



Unreal Engine Rendering Guide

Realtime Multi-Pass Rendering for Nuke Compositing

Created by Kyle Ragsdale 2021

This is a guide to rendering 3D from Maya in Unreal Engine (UE) for use in Nuke. Unreal Engine provides a high-quality option that will allow you to finish your student projects much quicker, without access to a render farm or having to compromise the results.

This guide will show how to import animated geometry and cameras from Maya into UE, set up materials and textures in UE, and render .exr passes from UE. It will not give you the same exact type of renders you may be used to in Arnold or other programs, but they can be done in a fraction of the time and you are still able to have control in Nuke. The ability to re-render quickly in Unreal gives a lot of flexibility to this pipeline. Some basic knowledge or UE is necessary, and lighting will not be covered.

CONTENTS:

- Importing Geo from Maya into UE.....02**
 - Export Geo from Maya.....02
 - Import Geo into UE.....03
 - Playing Animation in Sequencer.....04
- Creating Materials and Applying Textures in UE.....05**
 - Creating Master Material.....05
 - Creating Material Instances.....06
- Importing Cameras from Maya into UE.....08**
 - Export Maya Camera.....08
 - Import Camera to UE.....08
 - UE Camera Setup.....09
- Exporting Render Passes from UE.....10**
 - Beauty, Alpha, Lighting Only.....11
 - BaseColor, Post ToneMapColor, Normal, Depth.....13
 - Ambient Occlusion, Shadow.....14
 - Contact Shadow.....15

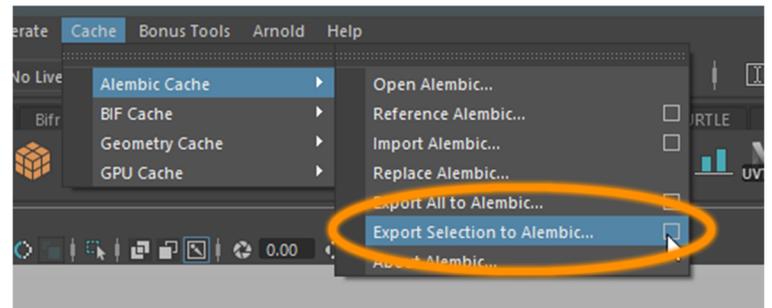
Importing Geo and Textures from Maya into Unreal Engine

Although it is possible to import .fbx files into UE, many rigs are not game-ready and will not import properly, or at all. For example, cinematic rigs that have more than one root will not import. The most *solid method to import animated geo into UE is to use Alembic Cache (.abc file).*

The Export Process (Maya):

- Make sure that materials are applied to objects.
- Select *geo only* for object, not a parent group. The objects can be grouped but must be selected individually. The easiest way to do this is *show Polygons only in the viewer* and drag select the geometry in the scene that you want to export.
- UE will not import Alembic geometry that includes n-gons, so you may have to run a mesh *cleanup* to fix before export.

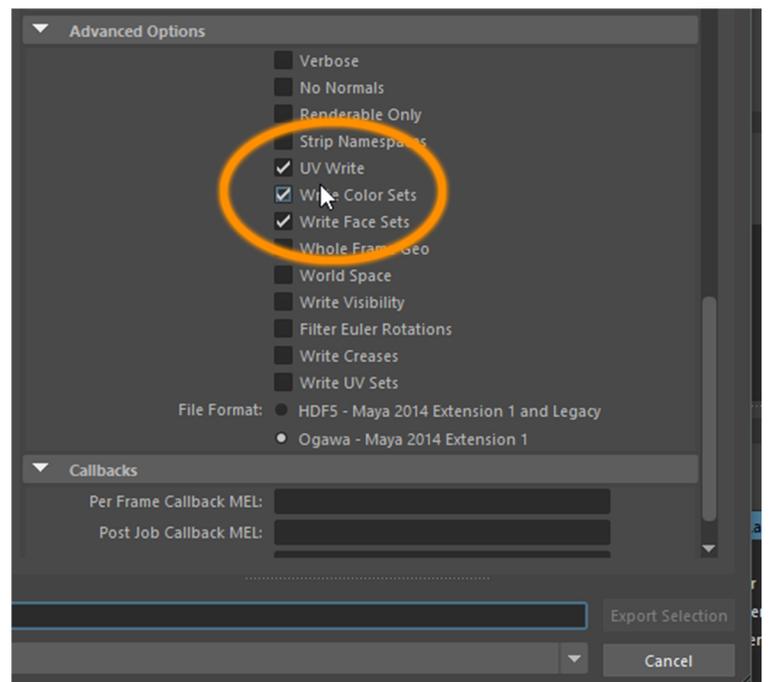
-Cache > Alembic Cache > Export Selection to Alembic



- In Alembic Export Options under Advanced Options, make sure *UV Write, Write Color Sets, and Write Face Sets are checked.*

If you have a problem later bringing in textures to UE, you may need to rewrite .abc with Write UV Sets checked, but usually this is not necessary.

-Name file and Export Selection



The Import Process (UE):

-Drag Alembic .abc file into the content browser. (The default location for Alembic files is in Maya Project > Cache > Alembic)

Import Options window:

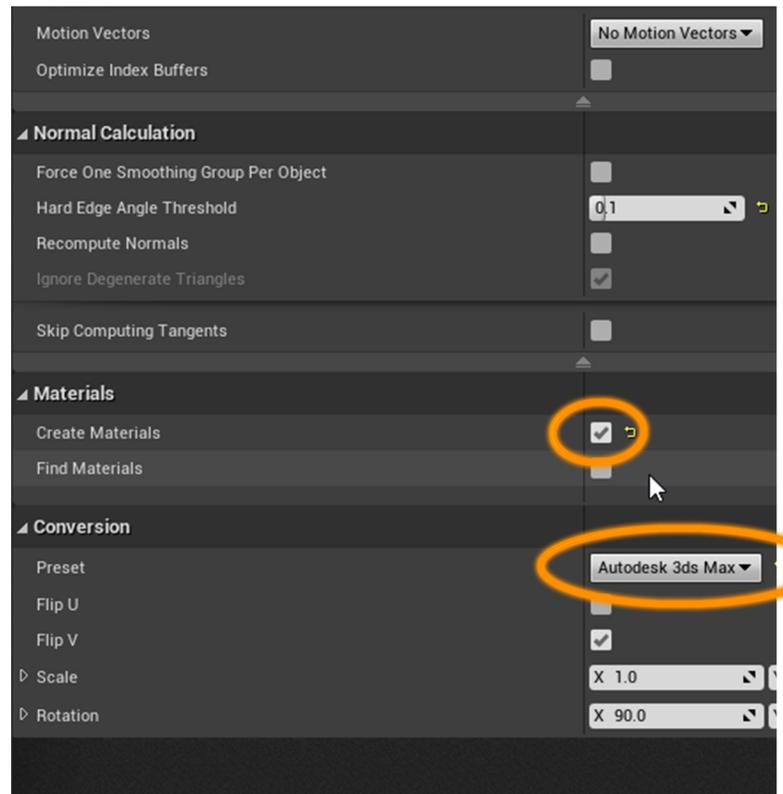
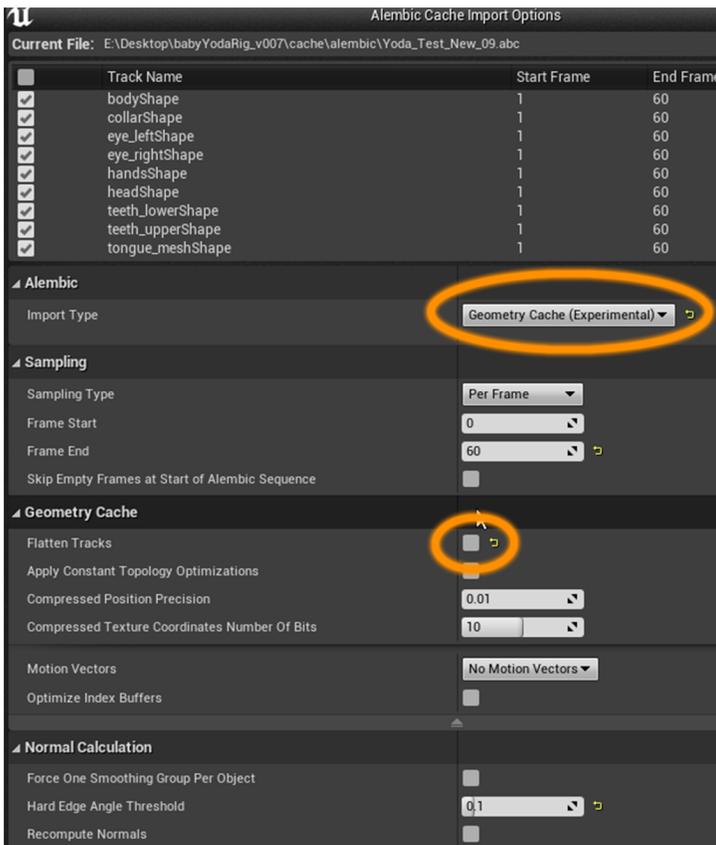
-Import Type: *Geometry Cache (Experimental)*

