

The logo features the AMD logo (a stylized 'A' with a square) in white, followed by 'RADEON PRO' in a smaller, spaced-out white font, and 'Software' in a large, white, sans-serif font. The background is a dark blue, abstract, leaf-like pattern.

AMD
RADEON PRO
Software

AMD Radeon ProRender Game Engine Importer for Unreal Engine

User Guide for CAD to VR

This guide offers step-by-step directions for the importing of CAD models, exported using AMD Radeon™ ProRender, into Unreal® Engine.

DISCLAIMER

The information contained herein is for informational purposes only and is subject to change without notice. While every precaution has been taken in the preparation of this document, it may contain technical inaccuracies, omissions, and typographical errors, and AMD is under no obligation to update or otherwise correct this information. Advanced Micro Devices, Inc. makes no representations or warranties with respect to the accuracy or completeness of the contents of this document, and assumes no liability of any kind, including the implied warranties of non-infringement, merchantability or fitness for particular purposes, with respect to the operation or use of AMD hardware, software or other products described herein. No license, including implied or arising by estoppel, to any intellectual property rights is granted by this document. Terms and limitations applicable to the purchase or use of AMD's products are as set forth in a signed agreement between the parties or in AMD's Standard Terms and Conditions of Sale.

©2018 Advanced Micro Devices, Inc. All rights reserved. AMD, the AMD arrow, AMD FirePro, AMD Radeon Pro, AMD Radeon ProRender and combinations thereof are trademarks of Advanced Micro Devices, Inc. in the United States and/or other jurisdictions. Unreal Engine is a trademark or registered trademark of Epic Games, Inc. in the United States and elsewhere. HTC, VIVE and the VIVE logo are trademarks or registered trademarks of HTC Corporation in the United States and elsewhere. Oculus, Oculus VR, Rift, and the Oculus logo are trademarks or registered trademarks of Oculus VR, LLC in the United States and elsewhere. Steam and the Steam logo are trademarks and/or registered trademarks of Valve Corporation in the United States and/or other countries. Other names are for informational purposes only and may be trademarks of their respective owners.

© 2004-2018, Epic Games, Inc. All rights reserved. Unreal and its logo are Epic's trademarks or registered trademarks in the US and elsewhere.

Table of Contents

Supported Platforms	1
Software.....	1
Operating System.....	1
Additional VR Requirements.....	1
Compiling Standalone Unreal Engine Applications.....	1
Step 1: Launch Unreal Engine VR Template	2
Step 2: Open Motion Controller Map	3
Step 3: Verify Plug-In Readiness	4
Step 4: Import AMD Radeon ProRender Scene	5
Step 5: Select Import Options	6
Other Import Options.....	7
Two Sided Materials	7
Scale Factor	7
Flip Windings	7
Import Lighting	7
Use Blueprint.....	7
Use Dynamic Physics.....	7
Convex decomposition	7
Step 6: Edit the Object’s Blueprint	9
Step 7: Set Physics & Gravity	10
Step 8: Compile the Blueprint	11
Step 9: Align Object in Coordinate Space	12
Step 10: Add Reflection Capture Object	13
Step 11: “Play” The Scene in VR	15
Advanced: Edits for “Grabbability”	17
Advanced: Building a Standalone VR App	0

Supported Platforms

AMD Radeon ProRender Game Engine Importer for Unreal Engine requires OpenCL™ 1.2 or higher.

Software

AMD Radeon ProRender Game Engine Importer for Unreal Engine can be used with any CAD software with AMD Radeon ProRender add-in/plug-in that supports the export of AMD Radeon ProRender scene files. (See [here](#) for AMD Radeon ProRender for SOLIDWORKS User Guide)

- Dassault Systèmes SOLIDWORKS® 2016 or higher (See [here](#) for updates about additional supported applications)
- Unreal Engine® 4.17 (Available [here](#))
- AMD Radeon™ ProRender Game Engine Importer for Unreal Engine (Available [here](#))

Operating System

AMD Radeon ProRender Game Engine Importer for Unreal Engine is designed to support the following Microsoft Windows® platforms:

- Microsoft Windows® 10 (64-bit)

Additional VR Requirements

For professional graphics card users working with CAD, VR, and GPU accelerated rendering, AMD suggests the Radeon™ Pro WX 7100, and recommends the Radeon™ Pro WX 9100. See [here](#) for more information.

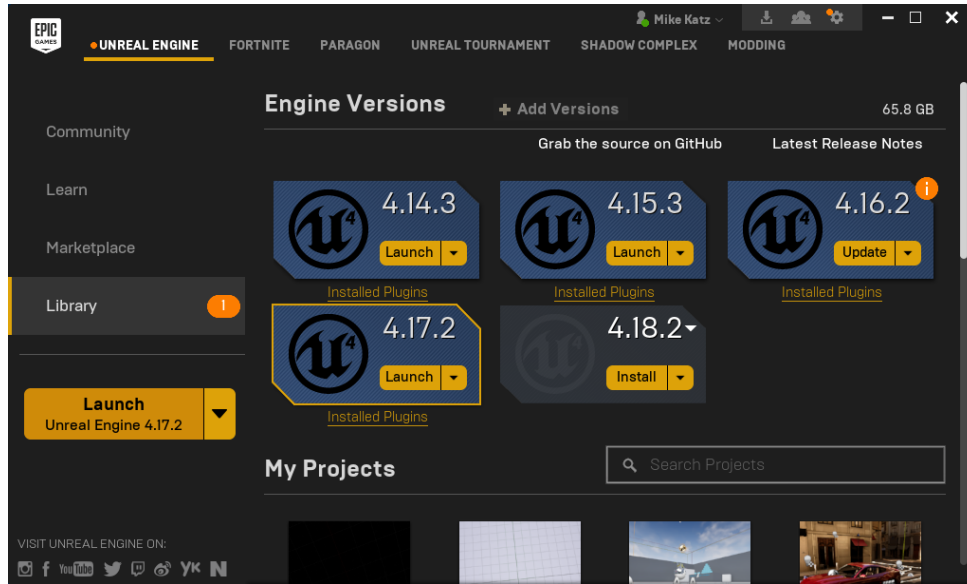
- PC support for SteamVR™ can be determined using this [tool](#).
- PC support for Oculus Rift can be determined using information and tools on this [site](#).
- VR headset and associated software supported by Unreal Engine – HTC VIVE™ or Oculus Rift.

Compiling Standalone Unreal Engine Applications

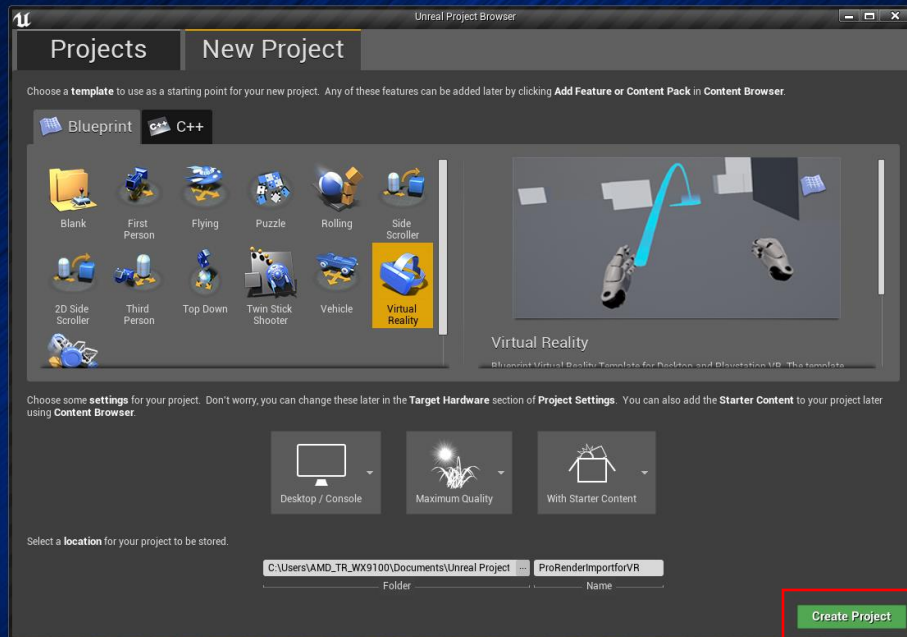
Install Microsoft Visual Studio® 2017 Community Edition – a free download available from [here](#).

Step 1: Launch Unreal Engine VR Template

Launch Unreal Engine 4.17 from the Epic Games Launcher.



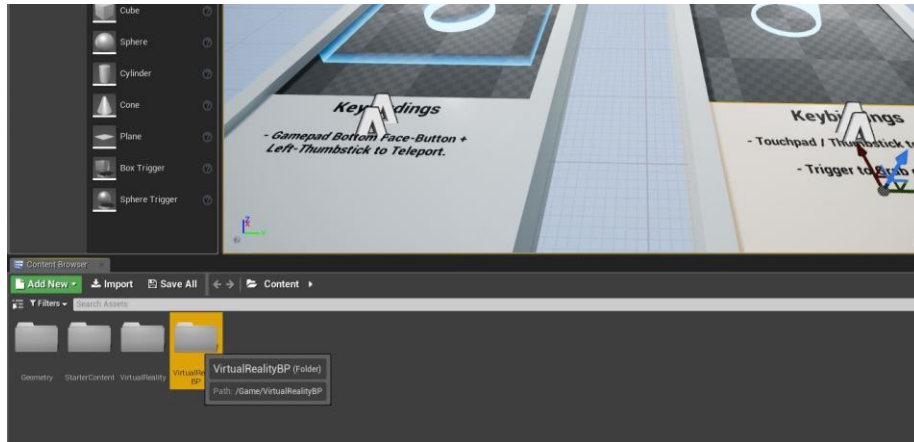
Select the "New Project" tab and choose the "Virtual Reality" Blueprint template and the "With Starter Content" option selected. By clicking "Create Project" you will now enter the Unreal Engine Level Editor.



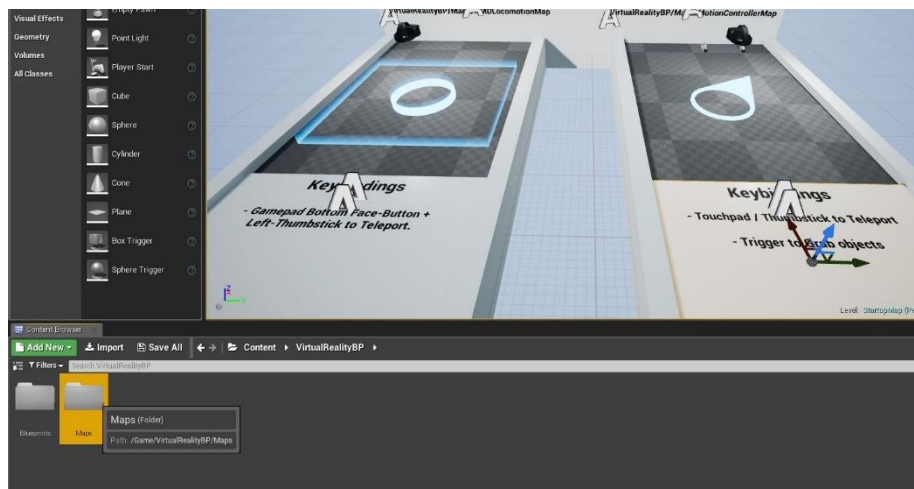
Creates your new project in the specified location with the specified template and name.

Step 2: Open Motion Controller Map

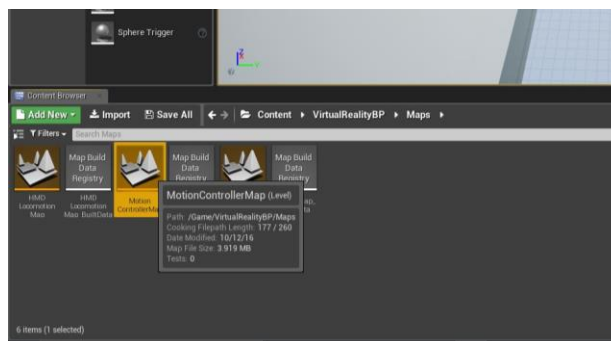
Using the Content Browser in the lower left corner, double click in on the “VirtualRealityBP” folder...



...then double click in on the “Maps” folder...

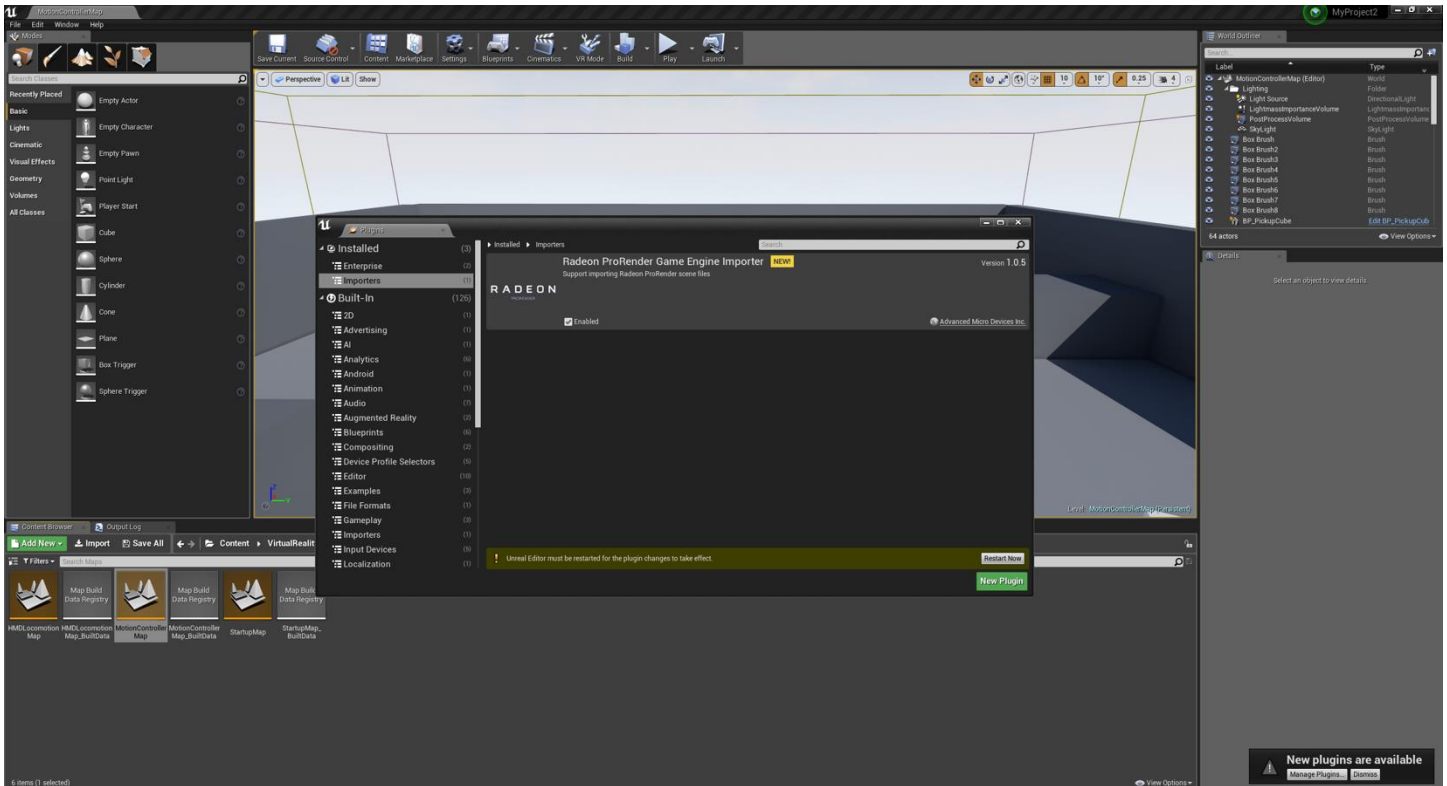


...then select the “MotionControllerMap”, which will open in the Level Editor window in the proper environment for using VR motion controls for teleportation and picking up and dropping objects in VR.



