script/OnlineSubsystemSteam.SteamNetDriver]

NetConnectionClassName="/Script/OnlineSubsystemSteam.SteamNetConnection"

LAN

[/Script/Engine.GameEngine]

!NetDriverDefinitions=ClearArray

+NetDriverDefinitions=(DefName="GameNetDriver",DriverClassName="/Script/OnlineSubsystemUtils.IpNetDriver",
DriverClassNameFallback="/Script/OnlineSubsystemUtils.IpNetDriver")

[OnlineSubsystem]

DefaultPlatformService=NULL

bHasVoiceEnabled=true

PollingIntervalInMs=20

VoiceNotificationDelta=0.2

AllowPeerConnections=True

AllowPeerVoice=True

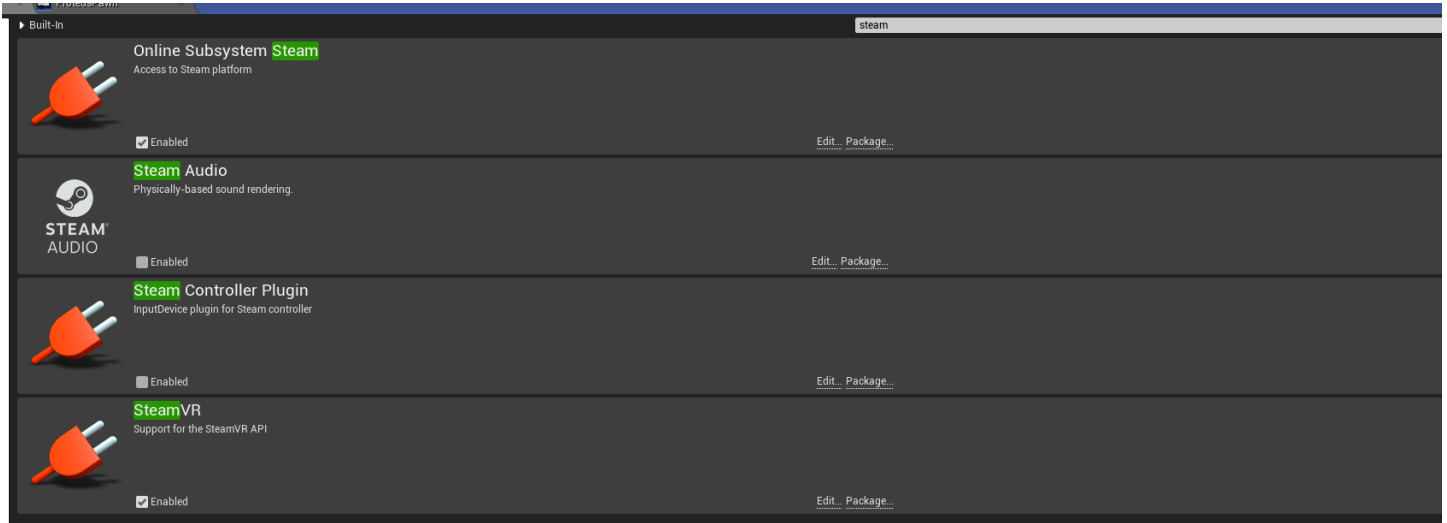
The Oculus Rift will use the RiftAppId found in Developer Dashboard FOR RIFT

The Oculus Go/QUEST will use GearVRAppId found in the Developer Dashboard FOR MOBILE

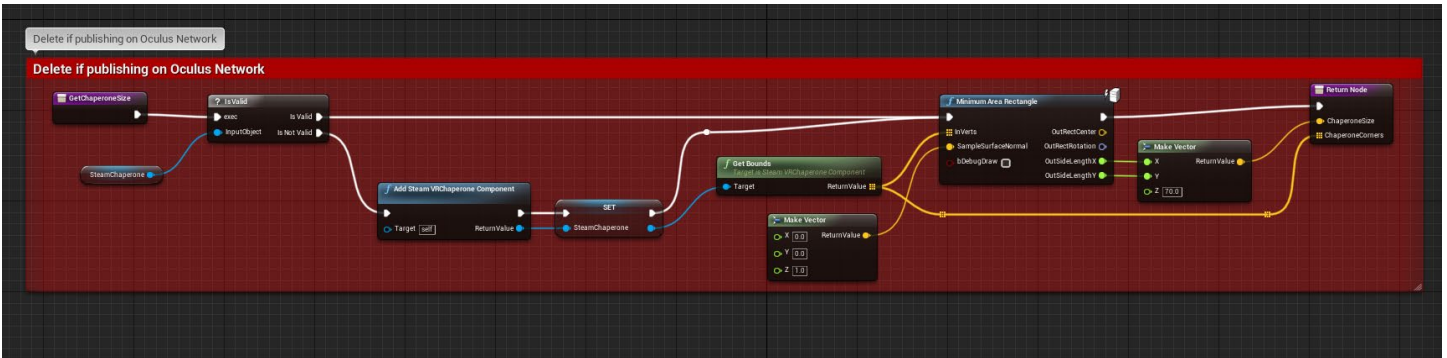
UPLOADING APP BINARIES TO OCULUS DASHBOARD

IMPORTANT:

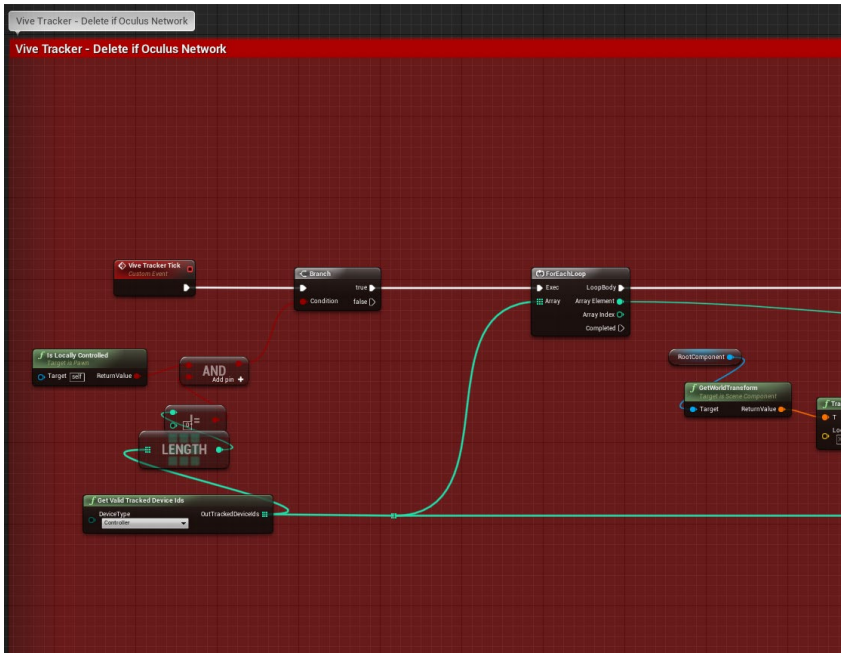
IN PLUGINS: UNCHECK STEAMVR AND STEAM SUBSYSTEM



In ProteusPawn, remove references to SteamVR



Here: remove SteamVR Chaperone



Here: remove GetValidTrackedDeviceIds



DEVELOPING FOR OCULUS GO AND QUEST

We recommend to do the following steps:

