# ShaderLab

## UnityShader基础

**常见流程：**

创建材质

创建Unity Shader，赋值给上一步创建的材质

材质赋值到要渲染的对象

在材质中调整Unity Shader的属性，以得到满意的效果

**Unity Shader 分类：**

Standard Surface Shader. //包含标准光照模型的表面着色器模版

Unlit Shader //不含光照，但是报含雾效的 基本定点/片元着色器

Image Effect Shader // 为我们实现各种屏幕后处理效果提供基本模版

Compute Shader // 特殊shader 旨在利用GPU的并行性进行常规渲染流水无关的计算。

Shader Variant Collection

**Shaderlab 基础结构：**

**Shader “ShaderName”{**

**Properties{. //属性**

**}**

**SubShader{ //显卡A使用的着色器**

**//真正意义上的Shader代码**

**//表面着色器（Surface Shader）**

**//定点/片元着色器（Vertex/Fragment Shader）**

**//固定函数着色器（Fixed Function Shader）**

**}**

**SubShader{ //显卡B使用的着色器**

**}**

**}**

**SubShader:**

**Unity 加载 Unity Shader 的时候会扫描所有的SubShader语义块，然后选择第一个能够在平台上运行的SubShader. 如果读不支持的话，会使用Fallback语义指定的UnityShader**

**SubShader{**

**//可选的**

**[Tags]**

**//可选的**

**[RenderSetup]**

**Pass{**

**}**

**// Other Passes**

**}**

**常见状态 [RenderSetup]**

**Cull. Cull Back|Front|Off. 设置剔除模式； 剔除 背面｜正面｜关闭剔除**

**ZTest ZTest Less Greater | LEqual| GEqual |Equal | NotEqual | Always. 设置深度测试时使用的函数**

**ZWrite ZWrite On|Off 开启/关闭深度写入**

**Blend. Blend SrcFactor DstFactor 开启并设置混合模式**

**标签：**

**字符串键值对：**

**Tags{“TagName1” = “values” “tagName2”=”value2”}**

**通知Unity渲染引擎，SubShader 希望怎样以及何时渲染这个对抗**



**表面着色器**

**Unity自定义的一种着色器**

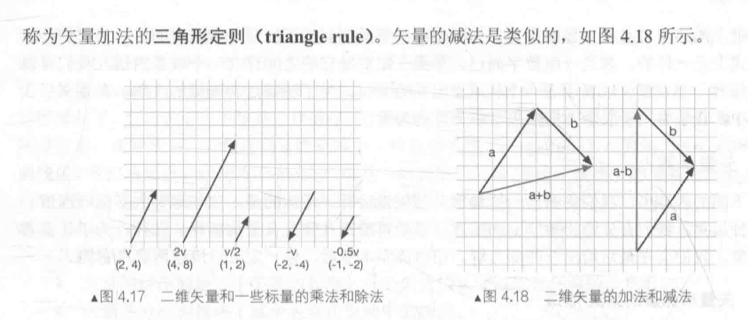
**着色器形式的选择：**

* **与光源相关，使用表面着色器，注意移动平台的性能表现**
* **光照数目少，使用顶点/片元着色器**
* **有很多自定义的渲染效果，选择顶点/片元着色器。**

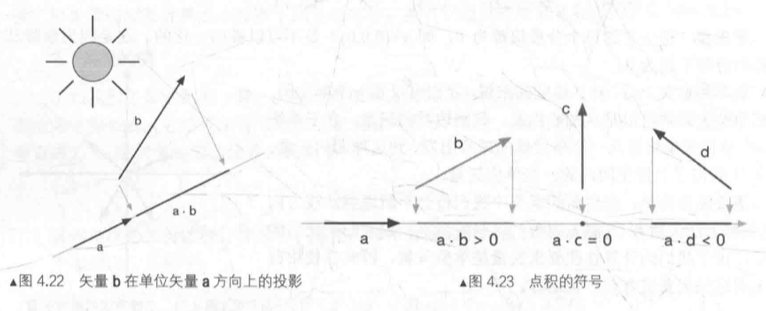
# ****Shader 数学知识****

**矢量：**包含 **模** 和 **方向** 的 **有向线段**

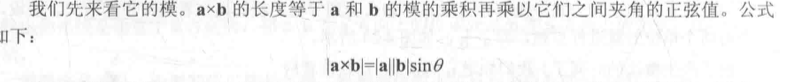
**标量：**只有模没有方向



矢量的点积：

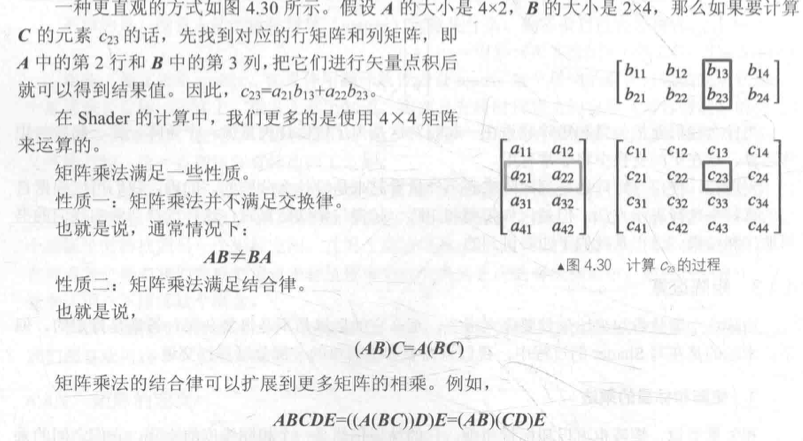


矢量的叉积：



**矩阵：**

**r n 矩阵 X n c 矩阵 => r c 矩阵**

****

**矩阵的几何意义：**

**缩放**

**旋转**

**错切**

**镜像**

**正交投影**

**平移矩阵**

**缩放矩阵**

**旋转矩阵**

**一般变换顺序：**

**先缩放 -> 旋转 -> 平移**