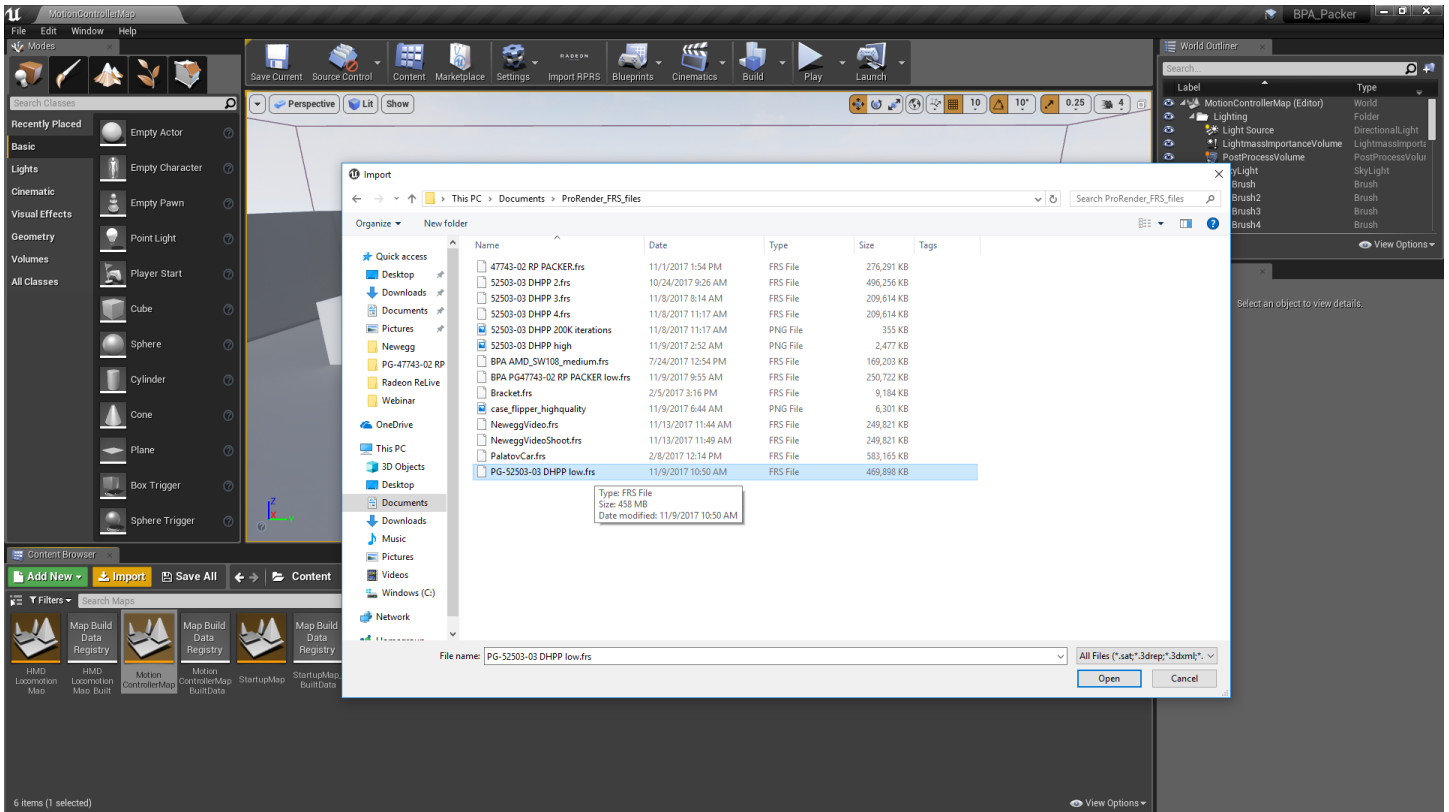
Step 3: Verify Plug-In Readiness

Verify that the AMD Radeon ProRender plug-in is installed and enabled by clicking “Edit...Plugins” from the Level Editor toolbar, then “Importers” from the list of plugin categories. “RprsImporter” should be installed and enabled (“Enabled” must be checked).



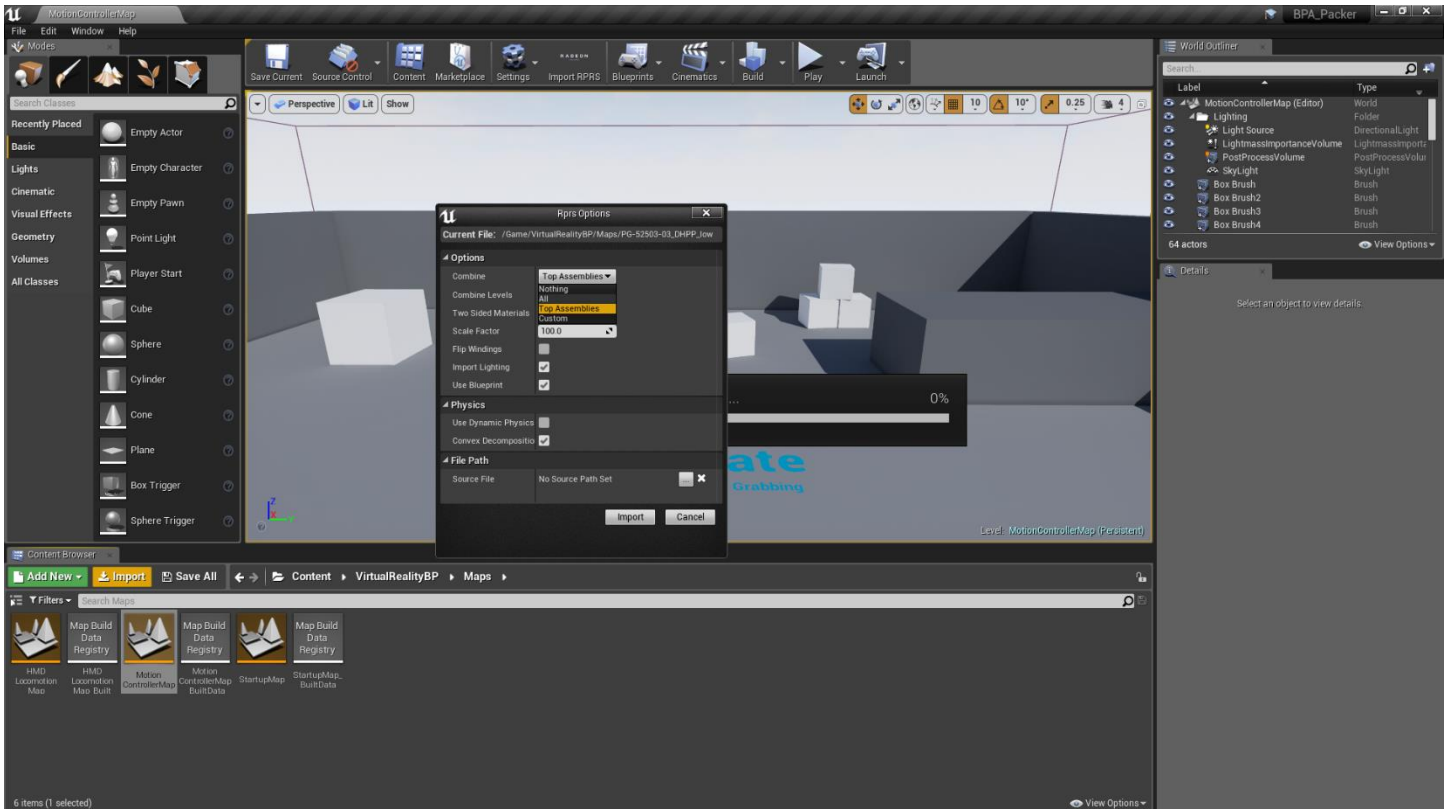
Step 4: Import AMD Radeon ProRender Scene

In the “Content Browser” pane in the lower left corner of the screen, click the “Import” button, and choose any AMD Radeon ProRender scene, which is a file with the “.frs” file extension. You may also use the “Import RPRS” button on the toolbar to invoke the same operation.



Step 5: Select Import Options

Choose the AMD Radeon ProRender import options via the dialog box. Selecting “All” for the “Combine” option will merge all the parts and assemblies of the CAD model to create a single object (aka a mesh actor) in Unreal Engine. Selecting “Top Assemblies” for the “Combine” option will import top level assemblies from the CAD model as distinct static meshes in Unreal Engine. Choosing the “Combine All” option is recommended for Unreal Engine beginners and offers the fastest path to VR visualization and manipulating your model as a single monolithic object.



Other Import Options

Two Sided Materials

Check this option if both sides of a CAD surface need to be viewed in Unreal Engine (i.e. exposed sheet metal). This option can also help in cases where the original model in SOLIDWORKS had problems with surface normals which can sometimes happen with parts that are imported as surfaces.

Scale Factor

The X, Y, and Z coordinate values of the CAD model are multiplied by this number. The default setting of 100 scales a SOLIDWORKS CAD object to align with the Unreal Engine coordinate space and is accurate to human scale in VR.

Flip Windings

This setting can be used in cases where all surface normal are inverted and may fix cases where, for example, reflections are incorrect.

Import Lighting

If checked, the lights defined in the AMD Radeon ProRender scene will be imported and translated into Unreal Engine lights. For most beginners, the default lighting in the Unreal Engine Virtual Reality template should be sufficient for VR viewing.

Use Blueprint

If checked, a Blueprint is created for the imported object. This is recommended, as Blueprints simplify the process of assigning gravity, physics, interactions, and other behavior and appearance properties of mesh actors.

Use Dynamic Physics

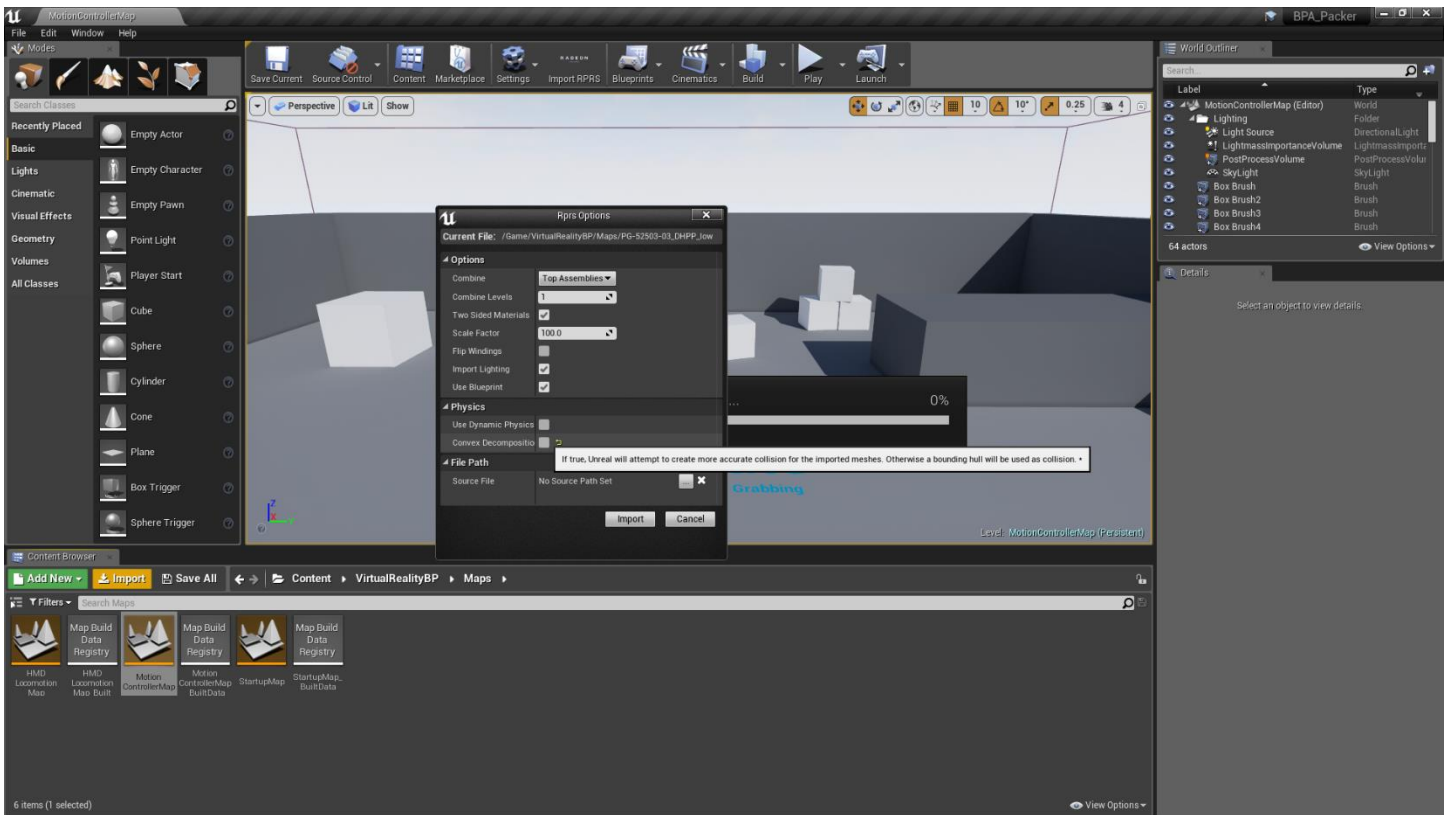
If checked, imported meshes will have physics properties enabled by default. Physics can also be enabled/disabled in the Blueprint for each mesh actor.

Convex decomposition

If checked, Unreal Engine will attempt to create a more precise collision zone that conforms to the geometry of the mesh. If unchecked, Unreal Engine will use a simpler bounding box to define the collision zone for the imported object. Convex decomposition is useful for surfaces that have openings that need to be traversed in VR (i.e. a door in a wall or a hollow opening in an object.)



Note: Convex decomposition requires additional computations during import but more importantly it can also impact runtime performance, so this option should be used sparingly.

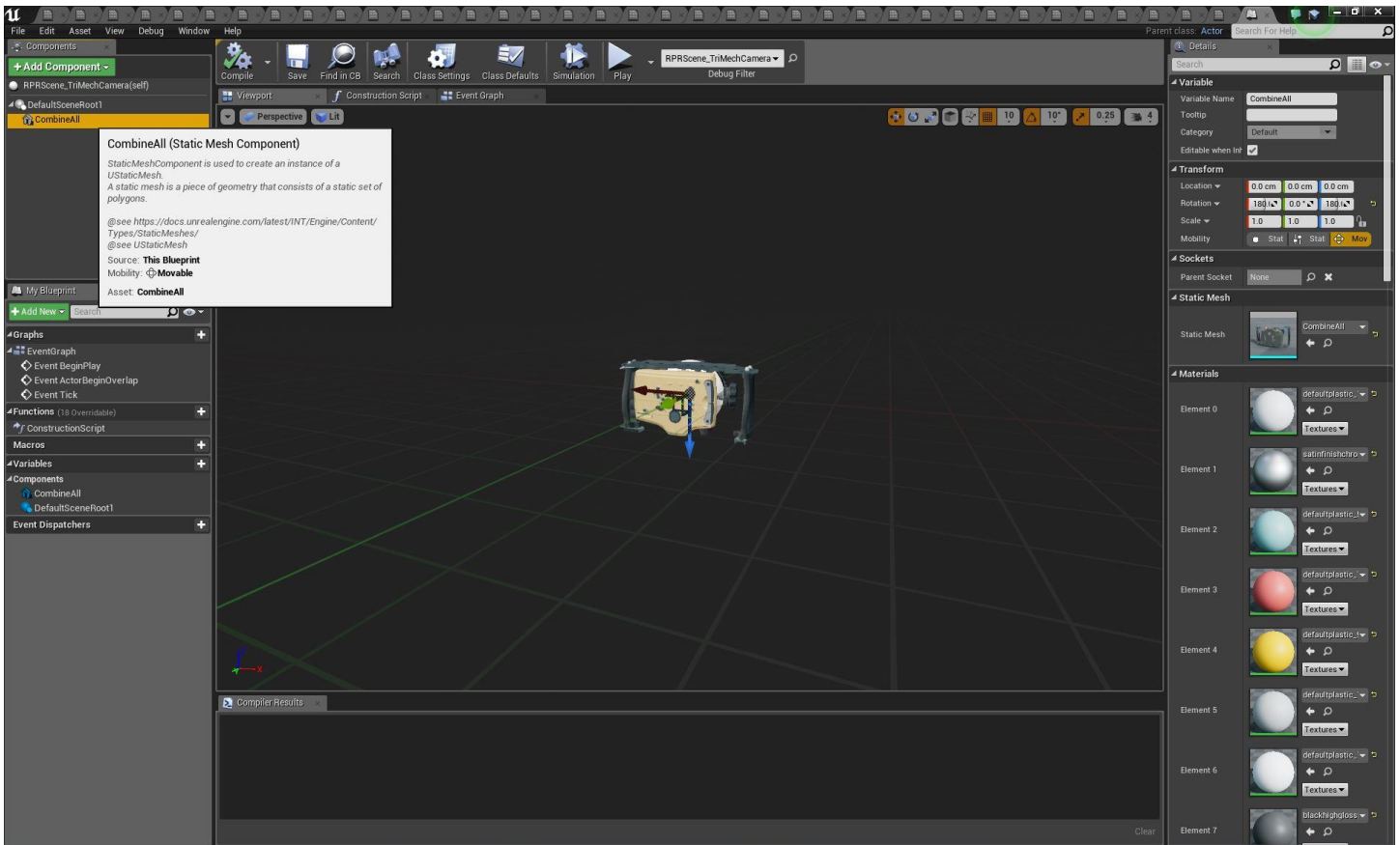


All attributes affecting material appearances (color, texture, refractivity, reflectivity, specular properties, etc.) assigned in the CAD application are automatically translated into Unreal Engine material shaders.

Factors that affect the duration of the AMD Radeon ProRender import process include the size and complexity of the geometry of the object, the number and complexity of materials contained in the object, and whether convex decomposition is enabled.