-Uncheck *Flatten Tracks* (otherwise only one material will be applied everything)

Hard Angle Threshold: lower value will create smoother normals

-Check *Create Materials*

-Preset: *Autodesk 3ds Max* (for some reason this works with Alembic files from Maya. Using the maya setting will import your geo with incorrect orientation).

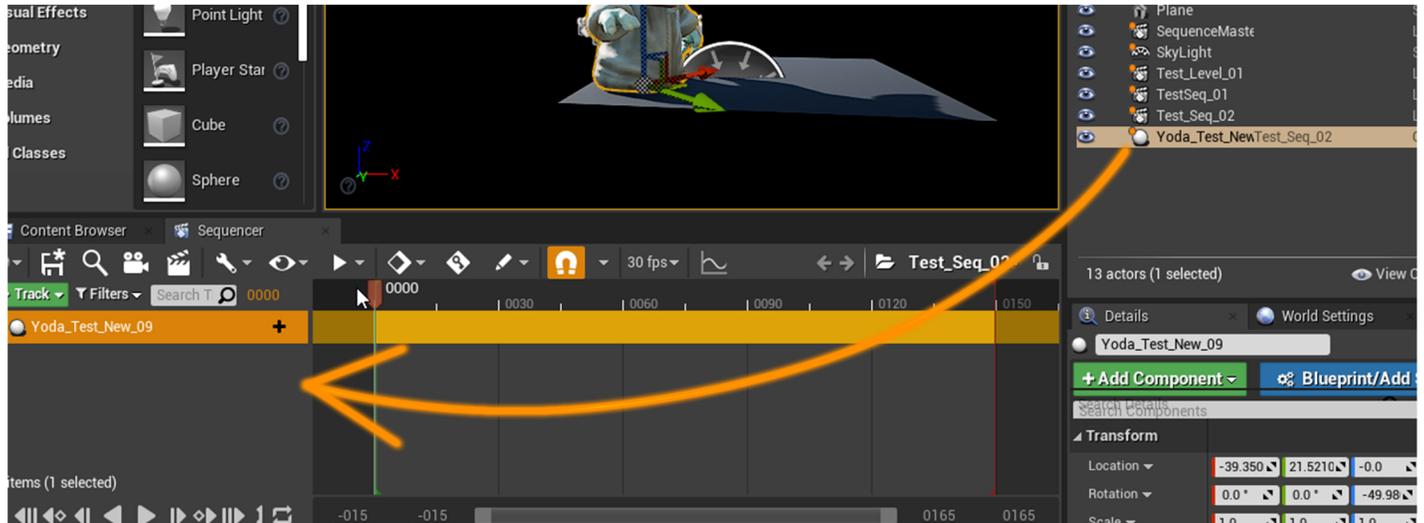


-Import

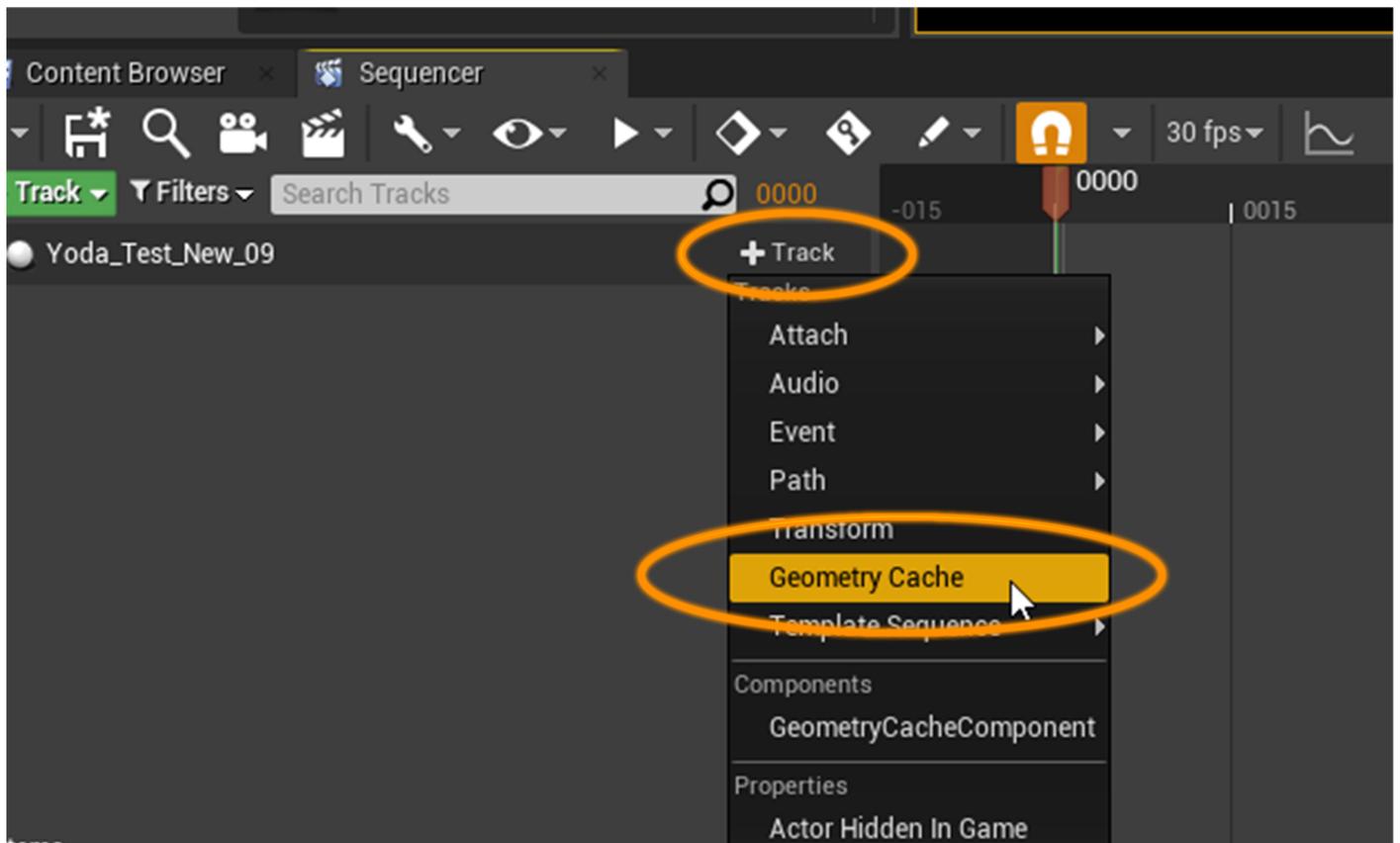
-Drag object from content browser into scene.

