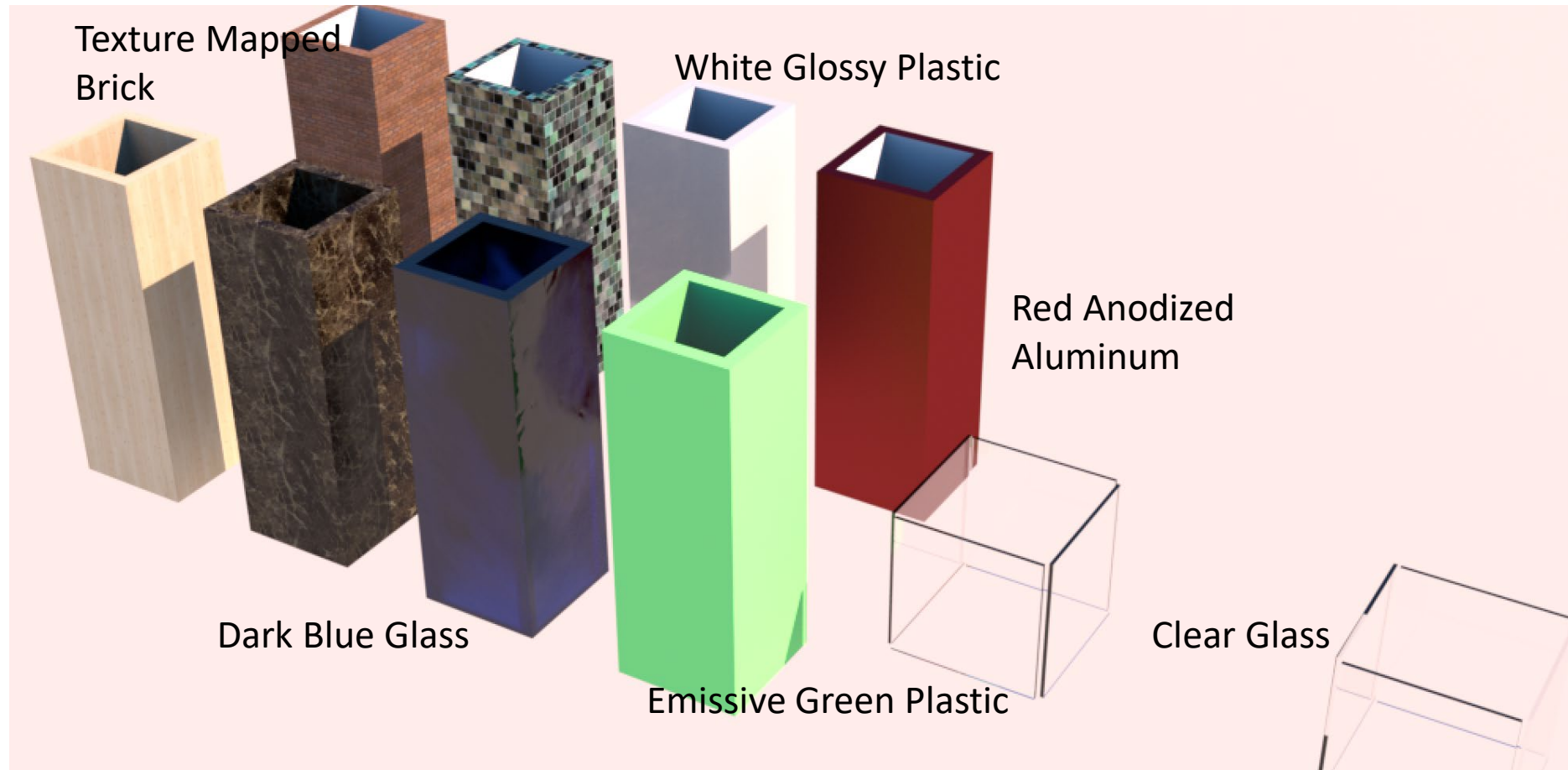
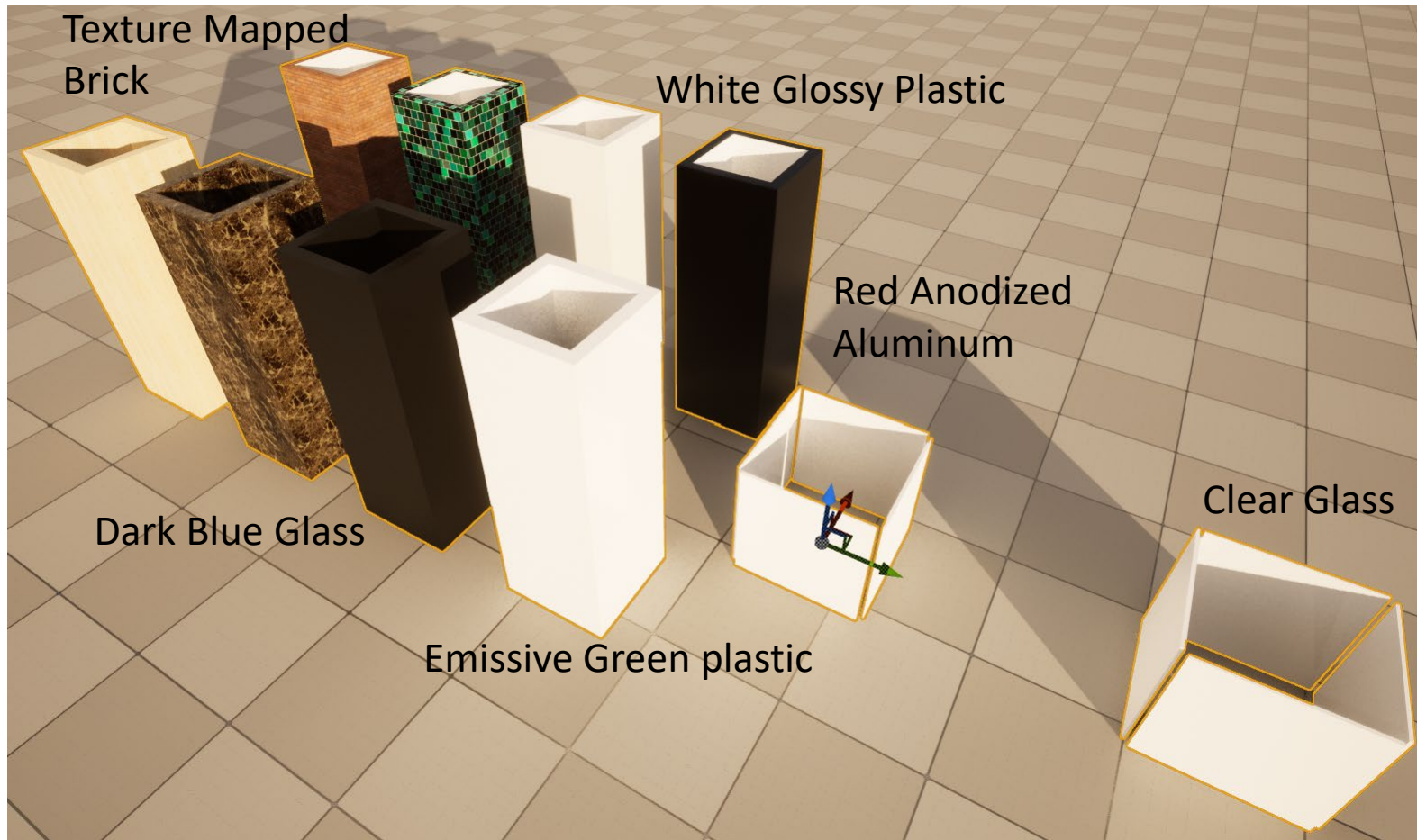


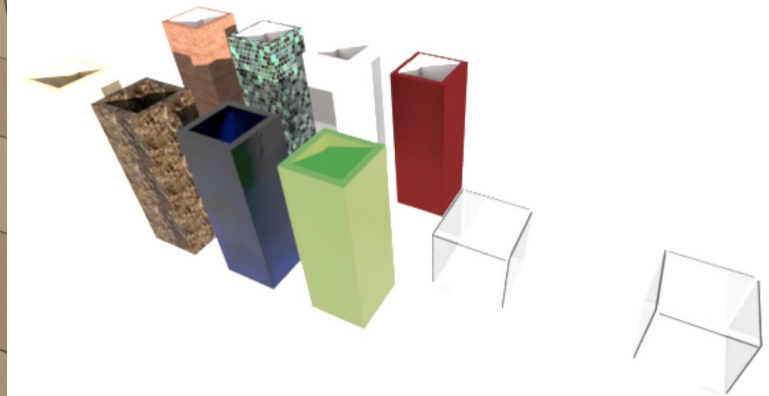
Original Octane Scene, materials from live DB – rendered in octane standalone



Orbx file after import via octane plugin into UE5, note only materials with texture maps appear similar to the original octane renderings.

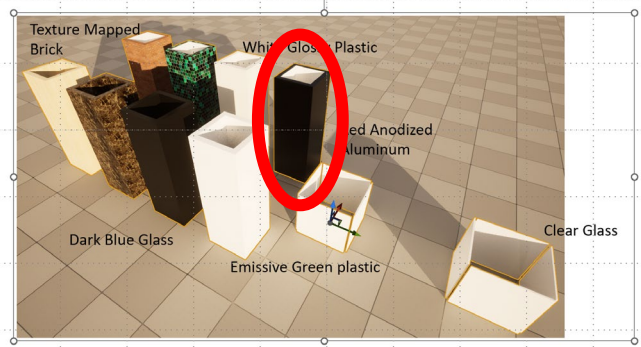


Below, Octane viewport rendering from inside UE5. Also note rendering is burnt out.

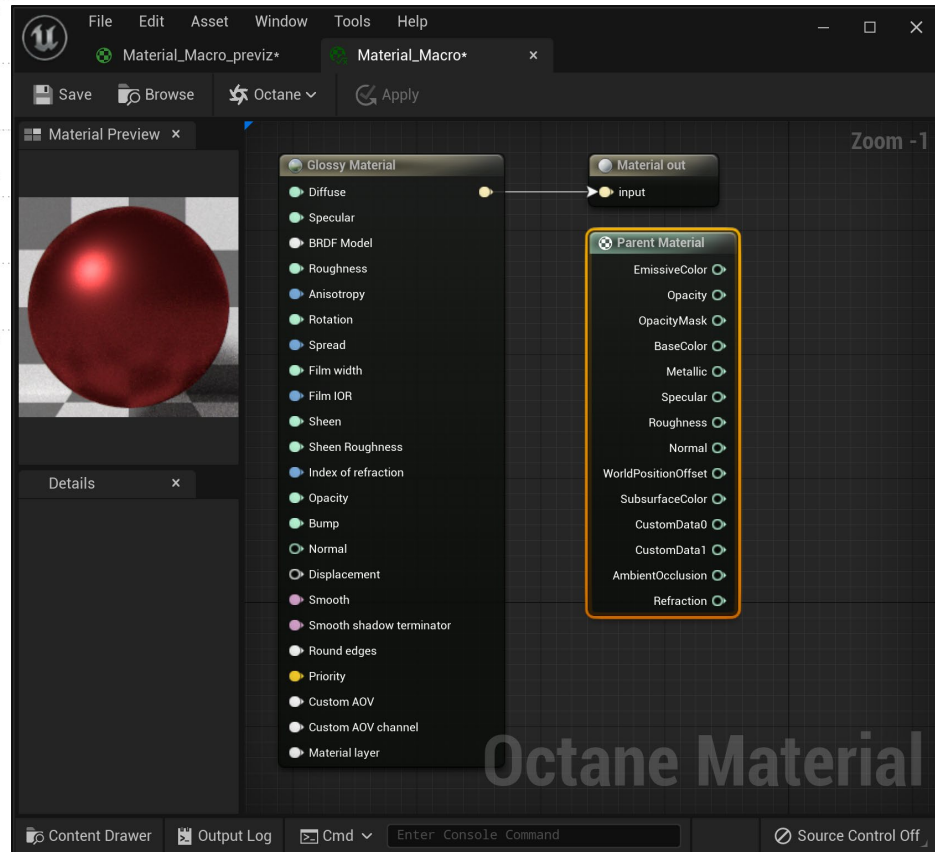


- The following slides look at both the Octane and the Octane converted material definition in UE5 for those materials that lost fidelity in the conversion.