Step 6: Edit the Object's Blueprint

The imported AMD Radeon ProRender asset opens by default in Unreal Engine Blueprint Editor in a separate window on top of the main Level Editor window. The Blueprint is where a mesh actor's appearance, interactivity, and physical behavior attributes are defined. Select the "CombineAll" static mesh component in the Components pane in the upper left corner of the Blueprint Editor window to edit the attributes of your imported AMD Radeon ProRender object.

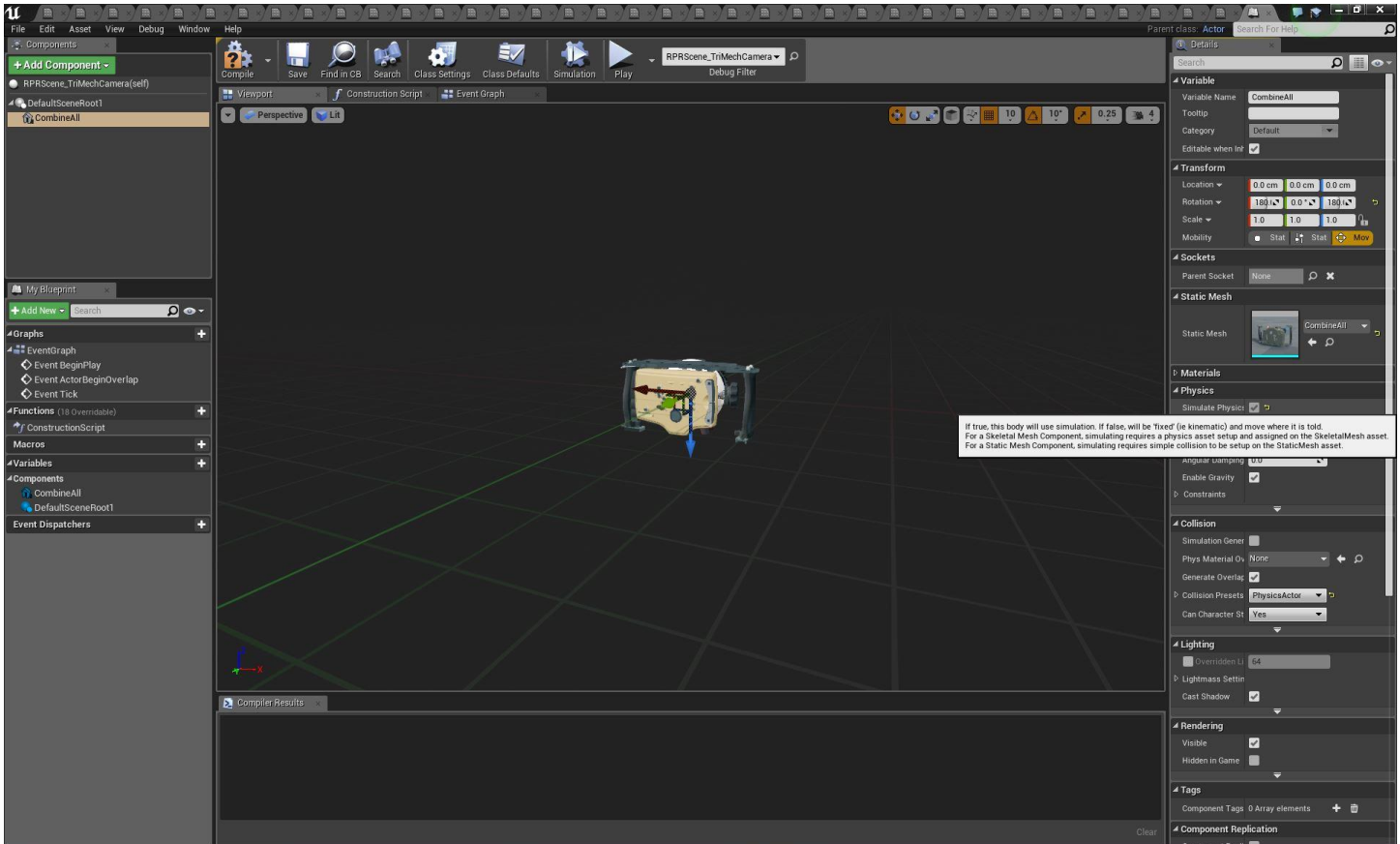


Step 7: Set Physics & Gravity

With “CombineAll” in the Components pane still highlighted, move to the Details pane on the right side, navigate to the Physics section (just below Materials) and check the “SimulatePhysics” option. (This box may already be checked if “Use Dynamic Physics” was enabled during import.)

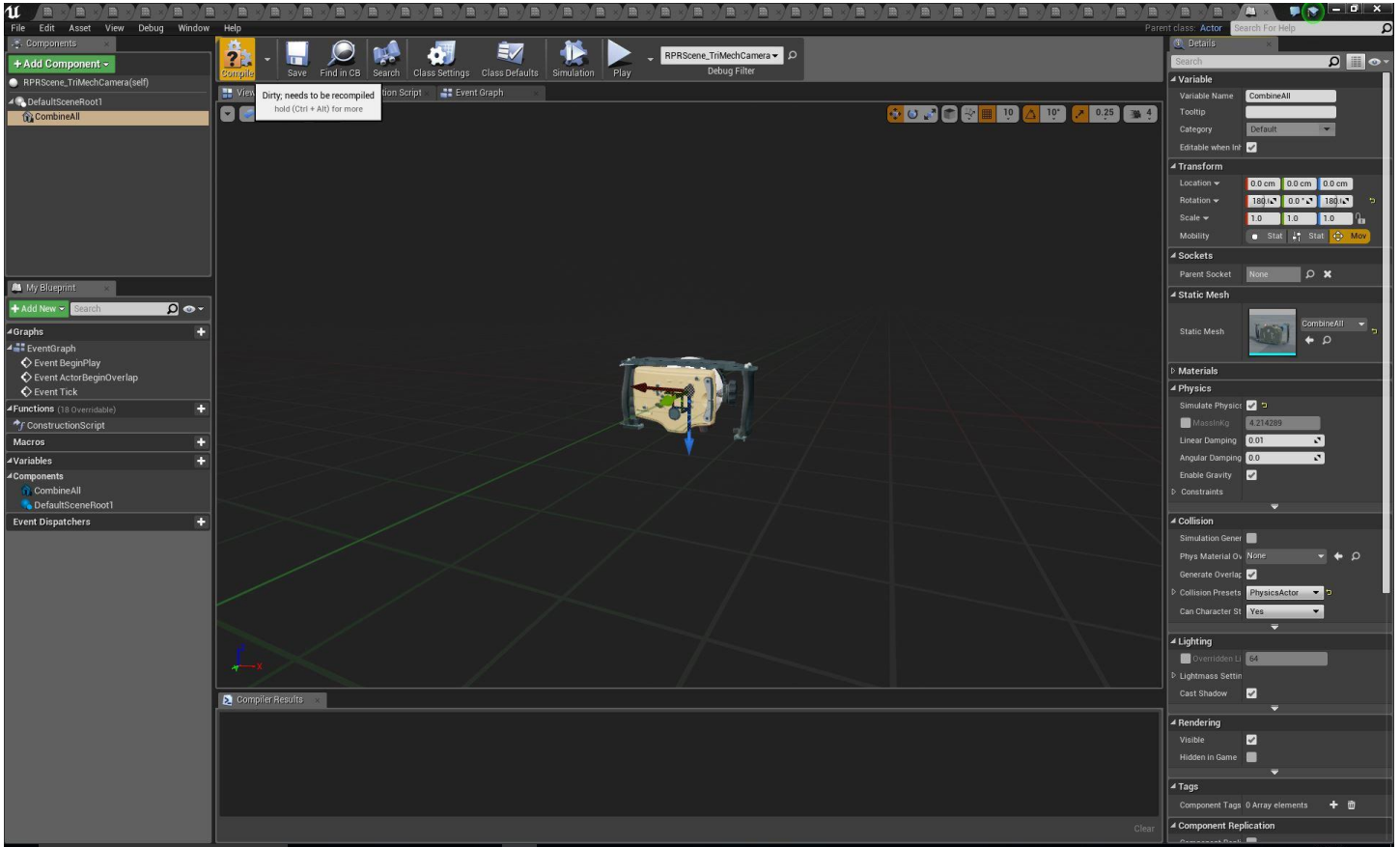


Note: Gravity can be toggled for the object.



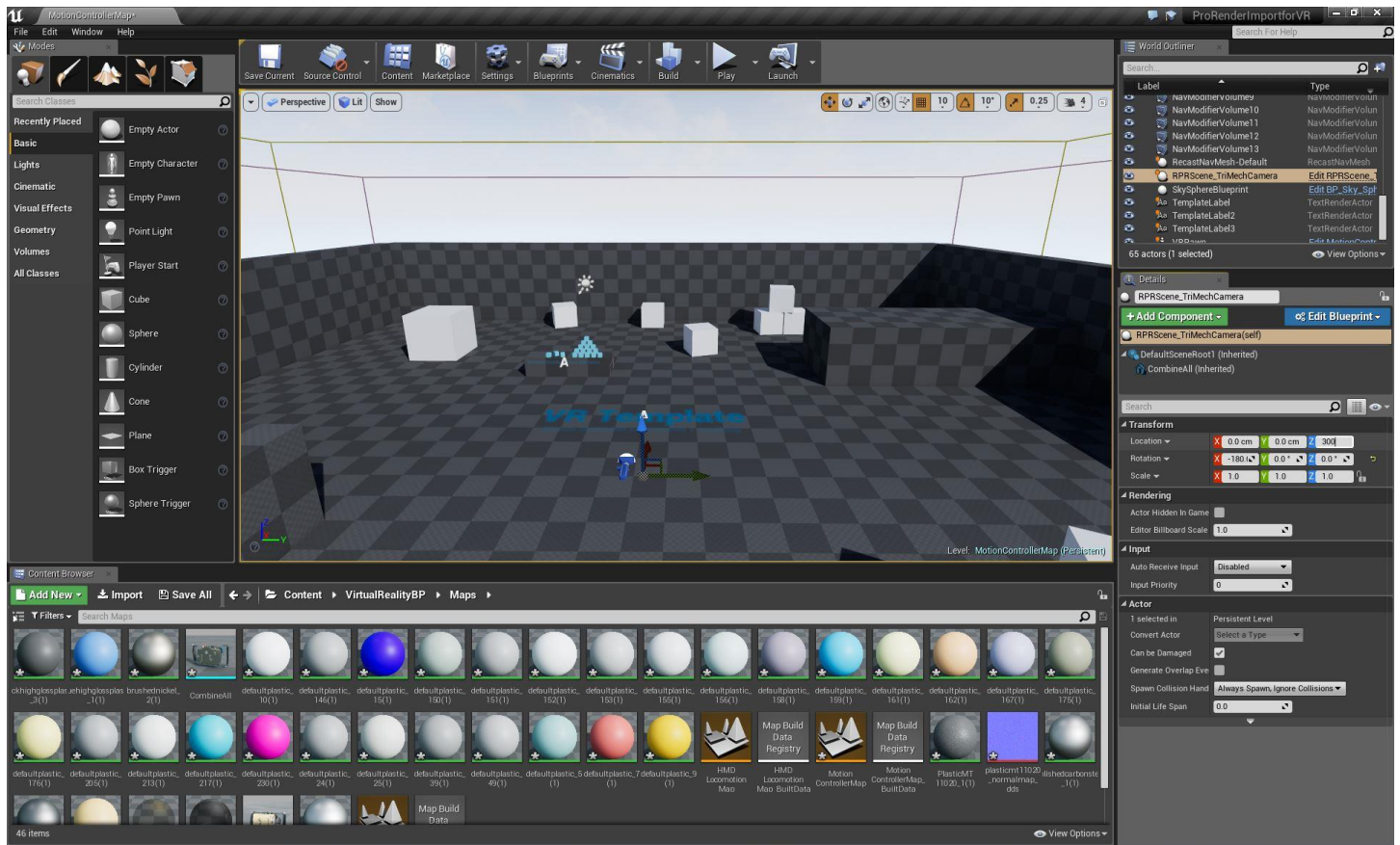
Step 8: Compile the Blueprint

By pressing the tool bar at the top of the Blueprint Editor, “Compile” the Blueprint. The icon on the Compile button will switch from a question mark to a **green check**.



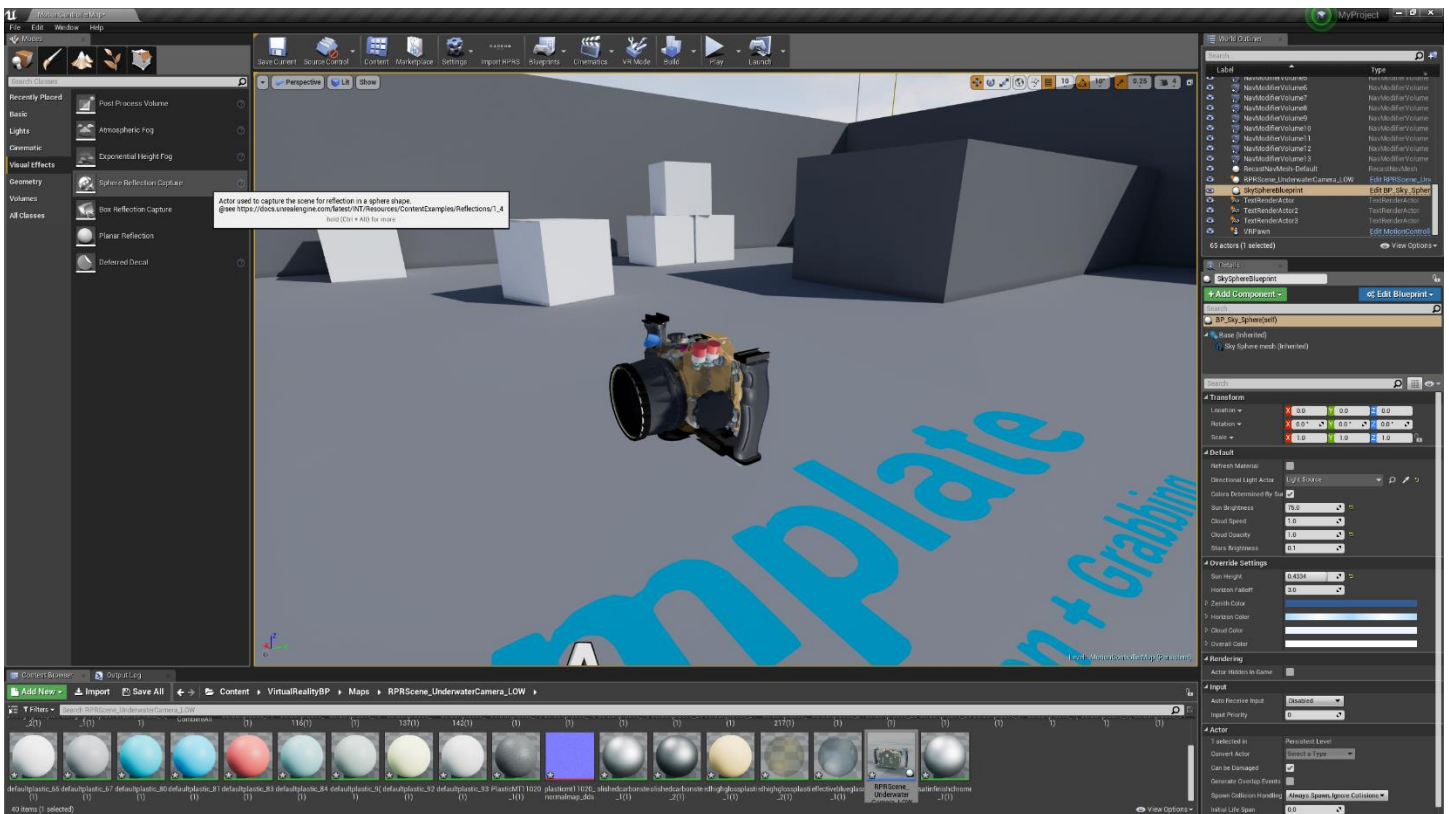
Step 9: Align Object in Coordinate Space

Minimize or close the Blueprint Editor and switch back to the Level Editor. The imported AMD Radeon ProRender asset will be located at the 0,0,0 position of the Unreal Engine coordinate plane, which places half of the object above the floor and half of it below (this will depend on the location of the floor in the scene). Raise the imported AMD Radeon ProRender asset above the Unreal Engine floor by selecting and dragging the blue arrow up, or by entering a value (in centimeters) in the Z location of the "Transform" section of the "Details" pane for the imported AMD Radeon ProRender object.

