



唐老狮系列教程

渲染路径概述



唐老狮系列教程-渲染路径概述

知识回顾



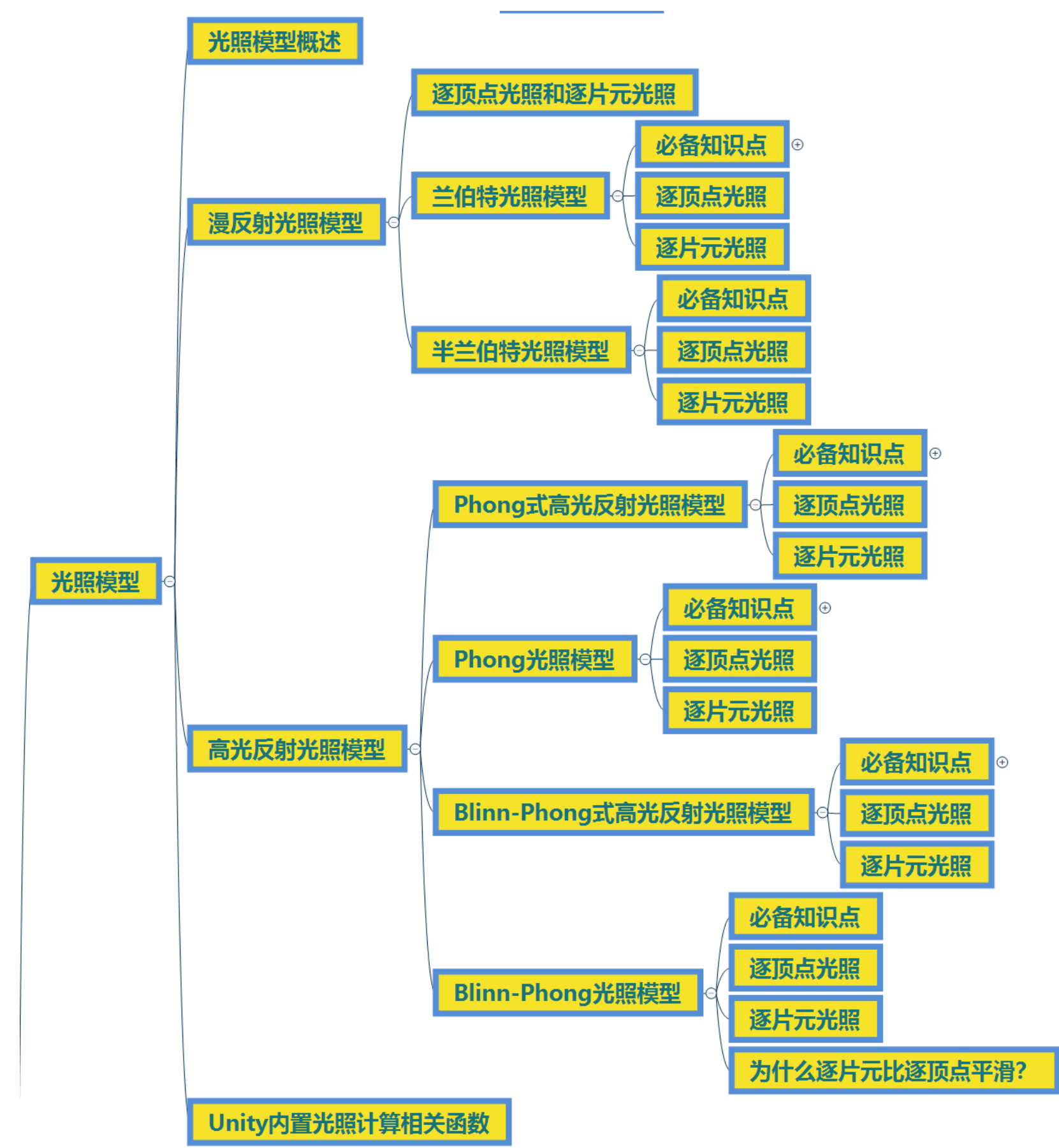
唐老狮系列教程-渲染路径概述

知识回顾

我们在之前的Shader入门相关知识的学习中已经学习过光照模型相关知识，我们目前已经可以实现出一些基础的光照表现效果

但是需要注意的是，我们之前实现的那些光照相关Shader都**不能直接应用到项目开发中**，因为它还不够完善。

比如，**之前的光照模型**相关知识中，我们在Shader中**只考虑了场景中的平行光**，如果我们的场景中**没有平行光会导致之前的Shader出现错误的表现效果**。