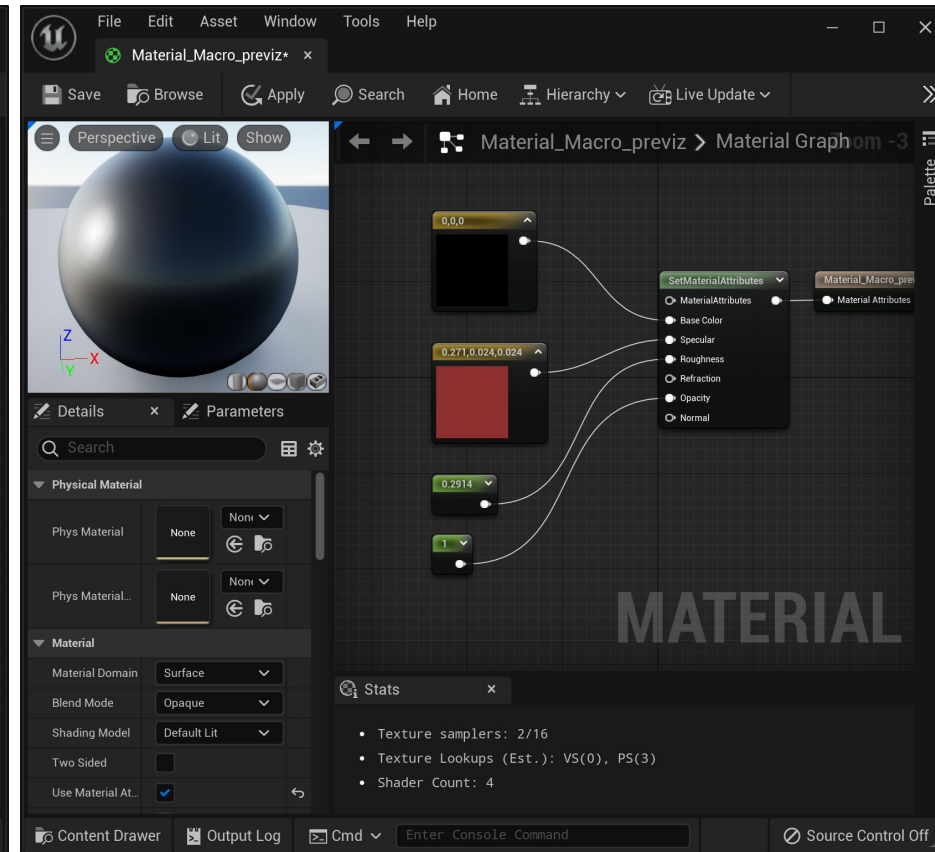
Red Anodized.



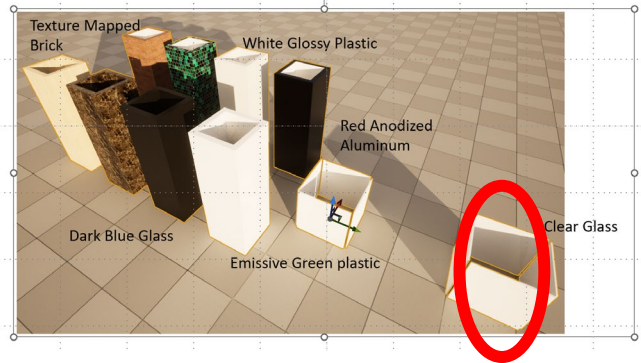
Octane material,
accurate representation of standalone
rendering



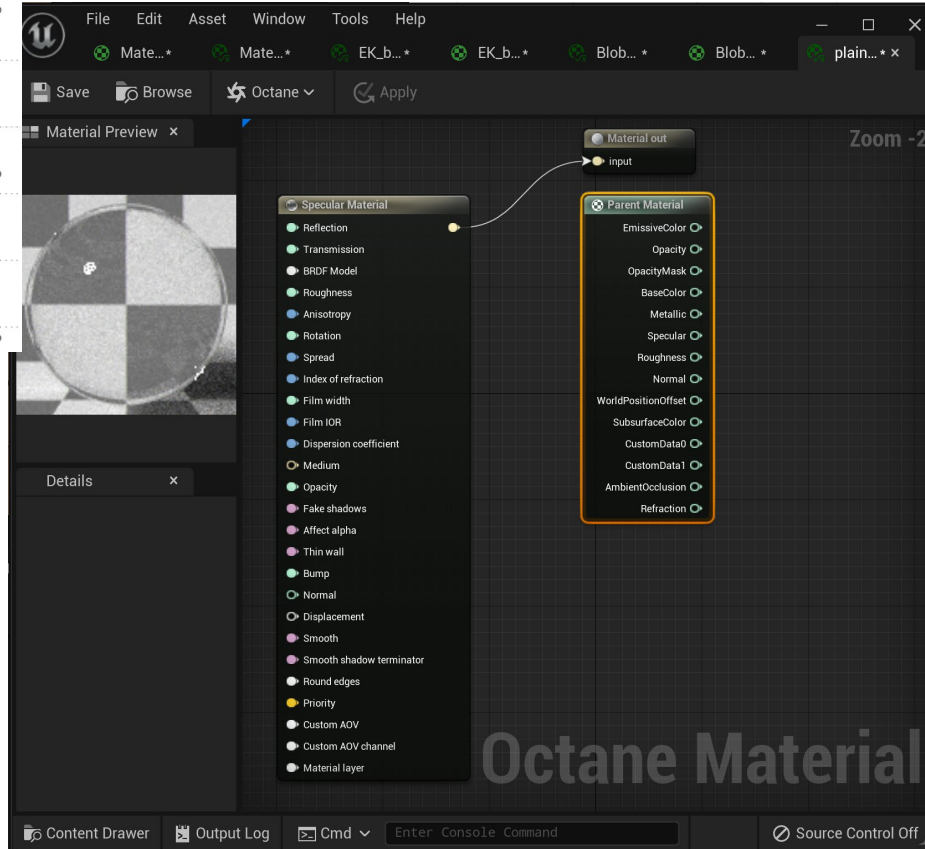
Previz material,
used in project but in accurate



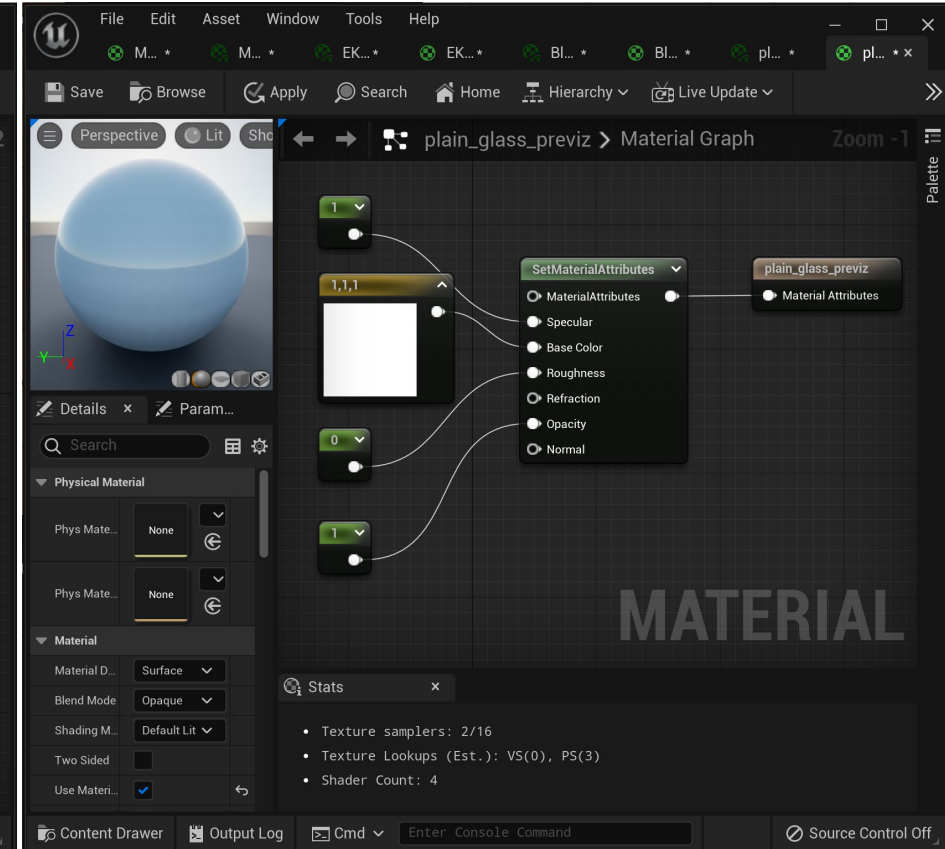
Clear Glass.