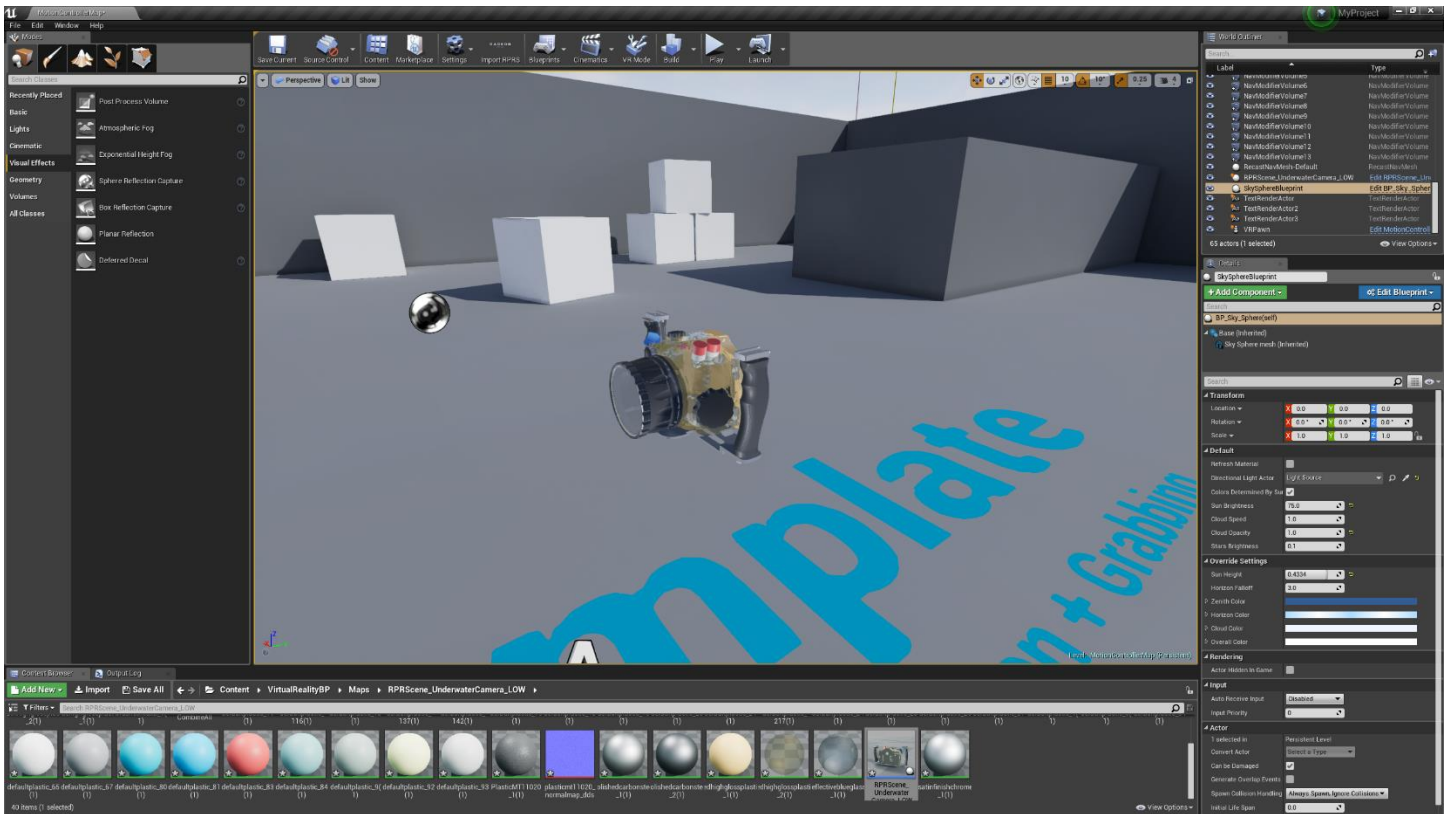


Step 10: Add Reflection Capture Object

The Unreal Engine Virtual Reality template has a default skylight defined, but a reflection map must be added to the scene to ensure that reflective materials interact correctly with Unreal Engine lights. This is especially important for not only glass and other glossy surfaces but also all metals; colors observed on metals come from reflections, even if the material has a very rough surface. In "Visual Effects", choose one of the reflection capture objects...



And drag the reflection capture object into the scene...



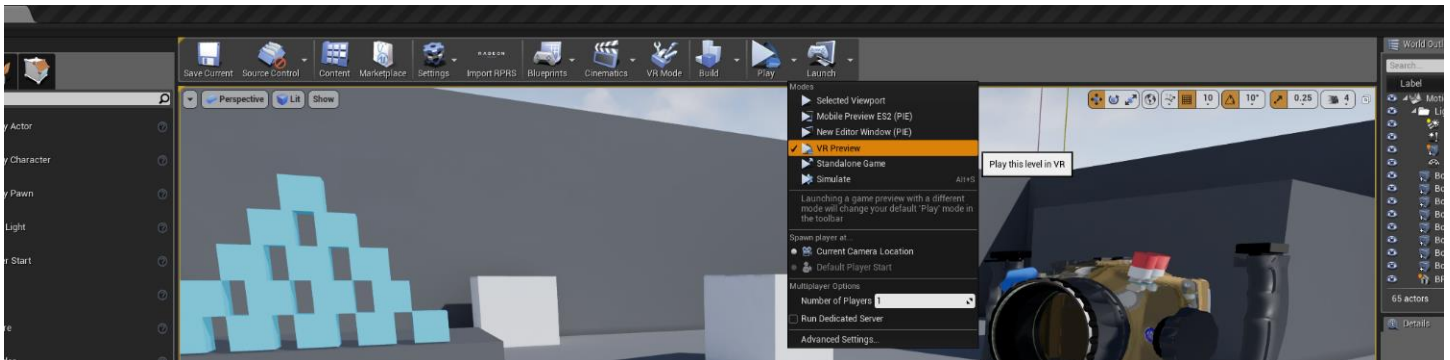
Note: AMD Radeon ProRender material appearances are automatically inherited by the asset's Unreal Engine reflection capture object.

Step 11: “Play” The Scene in VR

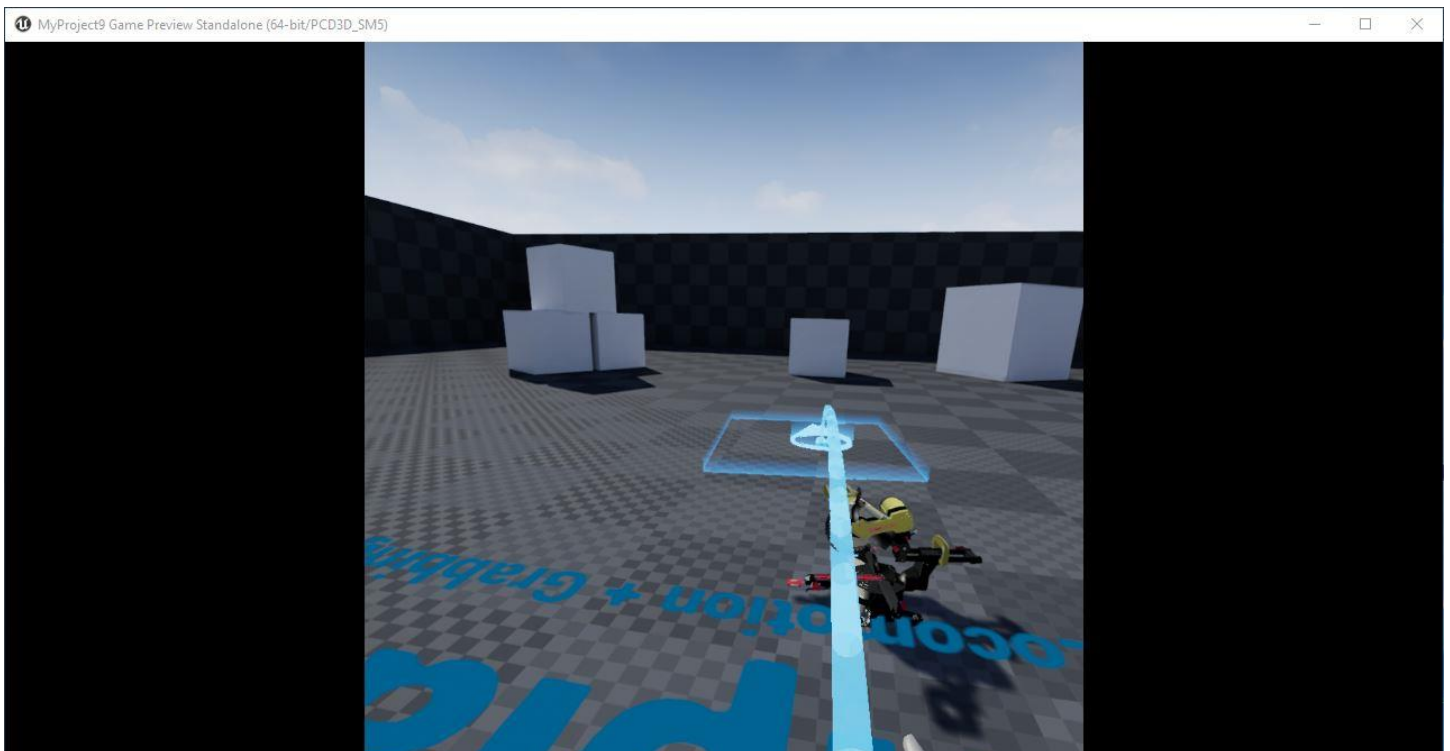
With prerequisite VR software running (SteamVR for HTC VIVE and Oculus Software for Oculus Rift) and headset and controllers tracking, “Play” the experience from the Level Editor toolbar in “VR Preview” for viewing through an HTC VIVE™ or Oculus Rift VR headset.



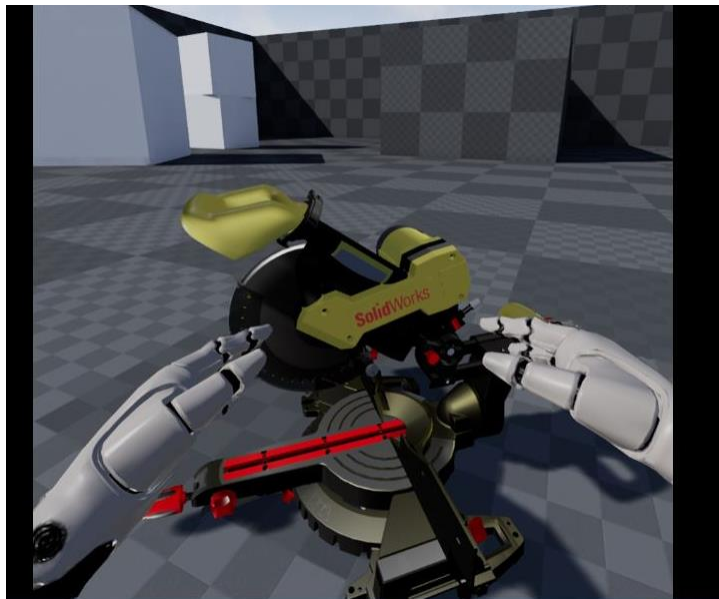
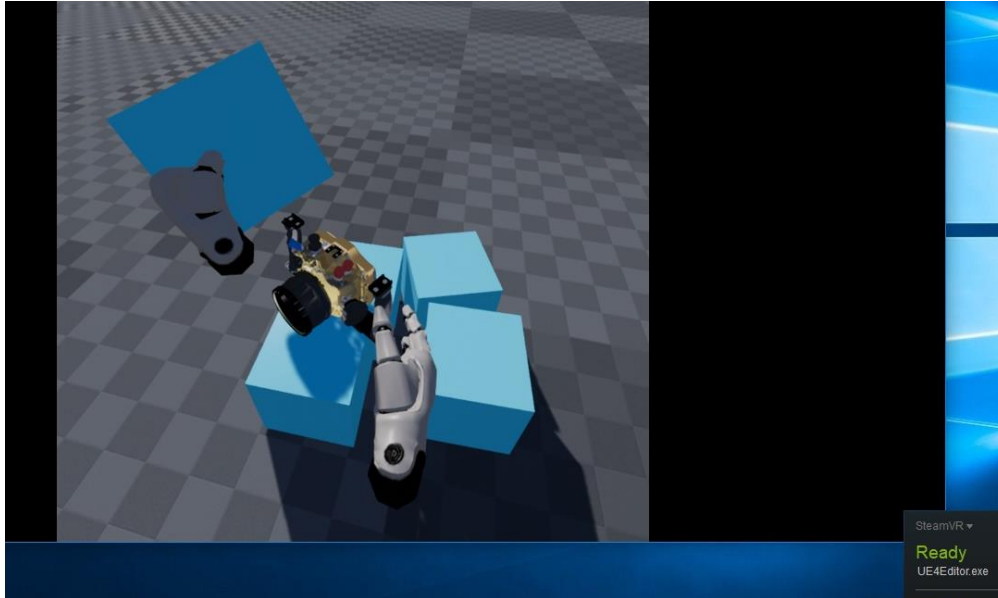
Note: There is a drop-down selector associated with the “Play” button. “VR Preview” must be selected from the list.



Teleport around the scene using the thumb pad of the left or right HTC VIVE controller. Push and hold the thumb pad to set the distance of the teleportation, and rotate your thumb around the thumb pad to indicate which direction the player will be facing when the teleportation is complete. Release the thumb to invoke the teleportation.



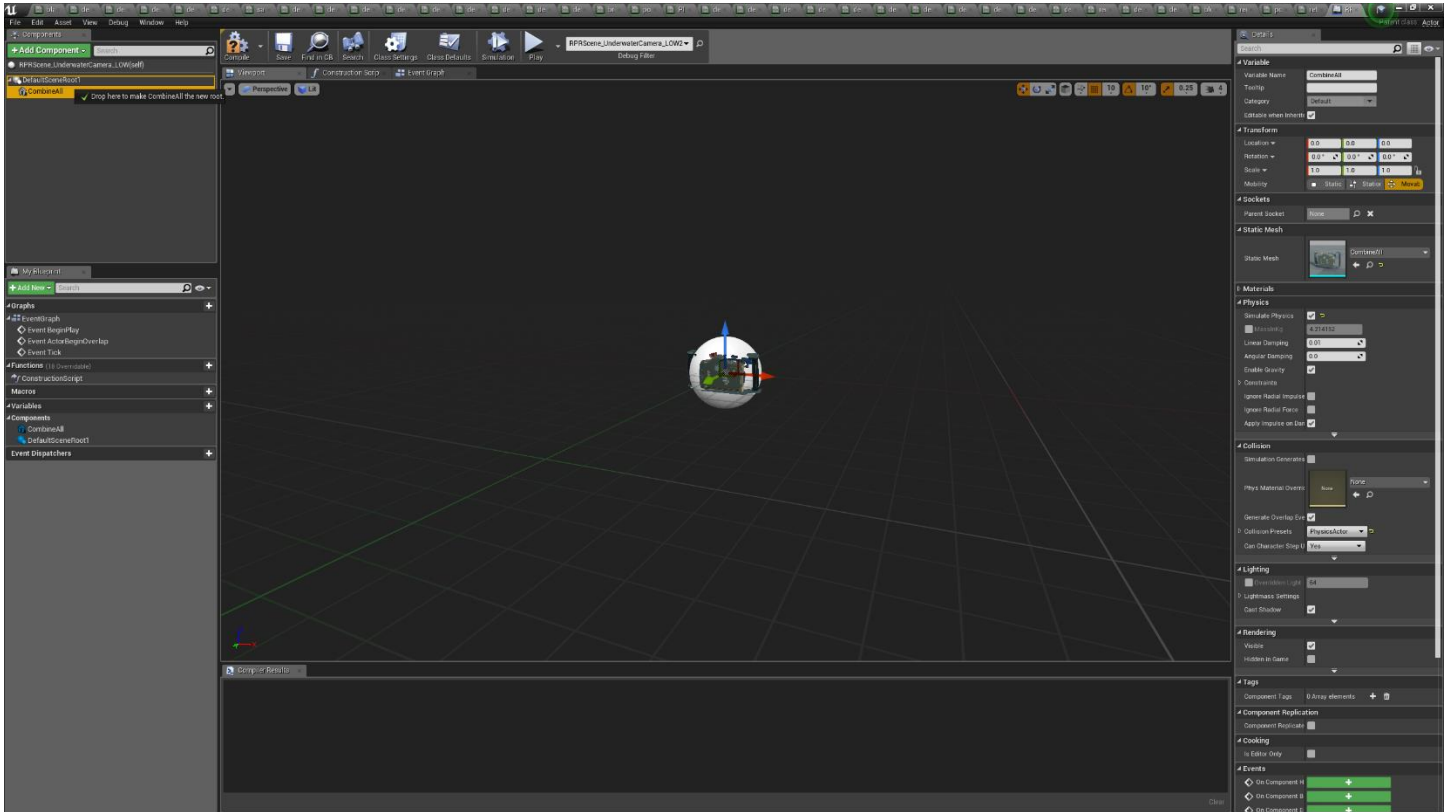
Pulling the trigger of either hand controller will invoke the “grab” action for that hand. The blue blocks in the scene can be picked up and manipulated with physics interactions. A clenched hand can be collided with the imported AMD Radeon ProRender asset, so the AMD Radeon ProRender object can be toppled and moved in a limited fashion. For more advanced interactions such as grabbing the AMD Radeon ProRender object, more advanced scripting is required in the Blueprint for that object.