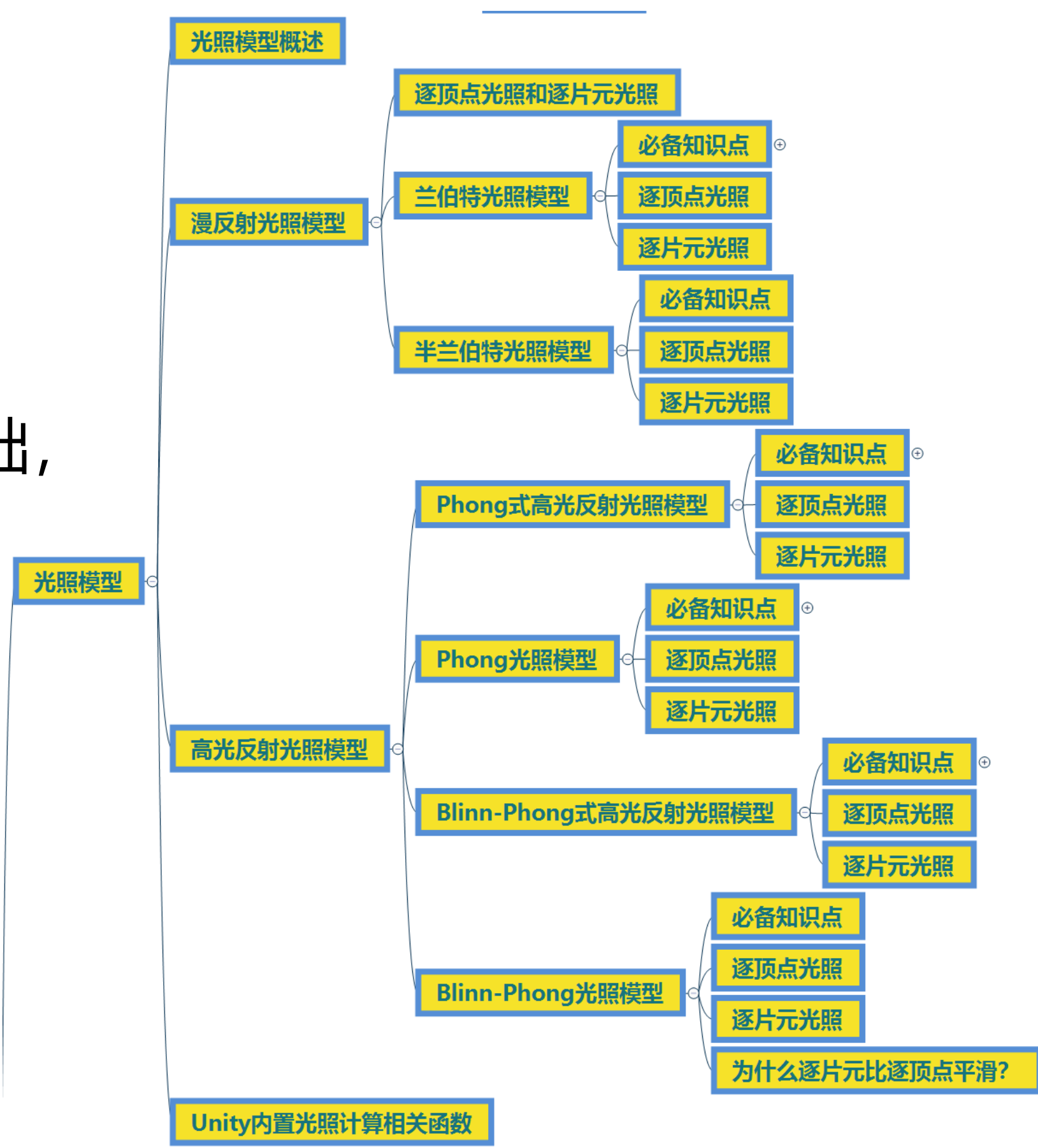


唐老狮系列教程-渲染路径概述

知识回顾

在实际的游戏开发中，我们场景中的光源肯定是更多、更复杂的！
一个平行光的处理，完全不能满足我们的需求。
因此之前关于光照模型的相关学习，只是为了给我们打下一个基础，
让我们能够理解光照处理的底层逻辑是光照模型的计算。

要处理更多的光源，我们就需要了解Unity底层是如何处理
这些光源的，我们将首先学习渲染路径相关的知识来了解这一点





唐老狮系列教程-渲染路径概述

| 主要讲解内容



唐老狮系列教程-渲染路径概述

主要讲解内容

1. 渲染路径是什么
2. 渲染路径的种类和设置
3. LightMode标签的作用



唐老狮系列教程-渲染路径概述

渲染路径是什么



唐老狮系列教程-渲染路径概述

渲染路径是什么

渲染路径（Rendering Path）是指在图形渲染过程中，图形引擎按照特定的步骤和顺序来处理场景中的几何、光照、材质等信息，最终生成屏幕上的图像的一种算法或策略。

它决定了图形引擎如何组织和执行渲染过程，以产生最终的视觉效果。

对于我们来说：在Unity中，**渲染路径决定了光照如何应用到Unity Shader中**，如果要在Unity Shader中和光源打交道，我们需要为每个Pass渲染通道匹配对应的渲染路径，这样**才能在Shader当中获取到正确的光源数据进行处理。**

总而言之：渲染路径会影响光照处理，从而影响最终的渲染效果（光照、阴影等）。

存在多种不同的渲染路径

为什么会影响：我们可以简单理解，使用不同的渲染路径时，Unity在Shader中准备光源数据的流程是不同的，那么我们在Shader开发时，获取光源数据的方式就会有所不同