Octane material,
accurate representation of
standalone rendering

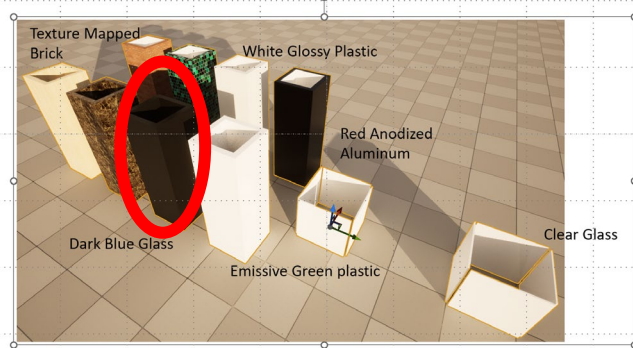


Previz material,
used in project but inaccurate, not
faithfully rendered by UE5

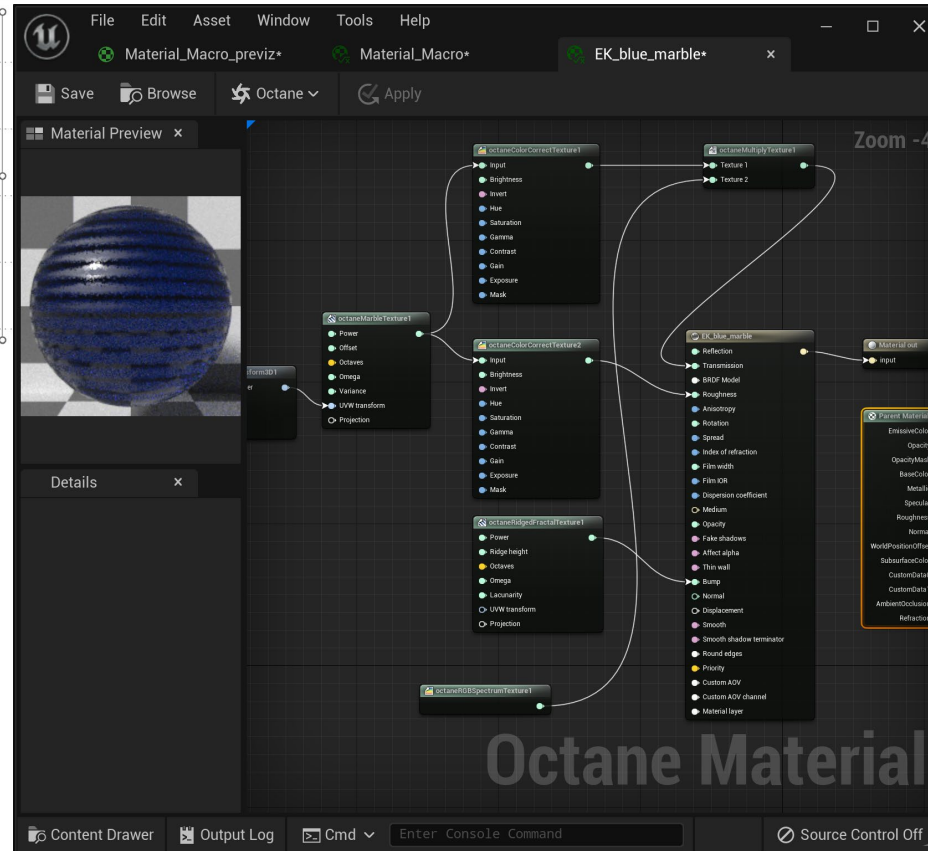


Previz material,
Lacks transparency, color is incorrect

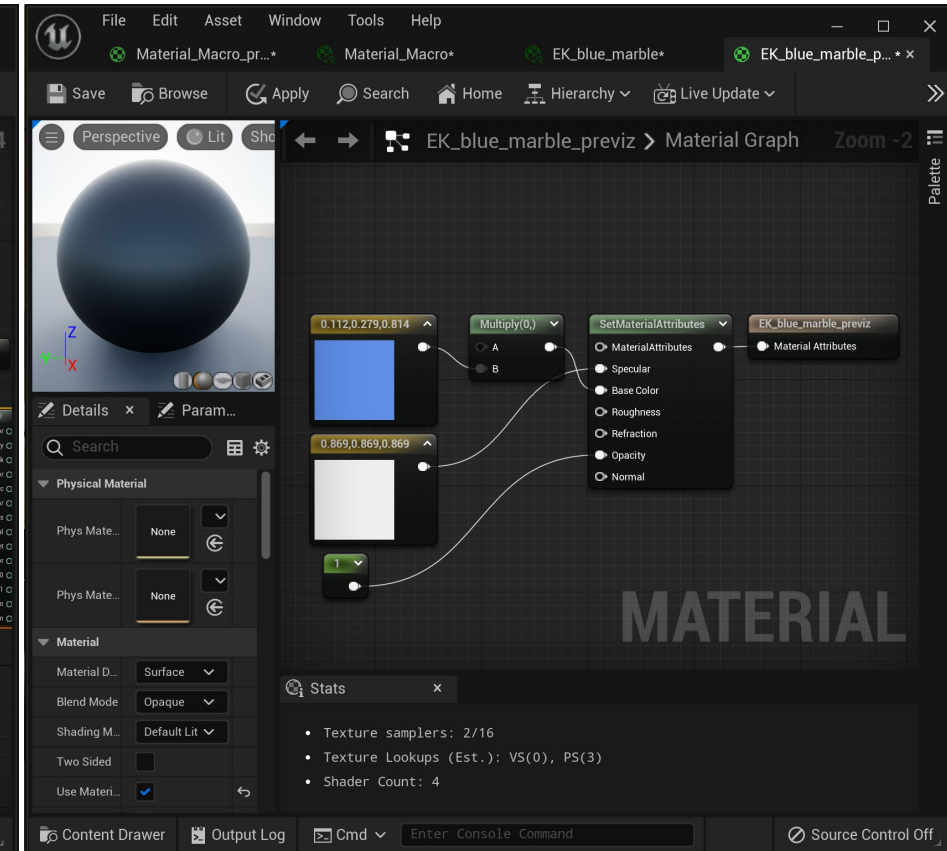
Dark Blue Glass.



Octane material,
accurate representation of standalone
rendering

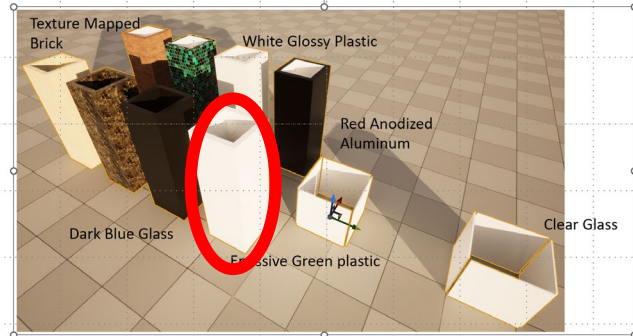


Previz material,
used in project but inaccurate

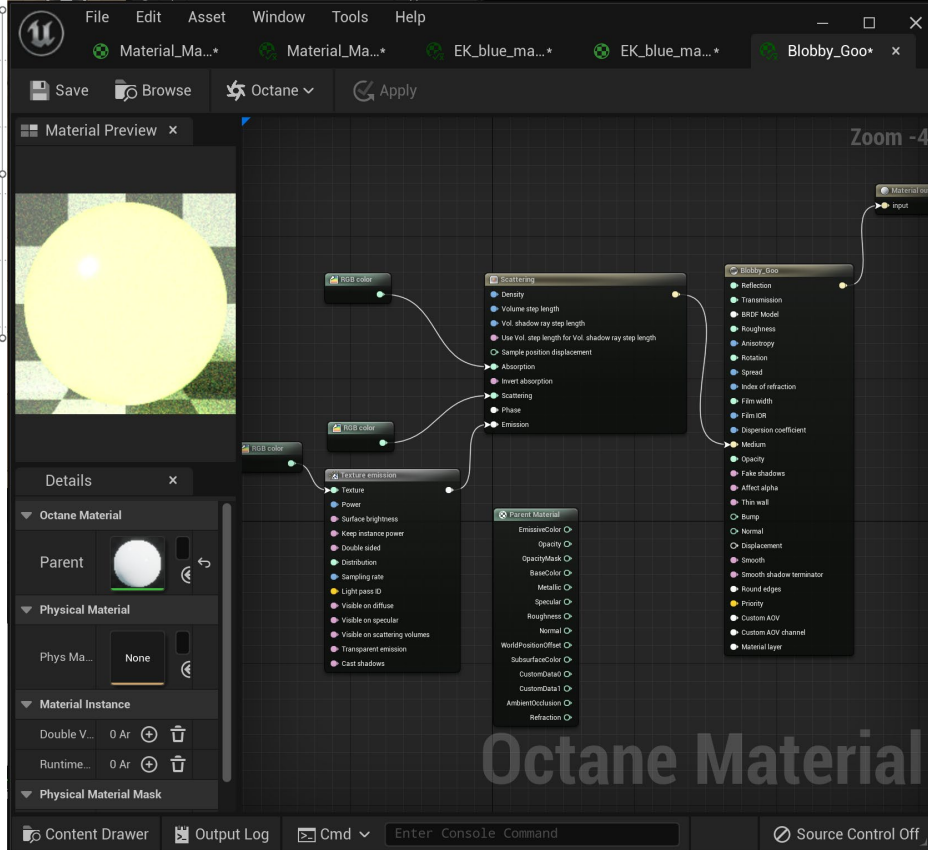


Previz material,
Lacks transparency and reflections

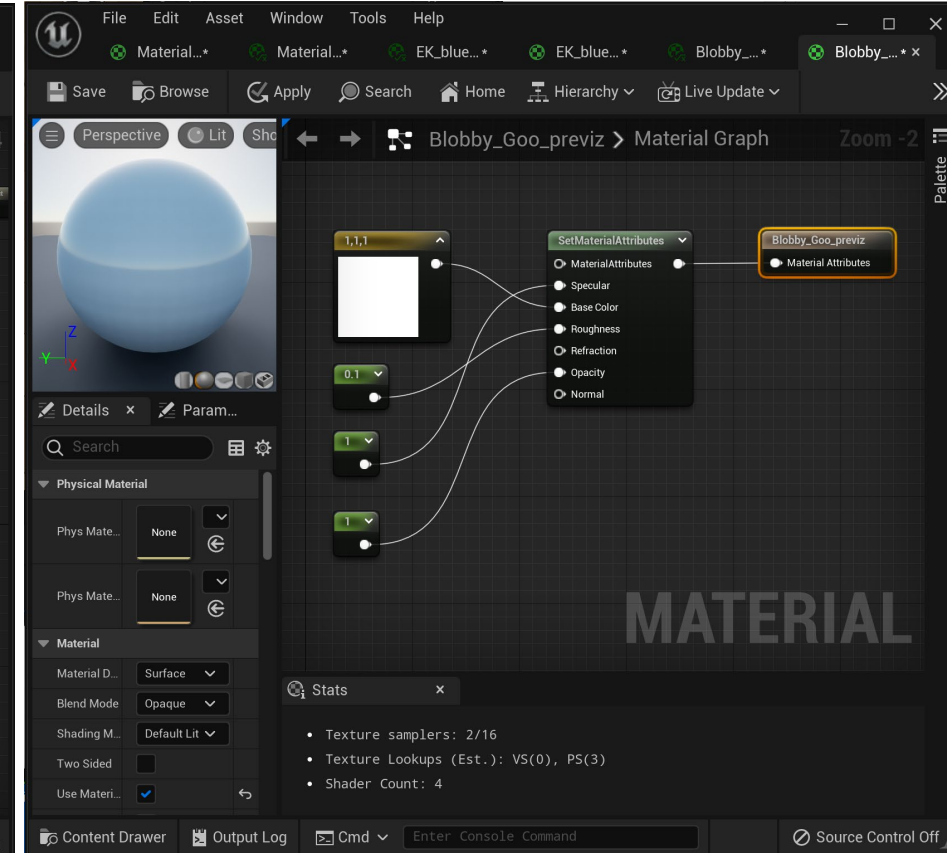
Emissive Green.