- 1) Package for Windows-64 bits
- 2) On the Oculus Developer Dashboard
 - a. Create a Rift App
 - b. Copy/ Paste App ID into GameEngine.ini RiftAppID=
 - c. Upload package (new binary) to a channel (not Store Channel) via browser or Unreal Oculus Windows Platform Tool
 - d. Set the Matchmaking Mode in the Developer Dashboard to:

View/Edit Matchmaking Pool

TestAvatar -

Pool Key?

Proteus1

Should Pool Manage Rooms?

☒ Yes

☐ No

Who can create rooms?

☒ Users can create rooms

☒ System can create rooms

Can unmatched users join matchmaking rooms?

☒ Yes

☐ No

Can the matchmaking service keep matching into the same room?

☒ Yes

☐ No

Enable host migration when room owner leaves the room?

☒ Yes

☐ No

Users per Match?

Min Users

2

Min Preferred Users

2

Max Preferred Users

2

Max Users

2

Skill Component?

None

Should Consider Ping Time?

☐ Yes

☒ No

[Show Advanced Tuning...](#)

Cancel

Save & Deploy



- 3) Create a keystore for Android
- 4) In Unreal, VR Preview, Stop; in output log, copy AvatarID
- 5) {QUEST ONLY} In MainMenuPC/Settings, paste it in Oculus Quest Dev Avatar ID
- 6) Ensure settings are correct for GearVR/Go in ProjectSettings, set keystore infos
- 7) Package for SHIPPING for Go/Gear VR, Android ASTC
- 8) On the Oculus Developer Dashboard
 - a. Create a GearVR/Go App
 - b. Copy/ Paste App ID into GameEngine.ini GearVRAppID=
 - c. Upload package (new binary) to a channel (not Store Channel) via Unreal Oculus Platform Tool
 - d. Each of your apps will currently have a different app-scoped ID for a given user. In order to enable them to co-mingle, you need to first set up an app grouping (see documentation). This allows both apps to be speaking in terms of the same user IDs. Group together Rift App and Go/GearVR App via App Grouping
- 9) {QUEST ONLY} Ensure settings are correct for Quest in ProjectSettings
- 10){QUEST ONLY if no access to Quest API} In MainMenuPC/Settings, check Use Oculus Avatars with Quest without Quest API

Following this, you'll be able to use Oculus Avatars with all Oculus headsets, and cross-platform with Windows and Mobile headsets.

Developer without access to Quest API cannot use Oculus platform features, including multiplayer.

How can I access Quest API?

Your game pitch has to be greenlighted by Oculus before accessing Quest API. Business/Educational users will be able to access much of Oculus platform features via the Go and Quest Enterprise Edition software suite, available in Q4 2019. More infos at <https://www.oculusforbusiness.com/>.

Custom Hand Poses

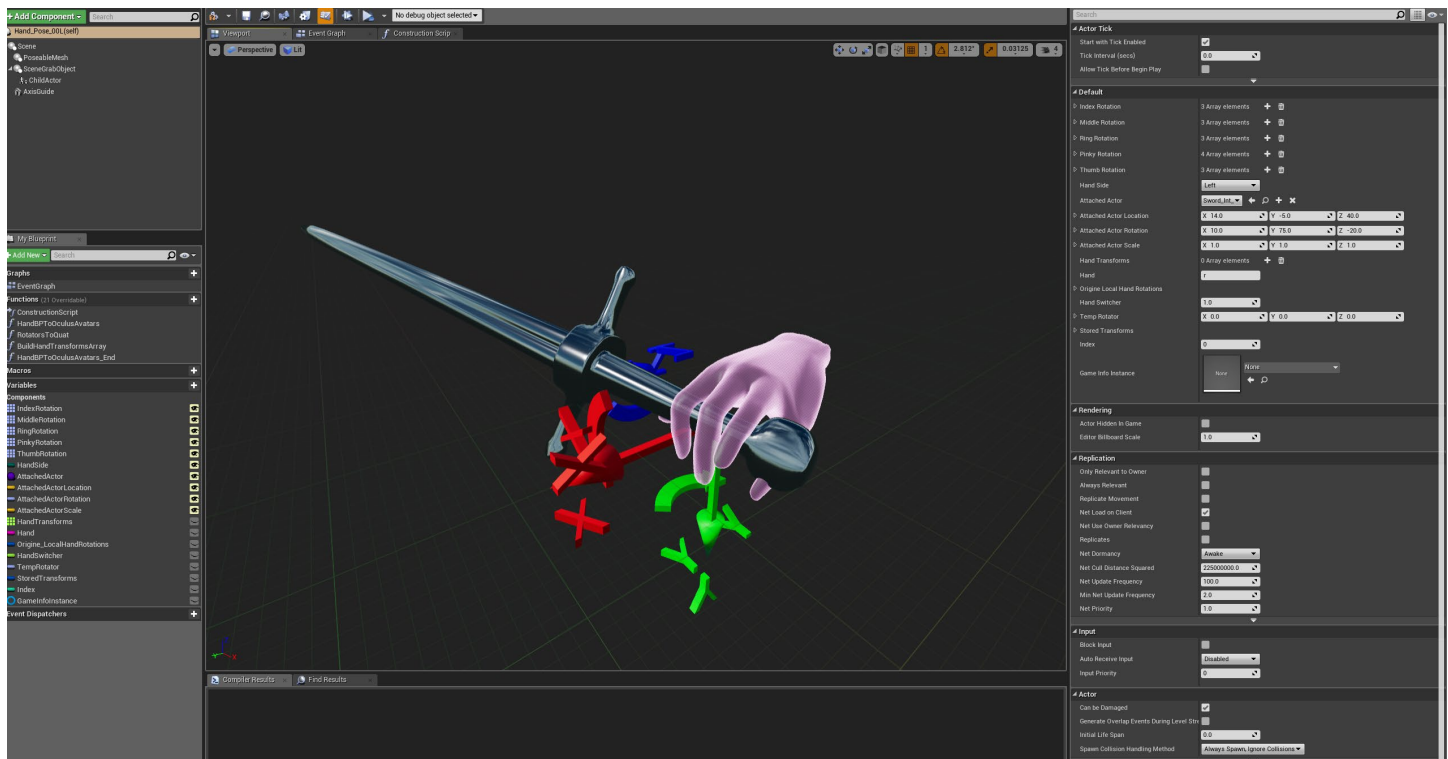
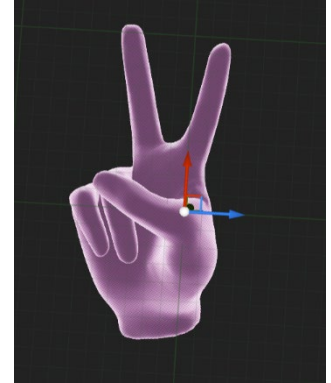
This works with both off-line (custom) and on-line (Oculus) Avatars.