唐老狮系列教程-渲染路径概述

渲染路径的种类和设置



唐老狮系列教程-渲染路径概述

渲染路径的种类和设置

我们可以在Camera组件中的Rendering Path（渲染路径）对其进行修改
在**内置渲染管线**中主要有3种渲染路径，分别是：

1. Forward（前向渲染路径）

默认的标准的渲染方式，适用于相对简单的场景和较少的光源

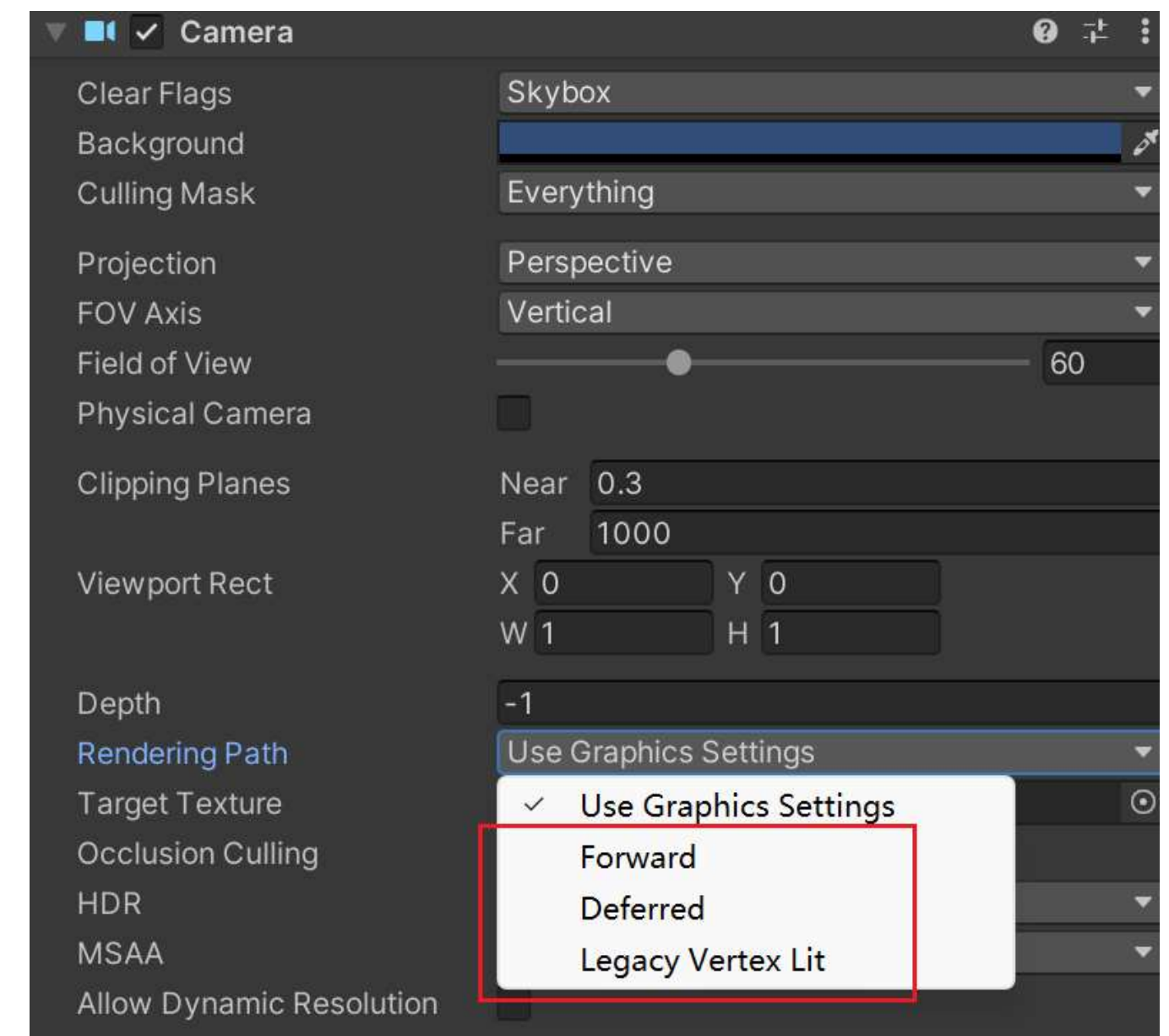
2. Deferred（延迟渲染路径）

可以处理较复杂的场景，有大量光源时可以提供更好的性能

3. Legacy Vertex Lit（(遗产)顶点照明渲染路径）

较适用于为简单的渲染方式，适用于性能受限的场景。

基本已经不会使用



注意：当显卡不支持选定的渲染路径时会自动选择一个较低精度的渲染路径
比如不支持延迟渲染路径时，前向渲染路径会被采用



唐老狮系列教程-渲染路径概述

| LightMode标签的作用



唐老狮系列教程-渲染路径概述

LightMode标签的作用

```
Pass
{
    Tags{ "LightMode" = "ForwardBase" }
```

我们之前在编写Shader时都会使用**LightMode**（光模式）标签。

它的主要作用就是来**指明该 Pass 匹配的渲染路径是哪种**

只要匹配正确，我们便可以获取到正确的光源相关数据

注意：

LightMode标签通常应该与Camera中的Rendering Path匹配

用于指定Pass在渲染过程中的哪个阶段

如果它们不匹配，可能导致渲染不正确



唐老狮系列教程-渲染路径概述

LightMode标签的作用

LightMode 标签支持的渲染路径设置选项有

标签名	描述
Always	不管使用哪种渲染路径，该Pass始终渲染，但不会计算任何光照
ForwardBase	前向渲染路径的基础照明阶段执行的Pass，该Pass会计算环境光、重要的平行光、逐顶点/SH光源和Lightmaps
