

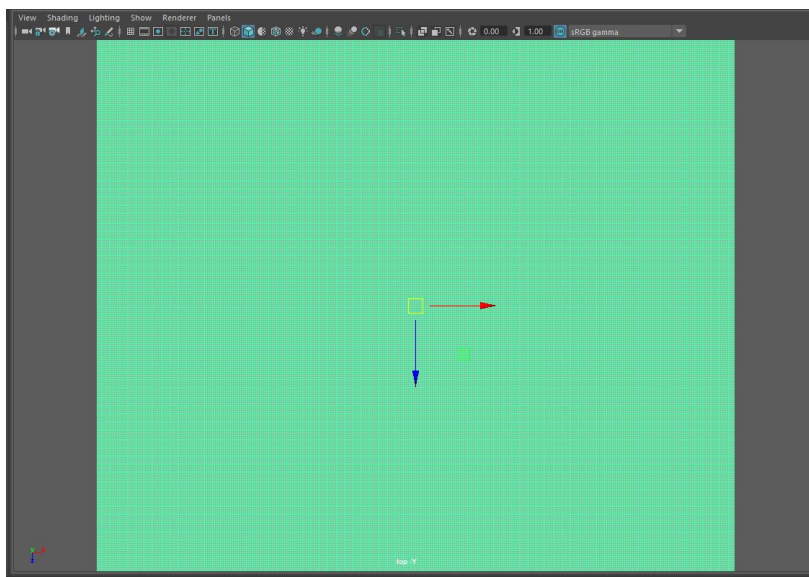
云海效果

效果图：



1、模型

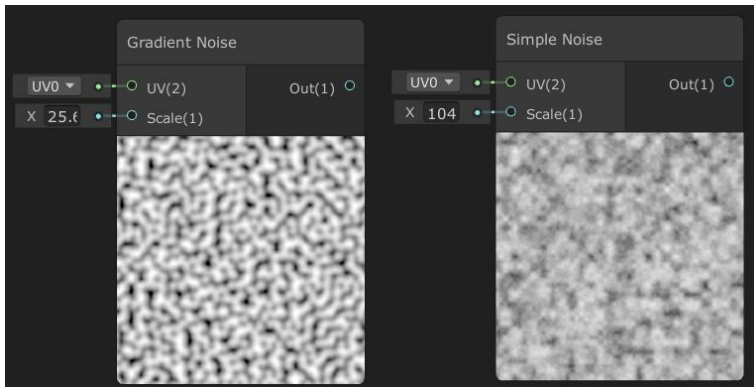
一个细分数较高的 Plan。



2、shader 制作

- (1) 使用 U3D 创建一个 Unlit Graph shader，并将该 shader 赋予给材质球。
- (2) 选择一张噪点图，用做控制云层的起伏效果和顶部与底部的颜色值。