Here's how to do it:

In Proteus Multi/Hand Poses, duplicate BlankPose actor blueprint, and make 2 copies for each pose (left and right hand)

I suggest to rename them left hand with even numbers, right hand with odd numbers, starting at 0. This will be also their index.

Inside HandPose:



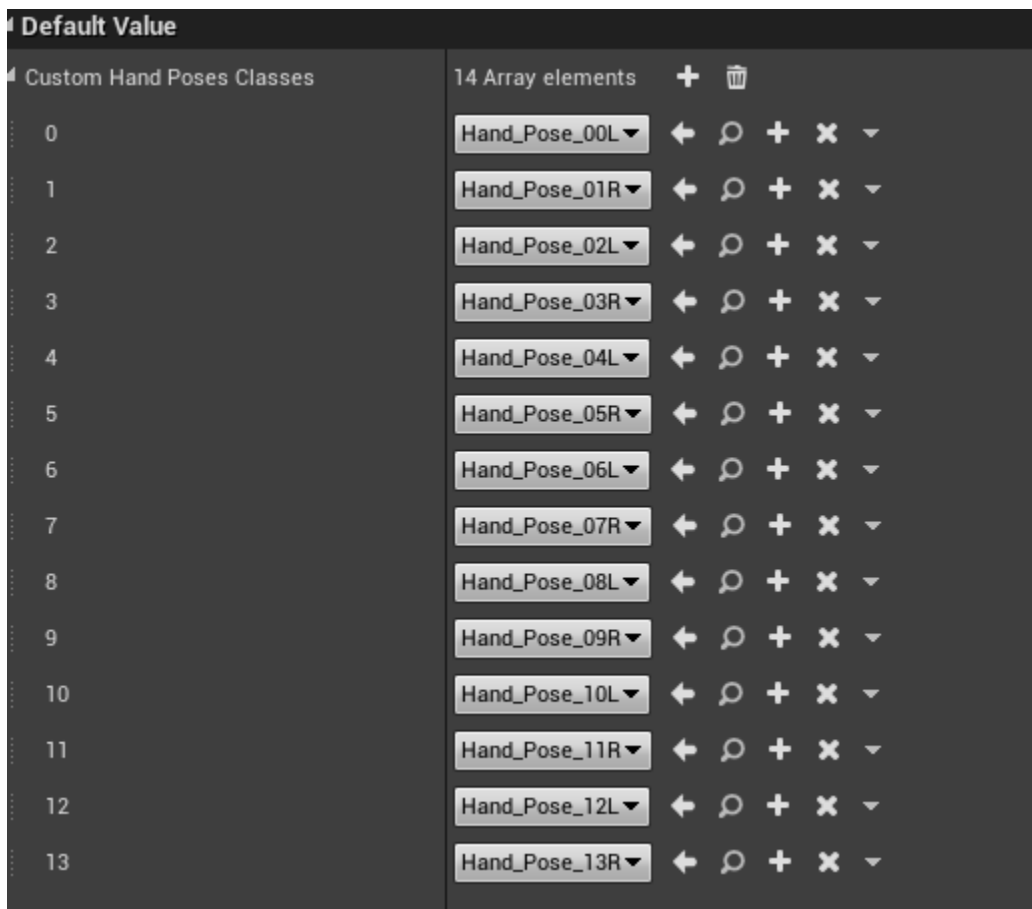
Change hand (l/r), fingers rotations, object reference (if any) to grab, object placement, object rotation. Do that for each hand.

You should only change these variables:

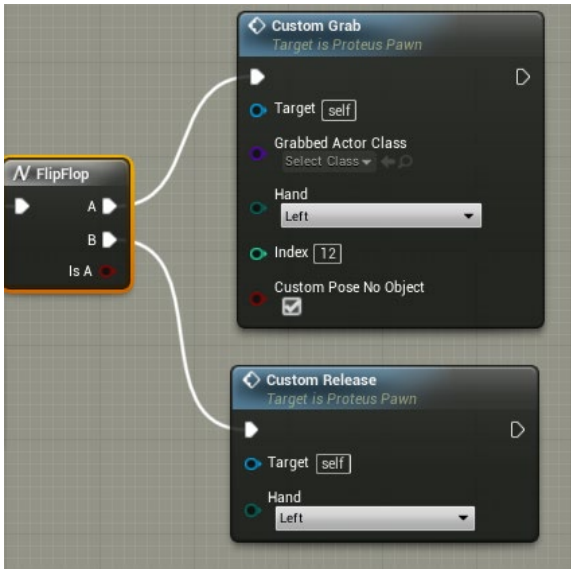
Index Rotation	3 Array elements	+	🗑️
▷ 0	X 10.0	Y 10.0	Z 0.0
▷ 1	X 40.0	Y 0.0	Z 0.0
▷ 2	X 0.0	Y 0.0	Z 0.0
Middle Rotation	3 Array elements	+	🗑️
▷ 0	X 10.0	Y 0.0	Z 0.0
▷ 1	X 20.0	Y 0.0	Z 0.0
▷ 2	X 10.0	Y 0.0	Z 0.0
Ring Rotation	3 Array elements	+	🗑️
▷ 0	X -10.0	Y 5.0	Z 0.0
▷ 1	X 0.0	Y 0.0	Z 0.0
▷ 2	X 10.0	Y 0.0	Z 0.0
Pinky Rotation	4 Array elements	+	🗑️
▷ 0	X 30.0	Y 0.0	Z 0.0
▷ 1	X -40.0	Y 40.0	Z 0.0
▷ 2	X 20.0	Y 0.0	Z 0.0
▷ 3	X 0.0	Y 0.0	Z 0.0
Thumb Rotation	3 Array elements	+	🗑️
▷ 0	X 0.0	Y 30.0	Z -20.0
▷ 1	X 0.0	Y 0.0	Z 0.0
▷ 2	X 0.0	Y 0.0	Z 0.0
Hand Side	Left		
Attached Actor	Sword_Int_ ◀ 🔍 + ✕		
▷ Attached Actor Location	X 14.0	Y -5.0	Z 40.0
▷ Attached Actor Rotation	X 10.0	Y 75.0	Z -20.0
▷ Attached Actor Scale	X 1.0	Y 1.0	Z 1.0

Set Attached Actor Scale at the same scale as your desired attachment actor to place the pose correctly

When satisfied, add the poses inside the array in MainMenuPC/HandPoses. Place them in order of their index number (even-left, followed by odd-right, etc).