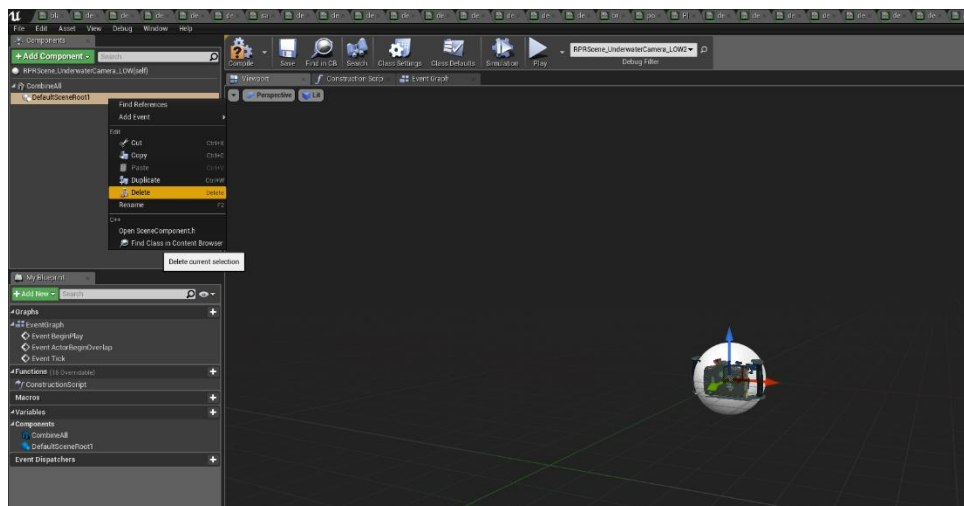
Continue reading to learn how to enable “grabbability” of your imported object.

Advanced: Edits for “Grabbability”

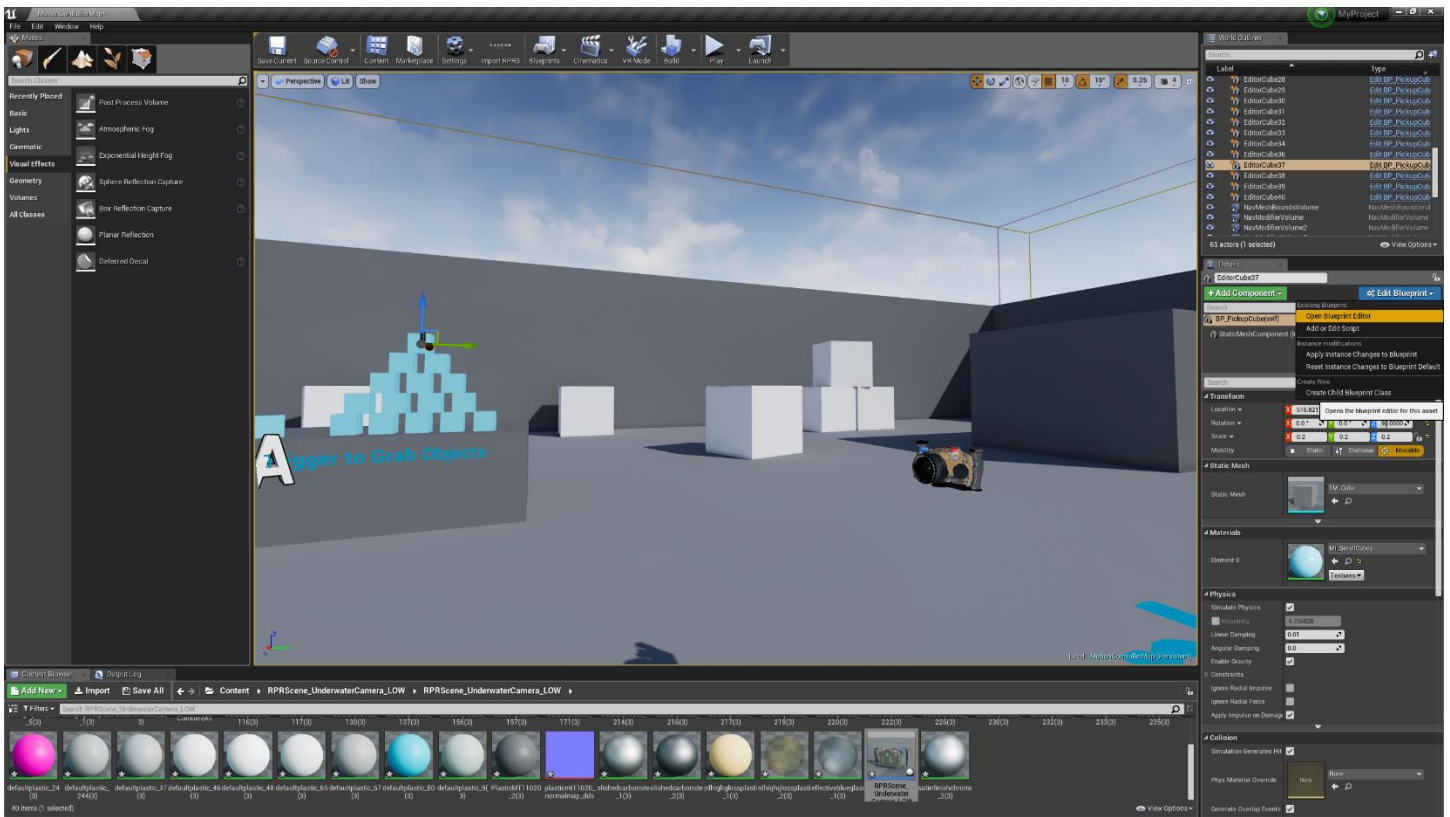
Open the Blueprint Editor for the object that was imported into Unreal Engine via the AMD Radeon ProRender Game Engine Importer. In the “Components” pane of the Blueprint Editor window, select the “CombineAll” object and drag it on top of the “DefaultSceneRoot” so that “CombineAll” appears at the top of the hierarchy.



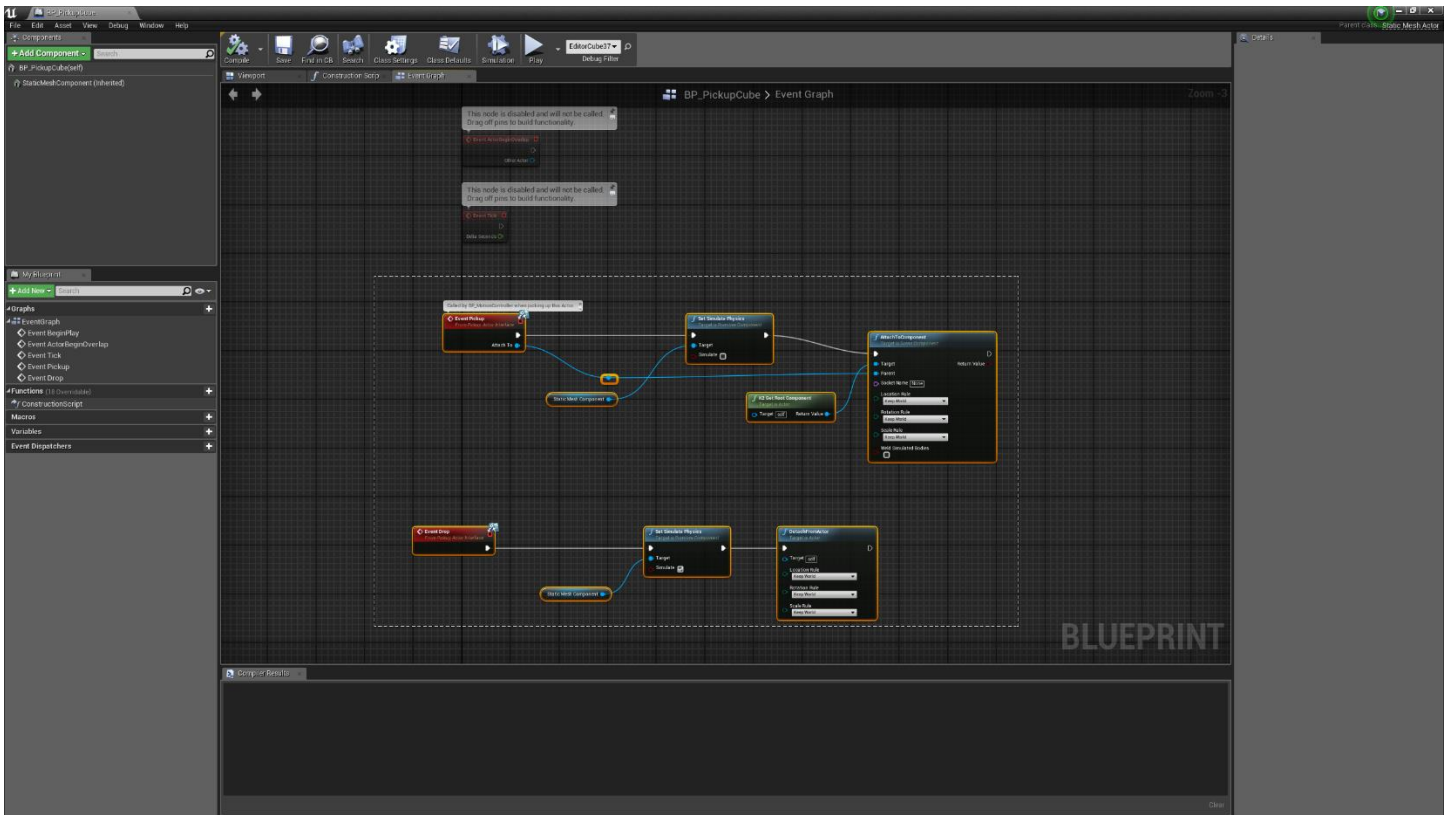
Then select “DefaultSceneRoot” and delete it.



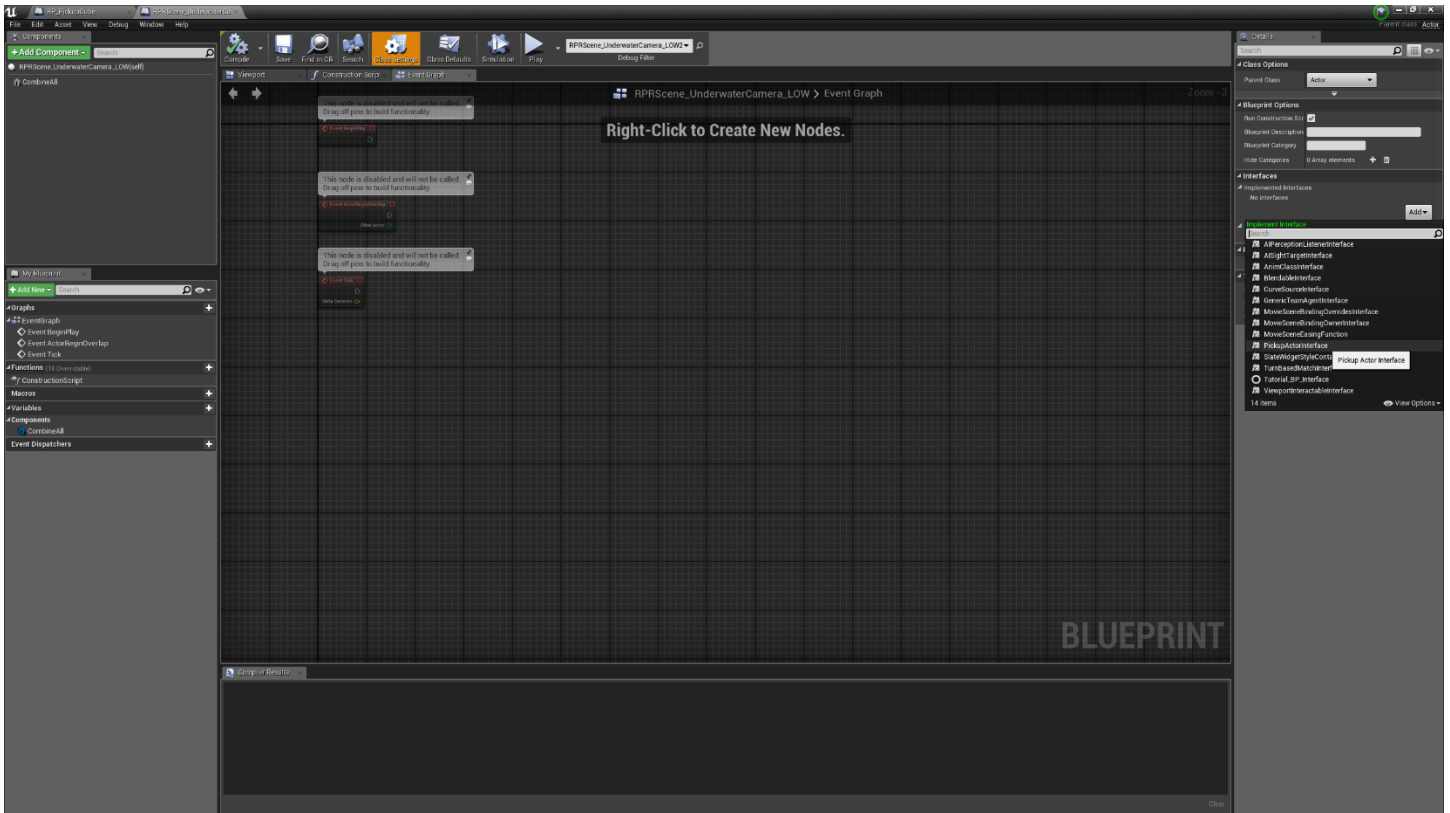
Switch back to the main Level Editor window, select one of the “grabbable” blue blocks (aka BP_PickupCube), and then choose “Edit Blueprint” for the block.



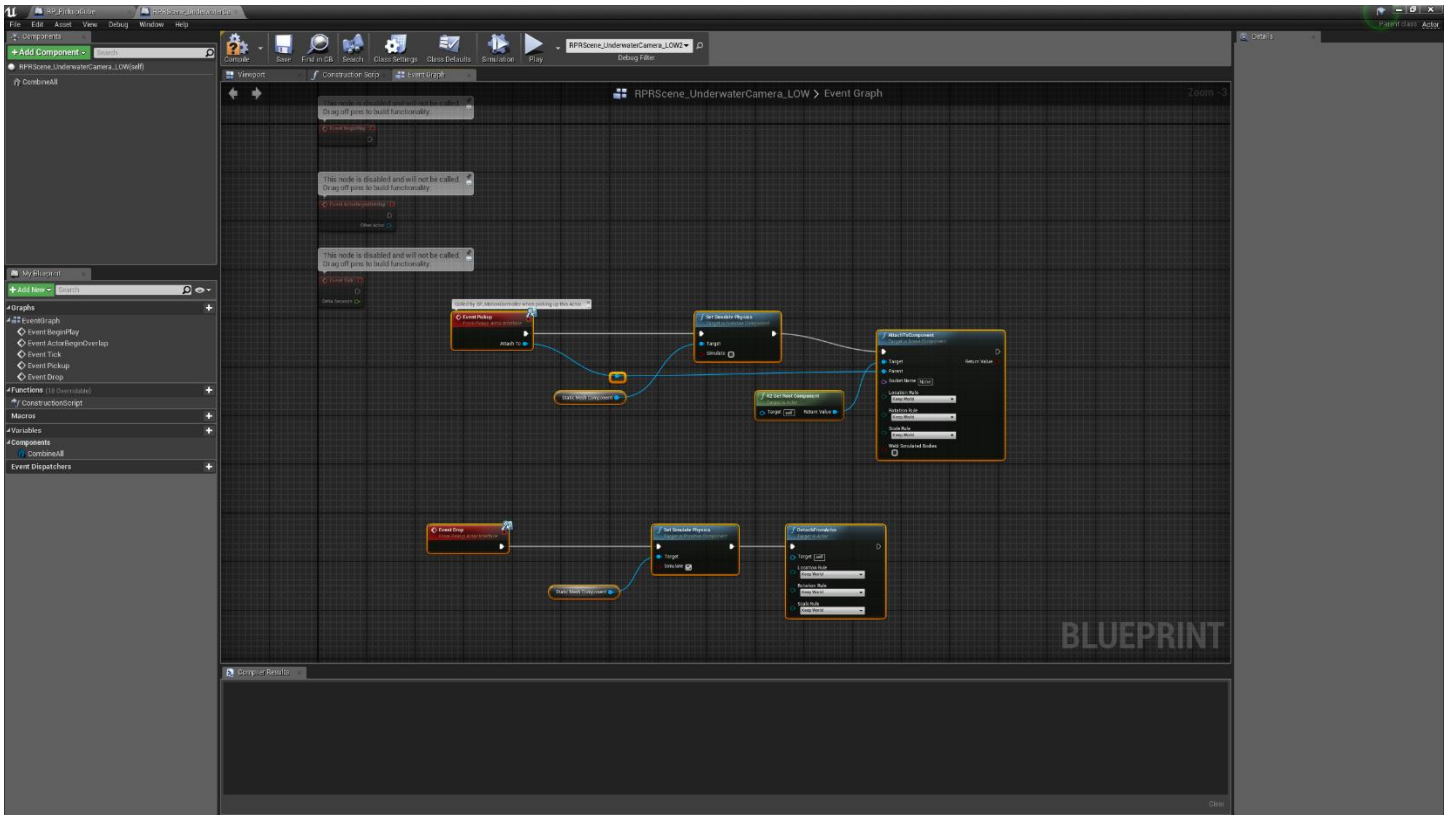
In the "Event Graph" tab of the Blueprint Editor for BP_PickupCube, select the entirety of the "EventPickup" and "EventDrop" graphs, and copy those graphs to the clipboard, either using "Control-C" or the right mouse menu "Copy" option.