Playing Animation in UE Sequencer:

- Create Level Sequence
- Put playhead at start
- Drag Geometry Cache object from world outliner into area left of timeline in sequence.



- Click +track > Geometry Cache



- Animation should play back when sequencer is played.

Creating Materials and Applying Textures (UE):

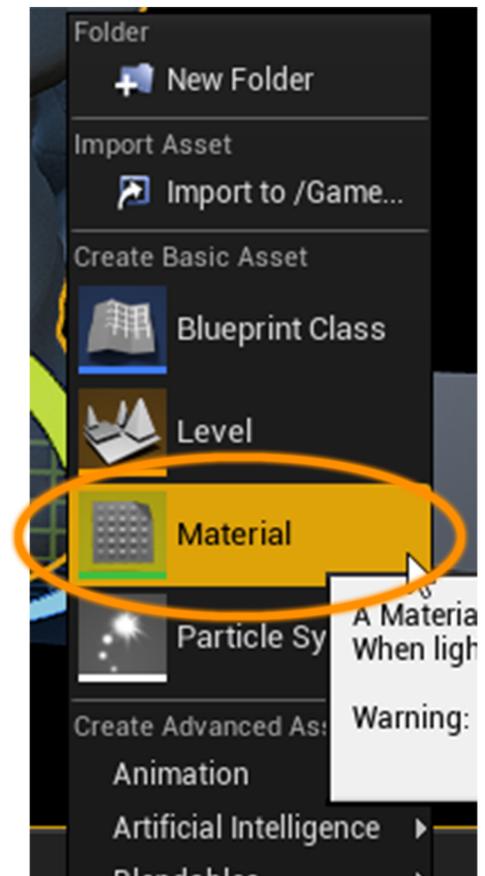
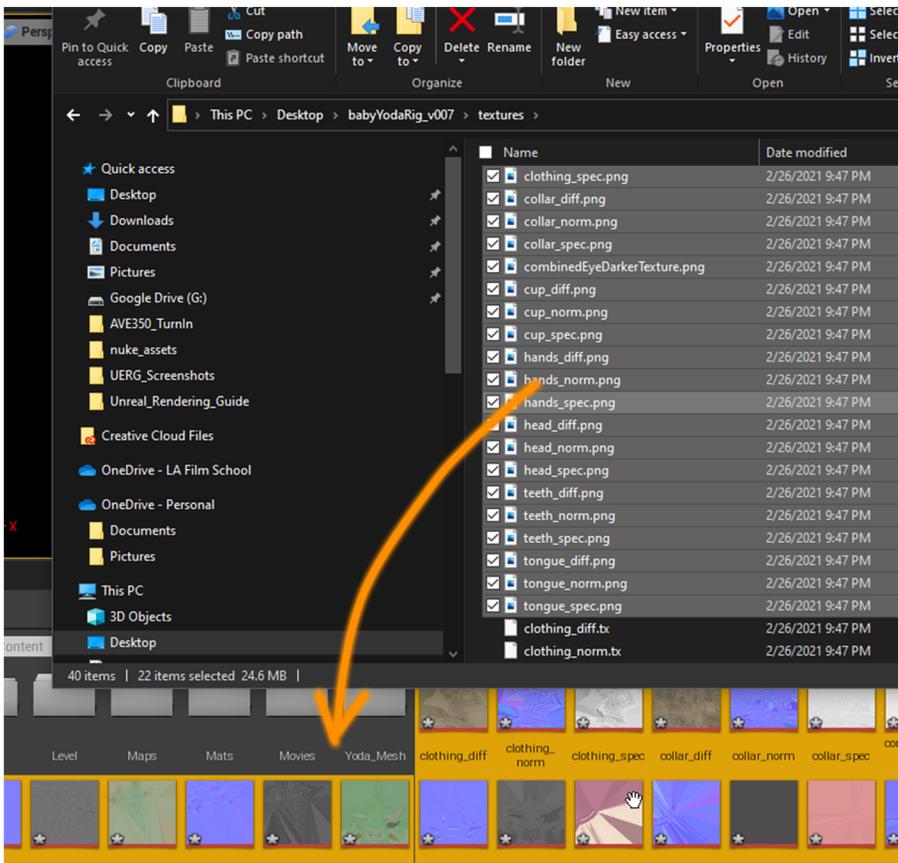
-If Alembic is successfully imported it will create separate elements for each material. They will all be set to WorldGridMaterial by default. **You will need to create material instances inside of UE and apply them to each of the elements.**

The Process:

-Drag all your texture files from your Maya project folder into the content browser.

Creating a separate folder is best practice.

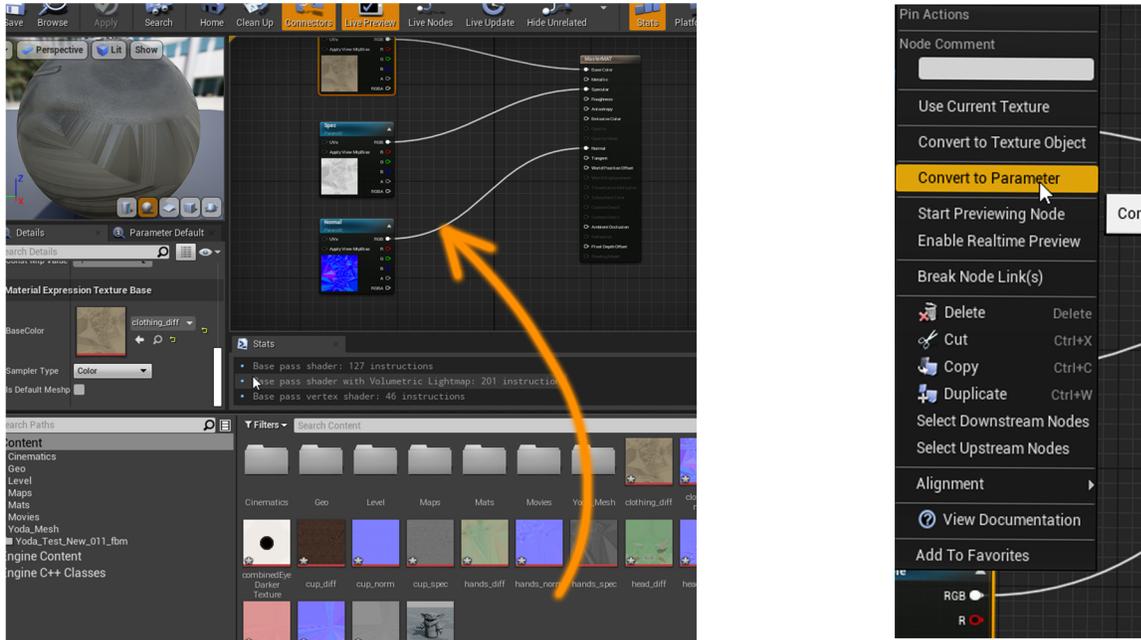
-Right-click in Content browser area and select *Material* under Create Basic Asst. Name this material MasterMAT. Creating one master material makes it easier to set up multiple material instances later.