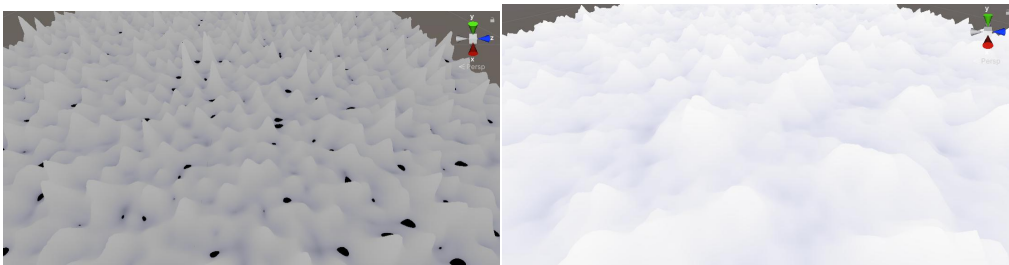
Unity ShaderGraph 里自带的两个噪点图节点



因为云层的上下起伏是非常不规则的，且不会存在完全平面的云层，左边的 GradientNoise 的白色区域的值过于平均，黑色的区域也较大。

右边 SimpleNoise 的颜色变化更高，不存在大量的黑色区域。

对比图：

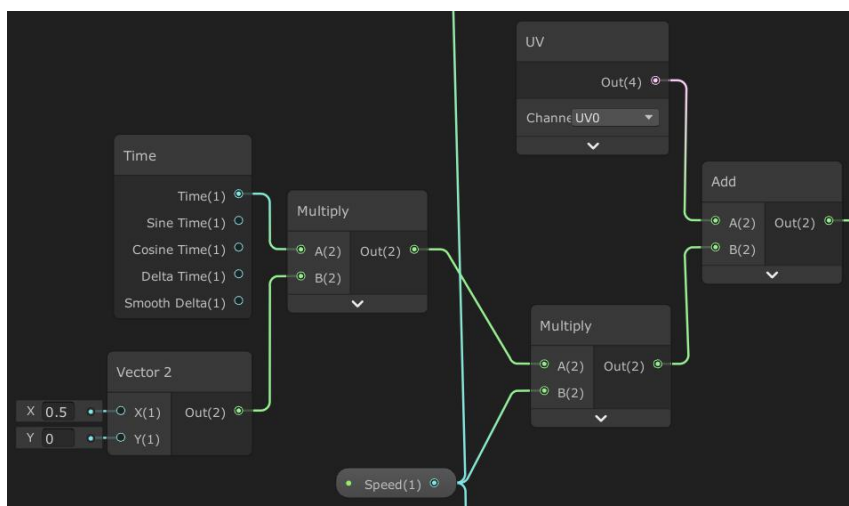


左：使用 GradientNoise，GradientNoise 的黑色区域使得云层的底部黑色区域是平整的，且颜色值也是 0.

右：使用 SimpleNoise，不存在完全平整和值是 0 的区域，变化更加丰富。

所以最终选择 SimpleNoise 图。

(3) 制作云层的流动效果



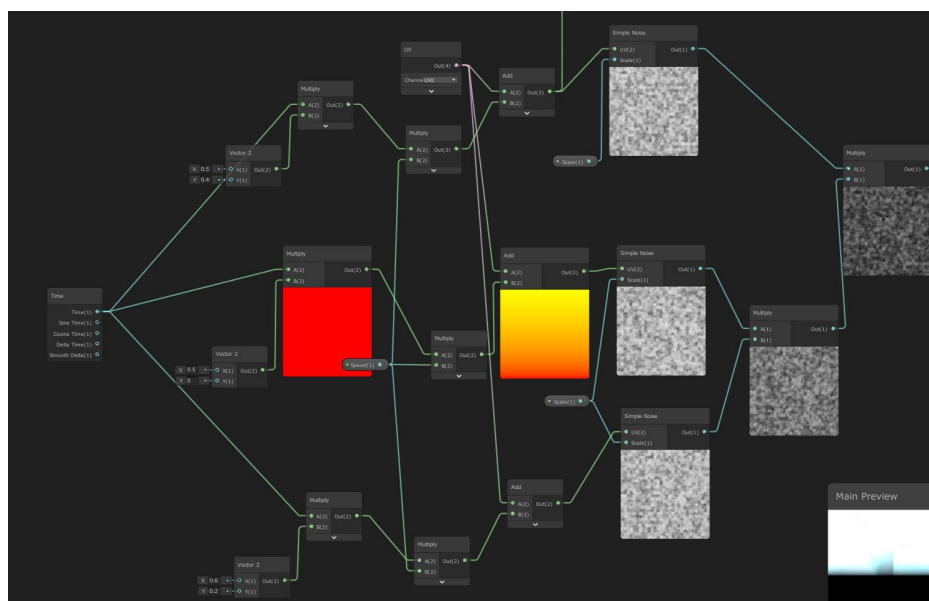
主要方法：使用 Time 与一个 vector2 相乘，再乘于一个 float，最后与 UV 相加来控制云层的运动。

Vector2: 主要控制云层的运动方向，当 X 的值为 0.5 时即云层在 x 轴做水平运动，当给 Y 增加值时，云层的运动也会相应的往 Y 轴进行偏移。当 X 或 Y 的值为负数时则沿该轴的负方向运动。

