Flipbooks

通过循环不同的基础纹理可以对Billboard粒子实现动画播放，Unity将其称为Flipbooks粒子，这通常使用常规网格布置的纹理图集完成。

启用Texture Sheet Animation，将Tiles设置为4×4，保证跟我们的网格纹理一样。Start Frame可以设置一个随机开始帧，因为我们是4×4，所以设置为0-15的随机帧，然后Time Mode使用默认的一个粒子的生命周期。

Flipbook混合

当粒子系统处于活跃状态时，粒子会循环几帧，因为Flipbook帧率很低，对于生命周期为5秒的粒子来说其每秒只有3.2帧。这可以通过在连续帧之间混合来平滑过渡，这需要我们在着色器中拿到第二组UV坐标和动画混合因子，我们通过在粒子系统的Renderer模块中启用Custom Vertex Streams来实现。它的作用是声明在材质的顶点着色器中配置哪些粒子的属性可用，我们添加UV2和AnimBlend。可以删除我们不使用的Normal Stream。

添加完自定义顶点流之后会出现一个错误提示，它指示粒子系统和当前使用的着色器不匹配，我们通过在着色器中使用这些顶点流来解决问题。

1. 首先在着色器的属性栏中添加一个切换开关来控制是否使用Flipbook混合，然后声明一个其关联的关键字。

2. 如果启用了Flipbook混合，在顶点输入结构体中声明2套UV，我们可以将它们合并存储在一个float4类型的向量中，随后声明一个混合因子。

3. 在片元输入结构体中声明一个float3类型的向量，用于在顶点函数中存储UV和混合因子。

4. 在InputConfig结构体中声明这个flipbookUVB向量和一个用于判断是否启用了Flipbook混合的bool值，默认为false。

5. 如果启用了Flipbook混合，则使用Flipbook的UV坐标对基础纹理进行二次采样，然后和第一次采样的数据根据混合因子进行插值混合。

6. 最后在片元函数中根据启用情况给这俩声明的属性赋值。