



# Creating VFX with Unreal Engine and Houdini

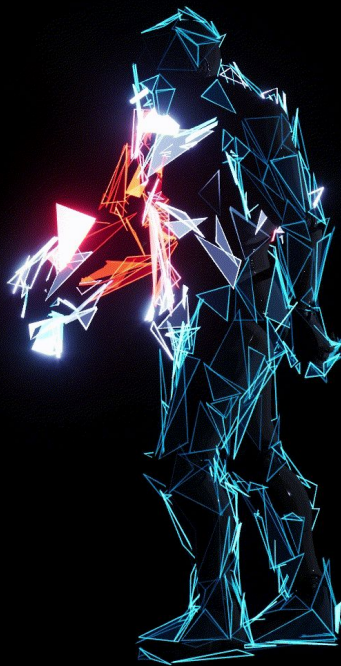
# Mesh Processing

Compelling visual

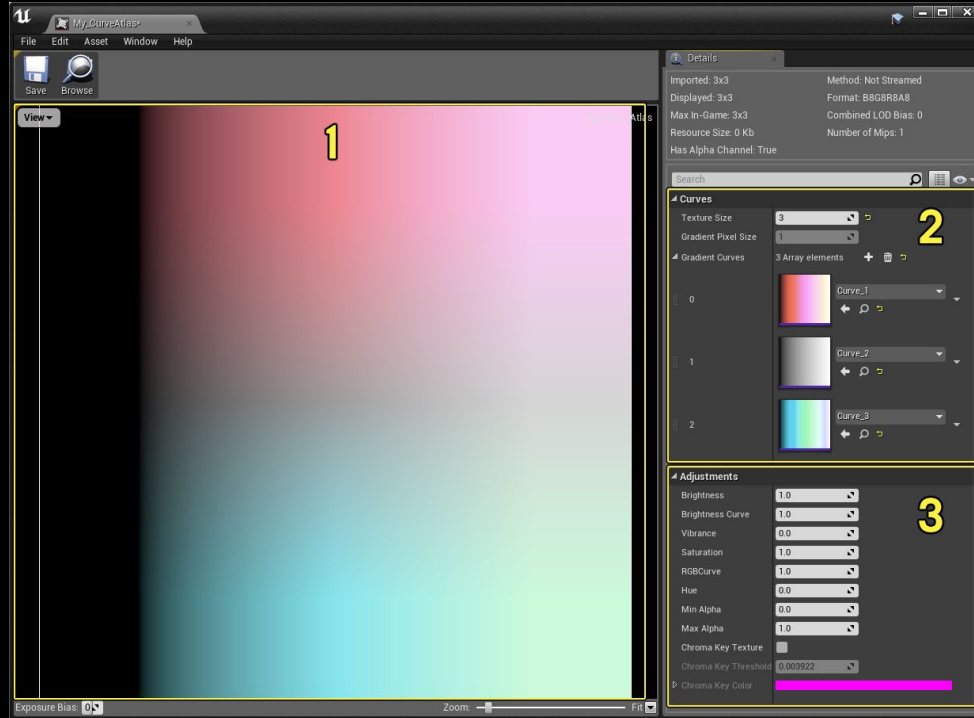
Instant gameplay feedback

Flexible animating ability

Good performance

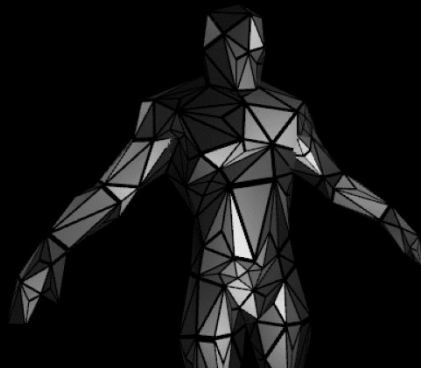


# Animate parameters in Material Editor

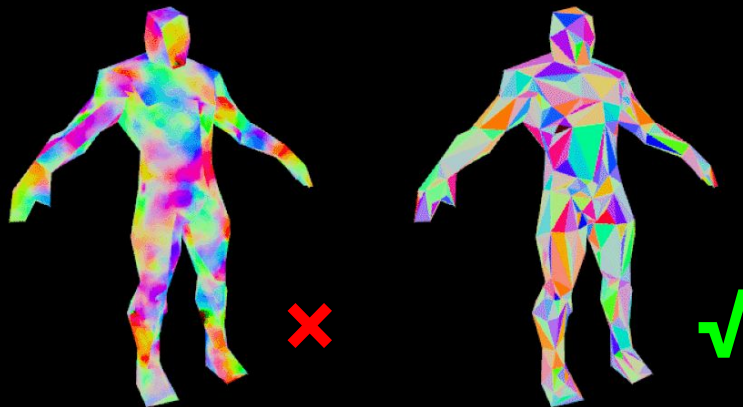


# Two challenges

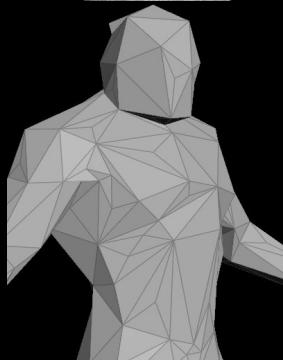
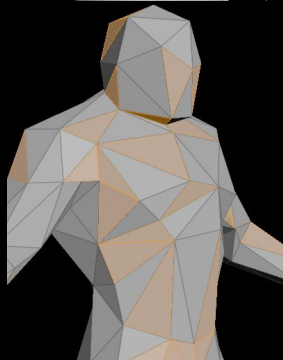
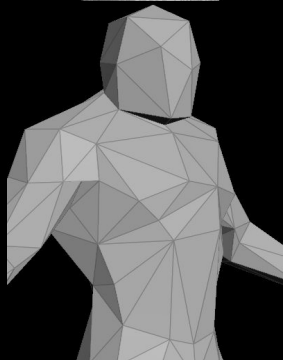
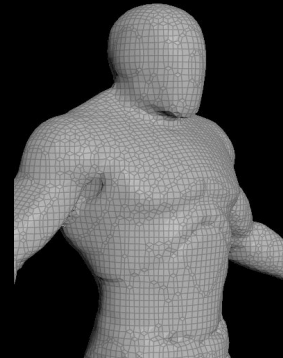
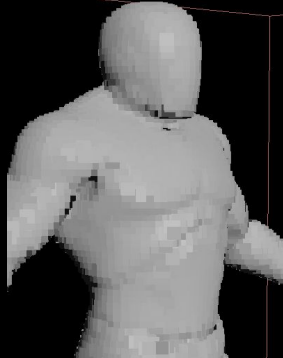
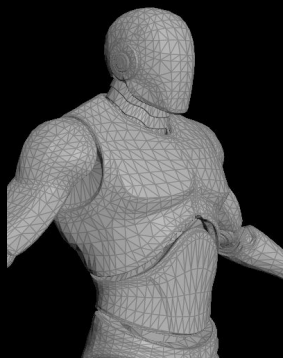
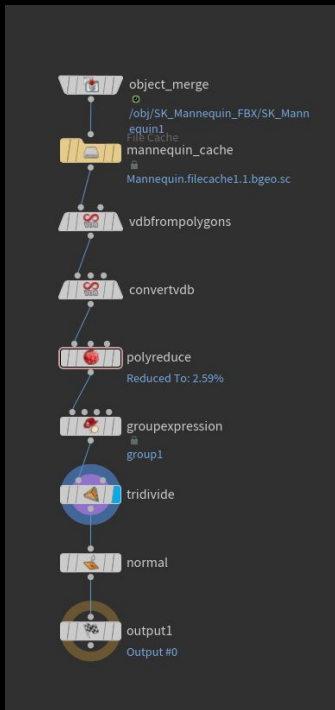
Animate the color and transformation of each triangles separately, using one single mesh