Applying Textures (UE) cont.

-Double-click masterMAT and *drag texture image files for first material into the blueprint*. Connect texture node to base color, normal, spec, etc depending on what textures you have. (use RGB output pin)

-right click on each Texture node and click *Convert to Parameter*. Name parameter appropriately (BaseColor, Spec, etc). This will have to be done for each texture node.



-Save and close masterMAT blueprint.

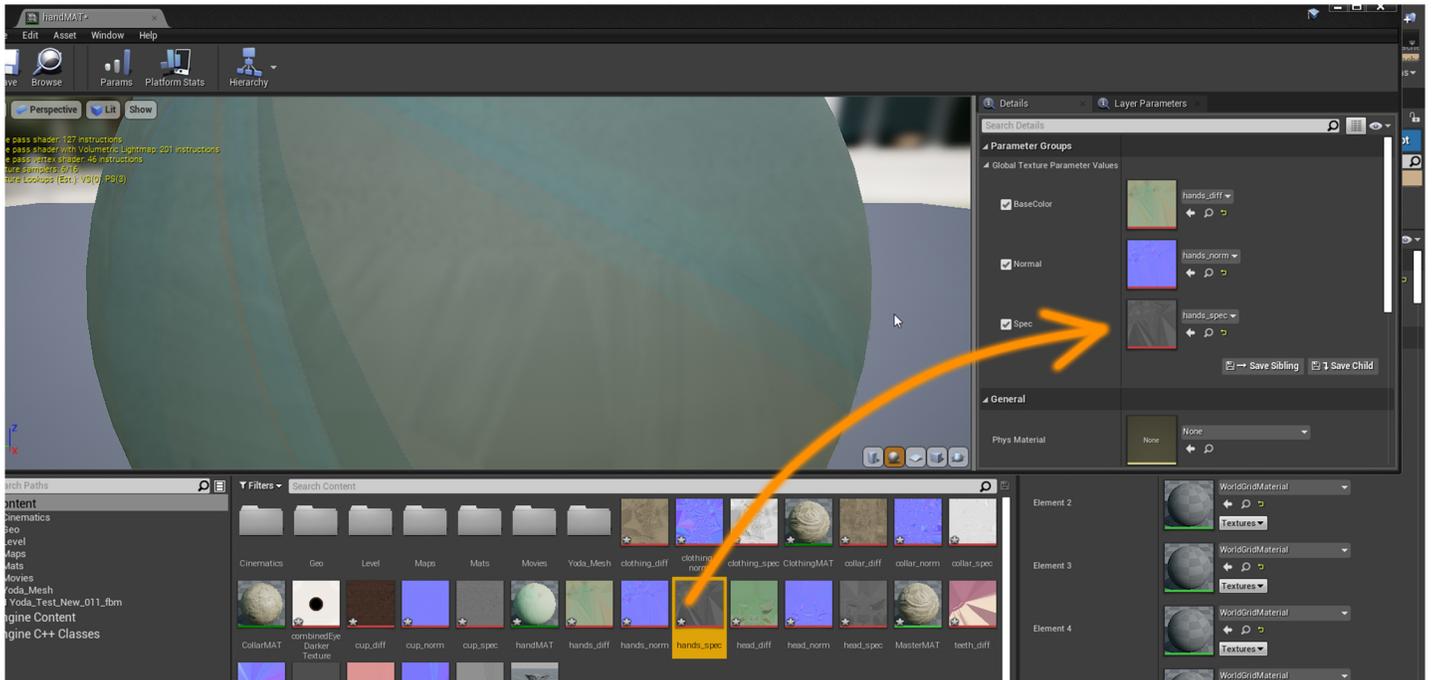
-In content browser right-click masterMAT and select *Create Material Instance*. You will need to do this for each material that you have in your scene. Name according to material set (example faceMAT, clothesMAT, etc).

-Inside each material instance, *check the parameters you want to change* (BaseColor, Normal, Spec..) These are the parameters you set up in the master material.

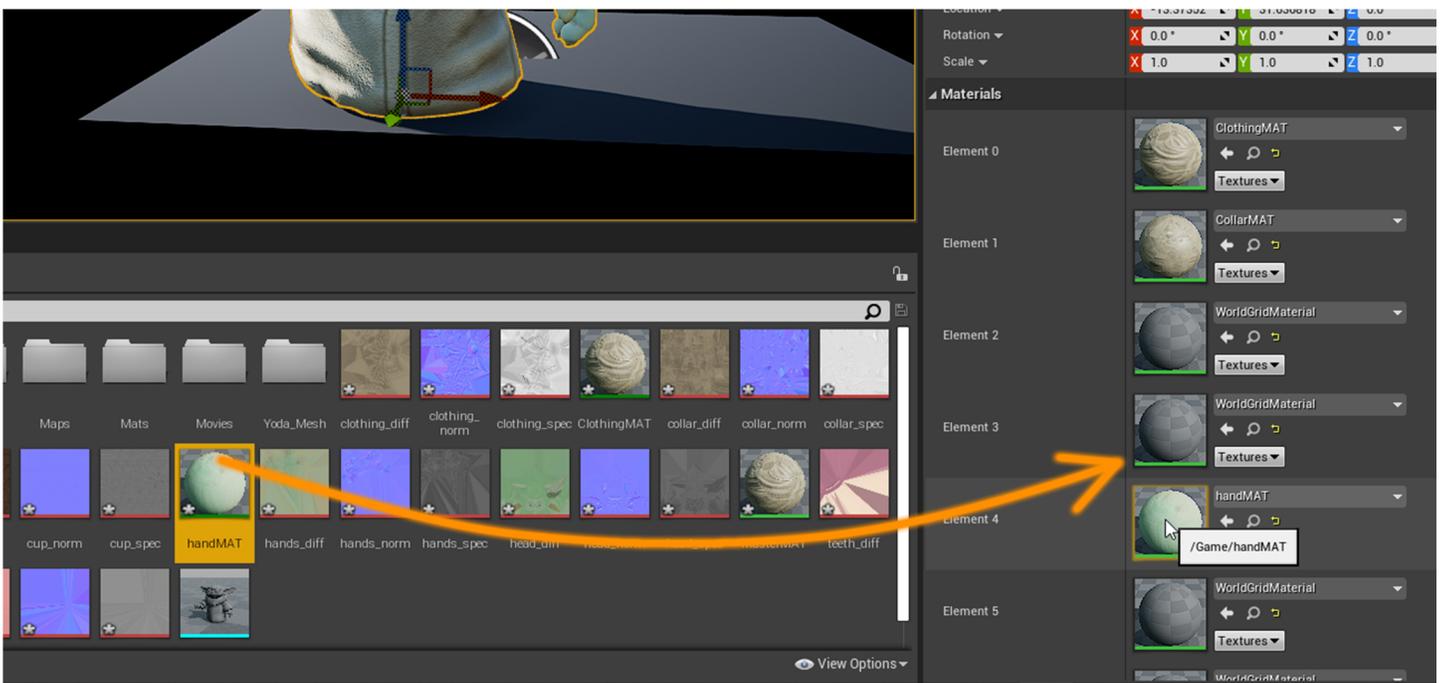


Applying Textures (UE) cont.