ForwardAdd	前向渲染路径的额外照明阶段执行的Pass，该Pass会计算额外的逐像素光源，每个Pass对应一个光源
Deferred	延迟渲染路径中执行的Pass，用于几何阶段之后执行光照计算，该Pass会渲染G缓冲（G-buffer）
ShadowCaster	用于生成阴影图的Pass，通常用于阴影投射。把物体的深度信息渲染到阴影映射纹理或一张深度纹理中
MotionVectors	用于生成运动矢量的Pass，通常用于运动模糊
Meta	用于在渲染对象元数据时执行的Pass
Vertex、VertexLMRGBM、VertexLM	用于遗留的顶点照明渲染（基本不使用了）



唐老狮系列教程-渲染路径概述

LightMode标签的作用

在Shader开发中如果我们**不进行LightMode渲染标签的设置**。
比如摄像机默认的是前向渲染路径，但是我们没有为Pass设置相关的标签，那么这个Pass会被当作一个顶点照明渲染路径的Pass。
这时光源相关的数据就不会被正确的进行赋值，我们计算出来的结果就会出现错误，从而**可能呈现出错误的渲染效果**。
因此

在进行Unity Shader开发时，我们需要对Pass渲染通道进行正确的LightMode标签设置，从而匹配当前使用的渲染路径

标签名	描述
Always	不管使用哪种渲染路径，该Pass始终渲染，但不会计算任何光照
ForwardBase	前向渲染路径的基础照明阶段执行的Pass，该Pass会计算环境光、重要的平行光、逐顶点/SH光源和Lightmaps
ForwardAdd	前向渲染路径的额外照明阶段执行的Pass，该Pass会计算额外的逐像素光源，每个Pass对应一个光源
Deferred	延迟渲染路径中执行的Pass，用于几何阶段之后执行光照计算，该Pass会渲染G缓冲（G-buffer）
ShadowCaster	用于生成阴影图的Pass，通常用于阴影投射。把物体的深度信息渲染到阴影映射纹理或一张深度纹理中