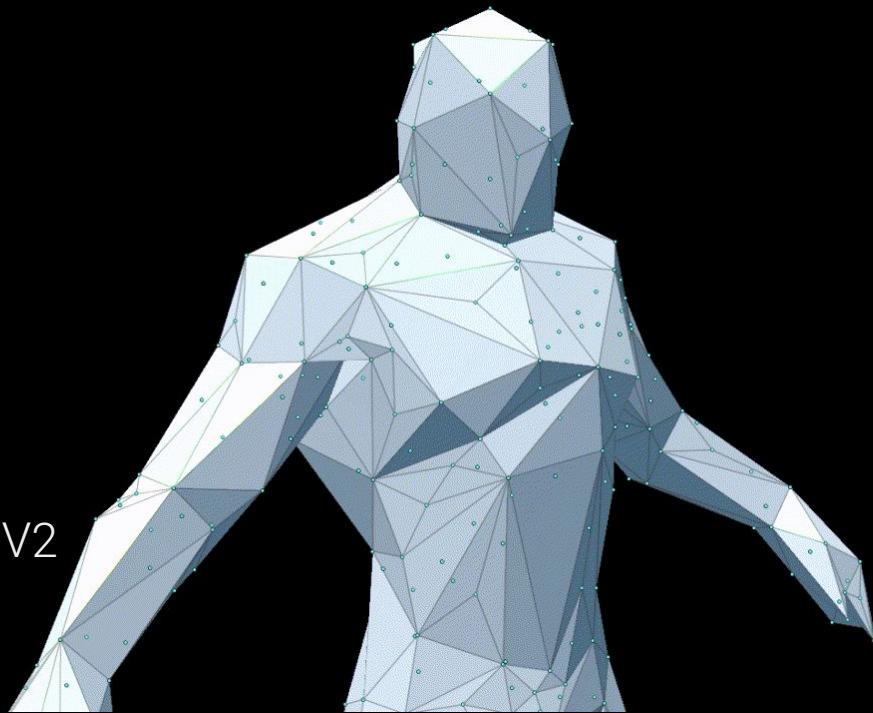
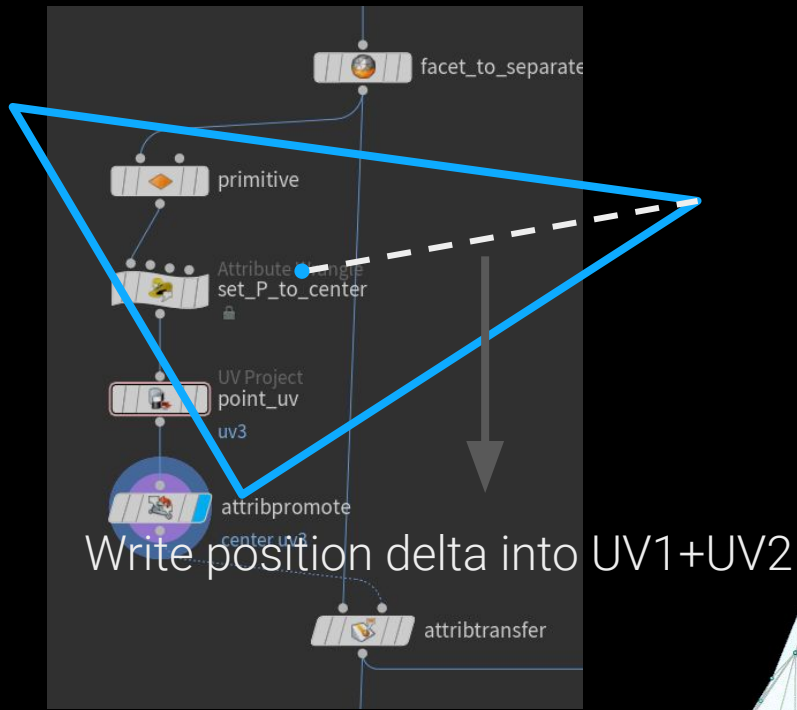
Add noise on top of color and transformation separately