-drag and drop the appropriate texture maps into the material instances to create your materials. Save each instance before closing.



-In content browser, drag newly created material instances into the elements of your geometry cache. It may take trial and error to assign correctly because the elements are not named. *Some geo might import FaceSet names from Maya, but usually not, and creating FaceSets in Maya pre-import may causes problems with rig animation.*



Importing Camera from Maya into Unreal Engine:

Export Camera from Maya:

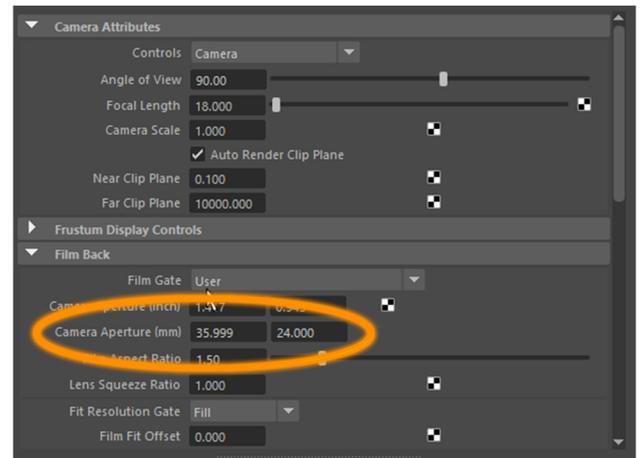
-Bake camera animation

-Unparent camera from any other objects in outliner

-Select camera only

-Export selection as .fbx

-write down Camera Aperture (mm) – both values – from Camera Attributes Editor. You will need these values to set up camera inside UE



Import Camera to UE:

-Create level sequence if you haven't already

-Create (drag and drop) *Cine Camera Actor* into sequence. Drop into area left of the timeline.

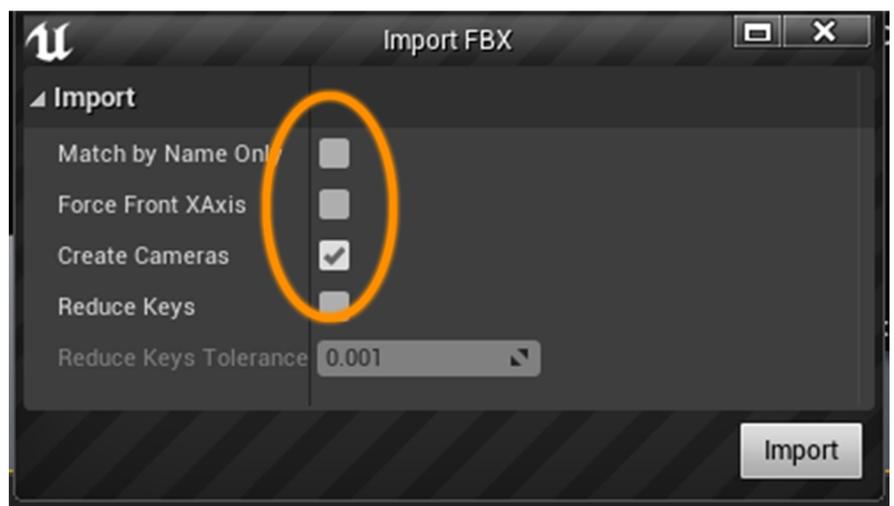
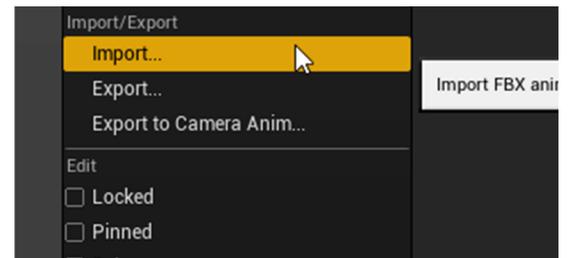
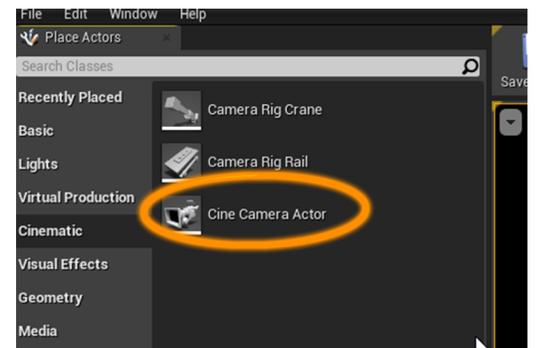
-Right click on Cine Camera Actor in sequencer then click *Import*

-Select .fbx camera file

-Uncheck "Match Name Only"

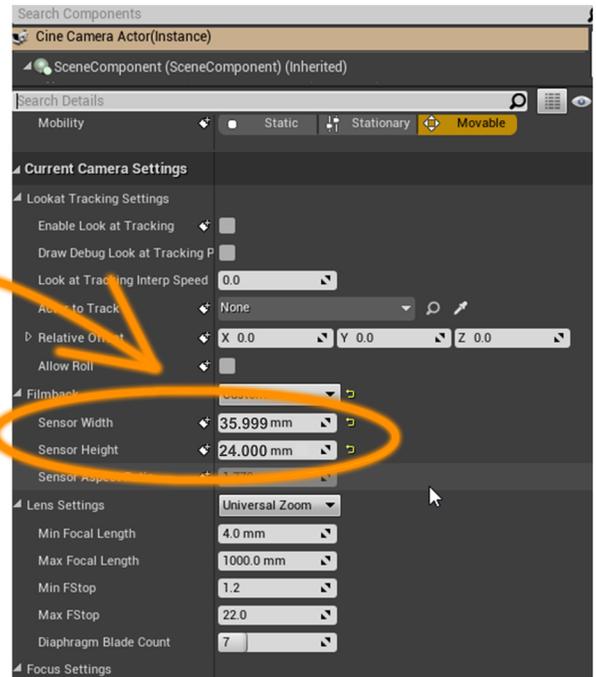
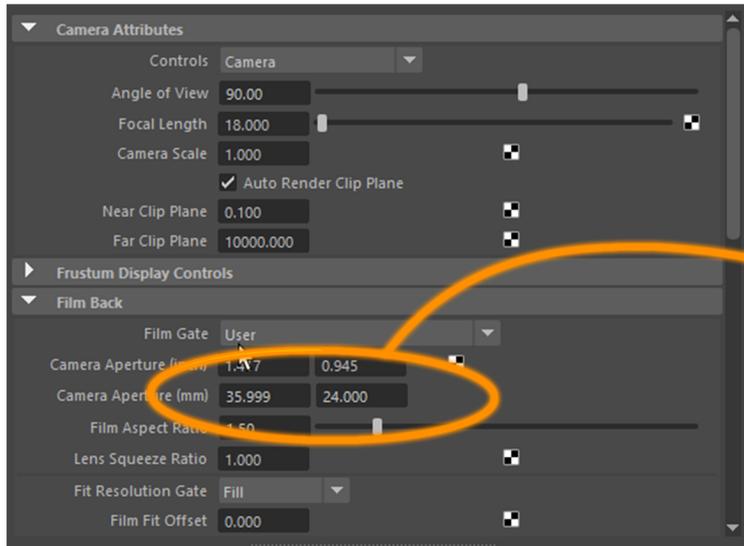
-Uncheck "Reduce Keys"