Float: 将时间与 Vector2 的乘积进行缩小，以达到降低云层移动速度的作用。

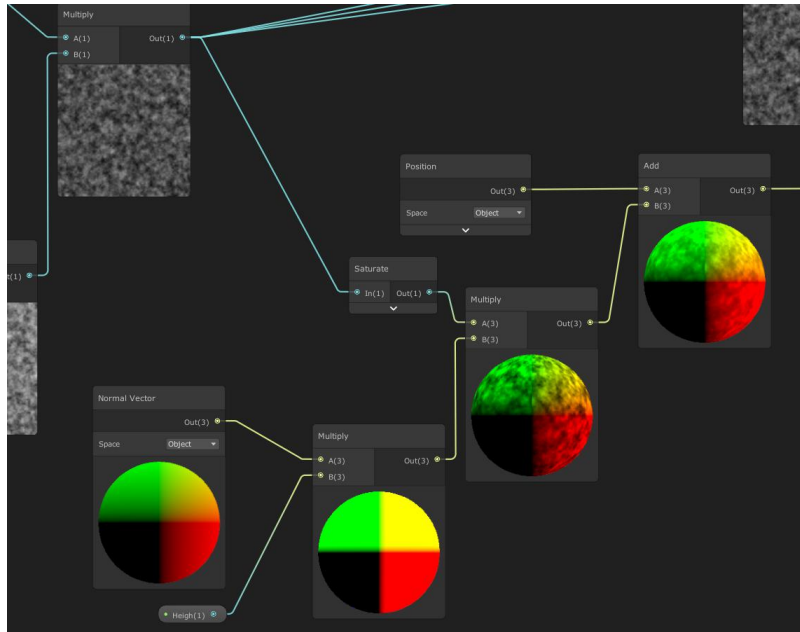
出现的问题：当只有一个 noise 图在 x 轴做水平运动时，云层的凸起部分就会一直保持不变的高度往 x 轴不断运动

解决方法：这里我多制作了几套 Y 轴也有一定的偏移量的 noise 图，并将他们相乘，得到一张颜色和方向变化都较大的 noise 图。

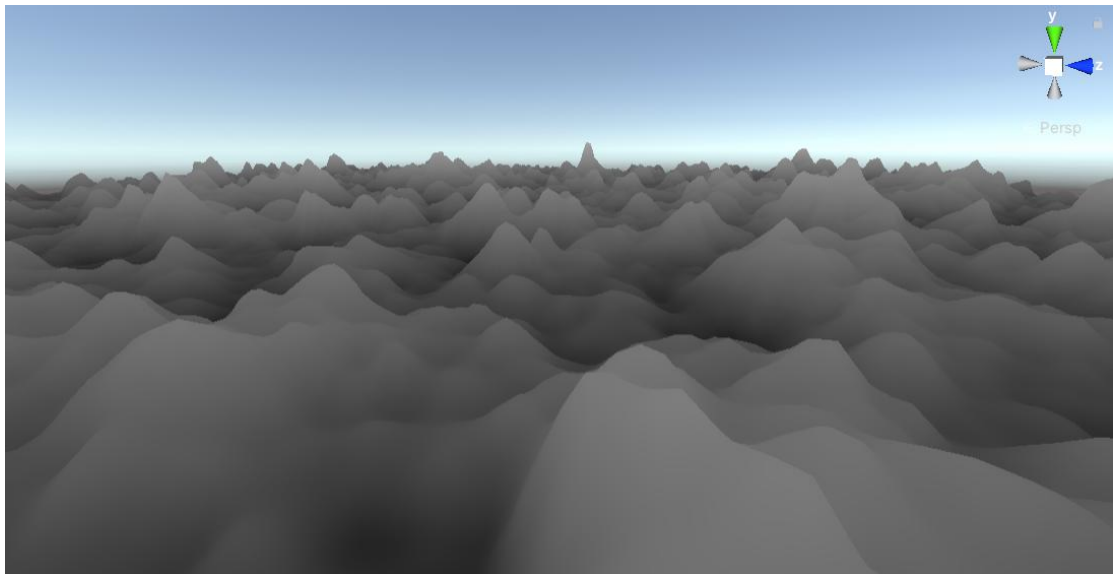


(4) 制作云层的起伏效果（顶点偏移）

将 noise 图与 Plan 的模型法线相乘，使得云层只沿模型的法线方向产生偏移，再将结果加上模型的顶点位置，Plan 就有了上下起伏的效果。最后将该 noise 图赋予给材质的 color 节点。