Finally, if you created handposes without an object reference (again one for each hand), you can assign them by their index number to an input in ProteusAvatar/Inputs. As an example, we put a peace sign pose at index 12 for left, 13 for right. Don't forget to check "Custom Pose No Object". That's it.

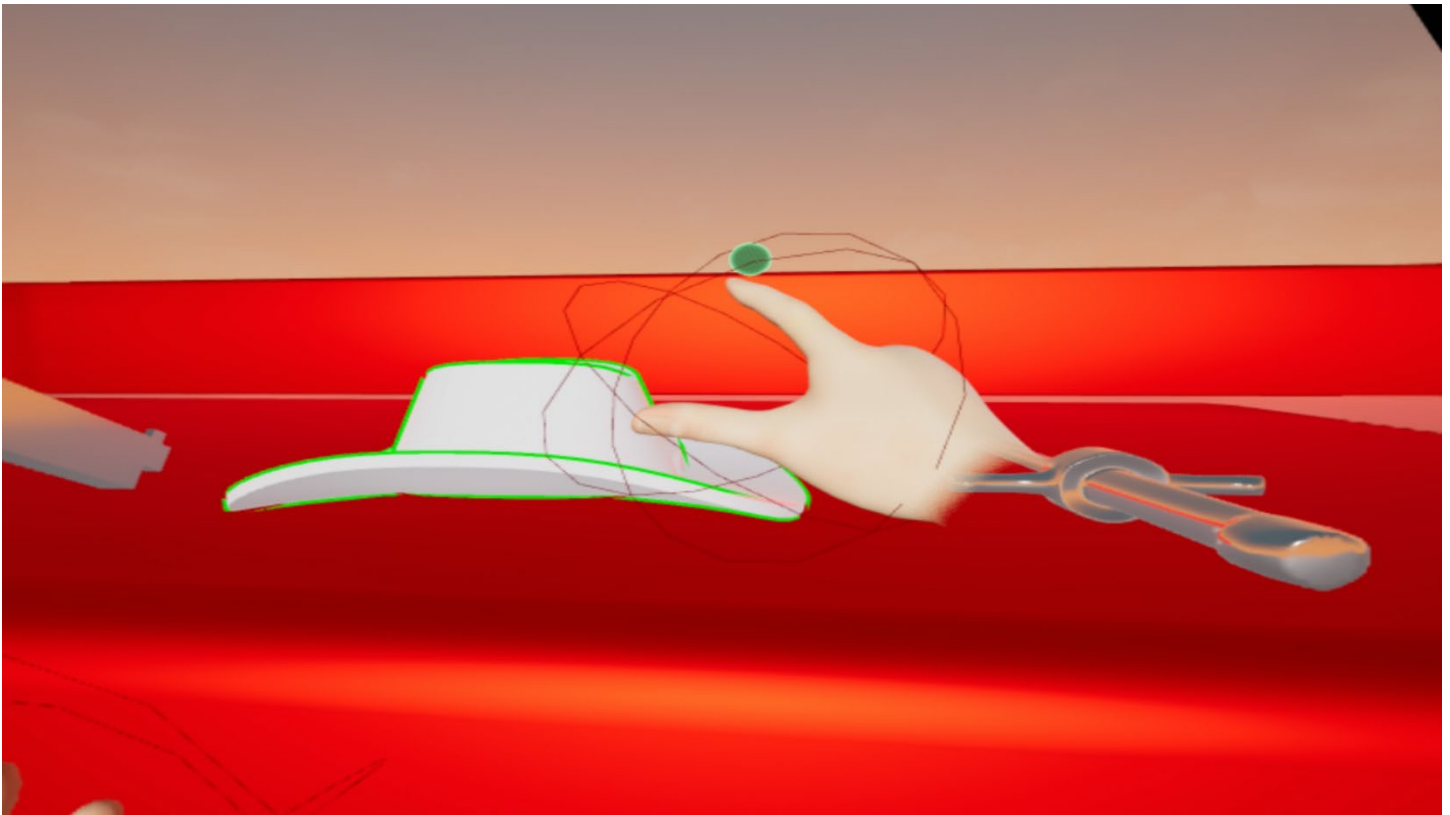


Highlighting objects

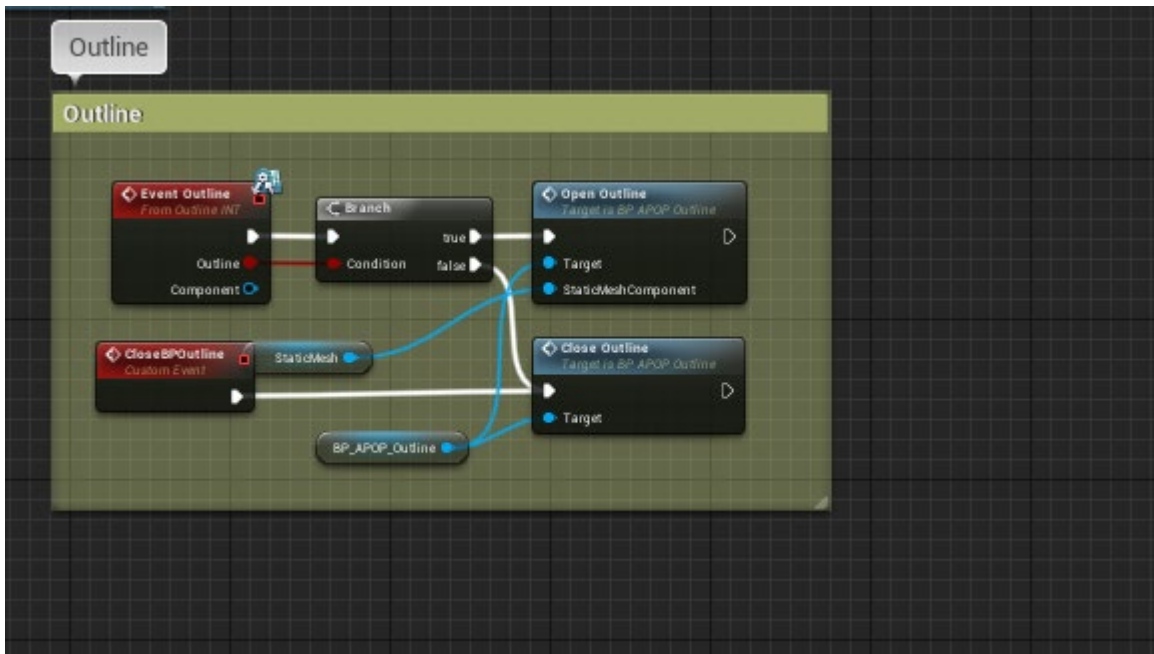
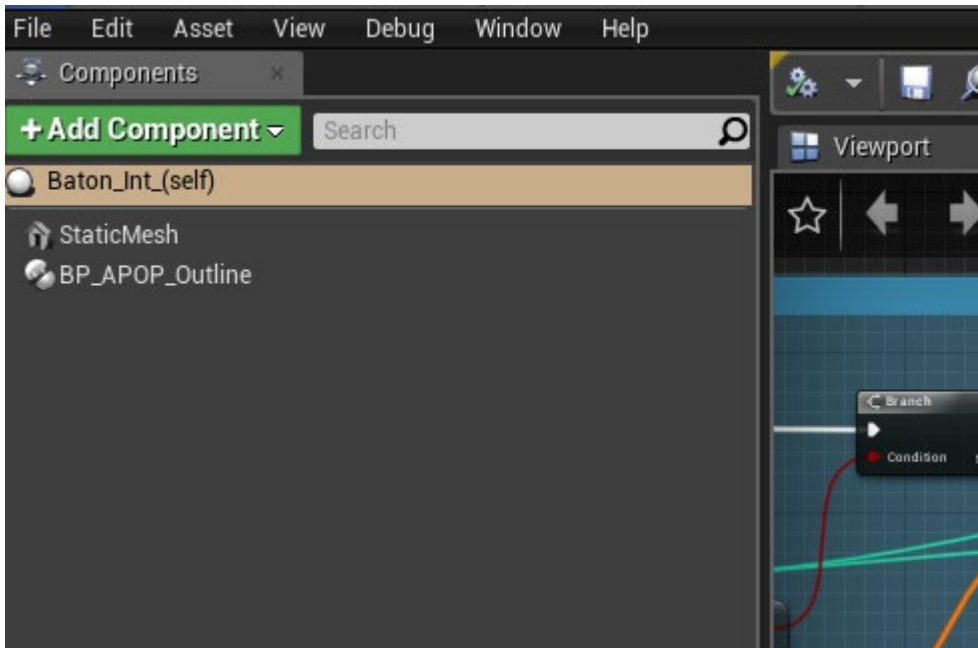