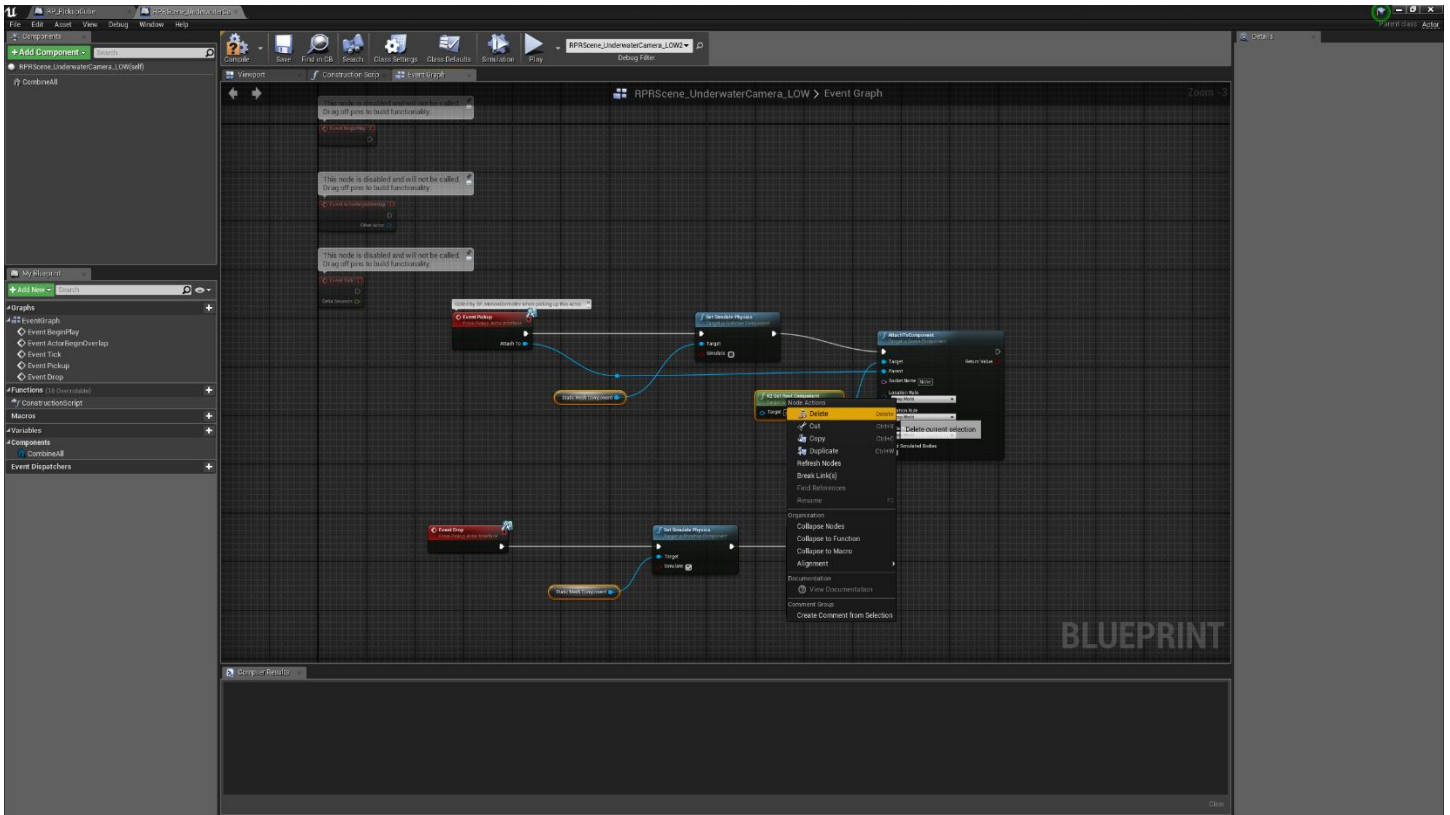
While still in the Blueprint Editor, toggle to the “Event Graph” tab of the object that was imported using the AMD Radeon ProRender Game Engine Importer. Click on “Class Settings” in the main toolbar, then add the “PickupActorInterface” to the “Implemented Interfaces” in the details pane on the right side of the Blueprint Editor screen.



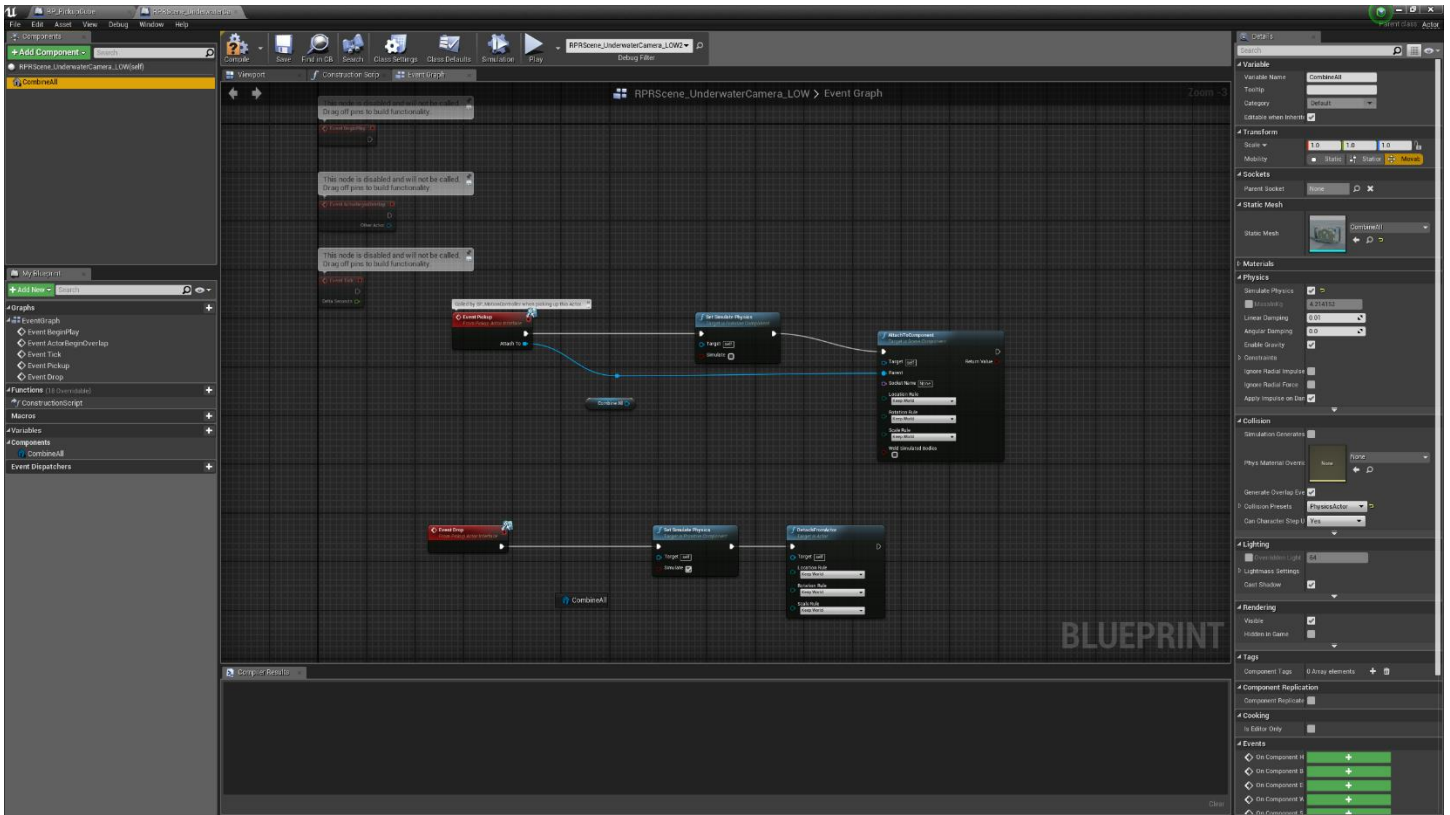
Paste the "EventPickup" and "EventDrop" graphs (from BP_PickupCube) into the "Event Graph" of the object that was imported using the AMD Radeon ProRender Game Engine Importer.



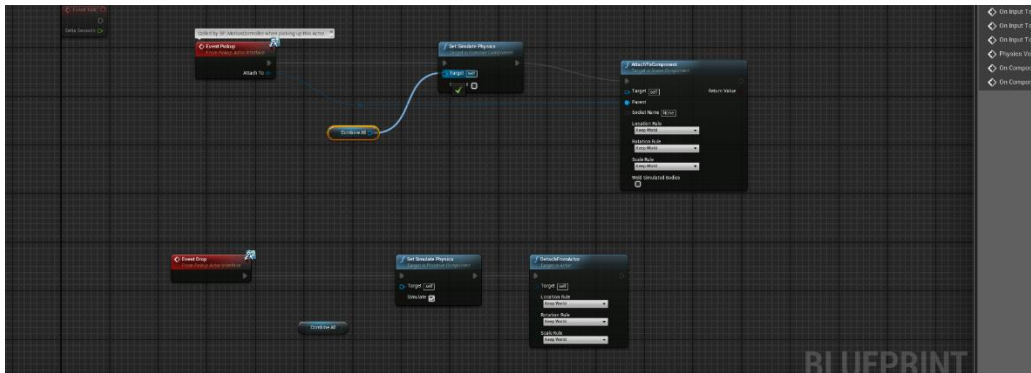
Delete the nodes highlighted below – “Static Mesh Component” and “K2 Get Root Component” from the “EventPickup” graph, and “Static Mesh Component” from the “EventDrop” graph.



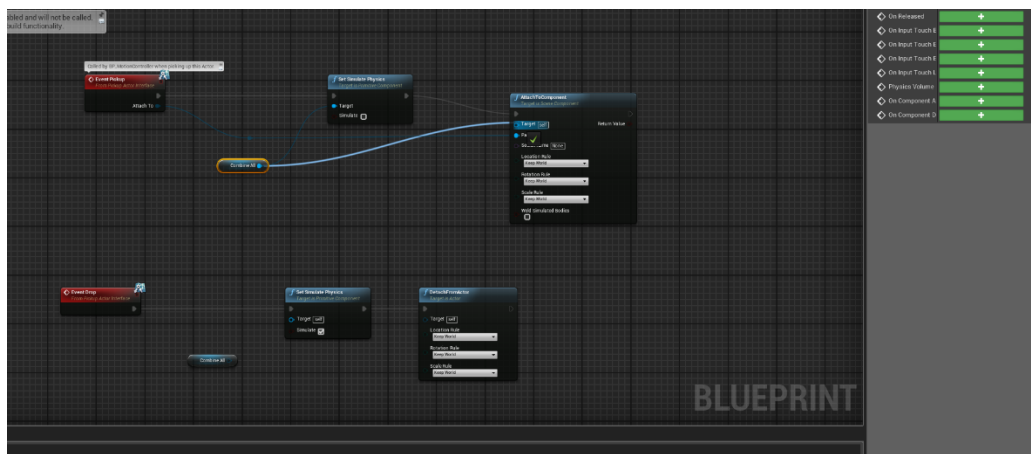
From the "Components" pane, drag two instances of "CombineAll" to the locations shown in the graphic below, aligned with the "EventPickup" and "EventDrop" graphs.



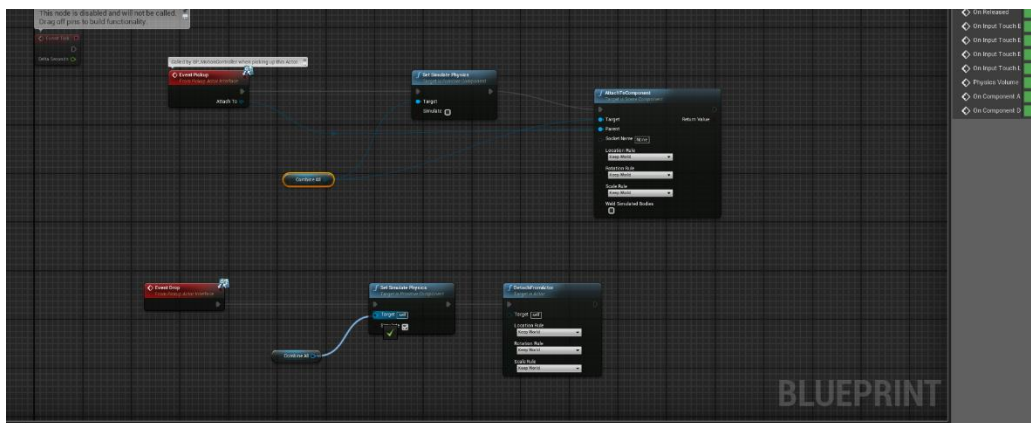
In the "EventPickup" graph, use a "click-drag-release" action to attach the "CombineAll" node to the "Target" field of the "Set Simulate Physics" node.



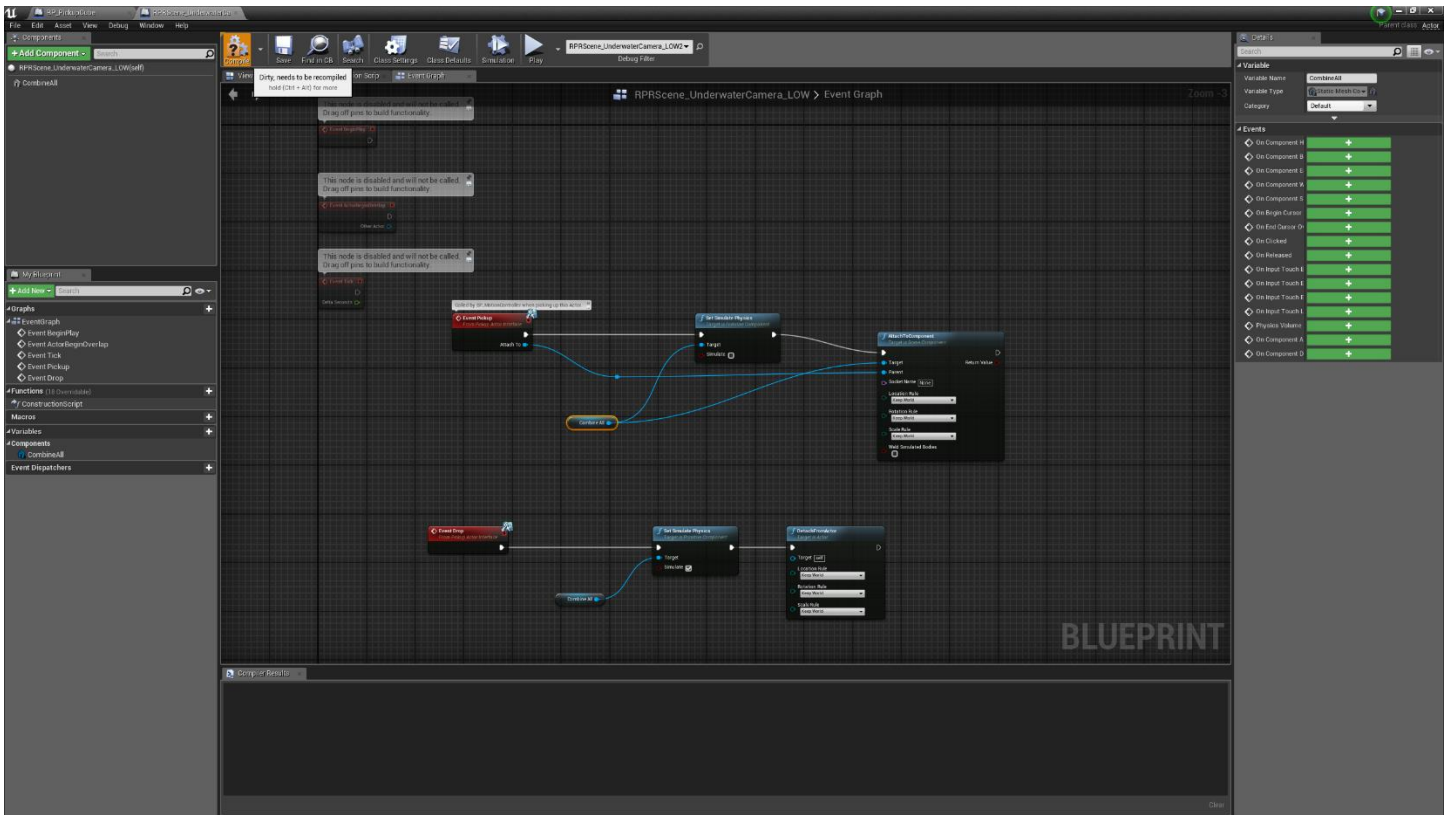
Repeat the same action to attach the same "CombineAll" node to the "Target" field of the "Set Simulate Physics" node.