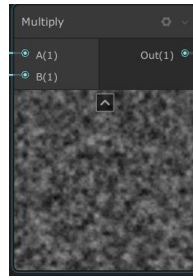


得到如下效果：



此时云层的大致效果已经出来了，最后开始调整云层的颜色。

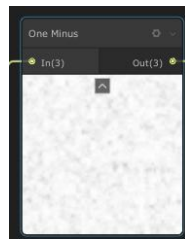
(5) 调整云层顶部与底部的颜色



我们前面计算得到的 noise 图，白色为凸起的区域，黑色为高度不变的区域，即云层底部区域。

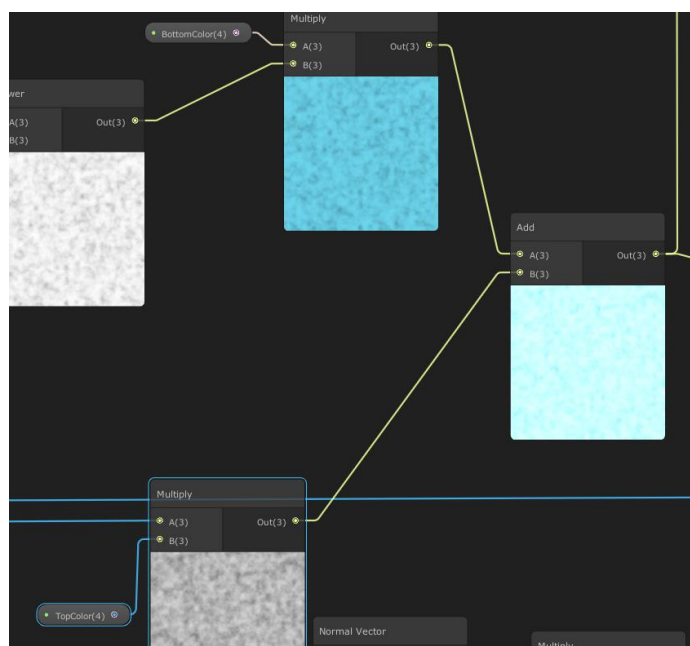
顶部颜色计算：使用该 noise 与 color1 相乘获得云层顶部的颜色。底部区域因为黑色的所以不会受到影响。

底部颜色计算：底部因为颜色是黑色的，即值为 0，此时与 color 相乘并不会得到相应的结果。所以这里我使用 OneMinus 节点对该 noise 图进行反转，将 0-1 变为 1-0, 1-0 变为 0-1, 得到反向后的 noise 图：



将该图乘以 color2 后得到云层底部的颜色。

最后将底部与顶部的颜色相加：



得到如下效果：



(6) 计算云层漫反射

兰伯特光照模型计算公式：

$$\mathbf{c}_{diffuse} = (\mathbf{c}_{light} \cdot \mathbf{m}_{diffuse}) \max(0, \hat{\mathbf{n}} \cdot \hat{\mathbf{l}})$$

\mathbf{c}_{light} : 环境光颜色

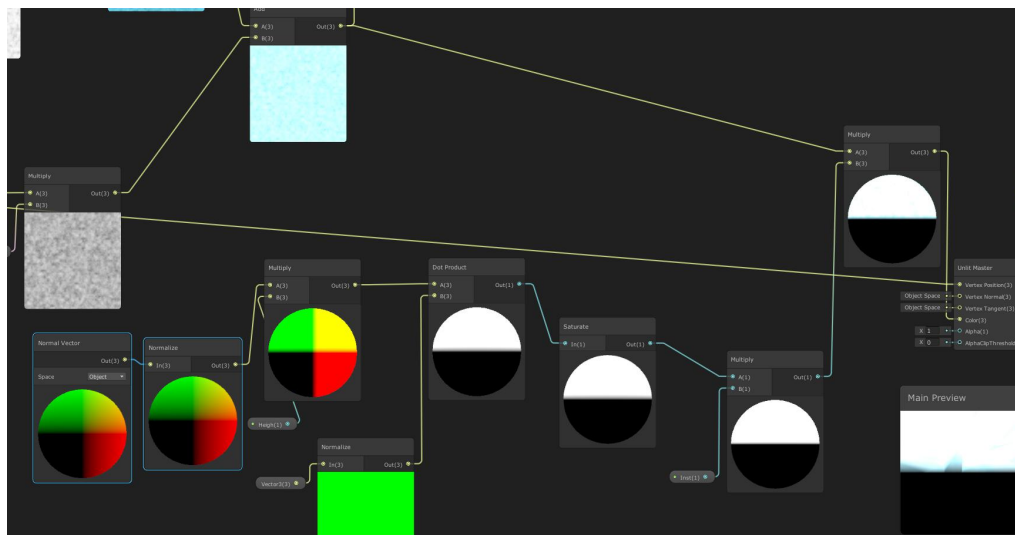
$\mathbf{m}_{diffuse}$: 材质的漫反射颜色

\mathbf{N} : 模型表面法线的单位矢量

\mathbf{L} : 光照方向的单位矢量

通过 normalize 节点将模型法线和光照方向进行归一化处理。

将光照方向与模型法线进行点乘，此时点乘得到的结果在模型表面法线与光照方向的角度大于 90 度时，得到的结果会出现负值，所以这里使用 saturate 节点将其结果的值限制在 0 - 1 的范围内。



最后与云层的颜色和环境光颜色相乘得到最终的效果。