In previous versions, we used a post-process material to highlight selected objects. However, this effect is not compatible with mobile VR (Quest, Go).

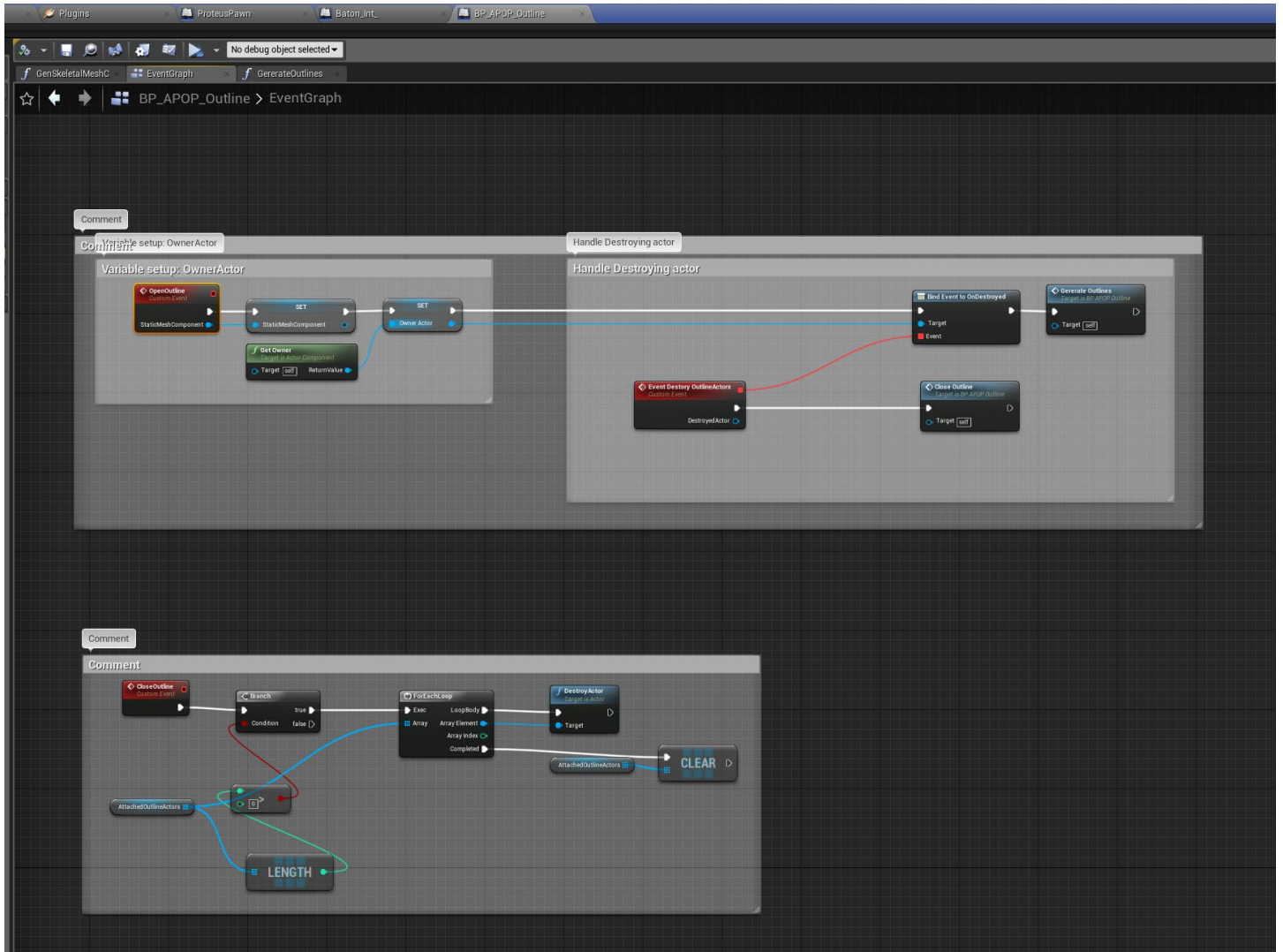
To solve this, we adopted the effects from the marketplace, "All Platform Outline Pack", at <https://www.unrealengine.com/marketplace/en-US/slug/all-platform-outline-pack> . It works well in Windows and Android-based VR.