In the "EventDrop" graph, repeat the "click-drag-release" action to attach the "CombineAll" node to the "Target" field of the "Set Simulate Physics" node.



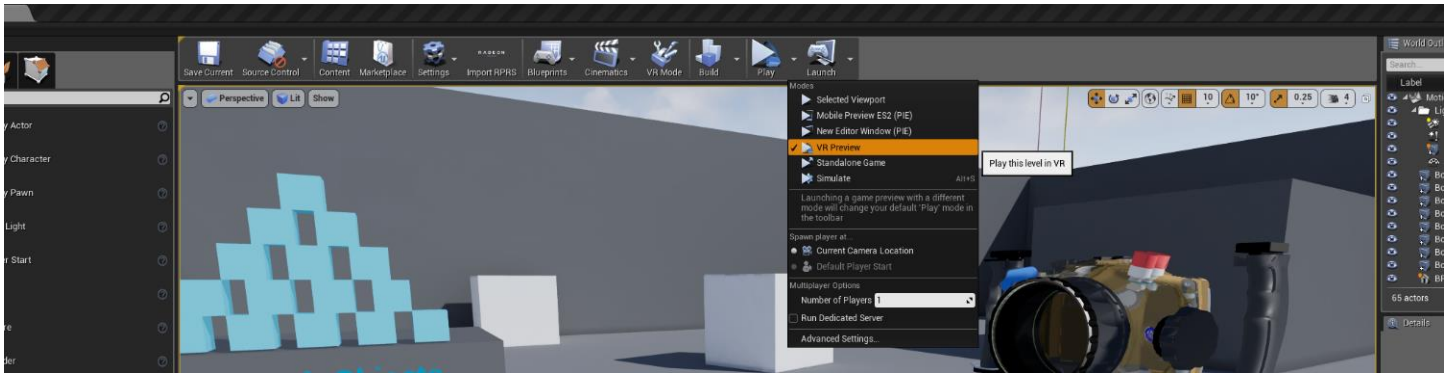
Click the “Compile” button on the toolbar to compile the Blueprint. If the compilation is successful, you will see a green check appear over the toolbar button. You may now close or minimize the Blueprint Editor window.



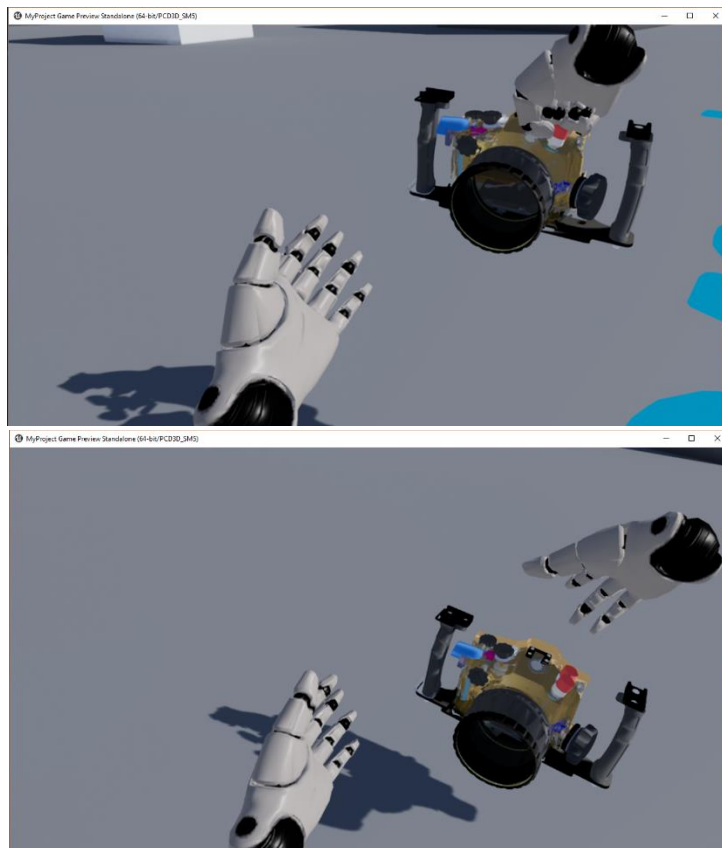
With prerequisite VR software running (SteamVR for HTC VIVE and Oculus Software for Oculus Rift) and headset and controllers tracking, “Play” the experience from the Level Editor toolbar in “VR Preview” for viewing through an HTC VIVE™ or Oculus Rift VR headset.



Note: There is a drop-down selector associated with the “Play” button. “VR Preview” must be selected from the list.



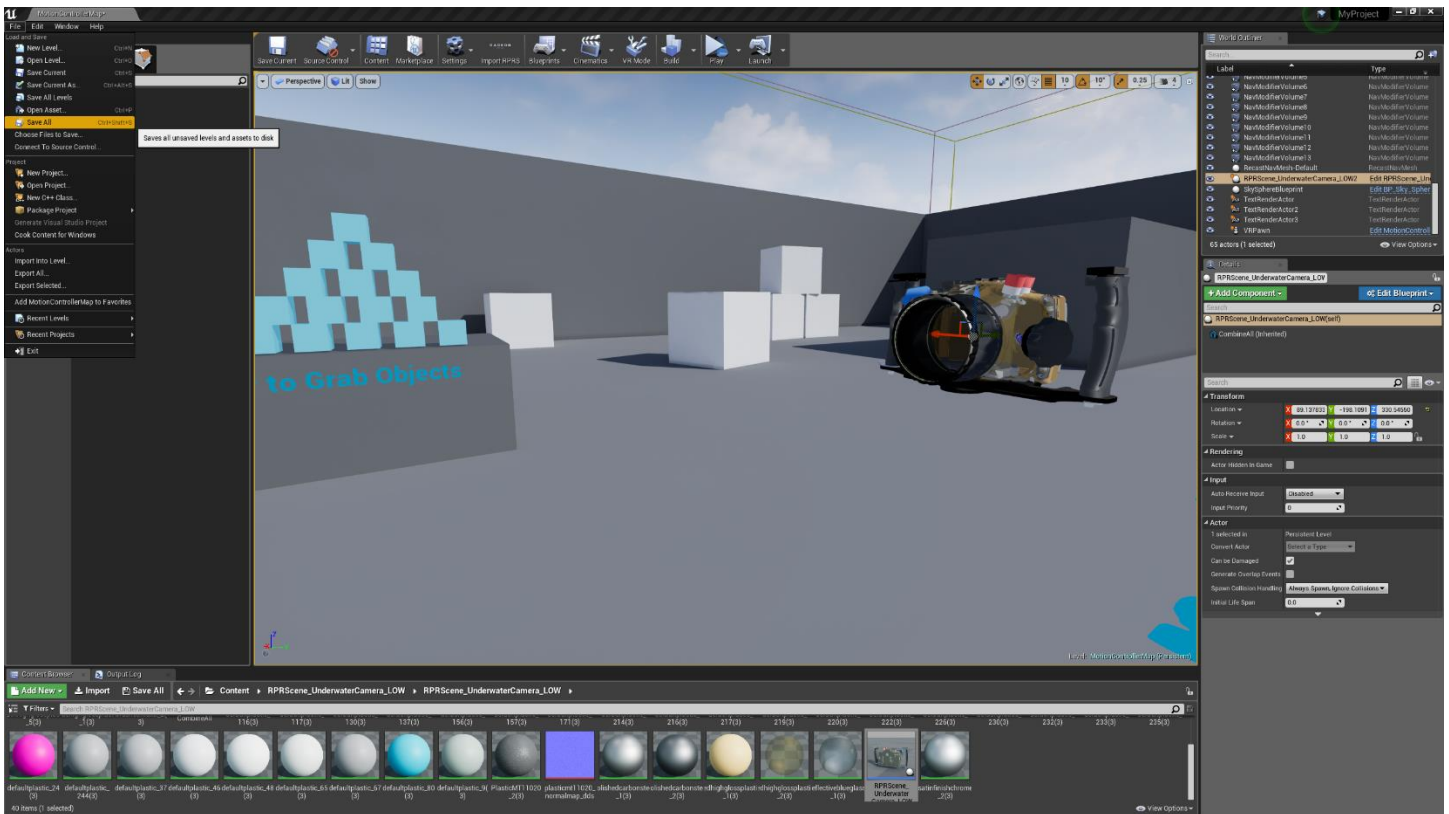
Using the trigger control of the VR controller, you can now pick up, drop, or even throw the object imported using the AMD Radeon ProRender Game Engine Importer. Holding the trigger control in the activated position will keep the object in your virtual “hand”, and releasing the trigger control will release the object from your virtual grip.



Advanced: Building a Standalone VR App

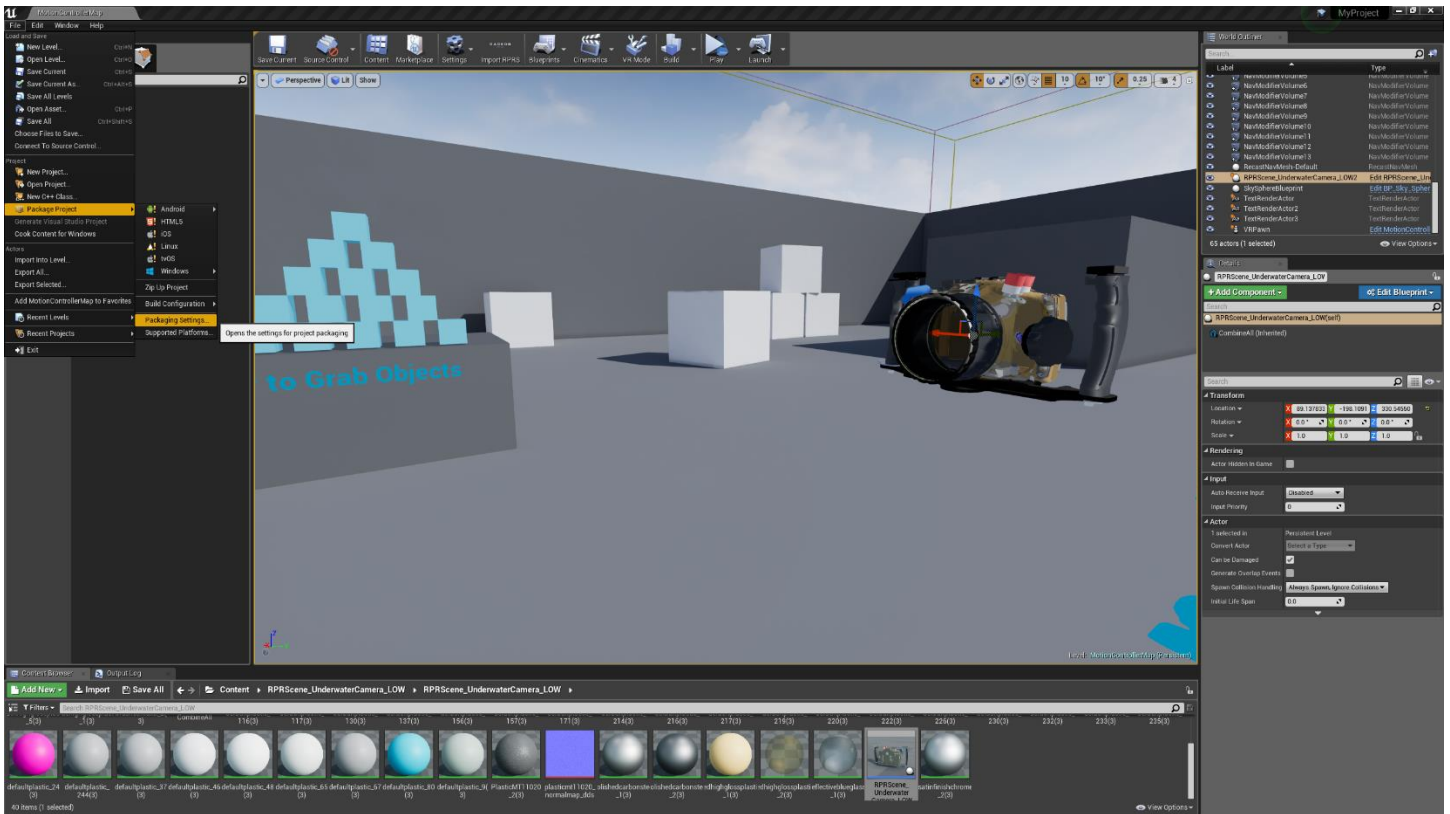
Running the VR visualization using the “VR Preview” feature from the Level Editor is convenient, but VR performance can be improved by compiling the Unreal Engine project as a standalone application that runs independent of the Level Editor. Compiled applications built with Unreal Engine can also be distributed to third parties, which is useful for sharing VR experiences with users who have a VR headset but who are not familiar with how to use Unreal Engine. Finally, sharing a VR experience as a compiled application obscures the precise coordinate geometry from third parties, which may be important for protecting intellectual property associated with the original CAD model.

Begin by invoking a “Save All” of your Unreal Engine project.

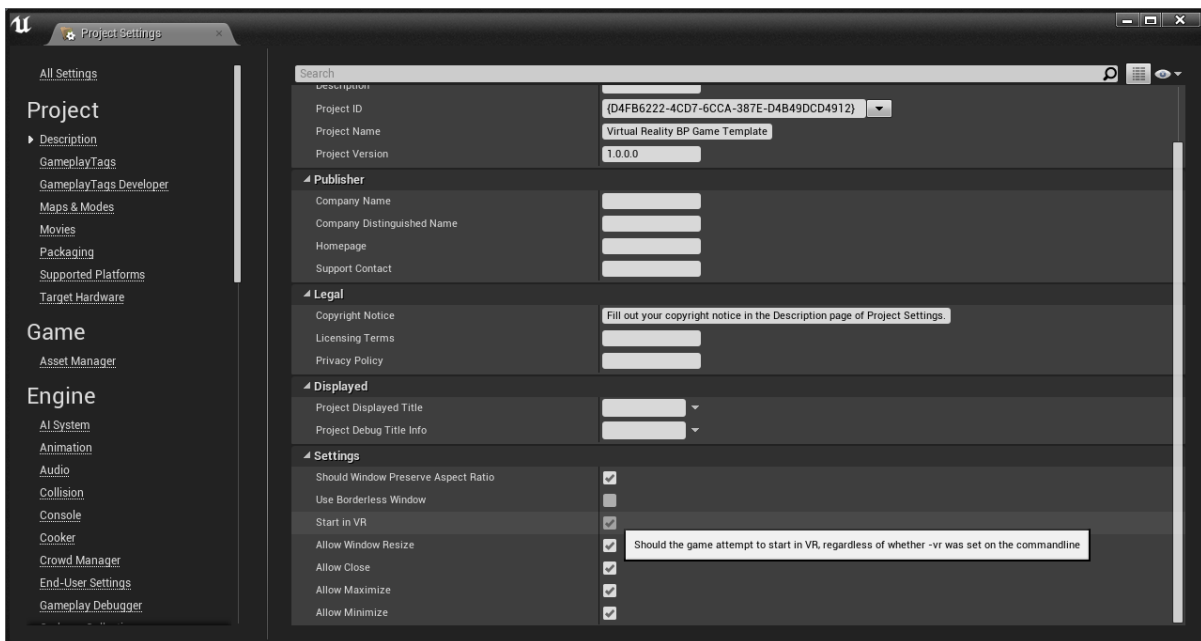


Verify that the free Visual Studio 2017 Community Edition is correctly installed per the prerequisites at the beginning of this document, and then take the following steps to build a standalone VR application.