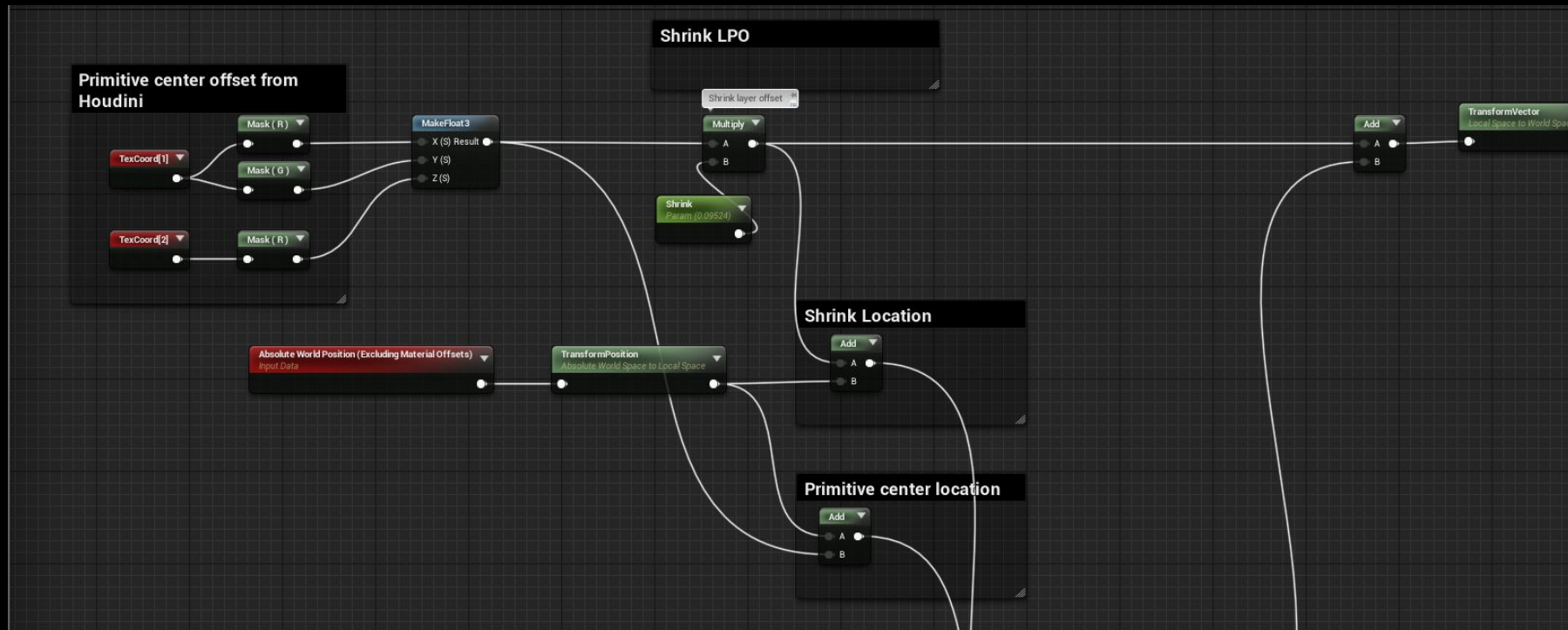
# Generate Low Poly Mesh



# Write vertices data into UV maps

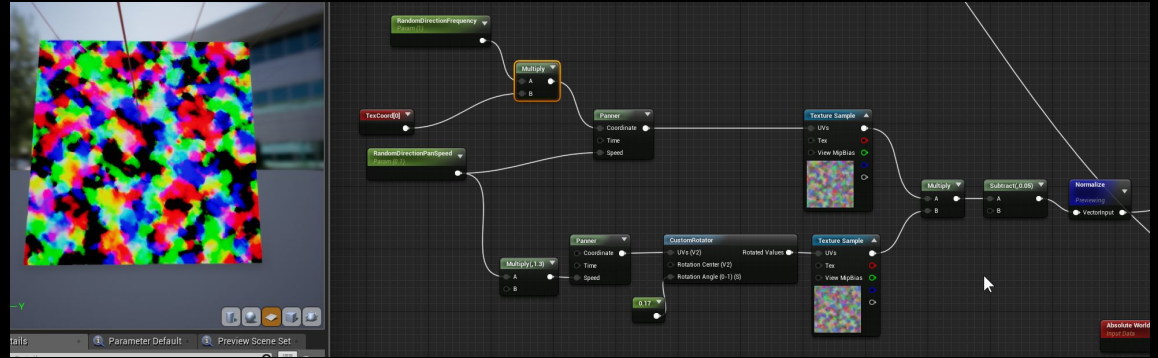


# Transformation in Unreal Engine

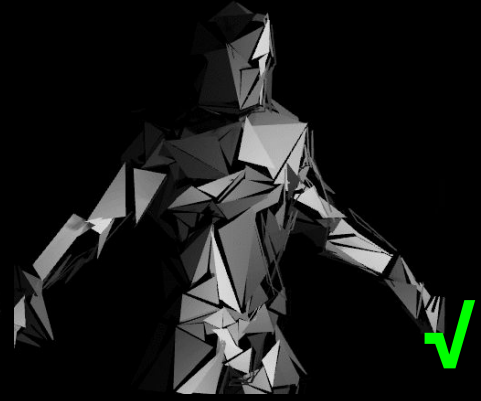
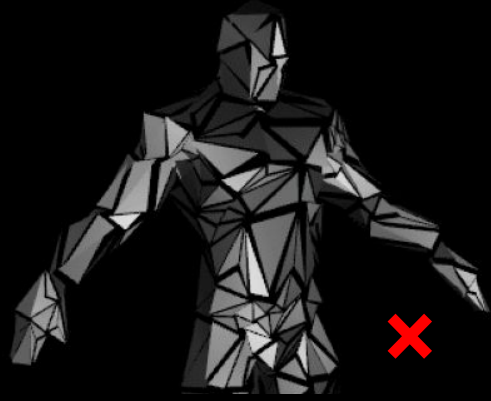
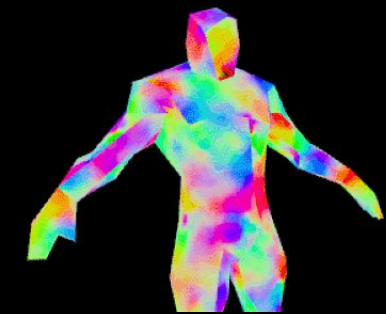


# Noises

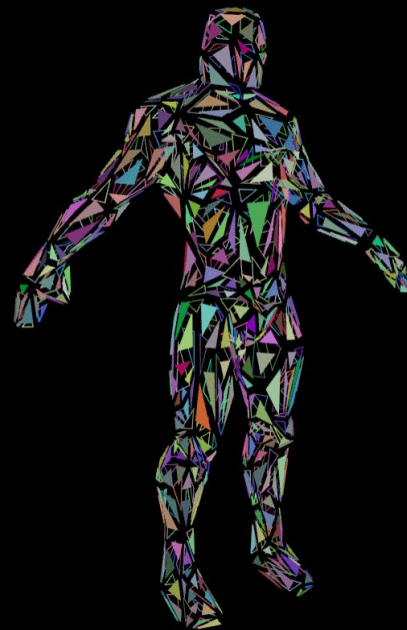
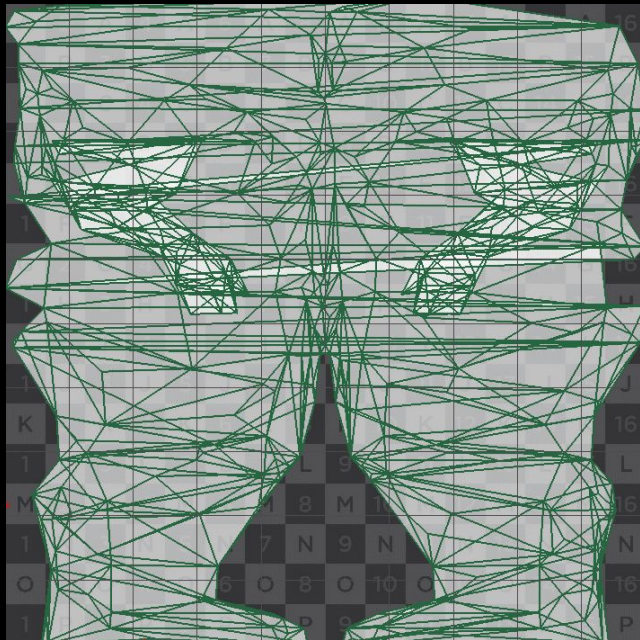
TexA \* TexB \* 2