-Import



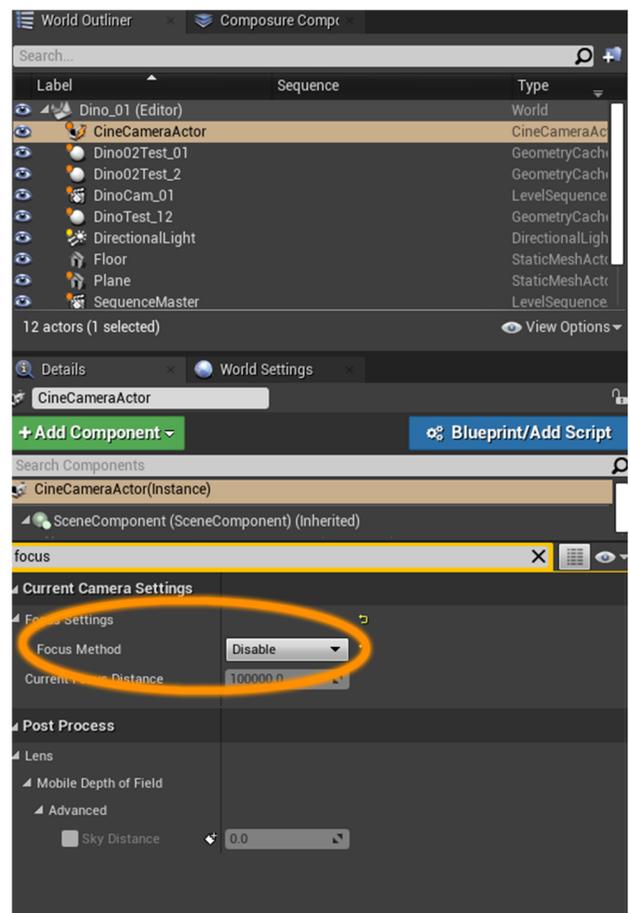
Importing Camera cont.

-In newly created Cine Camera Actor Details panel, change *sensor width and height to match Camera aperture settings (mm) from Maya* (In Maya the first number is Width).



Copy values from Maya camera and past into Unreal - otherwise camera will not match

-In Details panel for the Cine Camera Actor set *Focus Method to Disable*. This could be a bug, but I have found that 4k renders from the Cinematics Movie Render Queue and Sequencer Movie render were blurry unless I set Focus Method to Disable. For most projects, depth of field should be applied in comp stage using depth pass.



Exporting Render Passes from Unreal Engine:

These processes will generate the following passes for use in Nuke:

Beauty
Alpha
Base Color
Ambient Occlusion
Lighting Only
World Normal
Post ToneMapColor
Depth
Shadow
Contact Shadow

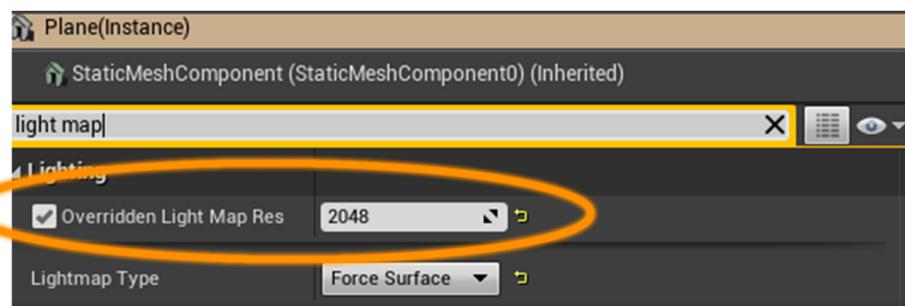
Notes:

-I am using *Unreal Engine 4.26.1*

- Cinematic Movie Render Queue and Sequence Movie Render automatically render the playback range of the sequence. For a single frame, the sequence must be at least two frames long (example playback range 001-002 will only render frame 001)

-Ambient Occlusion and Shadow passes require a plane or other geometry to be visible in the scene at time of render. It is better to use a basic static mesh plane than the default floor plane. Also, you will need to change the Light Map Res setting of the geometry that will be receiving shadows (floor plane). The Default is 128 and will give poor results. I was able to get good results with 2048 setting. This is found in the details panel of the object receiving shadow under the lighting tab. *Check Overridden Light Map Res.* I also had better results when setting the ground plane lightmap type to Force Surface. After

changing the settings and finalizing light positions (I had best results with stationary lights), ***you will need to Build Lighting. This can take a while but is essential to get quality shadows.***



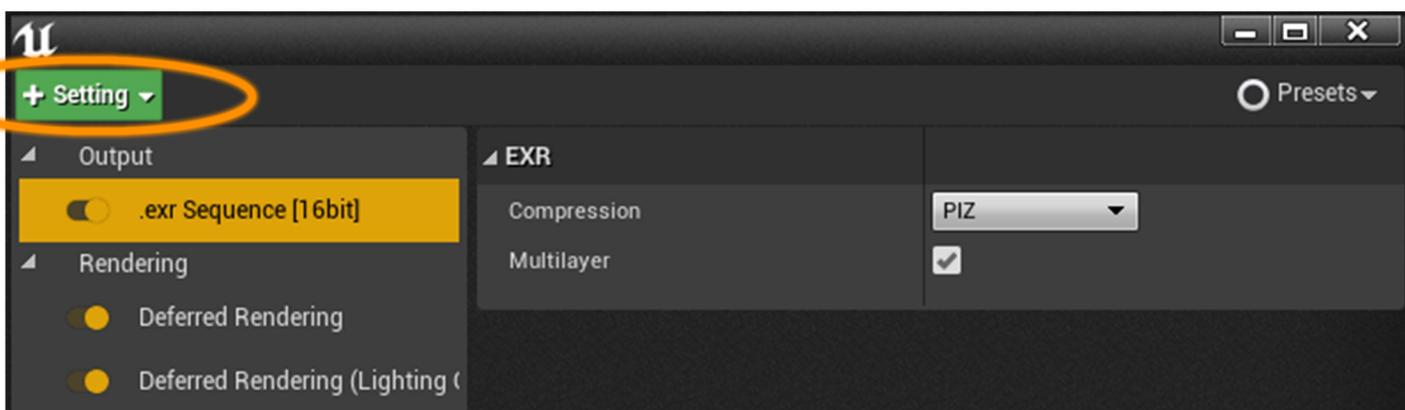
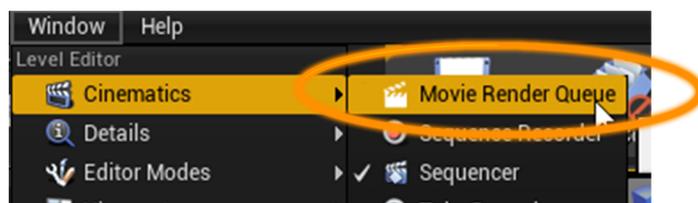
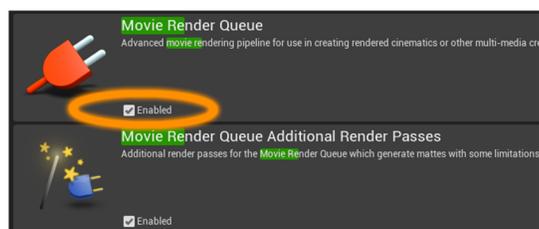
Exporting Render Passes from UE cont.