Octane material,
Moderately accurate representation of
standalone rendering

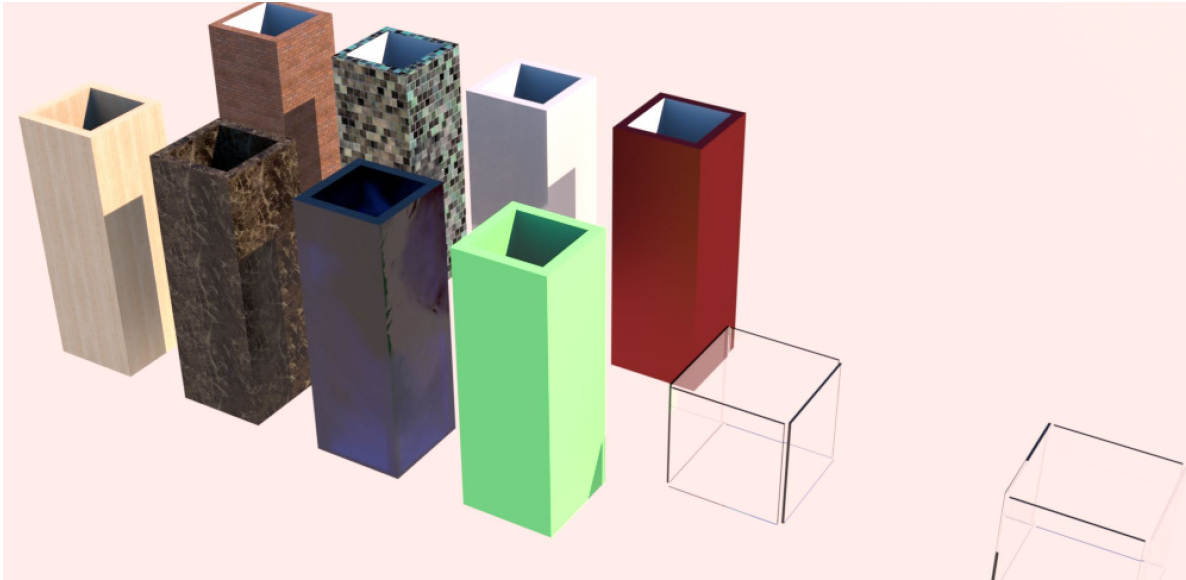


Previz material,
used in project but inaccurate

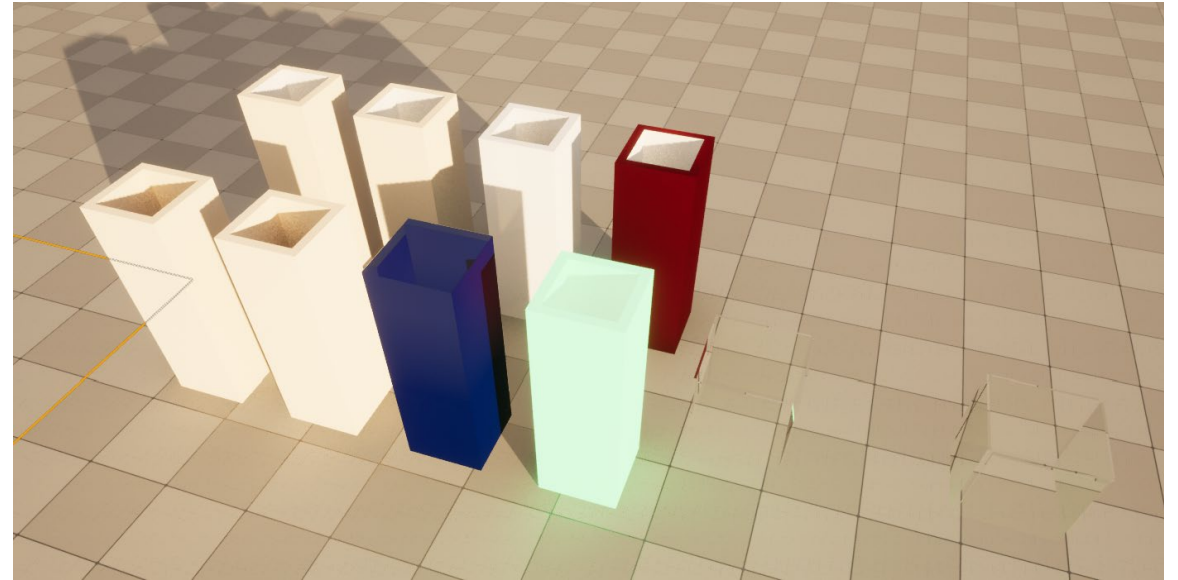


Previz material,
Lacks emissive, color incorrect

Octane Standalone



UE5 with revised materials as defined above

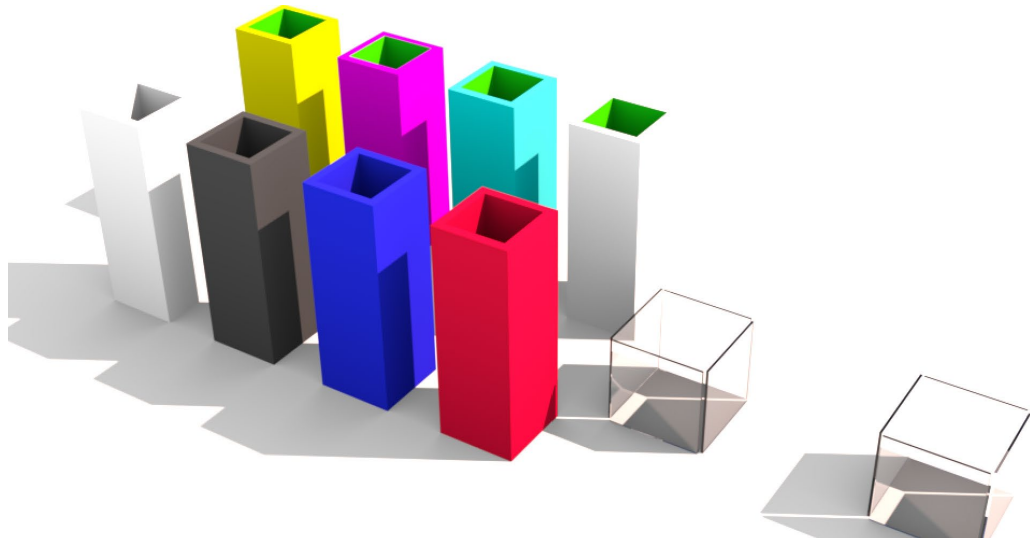


Testing basic material types.

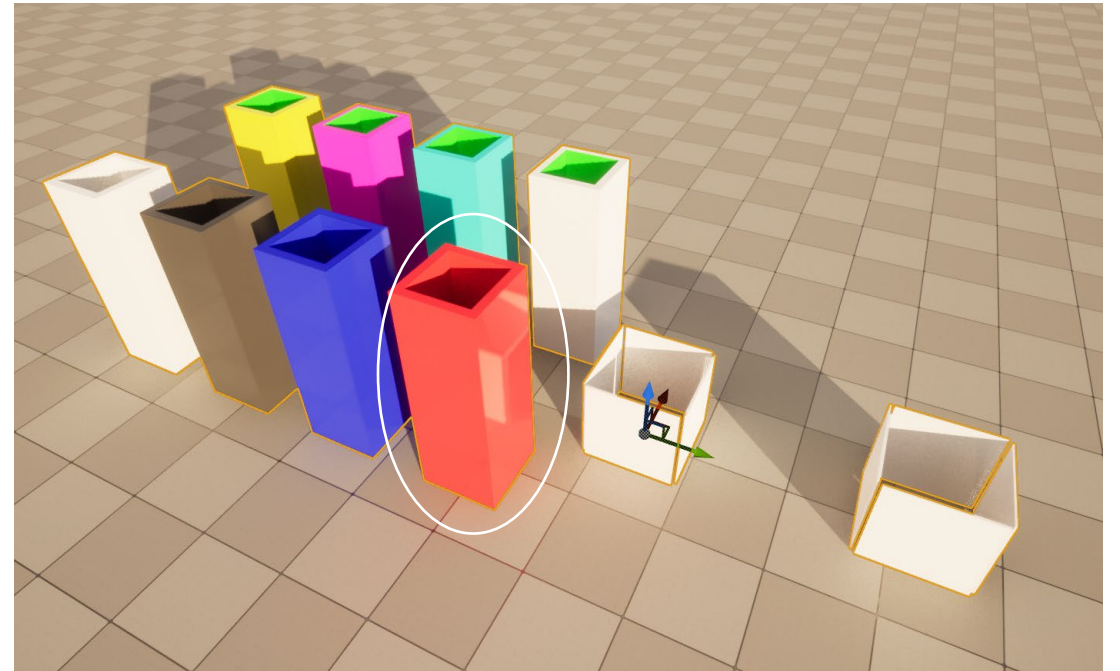
- The following slides show what happens to basic Octane material types, diffuse, glossy, specular, metallic, emissive when they are converted to UE5 via the orbix file format.