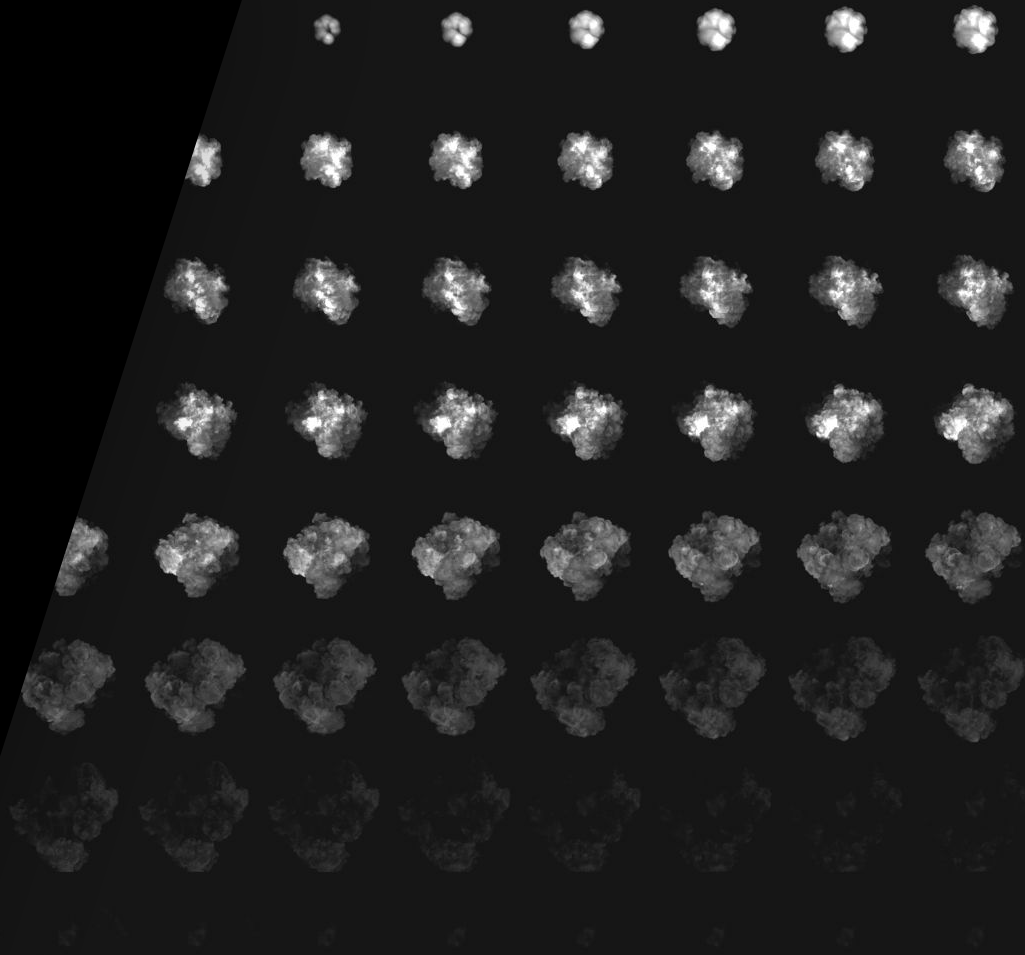
# UV for texturing



# Damage Mask



# Packed Flipbook



# Packed Flipbook

## Limitations

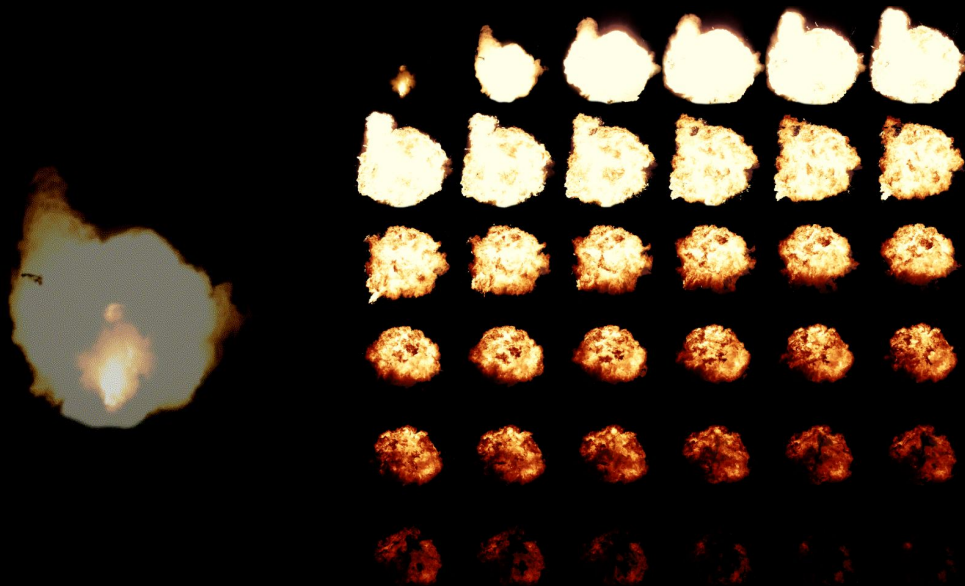
High res by nature

Can't be used for anything else

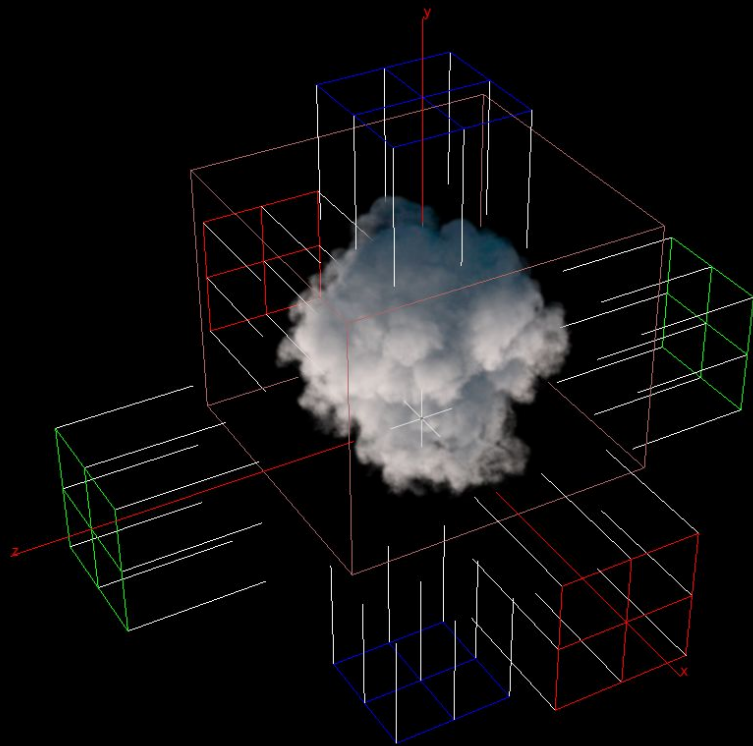
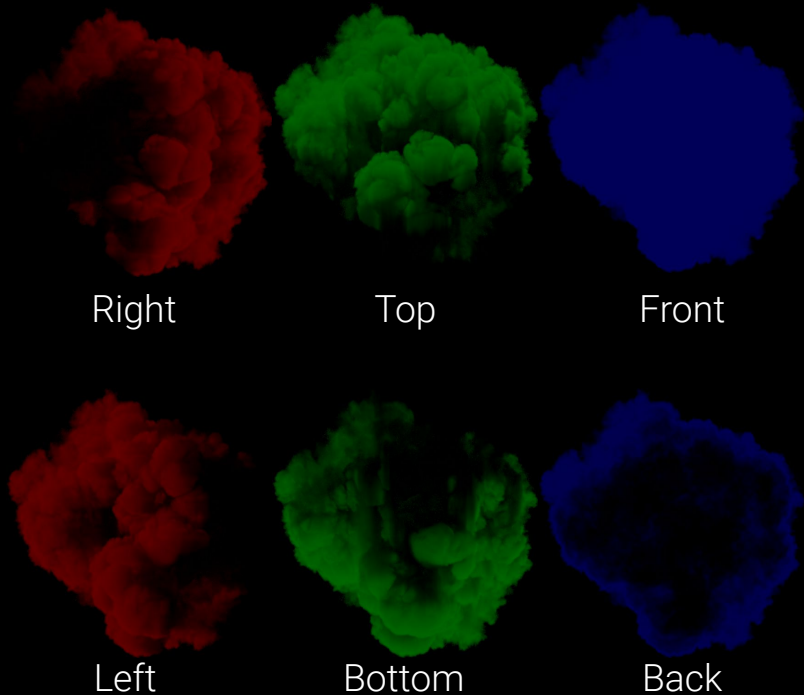
Limited length