For obvious reasons we removed code related to this asset, but we recommend to acquire and put it inside the template.





Here's how we did it:






Parent Socket

None







Default


Line Material





MI_APOP_SS_Opaque

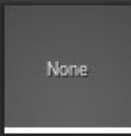
Line Dynamic Material





None

Static Mesh Component





None

Tags

Component Tags

0 Array elements

Component Replication

Component Replicated

☐

Activation

Auto Activate


☐


Cooking


Is Editor Only


☐

Events

 On Component



 On Component



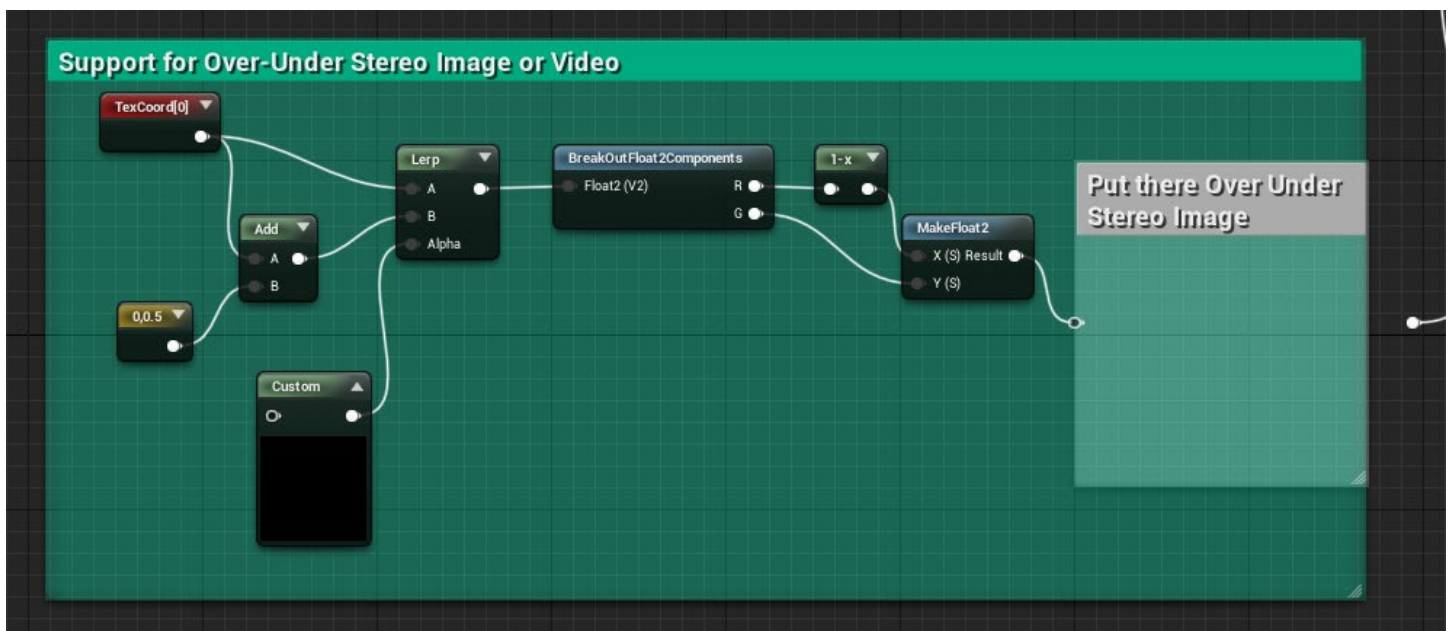
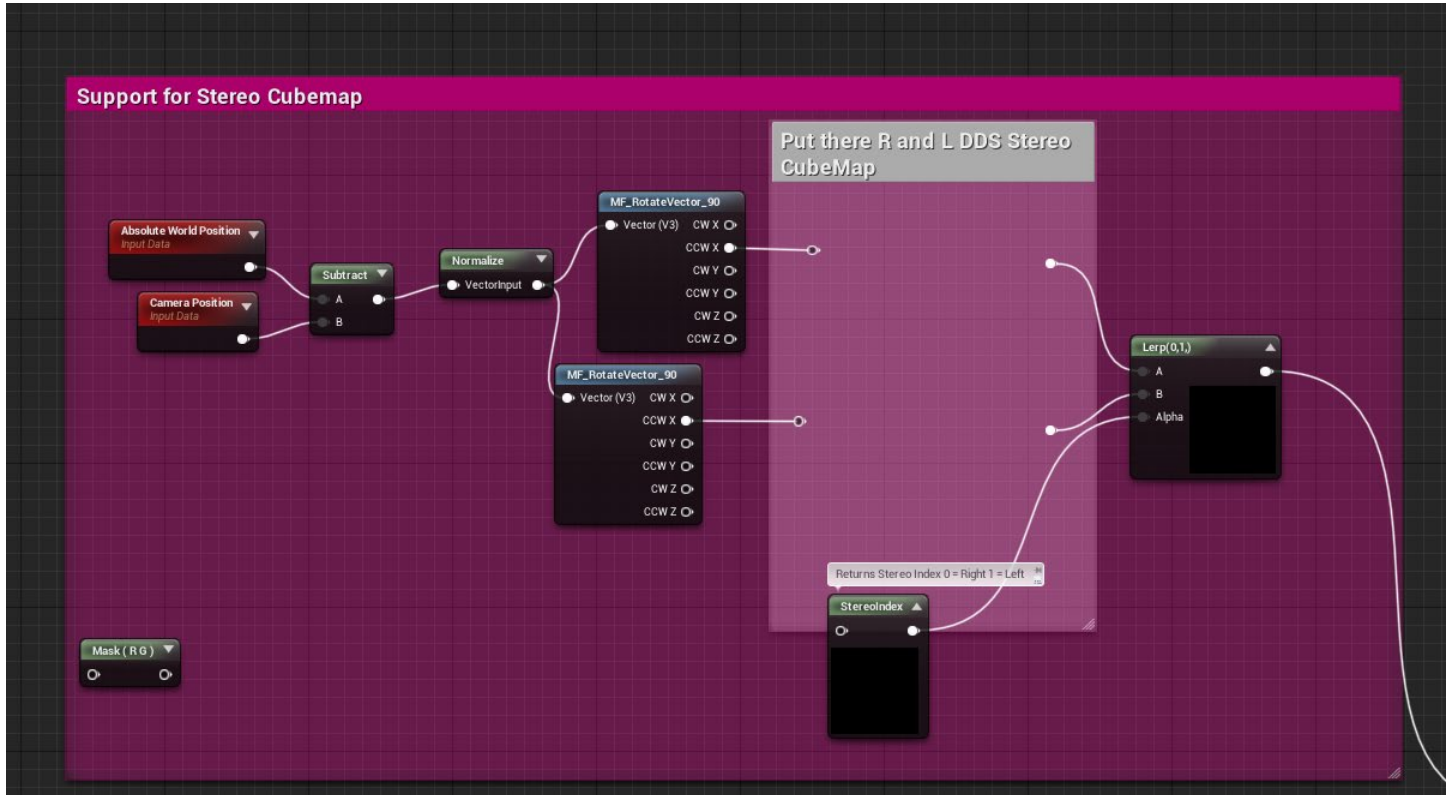
Asset User Data