all diffuse materials convert with a gloss

Render Octane Stand Alone

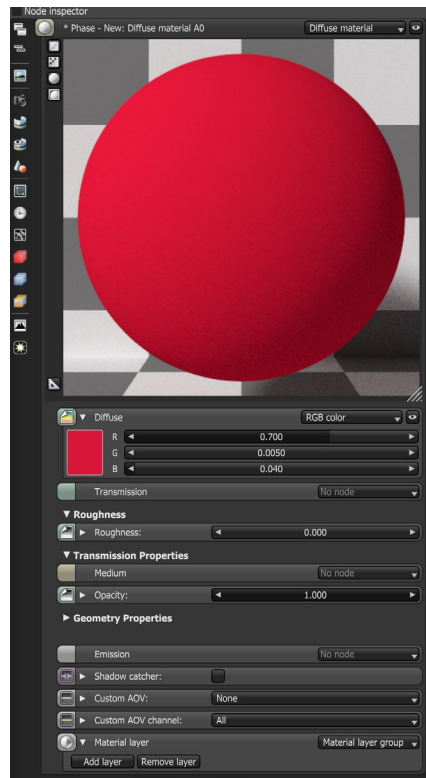


Orbx Import Rendered in UE5

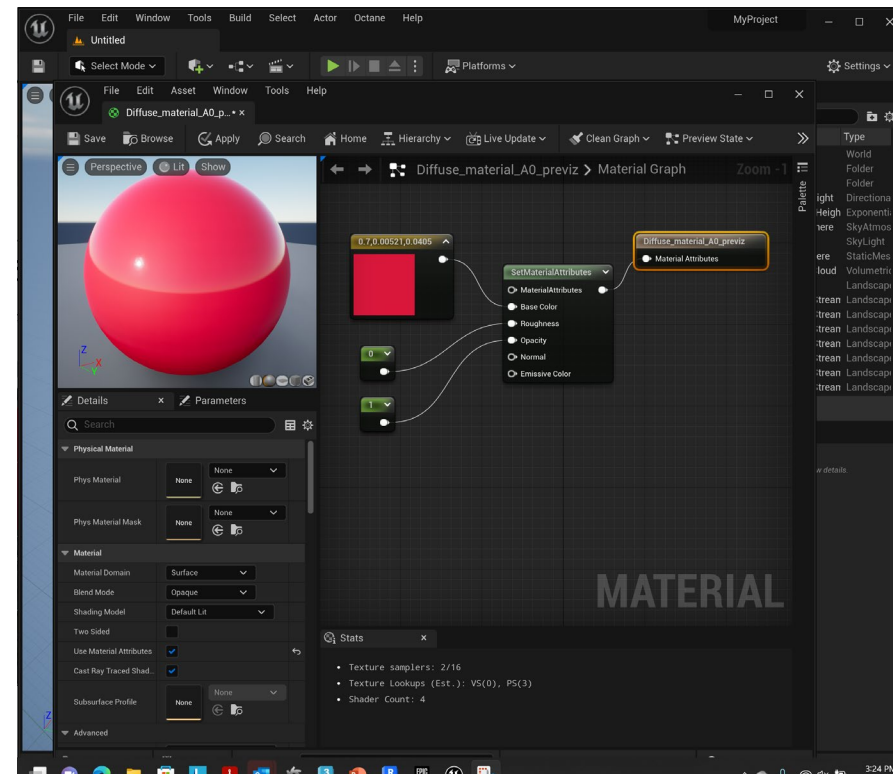


Octane diffuse

Octane Stand Alone parameters

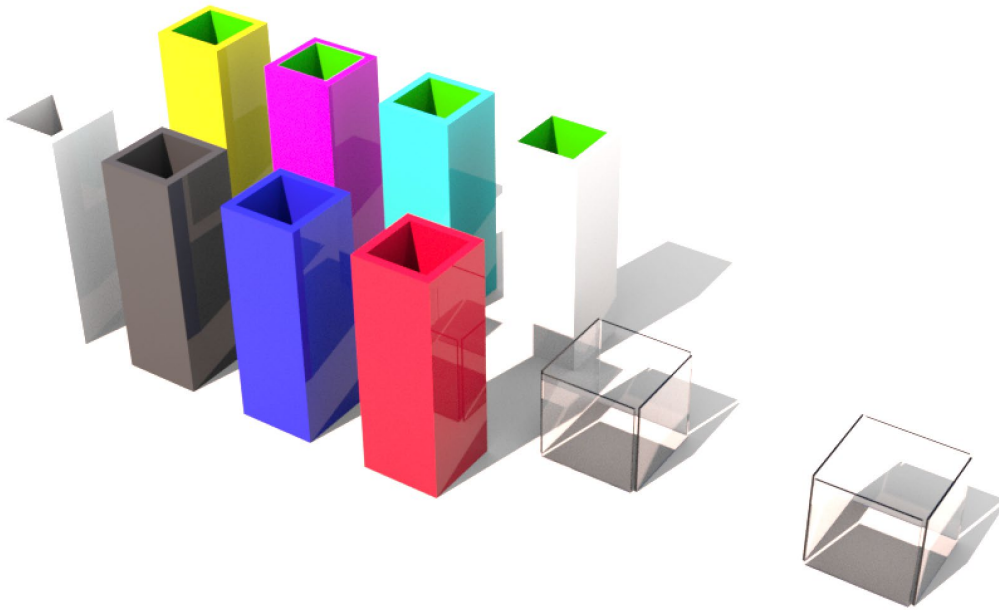


Orbx Import previz parameters

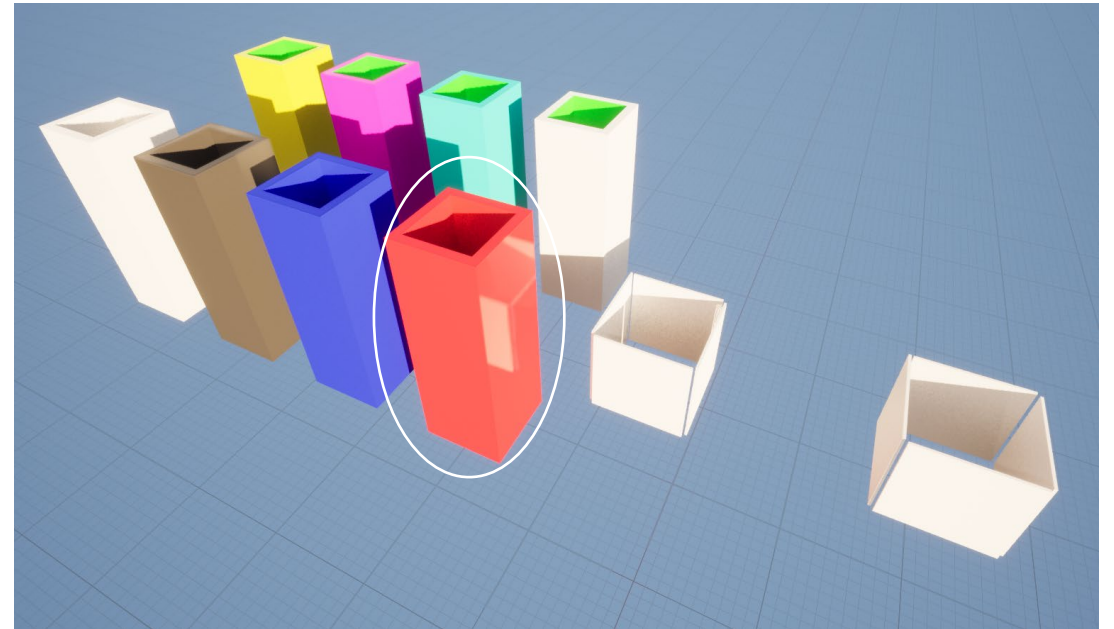


all gloss materials convert with a gloss

Rendered in Octane Stand Alone

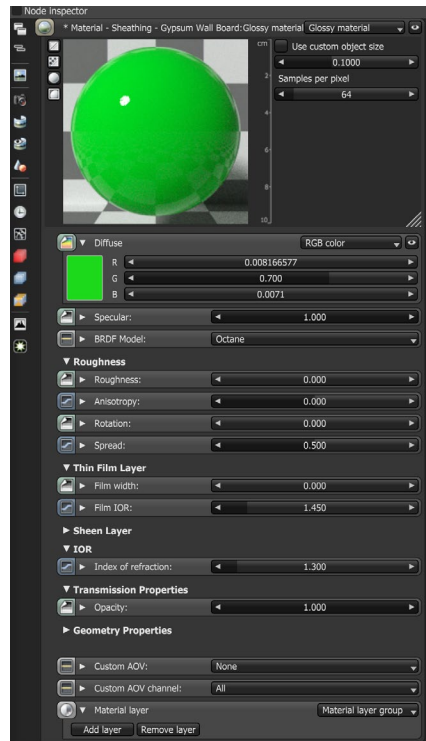


Orbx Import Rendered in UE5

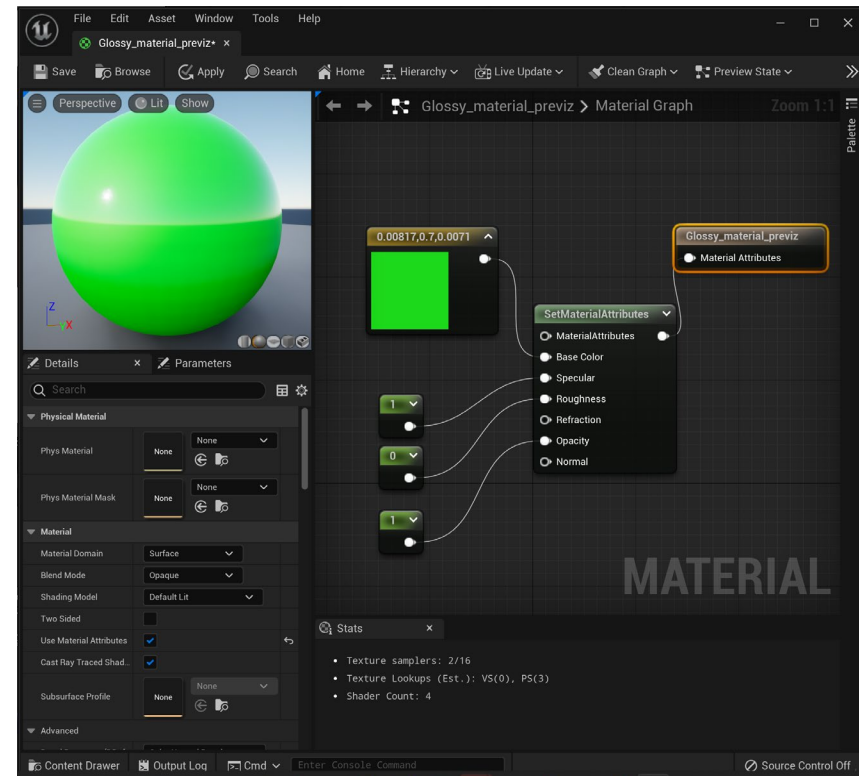


Octane Gloss

Octane Stand Alone parameters

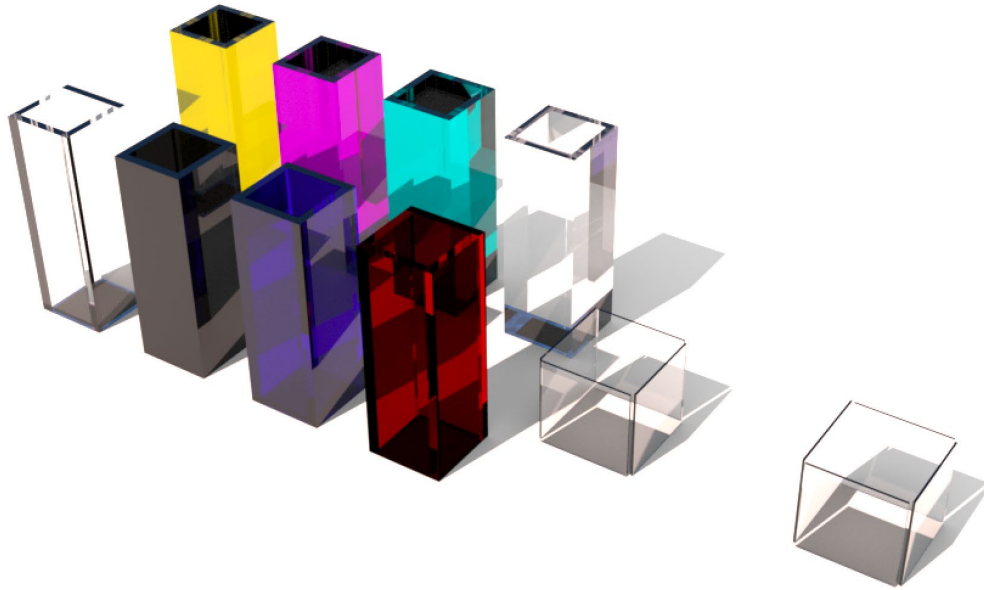


Orbx Import previz parameters

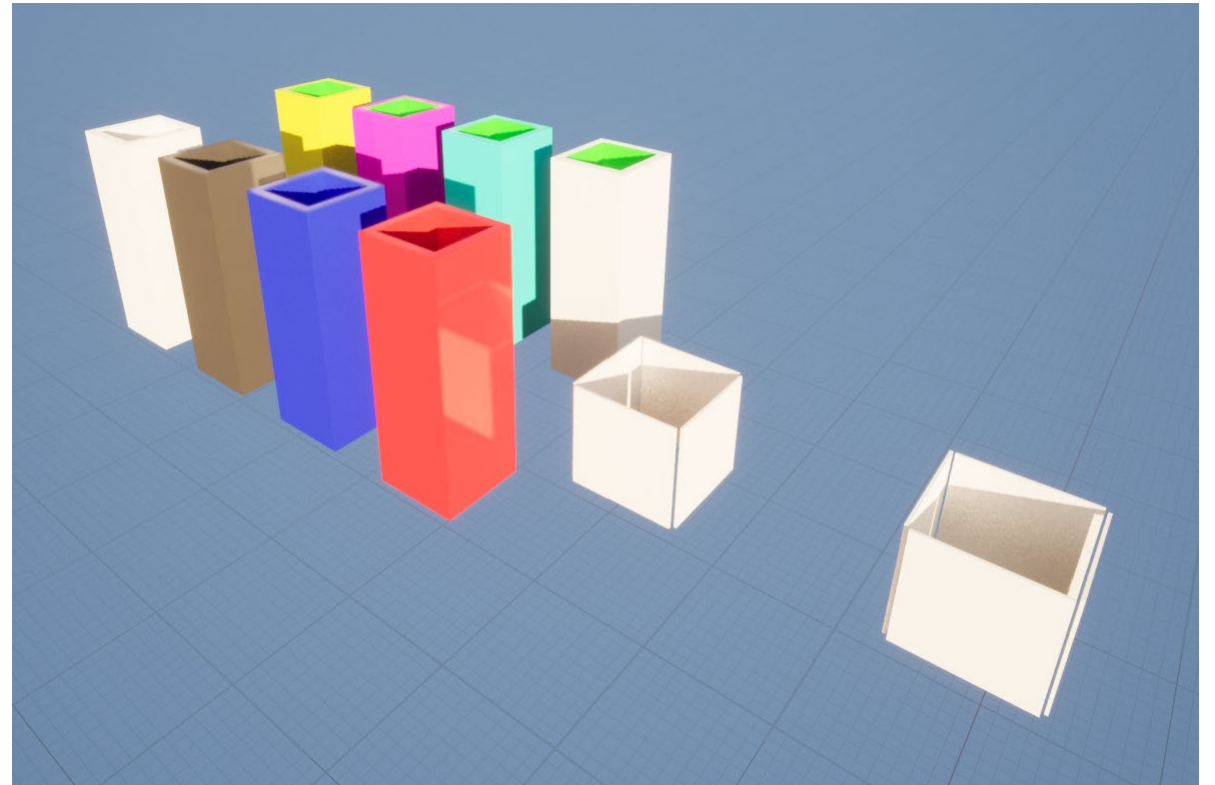


all specular materials convert without transparency