Unlit

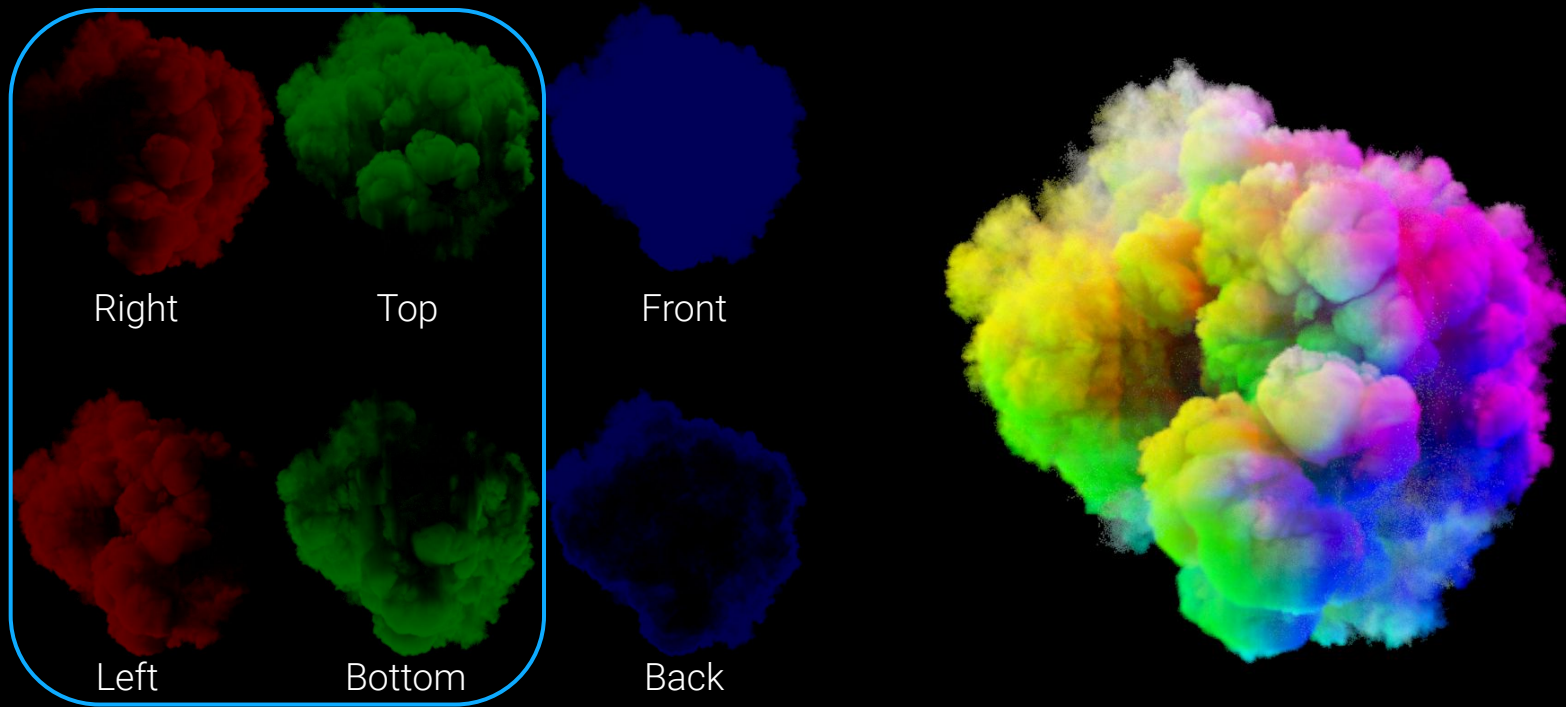
LDR



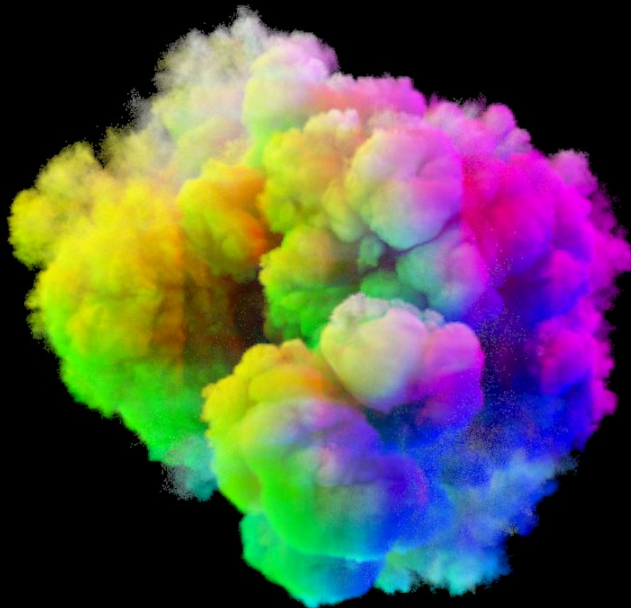
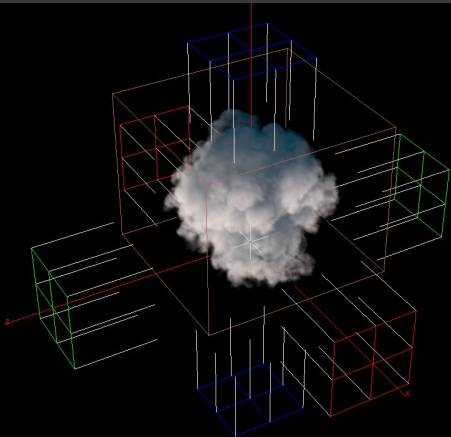
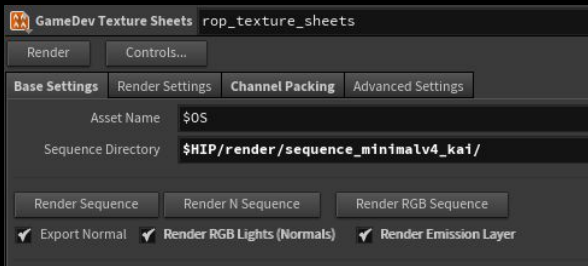
# Smoke: 6D / 4Direction Lightmaps



# Smoke: 6D / 4Direction Lightmaps

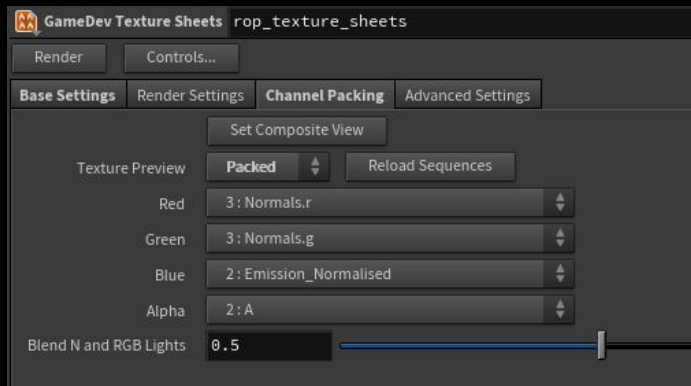


# Generate 6D Lightmaps





# Generate Normal Map



R: Normal.r

G: Normal.g

B: Emission (Temperature)

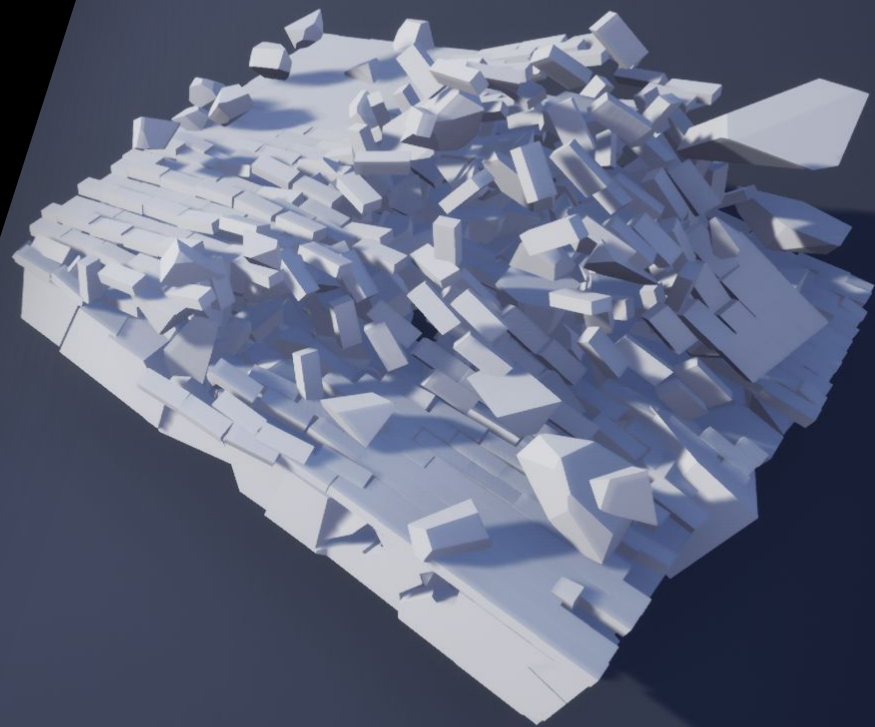
A: Opacity

# Temperature



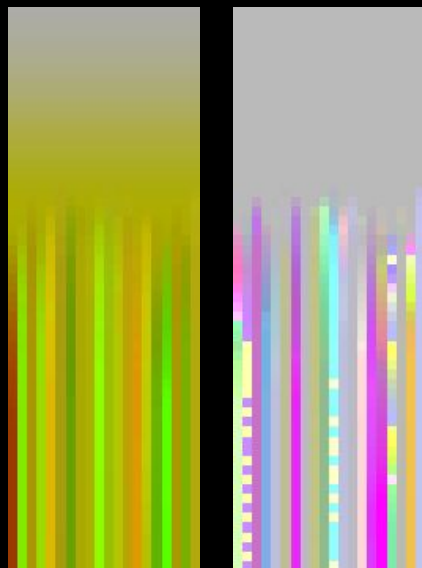


# Vertex Animation Texture



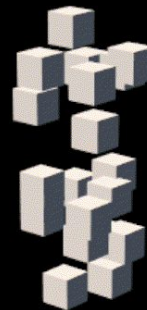
# Vertex Animation Texture

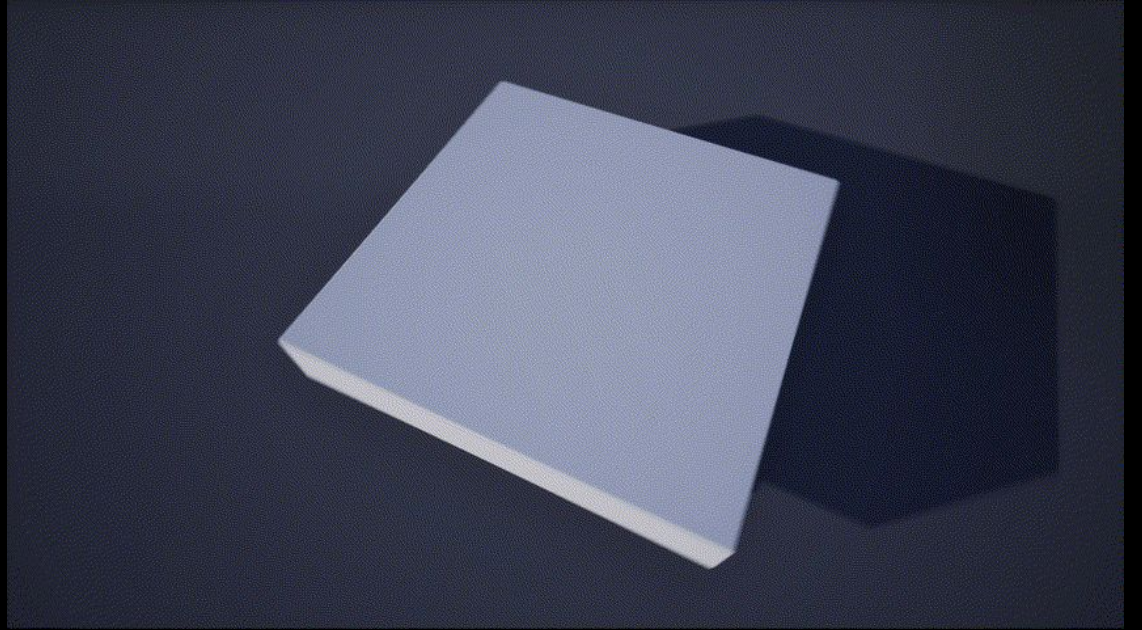
20 rows = 20 objects  
64 columns = 20 frames



