**Color Grading**

**LUT What：**

执行每个像素的所有颜色分级步骤工作量很大，我们可以将颜色分级烘焙到查找表中（简称LUT，Look-Up-Table），通过对其采样来转换颜色。

**LUT的格式**

LUT是一种3D纹理，通常为32\*32\*32。LUT是3D纹理，但常规着色器无法渲染到3D纹理。因此我们将2D切片连续放置一排，组合成一个宽的2D纹理来模拟3D纹理。这样LUT纹理的高度等于配置的分辨率，其宽度等于分辨率的平方。

**Log C空间值范围是多少？**

略低于59。

**如何将UV转换成颜色值？**

// params = (lut\_height, 0.5/lut\_width, 0.5/lut\_height, lut\_height/lut\_height-1) lut\_width = 32

real3 GetLutStripValue(float2 uv, float4 params)

{

    uv -= params.yz;

    real3 color;

    color.r = frac(uv.x \* params.x);

    color.b = uv.x - color.r / params.x;

    color.g = uv.y;

    return color \* params.w;

}

**如何将颜色变换成UV坐标？**

// scaleOffset = (1/lut\_width, 1/lut\_height, lut\_height-1)

// (1/32\*32, 1/32, 31)

real3 ApplyLut2D(TEXTURE2D\_PARAM(tex, samplerTex), float3 uvw, float3 scaleOffset)

{

    uvw.z \*= scaleOffset.z;

    float shift = floor(uvw.z);

    uvw.xy = uvw.xy \* scaleOffset.z \* scaleOffset.xy + scaleOffset.xy \* 0.5;

    uvw.x += shift \* scaleOffset.y;

    uvw.xyz = lerp(SAMPLE\_TEXTURE2D\_LOD(tex, samplerTex, uvw.xy, 0.0).rgb,

       SAMPLE\_TEXTURE2D\_LOD(tex, samplerTex, uvw.xy + float2(scaleOffset.y, 0.0), 0.0).rgb, uvw.z - shift);

    return uvw;

}

**Color Grading的工作流程？**

1. 将每一个UV值映射到一个颜色值，UV的范围为uv分别∈[0, 1]，对应的颜色值为rgb分别∈[0, 1]；
2. 将这个颜色值进行LutBuilderHdr内的一系列颜色变换处理，生成一个颜色值，存储到LUT贴图中；
3. 后处理时根据颜色值逆运算得到原来的uv，然后用UV插值采样LUT贴图。