Rendered in Octane Stand Alone

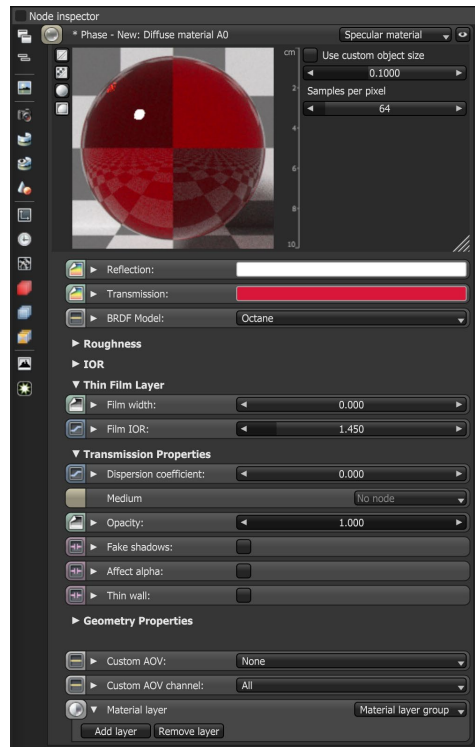


Orbx Import Rendered in UE5

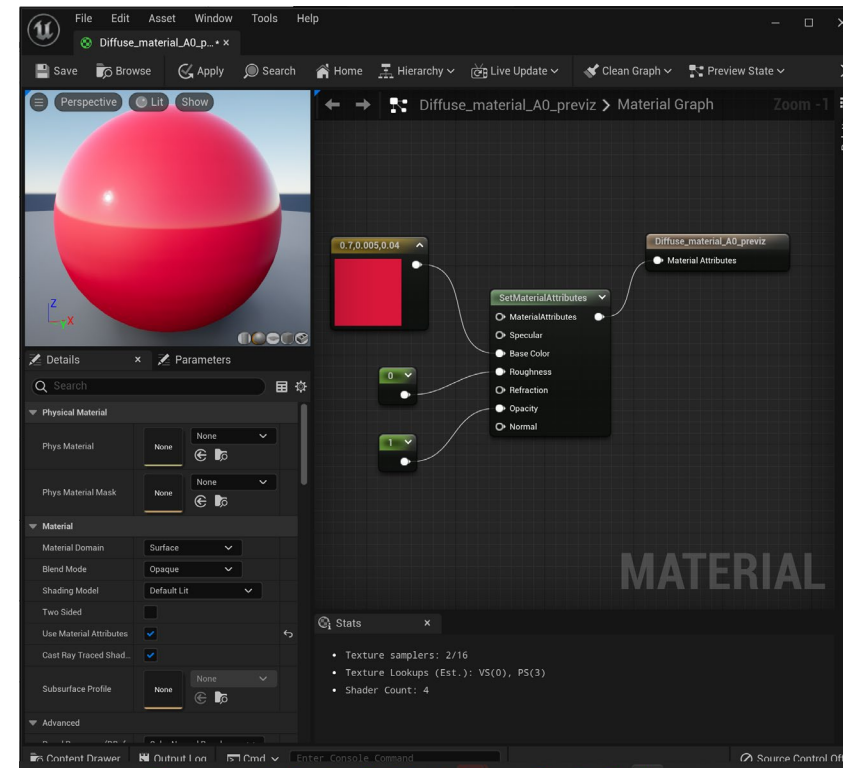


Octane Specular

Octane Stand Alone parameters

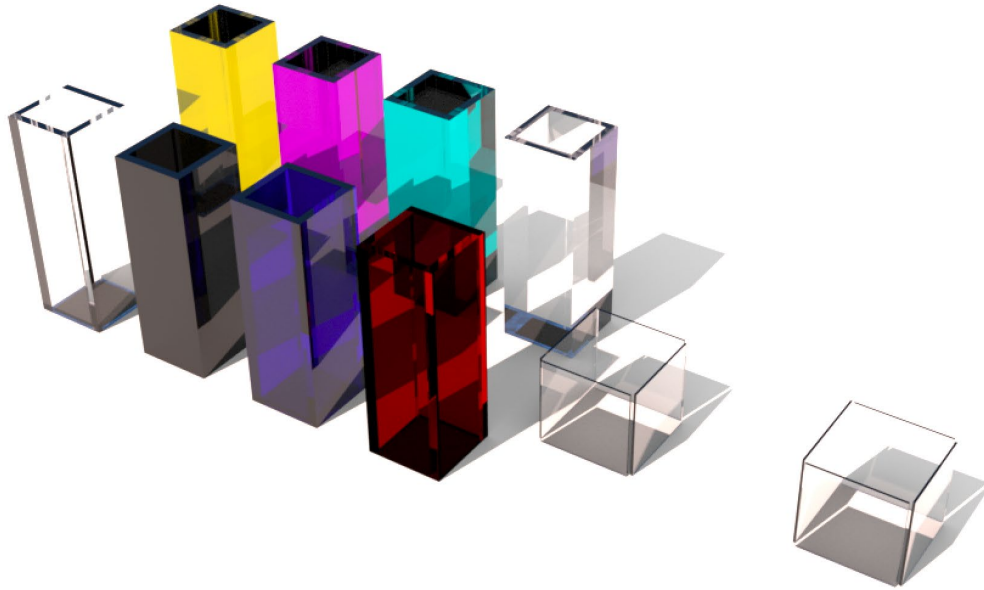


Orbx Import previz parameters

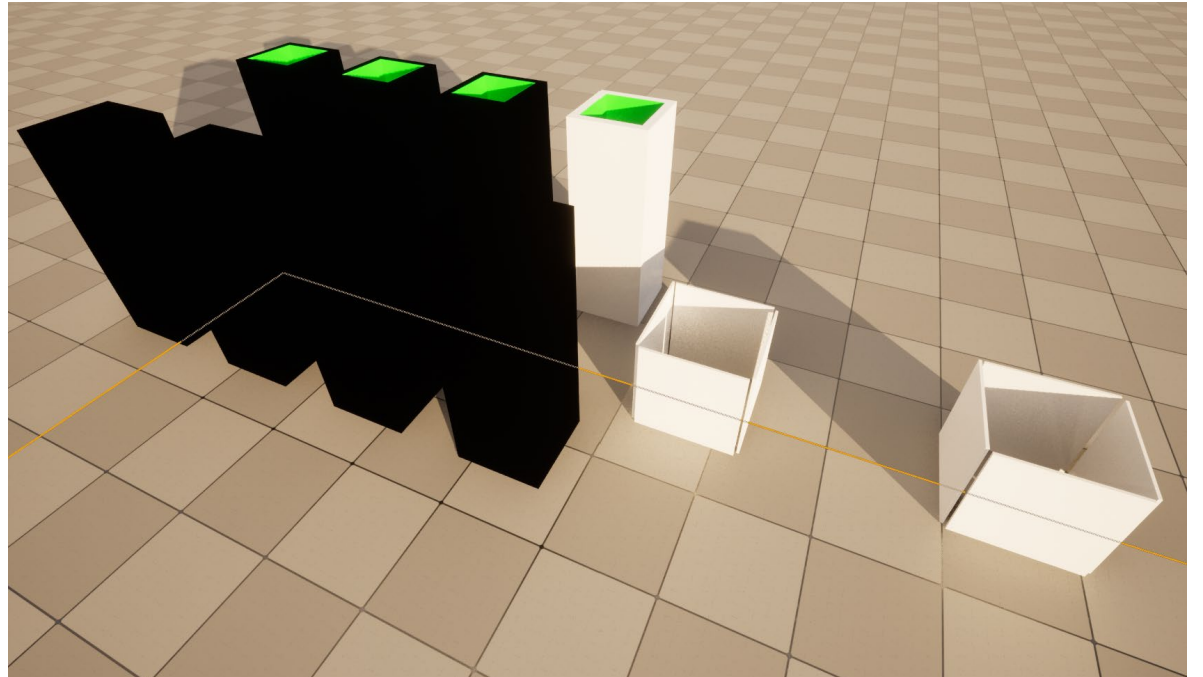


Issue – all metallic materials convert to black

Rendered in Octane Stand Alone

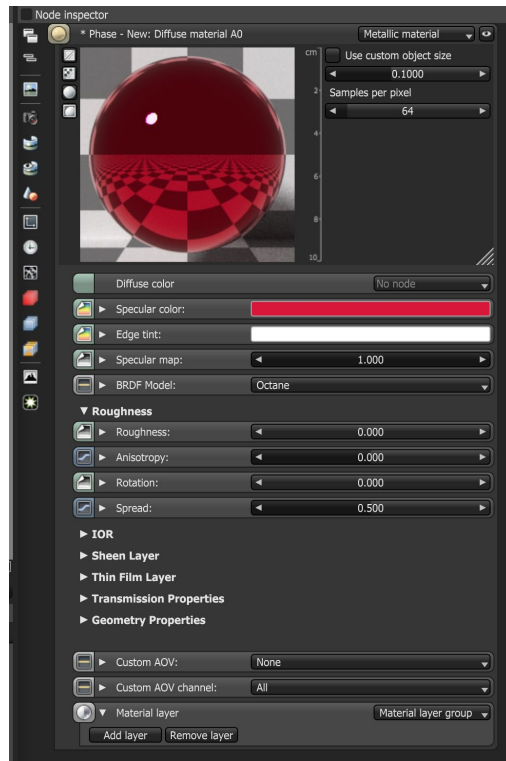


Orbx Import Rendered in UE5

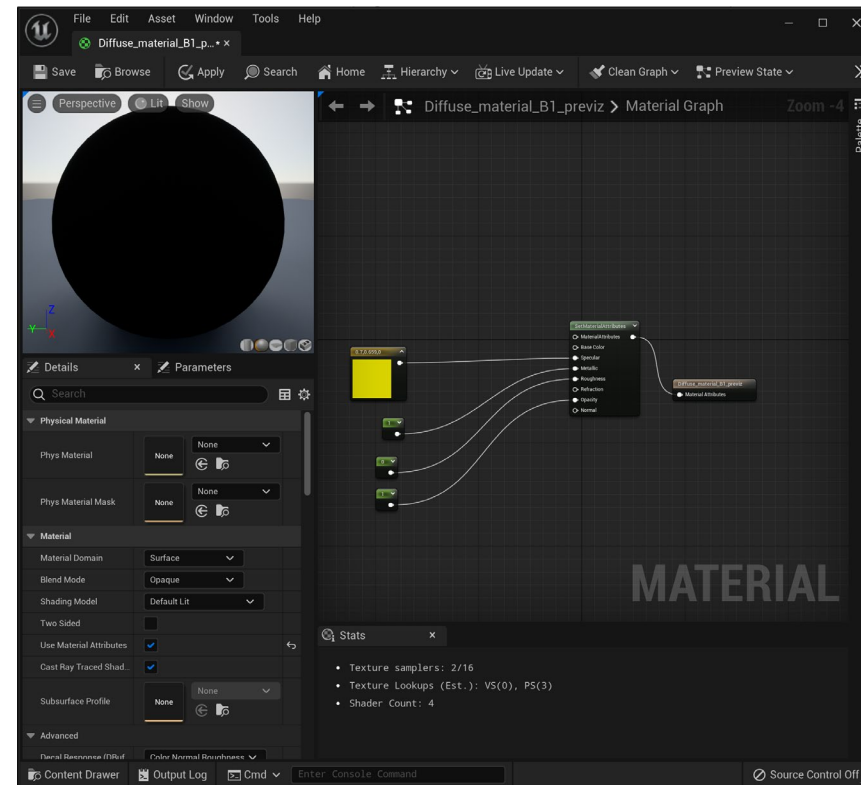


Octane Metallic

Octane Stand Alone parameters



Orbx Import previz parameters



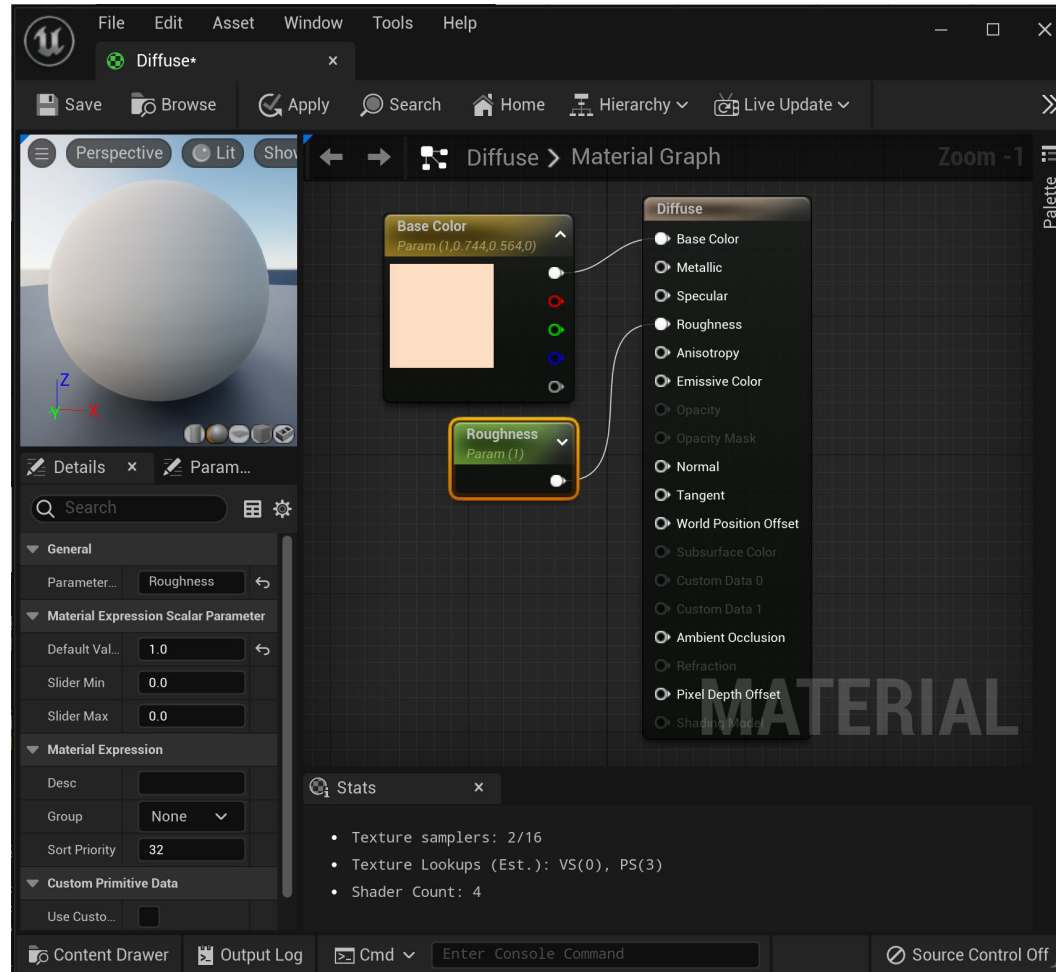
- The following slides show simple UE5 material definitions used to replace the previous Octane materials converted to UE5 for those materials that lost fidelity in the conversion.