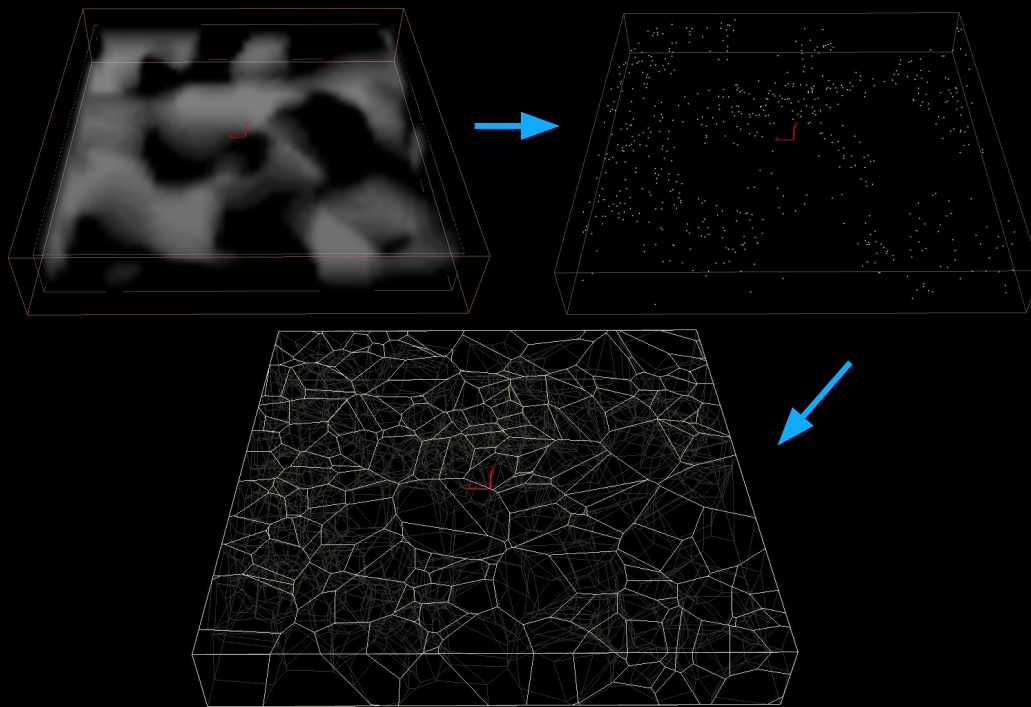
Position

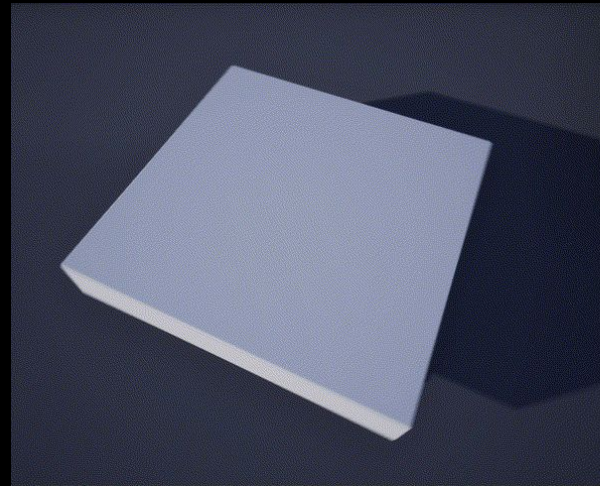
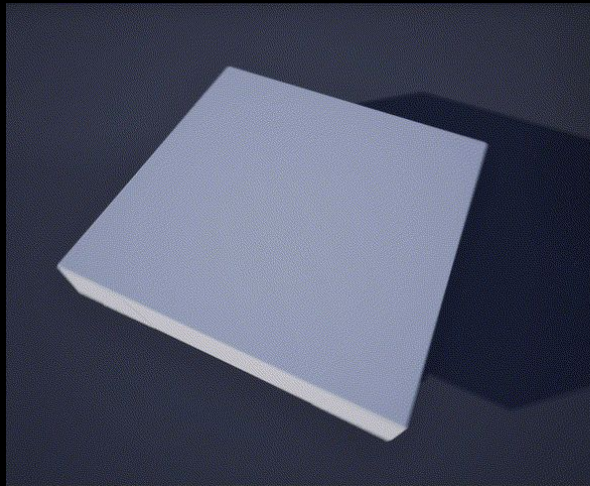
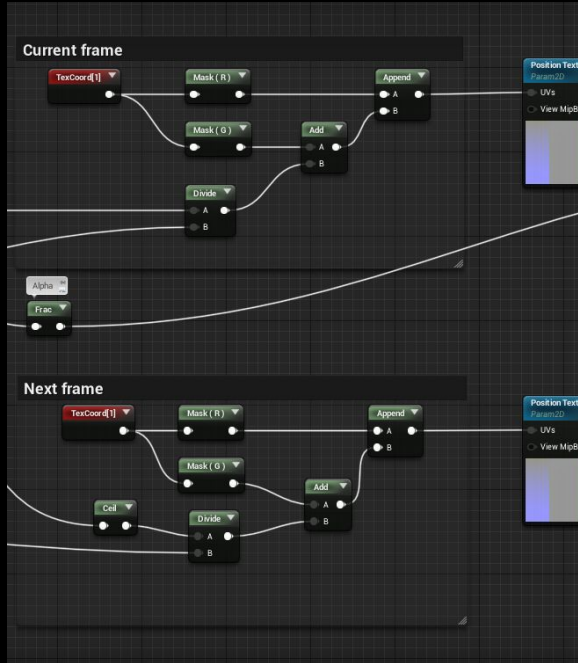
Rotation





# RBD Fracture









# Volumetric Rendering

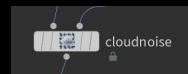
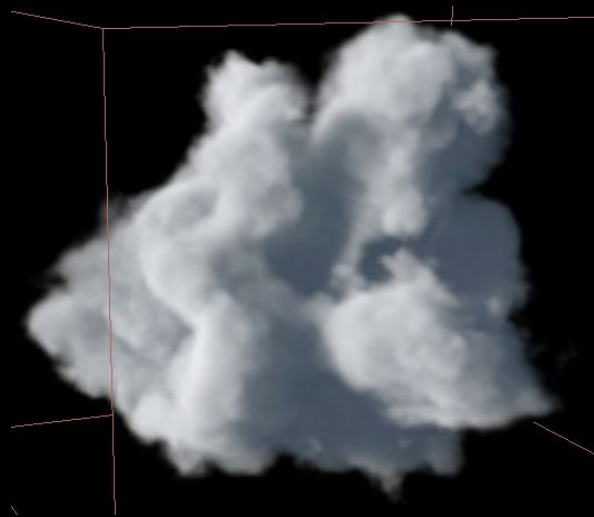
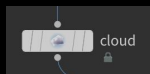
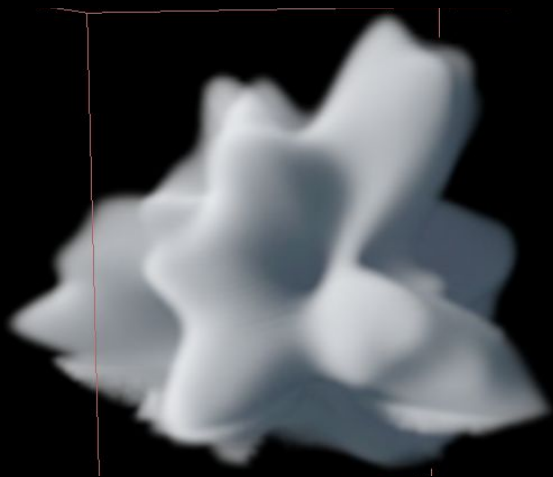
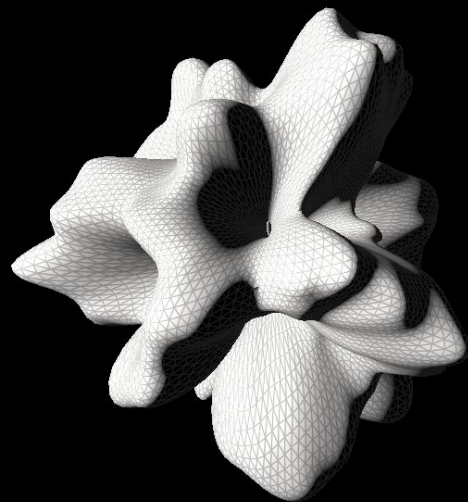
UEShaderBits-GDC-Pack