Beauty, Alpha, and Lighting Only

This process utilizes the **Cinematics Movie Render Queue to render a multichannel .exr** with beauty layer (RGBA) , and Lighting Only layer (RGBA). **Only the Cinematics Movie Render Queue can generate alpha passes** and you can get two different versions of the alpha (hard and softened) from this method. The alpha will be based on any geometry visible in the sequence, so **hide floor plane and any other geo if you do not want it included in the alpha**. The Lighting Only pass can be very useful in comp and can get you close to the Specular pass from Arnold.

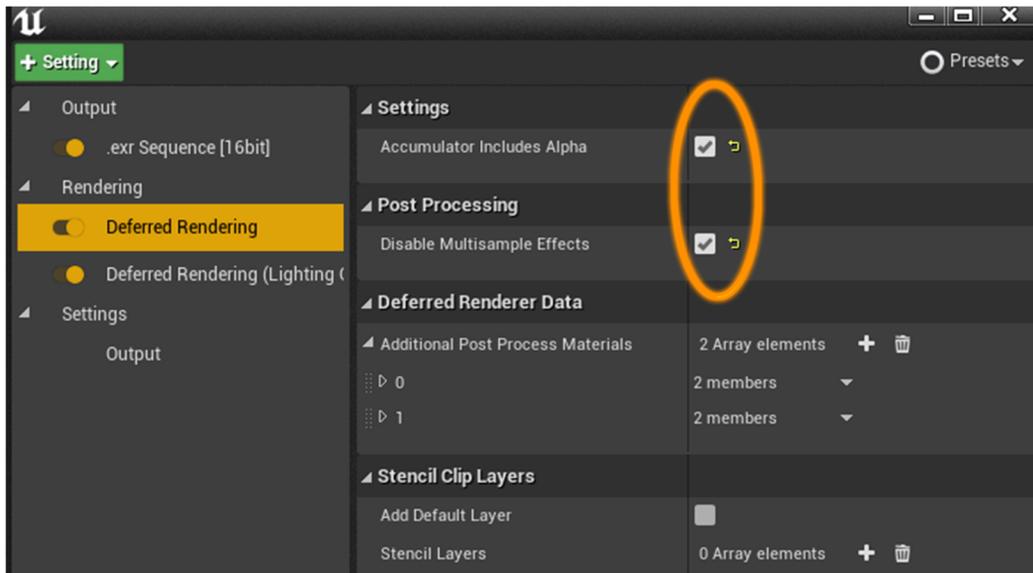
The Process:

- Enable Movie Render Queue and Movie Render Queue Additional Rednder Passes Plugins Found under Edit > Plugins (restart required)
- Window > Cinematics > Movie Render Queue
- Choose sequence to render
- click on unsaved configuration under settings
- Click into green +settings box drop down menu. Select .exr sequence, deferred render, and deferred render(Lighting Only). Delete other image sequence outputs (jpg, bmp etc.)
- .exr sequence settings: compression (PIZ is default and works well). Check multilayer.

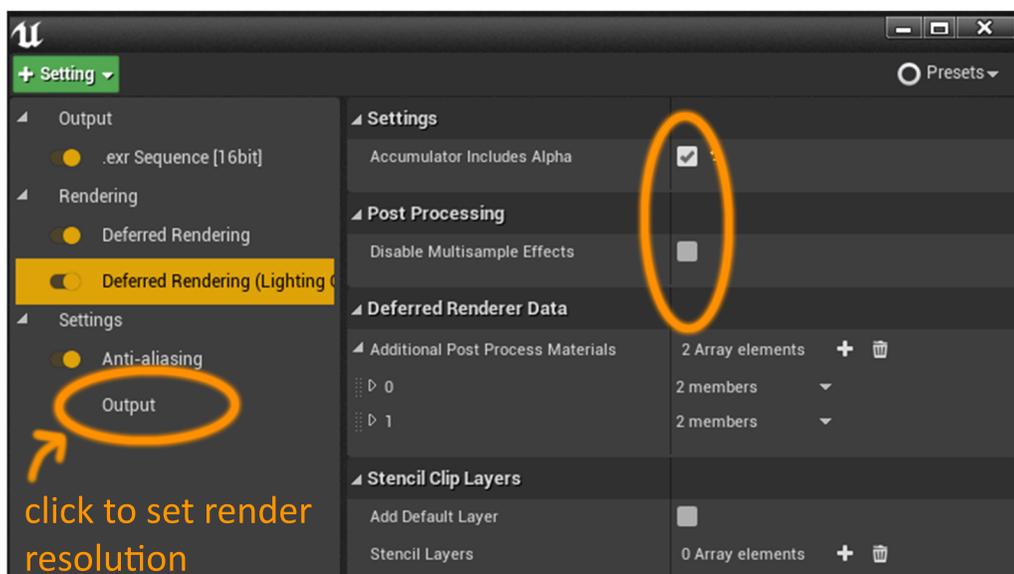


Beauty, Alpha, and Lighting Only cont.

-deferred render settings: *Check Accumulator includes Alpha. Check Disable post process* (disables motion blur on rgba beauty pass. Only Cinematic Movie Render Queue adds motion blur by default)



-deferred render(Lighting Only)settings: *Check Accumulator includes Alpha. Uncheck Disable post process* (enables motion blur). This will give you an alpha channel on the Lighting Only layer that has motion blur, creating softer edges. This provides a second option for the alpha that can be used in comp.



-Choose file resolution and write location in settings (below deferred render) > output
 note – by default your file will be named after the level, and frame number. It will not be name “beauty” unless you change the naming format.

-Accept

-Render Local

Base Color, Post ToneMapColor, Normal, Depth

This process utilizes the **Sequencer to render multiple single rgba layer .exr sequences** to be used in comp. *note -The shadows will be dealt with separately, so *any floor planes or other geo that you don't need rendered should be hidden* by unchecking Visibility and checking Actor Hidden in Game in the detail panel of selected actor (mesh object).*

-Select render icon above track listings in sequencer and set Render Movie Settings as follow:

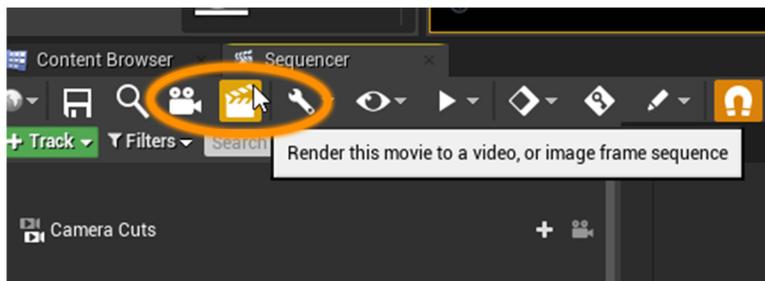


Image Output Format:
Custom Render Passes

Included Render Passes:
Base Color
Scene Depth
World Normal
Post Tonemap HDR Color