- It appears that creating simple UE5 materials based on the following types can improve fidelity. They conveniently map to Octanes Primary materials and important parameters. One could also create a 'Universal' material and connect/disconnect the wires differently according to following types.

1. Defuse
2. Gloss
3. Specular
4. Metallic
5. Emissive

FIX

Diffuse Material.

Only needs two UE5 parameters both of which are present in the octane material definition.



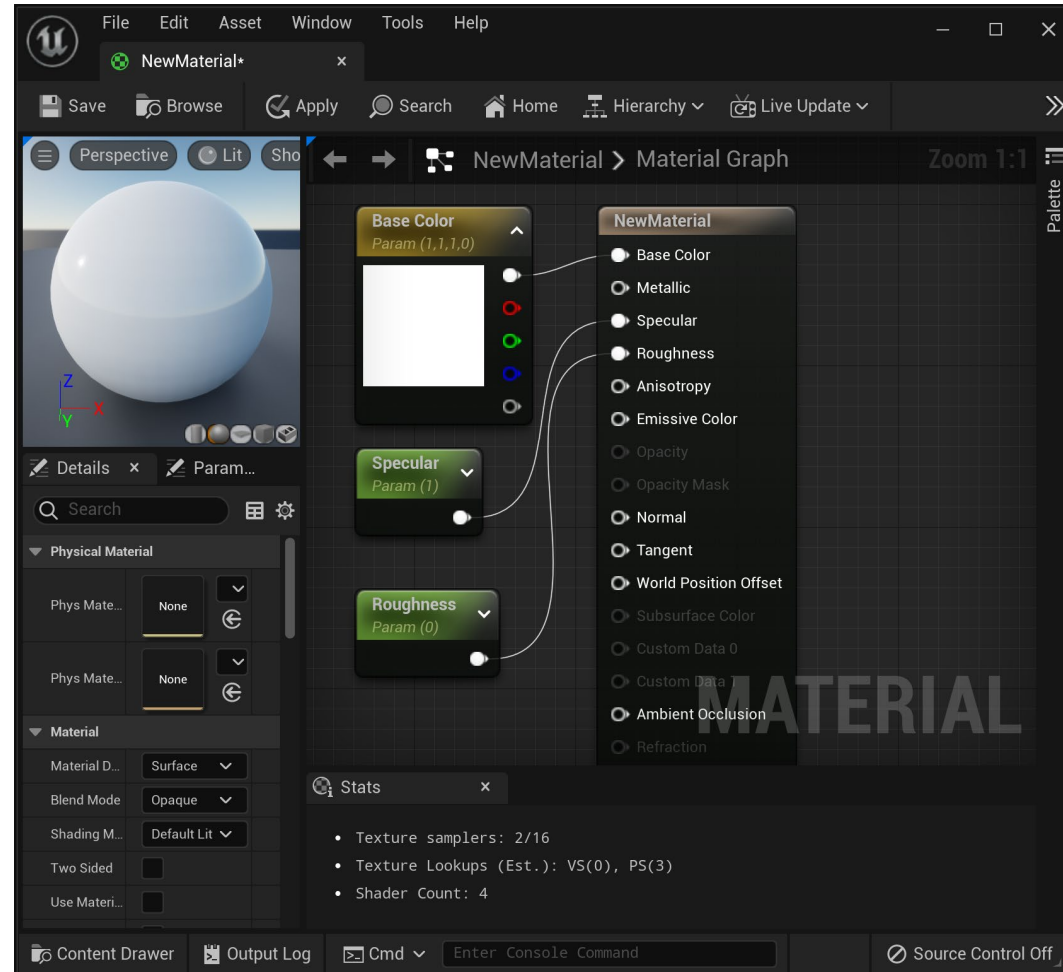
Parameters

- Base color
- Roughness

FIX

Glossy Material.

Only needs three UE5 parameter values all of which are present in the octane material definition.