Workflows

Optimizations



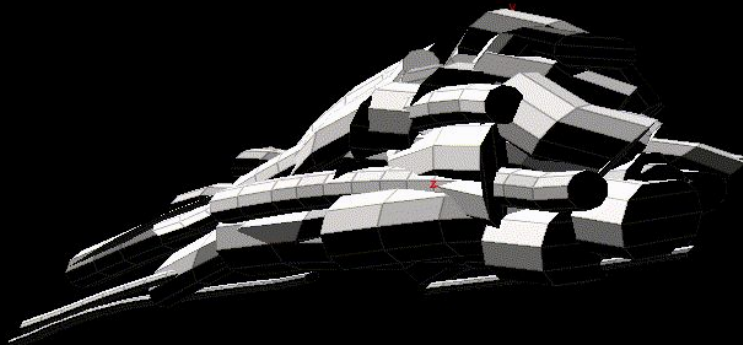
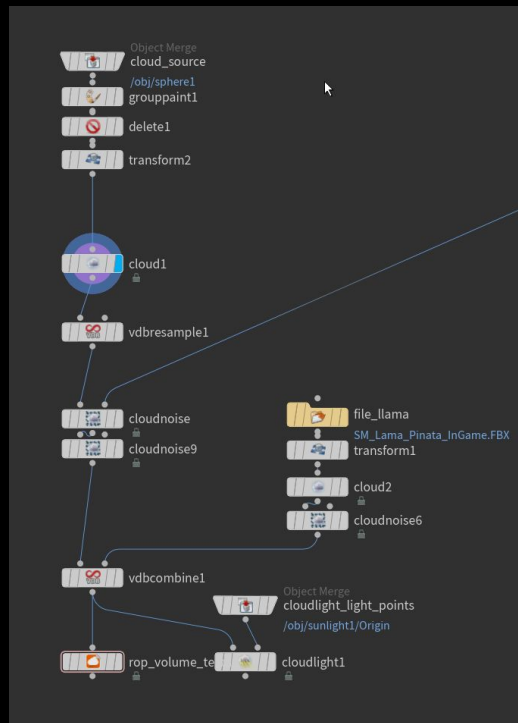
# Cloud in Houdini