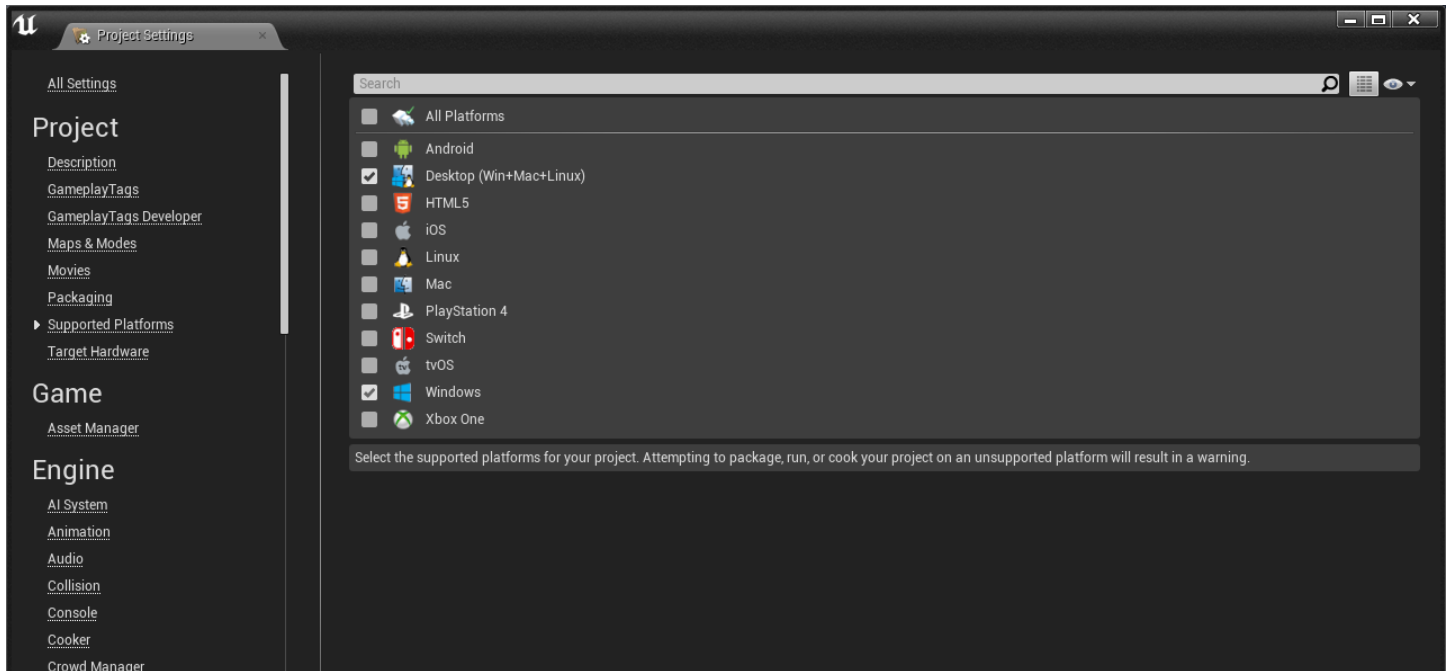
From the “File” menu, select “Package Project”, then choose “Packaging Settings...”



Check the option for “Start in VR” on the “Project - Description” page.



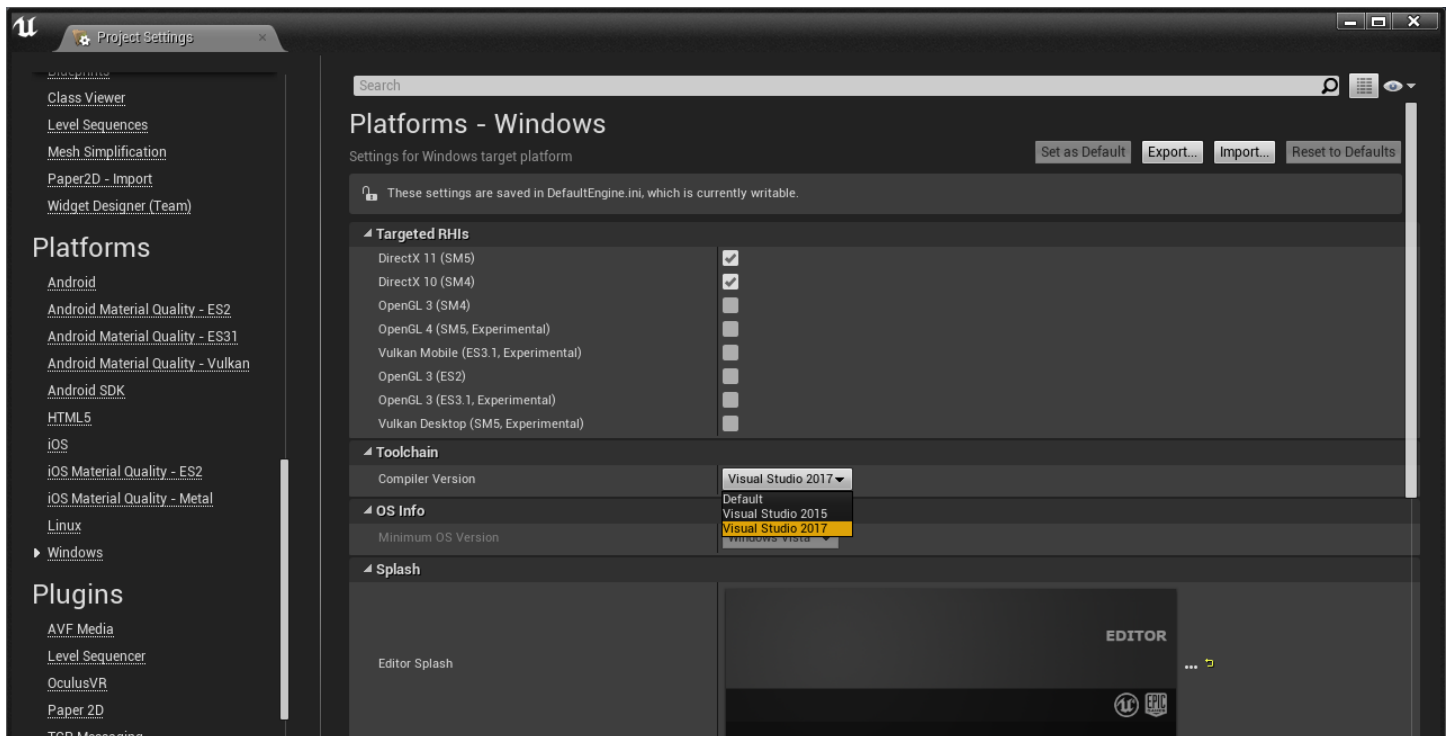
Check the options for “Desktop (Win+Mac+Linux)” and “Windows” on the “Project – Supported Platforms” page. Make sure that all other options are unchecked.



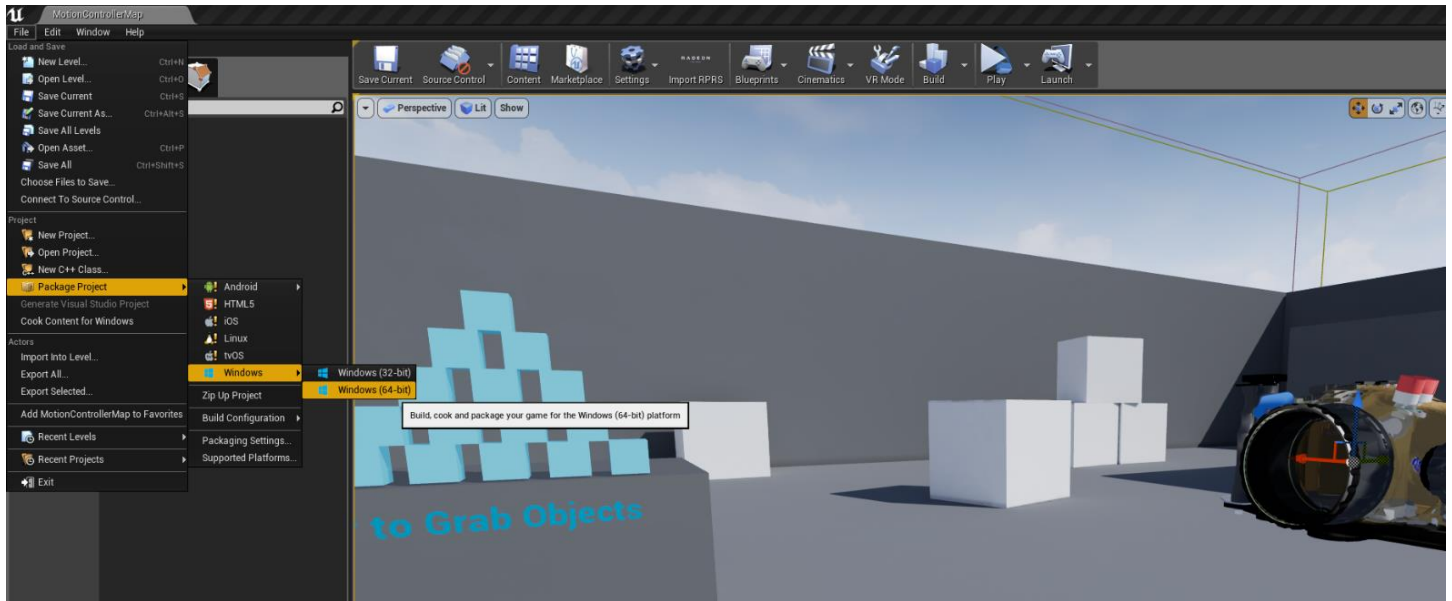
Verify that the “Compiler Version” in “Platforms – Windows” is set to match the version of Visual Studio on your system.



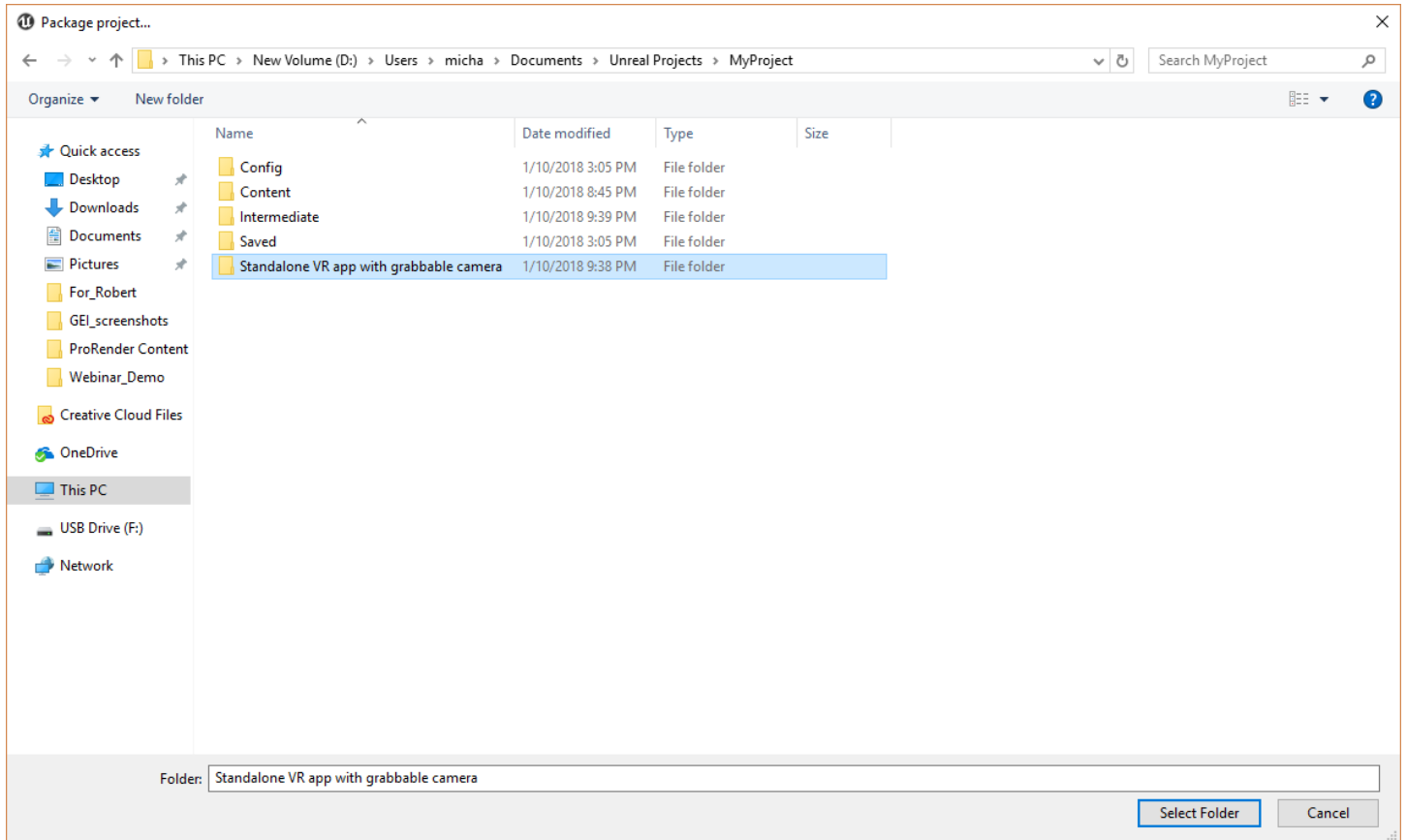
Note: Microsoft Visual Studio® Community Edition is free and recommended.



Close the “Project Settings” window. From the “File” menu, select “Package Project”, then choose “Windows”, then choose “Windows (64-bit)”.



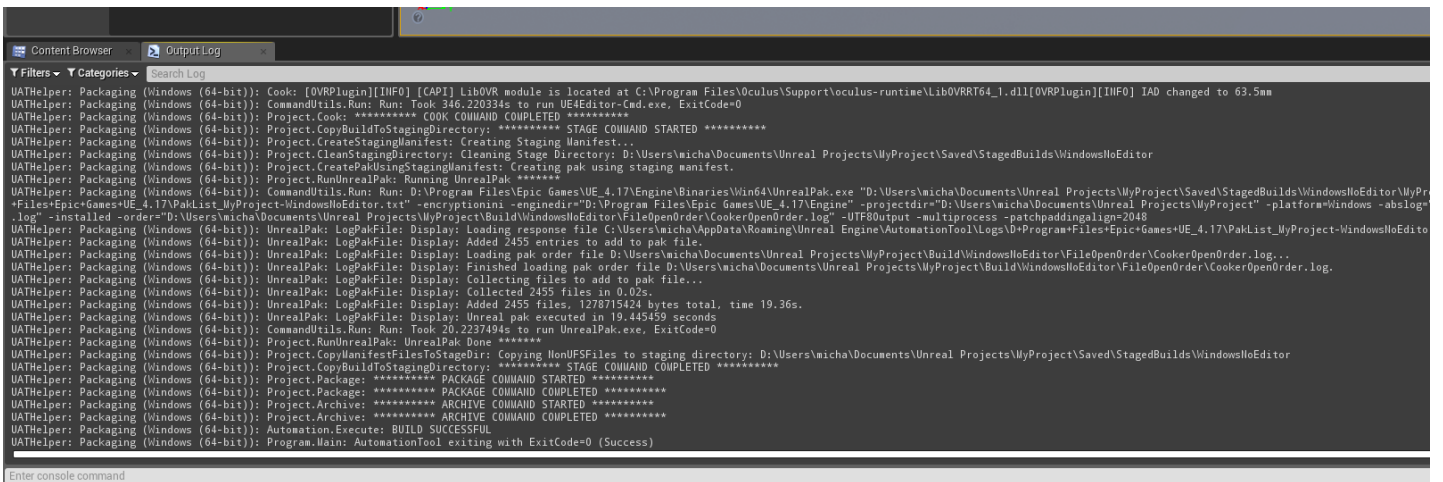
You will be prompted to choose the target directory where you want the compiled application to be saved.



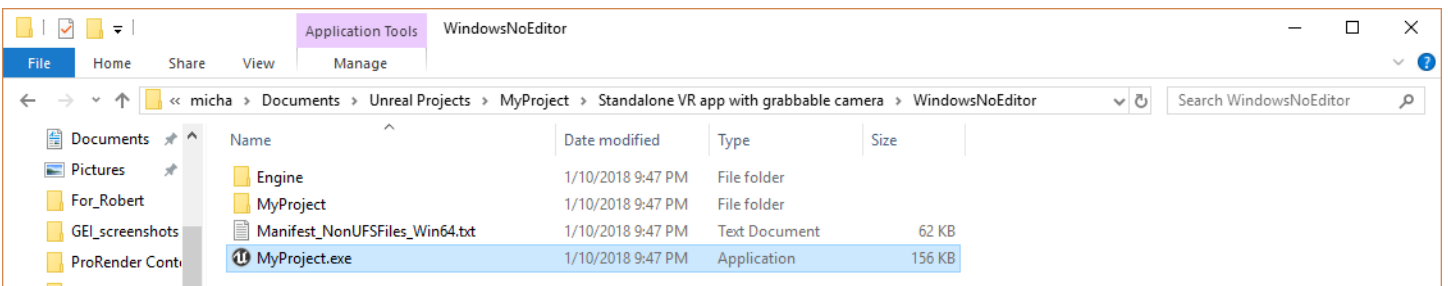
Unreal Engine will build, compile, and package the project automatically from this point. You may choose to see the output log while the process completes.



Upon completion, the Output Log will report (Success).



The contents of the target directory specified above contain everything necessary to run the application (assuming that the target system has a compatible VR headset.) To launch the app, run the .exe file that resides under the “WindowsNoEditor” directory.



Congratulations! You have completed the process of building a standalone VR application with “grabbability” of CAD objects imported using the AMD Radeon ProRender Game Engine Importer.





AMD Radeon ProRender Game Engine Importer for Unreal Engine

User Guide for CAD to VR

Written by: Michael Katz, Chris Hall, Annie Yu

08/13/2018

©2018 Advanced Micro Devices, Inc. All rights reserved.