**MSAA**

**MSAA的原理**

MSAA也会为每个像素点设置对应的次像素点。

同一个像素上的多个次像素点，共享像素中心点的像素计算结果。

对每个次像素点测试是否在三角形内，并进行深度-模板测试，当通过后，在像素中心位置采样的结果值就会写入到对应的次像素点。

**MSAA的使用情况**

由于带宽的限制，延迟渲染无法使用MSAA。

即使在手机上，目前使用 MSAA 的情况也比较少。可以说，MSAA 是一种比较经典，但是目前显得略有些过时的抗锯齿方式。

渲染比较细小的物体时。比如渲染头发，细绳子等物体时，因为物体太细，常常得到的图像宽度甚至不足一个像素，基于后处理的方式进行抗锯齿无法使用，这时就只能使用 MSAA 或者 TAA 进行处理了。

**什么是Alpha-To-Coverage？**

Alpha-To-Coverage 是一种借助 MSAA 来实现边缘半透明过渡的效果，需要硬件的支持。

次像素采样点上的颜色，会经过覆盖测试和深度-模板测试，当两个测试都通过后，次像素采样点会写入颜色。当Alpha-To-Coverage开启时，图形硬件会根据SV\_Target输出的 alpha 值，再做一次概率判断，如果这次概率判定成功，才会写入颜色。比如现在有某个像素中有4个次像素点，有两个点已经通过了覆盖测试和深度-模板测试，正在等待写入颜色，当 SV\_Target输出alpha 值为0.5，那么两个待写入的次像素点，很有可能只有一个被写入值，这样就实现了50%的半透明效果。