**实例：透明度测试**

**What:**

只要片元的透明度不满足条件，通常是小于某个阈值，那么该片元就会被舍弃。

**Why:**

**How:**

**需要关闭深度写入吗？**

不需要

**进行透明度测试的函数是什么？**

clip，如果给定参数的任何一个分量是负数，就会舍弃当前像素的输出颜色。

**透明度测试标签如何设置？**

”Queue” = “AlphaTest”

”IgnoreProjector” = “True”

”RenderType” = “TransparentCutout”

**实例：透明度混合**

**What:**

使用当前片元的透明度作为混合因子，与已经存储在颜色缓冲中的颜色值进行混合，得到新的颜色。

**Why:**

得到真正的半透明效果

**How:**

**需要关闭深度写入和深度测试吗？**

关闭深度写入，不关闭深度测试。

**为什么要关闭深度写入？有什么坏处？**

如果不关闭深度写入，半透明后的表面会被剔除

坏处：渲染顺序变得很重要

**渲染顺序是什么？**

1. 先渲染不透明物体；
2. 把半透明物体按它们距离摄像机的远近进行排序，然后按照从后往前的顺序进行渲染。

**为什么很难判断哪个物体距离摄像机更近？如何解决这个问题？**

深度缓冲中的值是像素级别的，但现在判断的是整个物体的顺序。

解决方法：拆分物体，分割网格。

**实例：开启深度写入的半透明效果**

**如何实现开启深度写入的半透明效果？**

使用两个Pass来渲染模型：第一个Pass开启深度写入，但不输出颜色，它的目的仅仅是为了把该模型的深度值写入深度缓冲中；第二个Pass进行正常的透明度混合。

Pass{

ZWrite On

ColorMask 0

}

ColorMask用于设置颜色通道的写掩码。

ColorMask RGB|A|0|其他任何RGBA的组合。

**实例：双面渲染的透明效果**

**如何实现双面渲染的透明效果？**

1. 透明度测试：使用Cull Off即可；
2. 透明度混合：使用两个Pass，第一个Pass只渲染背面(Cull Front)，第二个只渲染正面(Cull Back)。