Collision

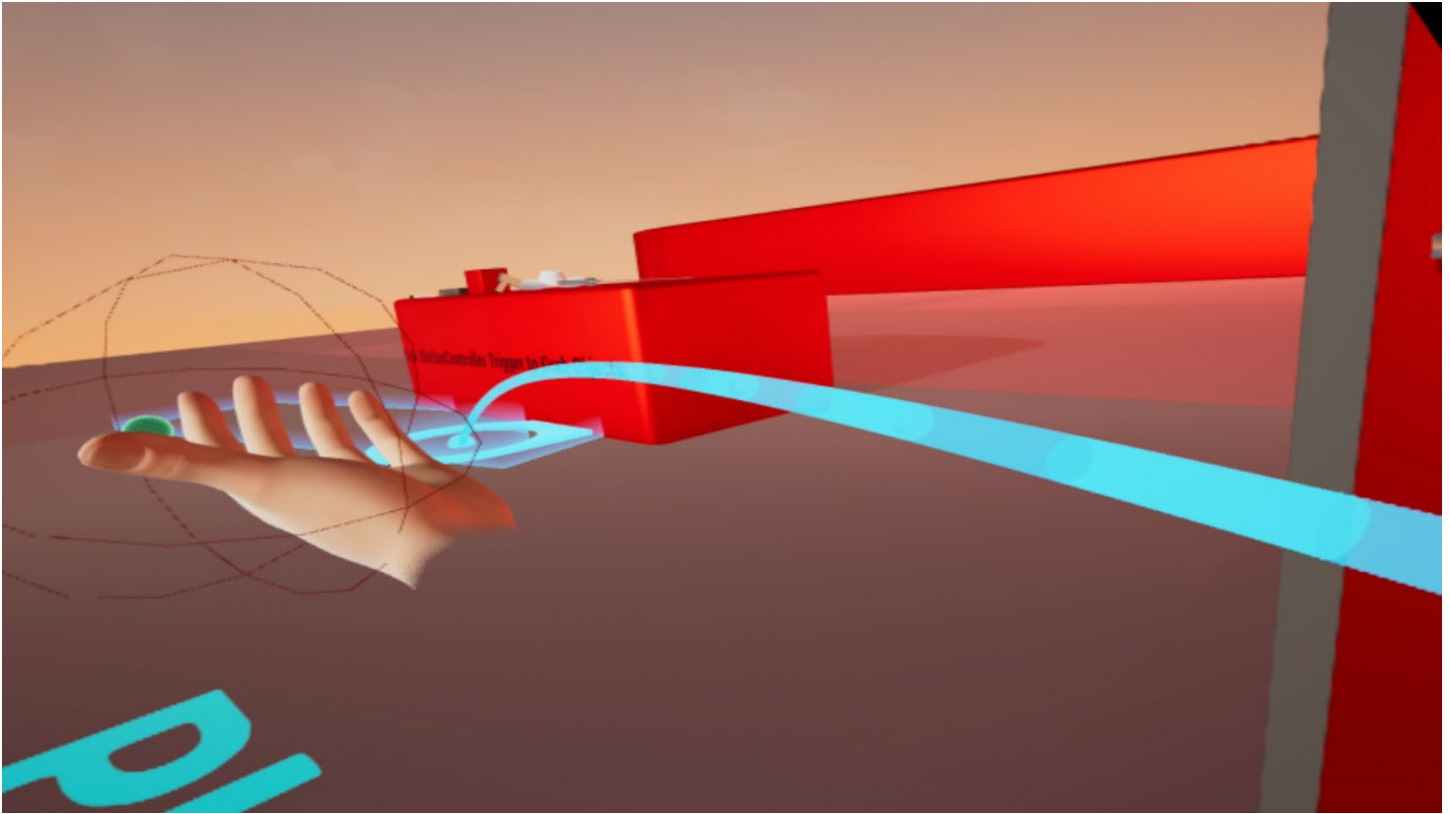
Playing Stereo CubeMap or Over-Under Images Video or Images

Note: Put videos in folder /Movies

Check SphereMat material for more details



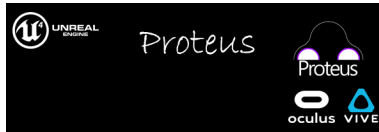
Teleporting



Pawns are able to teleport on surfaces when these 2 conditions are met:

- Has a NavMeshBoundsVolume;
- Is below the teleportation angle limit (see VR settings) – put 90 to this setting to disable it

IMPORTANT: Enable Project Settings / Engine / Navigation System / Allow Client Side Navigation for multiplayer teleportation; if not selected, clients won't be able to detect the Navigation Mesh.



REPORTED ISSUES AND BUGS (UNREAL 4.22.2)

STATUS

PATCHED: A CUSTOM SOLUTION EXISTS

FIXED: FIXED IN NEXT UNREAL VERSION

NOT FIXED: NO KNOWN SOLUTIONS

1. Retrieve Oculus ID and Verify Entitlement fail on Go/Quest

a. STATUS: PATCHED

b. Reference: <https://developer.oculus.com/bugs/bug/2343978258981741/>

c. See "I want to develop for Oculus Quest and/or Go "

2. Entitlement fail on Quest if no access to Quest API

a. STATUS: PATCHED

b. See "I want to use Oculus Avatars on the Quest, but I don't have access to Quest Platform API "

3. PlayDynamicForceFeedback has errors when nativizing assets

a. STATUS: PATCHED

b. Reference:

<https://answers.unrealengine.com/questions/831405/playdynamicforcefeedback-node-in-420-fails-to-cook.html>

c. See "I want to nativize assets without errors when compiling"

4. Oculus Go/Quest : No VOIP / or no LipSync with Oculus Avatars

a. STATUS: NOT FIXED

b. Reference: <https://developer.oculus.com/bugs/bug/471102320355137/>

c. Background infos: Android only allows access to the microphone from a single process. This wasn't an issue when networking avatars previously, as the mic input wasn't being used. But with the expressive update, we specifically need to run the mic through the OVRlipsync plugin to generate blend-shapes and drive the mouth shapes. Trying to hook up the mic to both VoIP and Lipsync therefore causes an inevitable race condition. The loser gets a bunch of zeros. So either there's no networked audio, or no blend-shapes.