Parameters

- Base color
- Specular
- Roughness

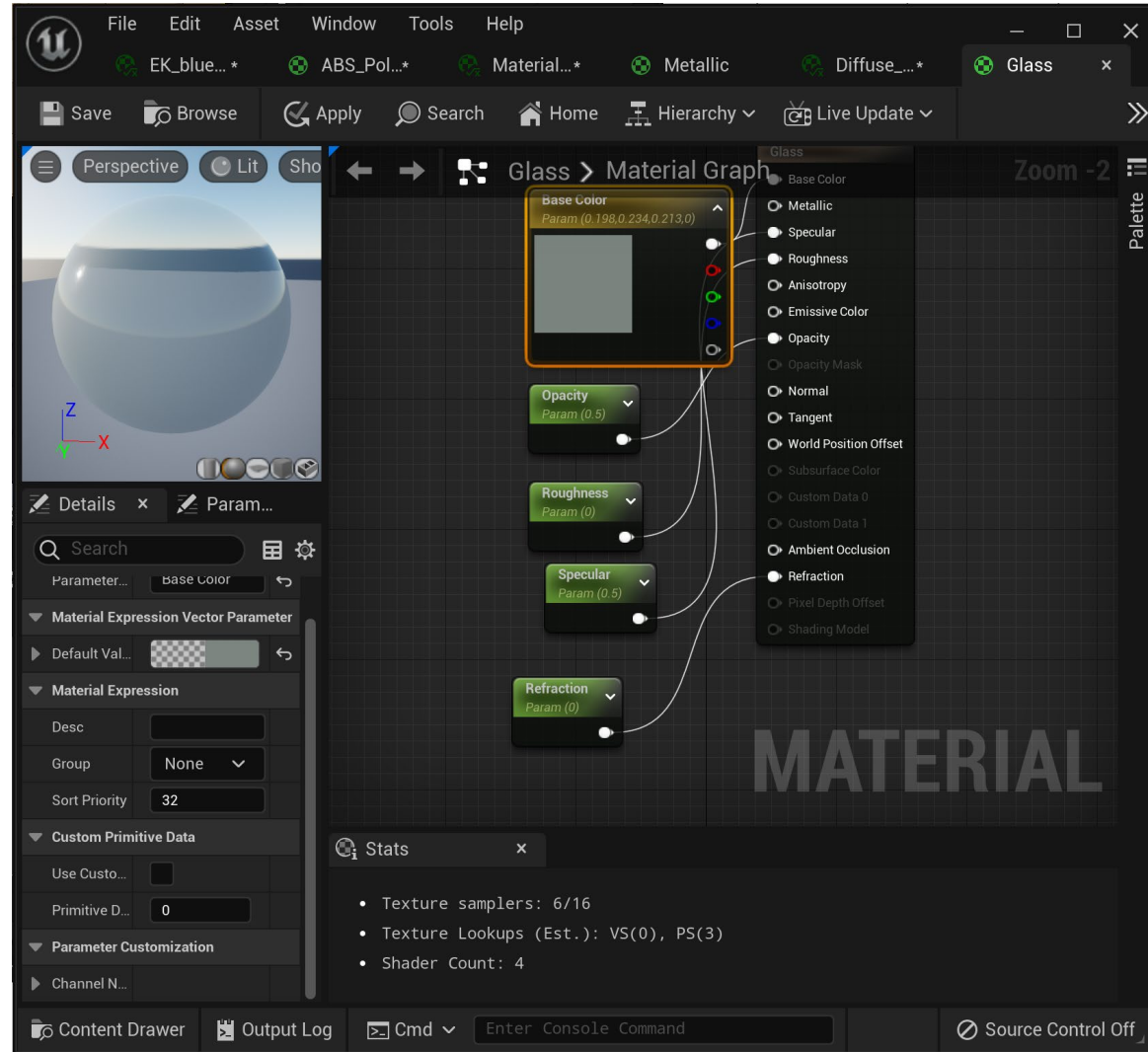
FIX

Specular Material.

Needs five UE5 parameter values all of which are part of the original octane material definition.

However, the opacity value can not be used directly from octane and needs to be calculated from the transmission value.

Calculate the opacity based on the Transmission pin values

$$\text{Opacity} = (1 - (\text{tblTransmissionValues}[1] + \text{tblTransmissionValues}[2] + \text{tblTransmissionValues}[3]))/3$$


Parameters

- Base Color
- Opacity
- Specular
- Refraction
- Roughness

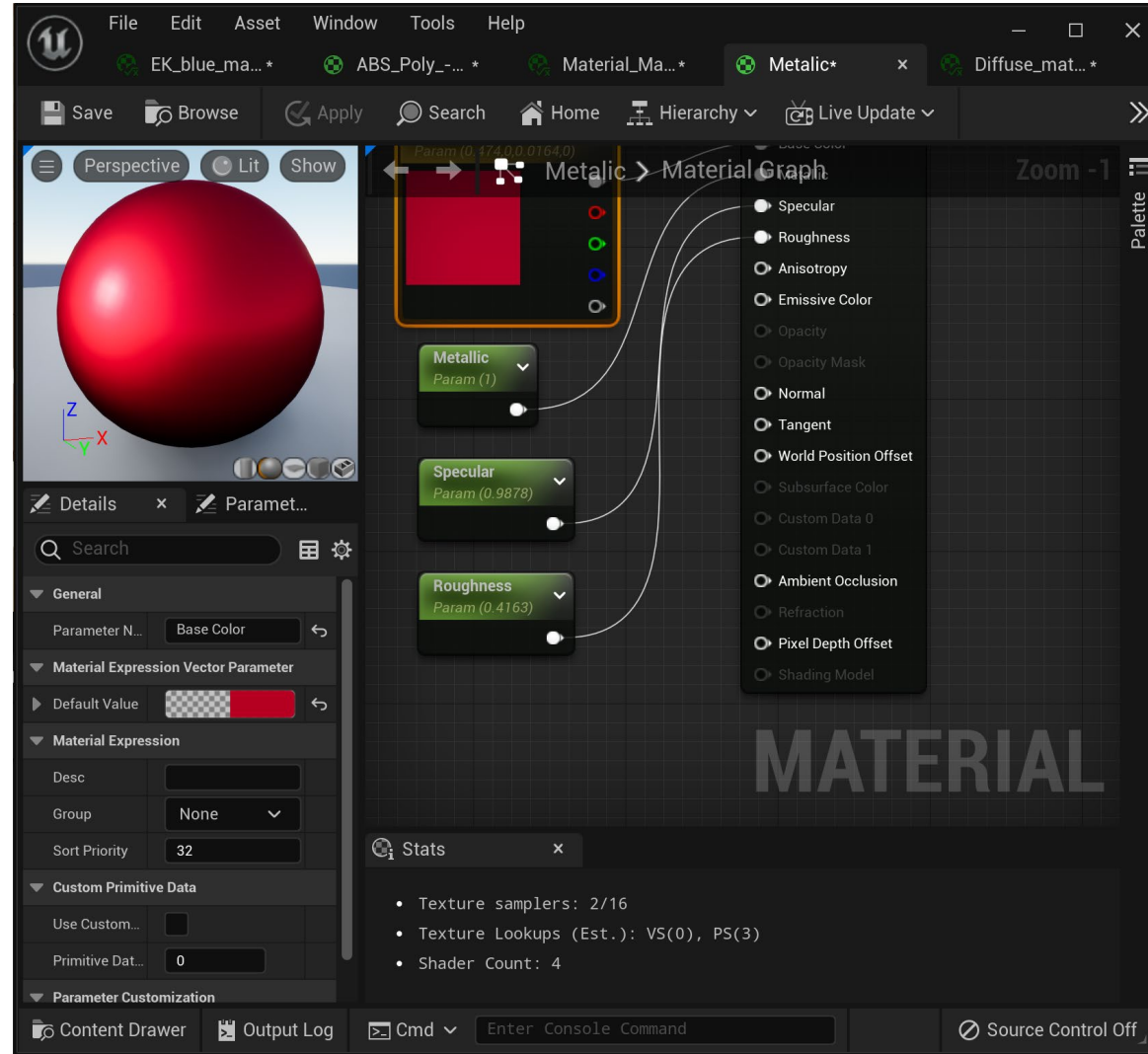
NOTE: Opacity in UE5 requires 2 other parameter to be set to the correct values in order to show.
Blend Mode must be set to *Translucent*, and Lighting Mode set to *SurfaceForwardShading*

FIX

Metallic Material.

Needs four UE5 parameter values all of which are present in the octane material definition.

Octane supports transparent metallic materials in which case its opacity pin value can be used.



Parameters

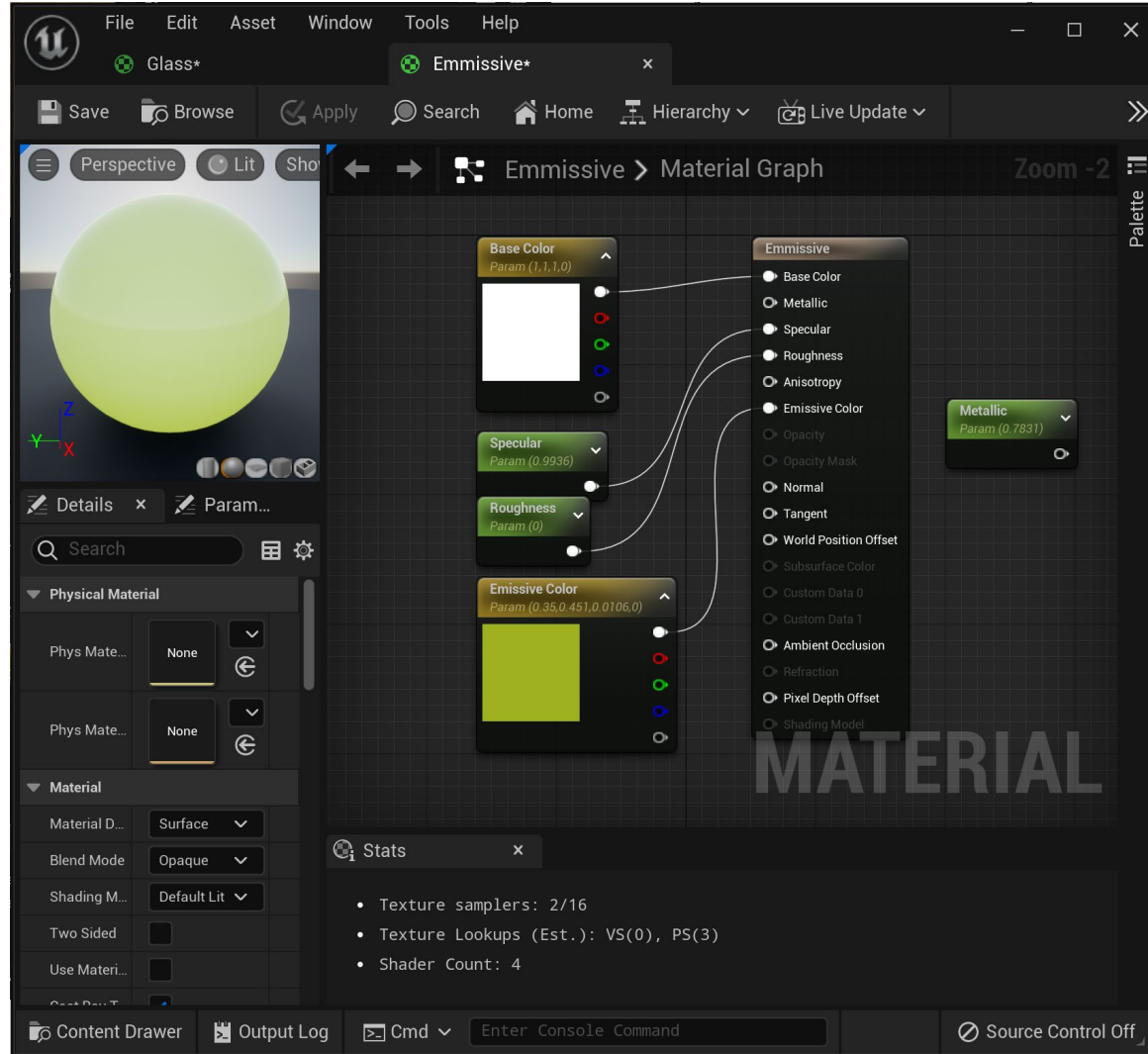
- Base Color
- Metallic
- Specular
- Roughness
- *Opacity*

NOTE: Opacity in UE5 requires 2 other parameter to be set to the correct values in order to show. Blend Mode must be set to *Translucent*, and Lighting Mode set to *SurfaceForwardShading*

FIX

Emissive Material.

Only needs five UE5 parameter values all of which are present in the octane material definition.



Parameters

- Base color
- Opacity
- Specular
- Roughness
- Emissive color