第五章开始UnityShader学习之旅笔记

1. 在directX10以后，有了一种新的语义类型，系统数值语义（system-value semantics），这类语义以SV（system-value）开头，例如SV\_POSITION、SV\_TARGET
2. SM2、SM3可以用8套TEXCOORD，SM4、SM5可以用16套TEXCOORD
3. Shader调试方法
   1. 假彩色图像：把需要调试的变量映射到[0, 1]之间，把它们作为颜色输出到屏幕，然后通过屏幕上显示的像素颜色来判断这个值是否正确
   2. Visual Studio Graphics Debugger
   3. RenderDoc
   4. Frame Debugger
   5. Inter GPA、NVIDIA NSight、AMD GPU PerfStudio
   6. SnapdragonProfiler（无需手机Root）
      1. Requires a Snapdragon 820 (or later) processor
      2. Requires Android N (or an Android 6.0 device with a graphics driver that supports Vulkan)
      3. Requires a Snapdragon 805 (or later) processor and Android 6.0 (or later)
4. any(x)——输入参数只要有其中一个不为0，则返回true。||运算
5. saturate(x)——如果x 小于0，返回0；如果x 大于1，返回1；否则，返回x
6. 渲染纹理的坐标差异 详见P115
7. 使用fixed类型存储颜色和单位矢量，如果要存储范围更大的数据可以旋转half类型，最差用float
   1. 大多数现代桌面CPU把所有计算都按最高的浮点精度处理；但在移动平台GPU上，不同类型的精度范围不同，运算速度也有差异
8. 慎用分支和循环语句
   1. 尽量把计算向流水线上端移动，例如把放在片元着色器中的计算放到顶点着色器，或者在cpu中进行预计算，再把结果传给shader
   2. 分支判断语句的条件变量最好是常数
   3. 每个分支中包含的操作指令书尽可能少
   4. 分支的嵌套层数尽可能少
9. 扩展阅读
   1. 《GPU精粹2》中的GPU流程控制一章，深入介绍流程控制语句在GPU上的性能影响