5. Multiplayer Oculus Avatar is broken with latest Oculus Avatar Plugin

a. STATUS: PATCHED

b. Use included Avatar plugin

6. Oculus Rift / S: Enabling Splash in Project Settings / Oculus Settings crash when loading next level

a. STATUS: PATCHED

b. Use blueprints to Set and Show Splash Screen

7. SteamVR crashes if using Splash Screen

a. STATUS: FIXED IN 4.23

b. Reference: <https://issues.unrealengine.com/issue/UE-62486>

c. In 4.22, don't use Splash Screens with SteamVR

8. Stereo Layers are translucent in ES3.1 (Oculus Go/Quest) / Vulkan (Oculus Quest)

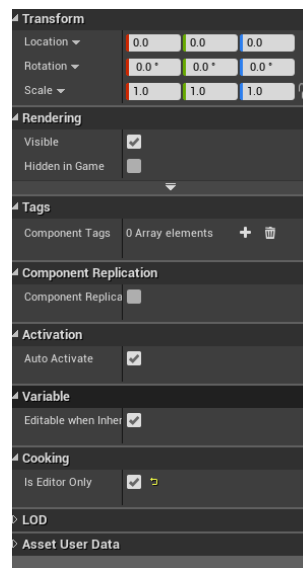
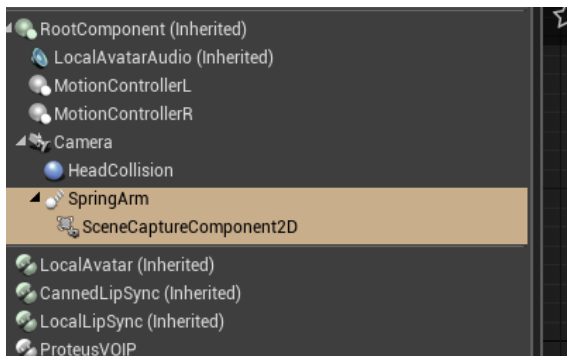
a. STATUS: NOT FIXED

9. Performance issues when a Render target is added to Spectator screen (VR) in a packaged game

a. STATUS: FIXED IN 4.23

b. Reference: <https://issues.unrealengine.com/issue/UE-70352>

c. IN 4.22, don't use Spectator Screens, or use them only in editor:



10. Cannot access Oculus Quest platform features

a. You have to be greenlighted by Oculus before doing so

b. Oculus Business users will have access to Business Suite Q4 2019

11. When a stereo layer is present rendering a widget it cannot be destroyed.

a. STATUS: NOT FIXED



b. Reference: <https://answers.unrealengine.com/questions/829084/stereo-layer-isnt-destroyed-on-end-play.html>

c. Avoid using stereo layers if you need to destroy them

12. 75Hz is not available with Oculus Quest, only 60 and 72Hz

a. STATUS: NOT FIXED

b. This bug has been introduced in Oculus plugin 1.39

13. Quest: Oculus Audio use 32-bits Android libraries (armeabi-v7a); no arm64 librairies available

a. STATUS: NOT FIXED

b. Use 32-bits libraries on Quest if using Oculus Audio



MULTIPLAYER ISSUES

What are the things possible?

- Oculus Network: Only possible with Oculus headsets
- Oculus Avatars: Only possible through Oculus Network
- To put the app on the Oculus Store, turn off Steam Subsystem and SteamVR plugins (and delete related functions)

Oculus Rift, Go, Quest : You don't use Stereo Layers?

- We are not using Stereo Layers, because they cannot be destroyed !!! (BUG)
- Stereo layers persist in HMD after VR Preview, BUG #UE-63210

Oculus Go / Quest

- Be gentle on the tick events, as you can easily crash the device
- Oculus Go cannot use Render Targets (i.e. the mirror)
 - That's life.



Q&A

Why did you not use the launcher version of UE4?

- 1) Because the launcher version is always behind the Oculus branch for using latest Oculus SDK implementations;
- 2) Because it's always better to use UE4 source-based, to be able to make some changes to the engine;
- 3) Because the Oculus-branch version has some nice features, i.e. the Project Settings Menus

I would like to log my users through Oculus Home, but I don't want to use Browse Matchmaking!

Browse matchmaking is the only way to match players while using blueprints, as of Unreal 4.21. We'll bring more ways soon through code (alternatively, there are C++ methods posted in the forums doing that).

My project doesn't work / crashes / doesn't compile! Your template is crap!

There can be a zillion reasons why your project doesn't compile / crashes. Are you using the latest version of Visual Studio? Correct UE4 version? Windows 10, updated? Is it cold outside? etc. We just want to be sure it's not our template that is crashing your game. Passed that, we're of limited help since, as I said, there could be a zillion reasons why your game does not work.

Other settings

- Follow latest guidelines for Rift / Go / Quest published on both Unreal and Oculus websites
- Try to avoid any other materials than opaque and masked
- Avoid fancy collision boxes
- Eliminate / reduce to minimum dynamic lights and shadows
- Avoid meshes with high poly count
- Reduce the number of animated objects

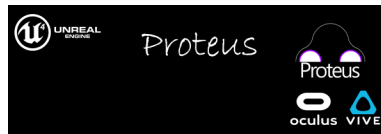


Supplemental resources

- UE4 Forum/VR development: <https://forums.unrealengine.com/forumdisplay.php?27-VR-Development>
- UE4 Virtual Reality development: <https://docs.unrealengine.com/latest/INT/Platforms/VR/>
- UE4 Networking and Multiplayer: <https://docs.unrealengine.com/latest/INT/Gameplay/Networking/>
- Tom Looman getting started in VR: <http://www.tomlooman.com/getting-started-with-vr/>
- Cedrik Neukirchen UE4 Multiplayer Network Compendium: <http://cedric-neukirchen.net/2017/02/14/multiplayer-network-compendium/>
- Mitch McCaffrey Unreal Engine VR Cookbook: <http://ue4vrcookbook.com/>
- Oculus UE4 GitHub: <https://github.com/oculus-vr/unrealengine>
- Oculus UE4 Developer Guide: <https://developer3.oculus.com/documentation/game-engines/latest/concepts/book-unreal/>
- Vive Tracker for developers: <https://www.vive.com/ca/vive-tracker-for-developer/>
- SteamVR Developer Hardware: <https://steamcommunity.com/app/358720/discussions/>

Questions?

Don't hesitate to contact me mathieu.beaulieu@proteus-vr.com .



What are the license terms?

Assets created by Epic can be freely reused in any Unreal Engine 4 project.

Assets created by Proteus are under the MIT license terms.

MIT licence terms (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Proteus