MotionVectors	用于生成运动矢量的Pass，通常用于运动模糊
Meta	用于在渲染对象元数据时执行的Pass
Vertex、VertexLMRGBM、VertexLM	用于遗留的顶点照明渲染（基本不使用了）



唐老狮系列教程-渲染路径概述

| 总结



唐老狮系列教程-渲染路径概述

主要讲解内容

1. 渲染路径是什么

它决定了图形引擎如何组织和执行渲染过程，以产生最终的视觉效果

不同的渲染路径会影响光照处理，从而影响最终的渲染效果

2. 渲染路径的种类和设置

在Camera上可以设置前向渲染路径（Forward）延迟渲染路径（Deferred）顶点照明渲染路径（Legacy Vertex Lit）

3. LightMode标签的作用

进行Unity Shader开发时，我们需要对Pass渲染通道

进行正确的LightMode标签设置，从而匹配当前使用的渲染路径

标签名	描述
Always	不管使用哪种渲染路径，该Pass始终渲染，但不会计算任何光照
ForwardBase	前向渲染路径的基础照明阶段执行的Pass，该Pass会计算环境光、重要的平行光、逐顶点/SH光源和Lightmaps
ForwardAdd	前向渲染路径的额外照明阶段执行的Pass，该Pass会计算额外的逐像素光源，每个Pass对应一个光源
Deferred	延迟渲染路径中执行的Pass，用于几何阶段之后执行光照计算，该Pass会渲染G缓冲（G-buffer）
ShadowCaster	用于生成阴影图的Pass，通常用于阴影投射。把物体的深度信息渲染到阴影映射纹理或一张深度纹理中
MotionVectors	用于生成运动矢量的Pass，通常用于运动模糊
Meta	用于在渲染对象元数据时执行的Pass
Vertex、VertexLMRGBM、VertexLM	用于遗留的顶点照明渲染（基本不使用了）



唐老狮系列教程

Thank
感谢您的聆听