# Using Blocks



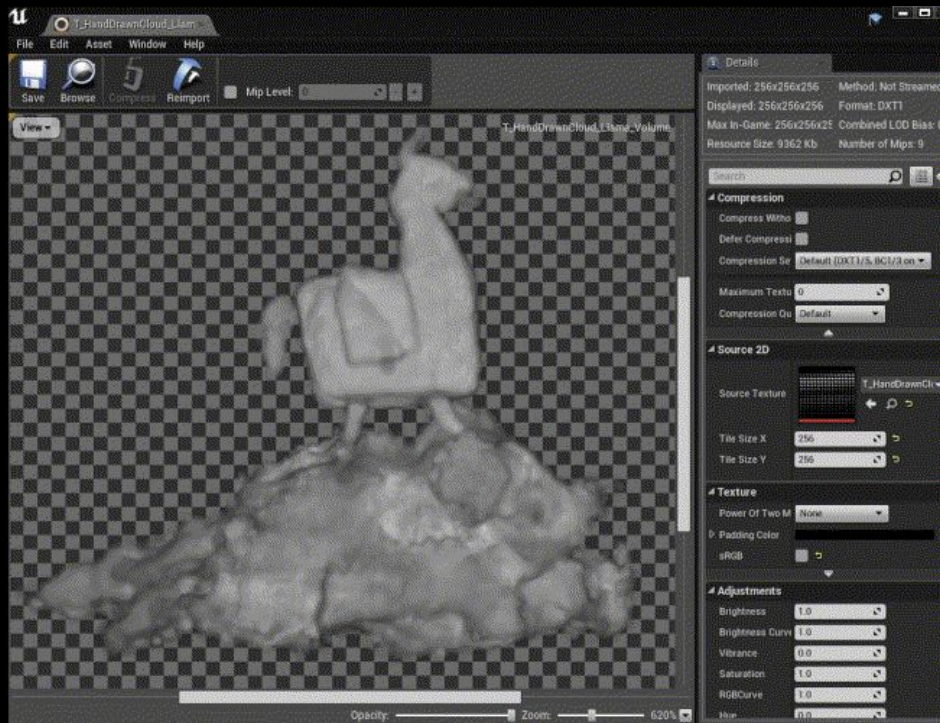
# Using Blocks



# Volume Texture

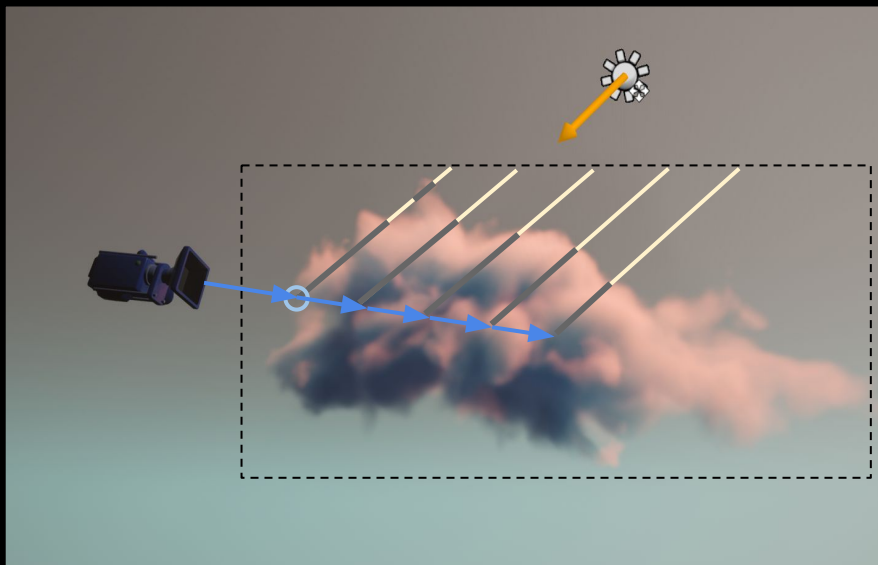


# Volume Texture



## Volume Texture Preview

# Volumetric Raymarcher



Marching ray



Shadow added to this ray

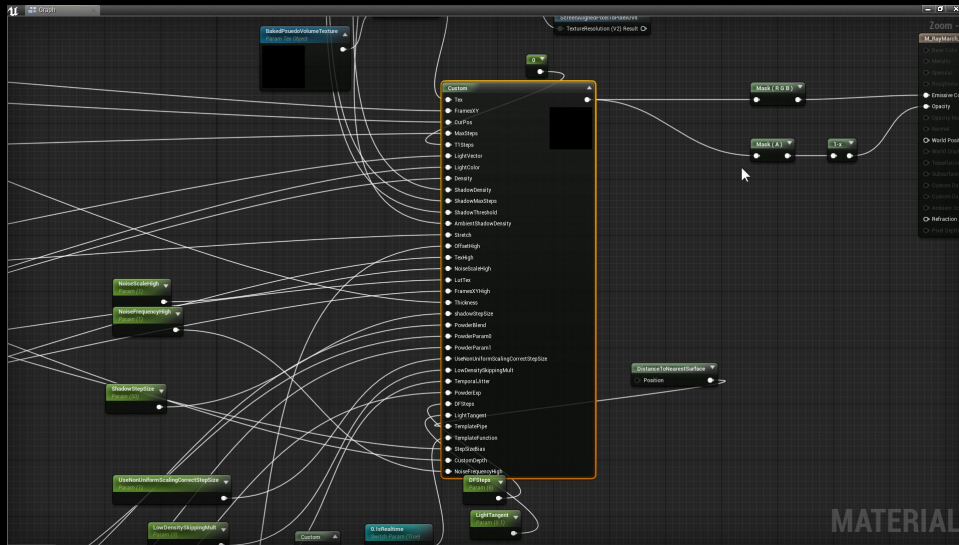


No shadow added



Bound box





### Material Expression Custom

Code

```
#include "/Project/RayMarchCompute.usf"
return 0;
```

Output Type

CMOT Float 4

Description

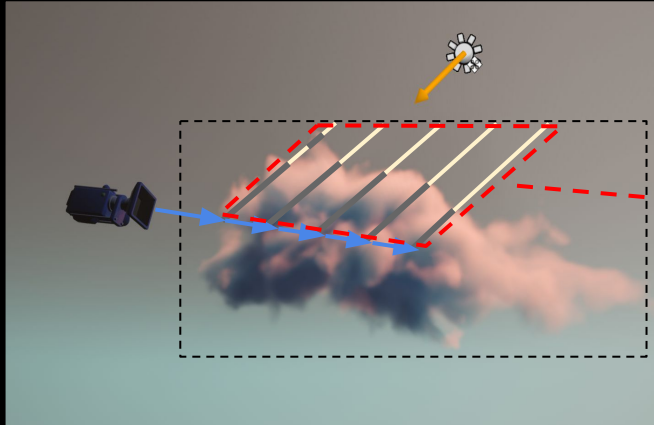
Custom

Inputs

34 Array elements + -

/UE4\_Project/Shaders/RayMarchCompute.usf

# Baking light energy



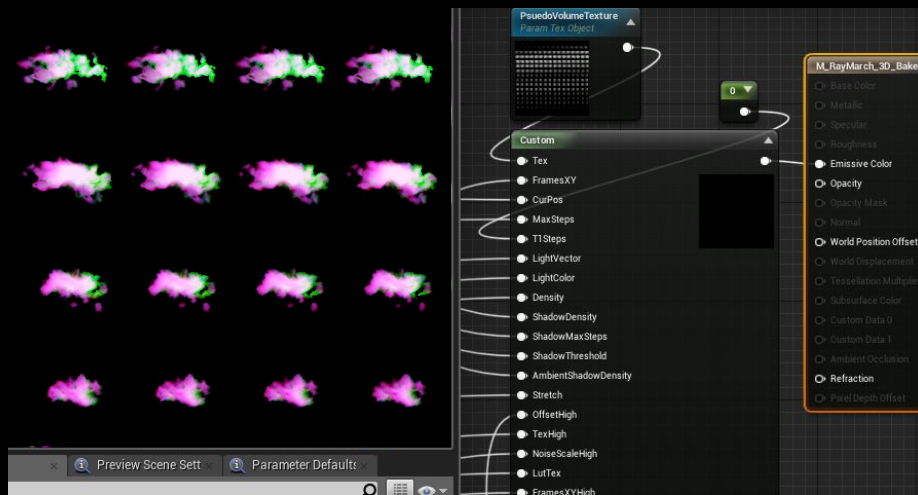
```
for(int i=1; i <= MaxSteps; )
{
    float curDensity = saturate(Density *
PseudoVolumeTexture(Tex, TexSampler, func.LocalToUVW(CurPos,
Stretch), FramesXY, framesN).x);
    float3 curLightPosition = CurPos;
    float shadowDistance = 0;
    for(int s=0; s<ShadowMaxSteps; s++)
    {
        curLightPosition += lightVectorStep;
        float d=PseudoVolumeTexture(Tex, TexSampler,
func.LocalToUVW(curLightPosition), XY, N).x;
        shadowDistance += d / ShadowMaxSteps;
    }
    transmittance *= 1-curDensity;
    energy +=
    exp(-shadowDistance * ShadowDensity) // Beer's Law
    * curDensity * transmittance * LightColor;
}
```

Baked volumetric lightmap:

R: Density

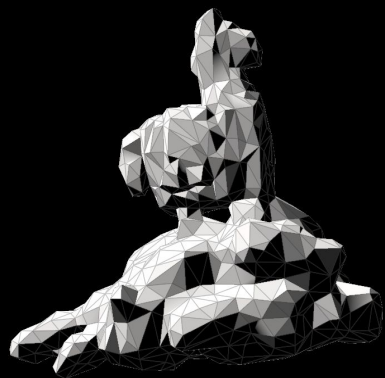
G: Shadow Distance

B: Sky Light Shadow Distance

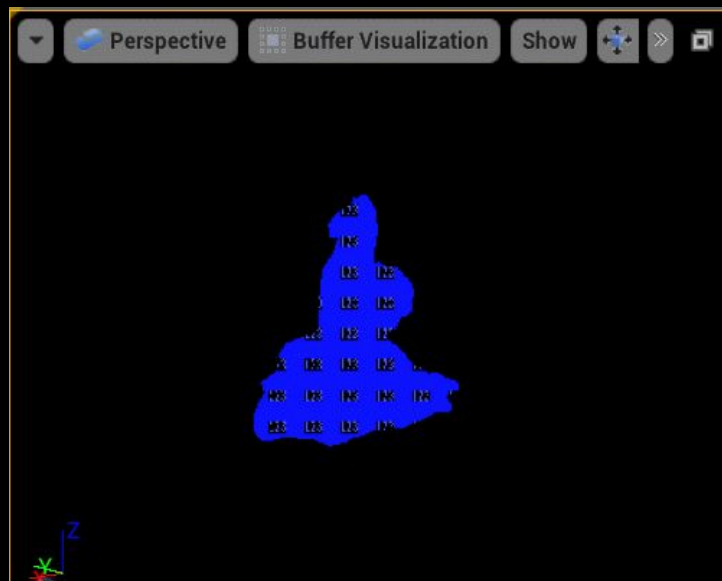


# Blutility

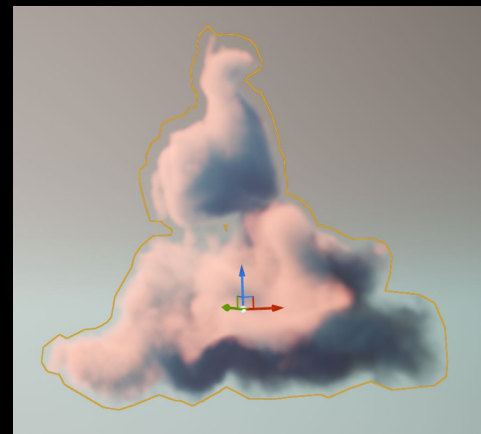
# Shell mesh



Low poly shell mesh

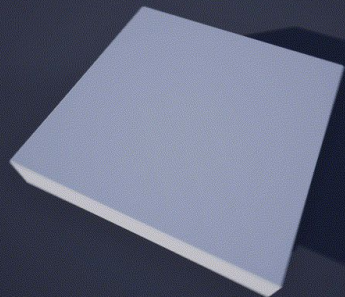


Custom Stencil



Custom Stencil as  
render mask

# Q&A



Thank you!

[asher.zhu@epicgames.com](mailto:asher.zhu@epicgames.com)

Unleashing Houdini for AAA Mobile Games Production

Martine A., Beeckmans P., Stroukoff A.

Smoke Lighting and texture re-usability in Skull & Bones

Mederic

Creating a Volumetric Ray Marcher

Ryan Brucks

All