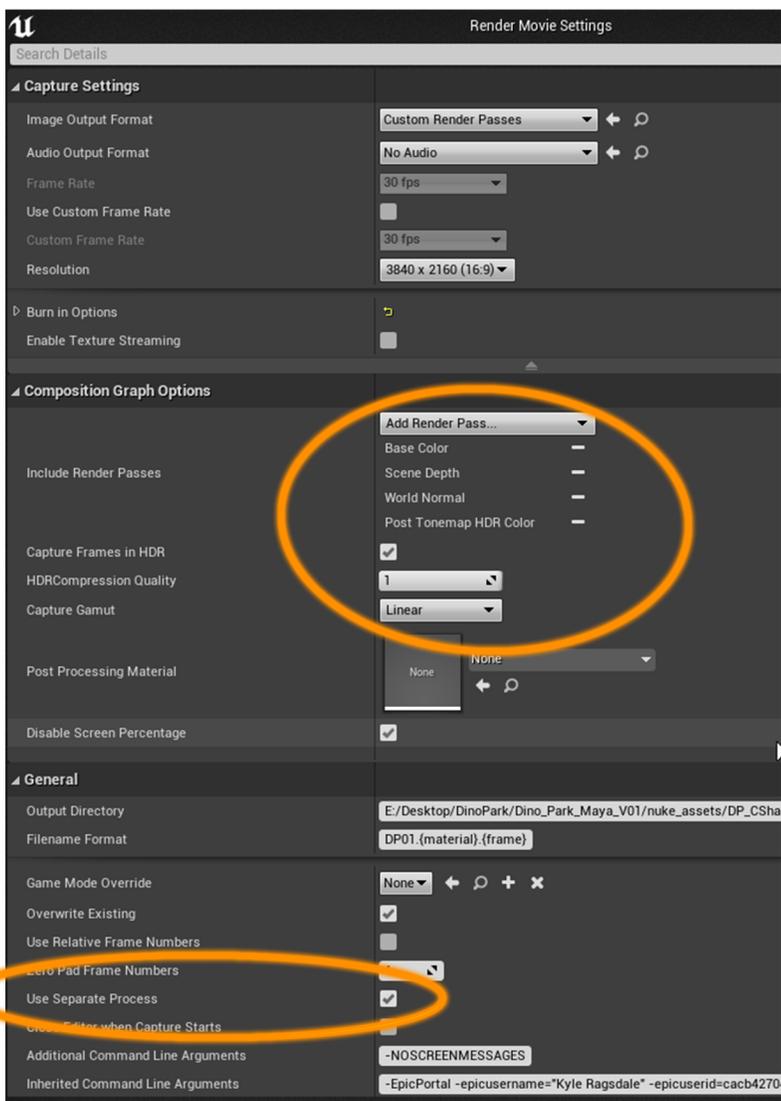
Check Capture Frames in HDR
(this changes output to .exr files)

HDRCompression Quality:
Default is 1 and gives great result

Capture Gamut:

Rec. 709 / sRGB is standard and will work, but to avoid having to change colorspace in Nuke, select *Linear*.

Check Use Separate Process
(will not render correct resolution without this checked)

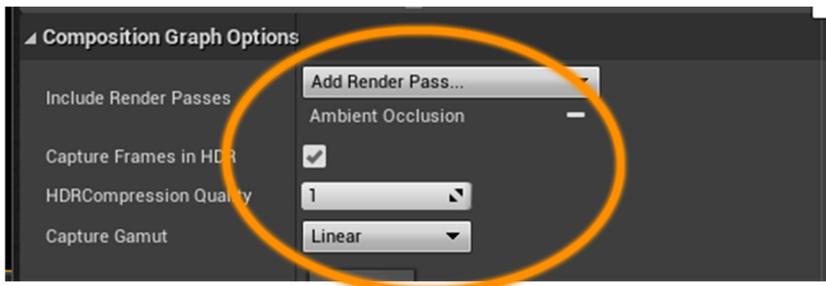


-Click *Capture Movie* *note – this will open a fullscreen render window that may appear black if your output resolution is larger than your viewer screen and your objects are not in the visible part of the frame. The render process will take a few minutes depending on your project and settings.*

Ambient Occlusion:

Before rendering AO pass, make sure ground plane is visible and not hidden in game. Refer to note on page 10 regarding *Override Light Map Res.*

The process is exactly the same as the previous render passes (color etc..) using the Sequencer's Render Movie Settings, but with only Ambient Occlusion Render pass selected.

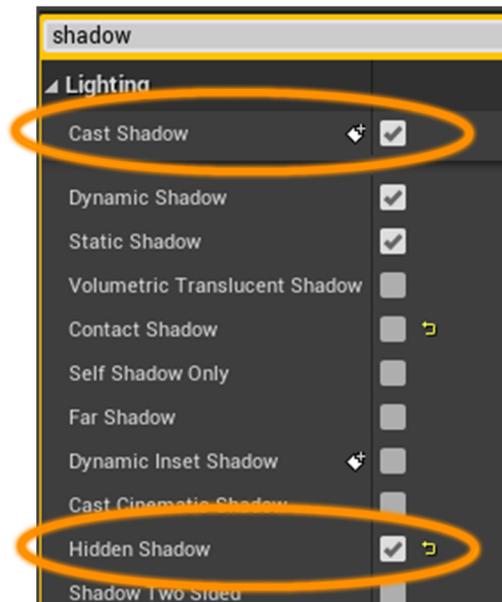
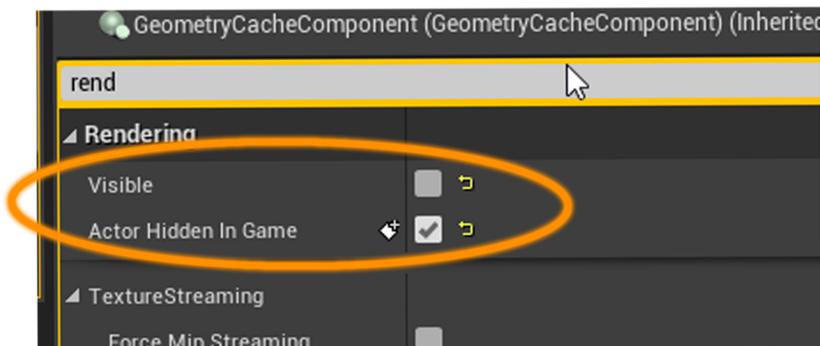


The values of the AO pass are very low so you will need to adjust black point and white point in Nuke to be able to use.

Shadow:

The best method I have found for creating a cast shadow layer is to render a **Final Image render pass though the Sequencer with the object hidden, but the shadows shown**. Utilizing a simple roto and grade in Nuke, you can get a matte to use for shadows.

To achieve this, go to the Details panel of the object you want to cast a shadow, *uncheck Visible, check Actor Hidden In Game, make sure Cast Shadows is checked, and check Hidden Shadow*. This only works with cast shadows as contact shadows will not be rendered when object is hidden.



Contact Shadow:

Rendering contact Shadows from UE are a challenge. For Contact Shadows, I also use a *Final Image* render pass similarly to the cast shadow, but with the *object Visible*, and *Cast Shadows turned off*.

Make the object visible (*check Visible, uncheck Hidden In Game*). *Uncheck Cast Shadow*.

The correct control for the contact shadow is on the light(s).

In the Details panel of the light, change *Contact Shadow length* to get close to the results you want. The quality is limited but gives you something to work with in Nuke. You will have to mask out the main object and use roto to achieve desired effect, but at least you have something to work with.

Additional Resources:

<https://docs.unrealengine.com/en-US/WorkingWithContent/Importing/AlembicImporter/index.html>

<https://docs.unrealengine.com/en-US/AnimatingObjects/Sequencer/HowTo/GeometryCacheTrack/index.html>

<https://docs.unrealengine.com/en-US/RenderingAndGraphics/Materials/HowTo/Instancing/index.html>

Export Alembic Animations to Unreal Engine the correct way
https://youtu.be/b_FvqQyyYFk

How to IMPORT an FBX Camera to Unreal (Maya)
<https://youtu.be/qtxLmWquOOY>

Created by Kyle Ragsdale
The Los Angeles Film School - 2021
kyleragsdalevfx.com
kyleragsdalevfx@gmail.com