3D Game Programming
Color, Material, and Lighting

Ming-Te Chi
Department of Computer Science,
National Chengchi University
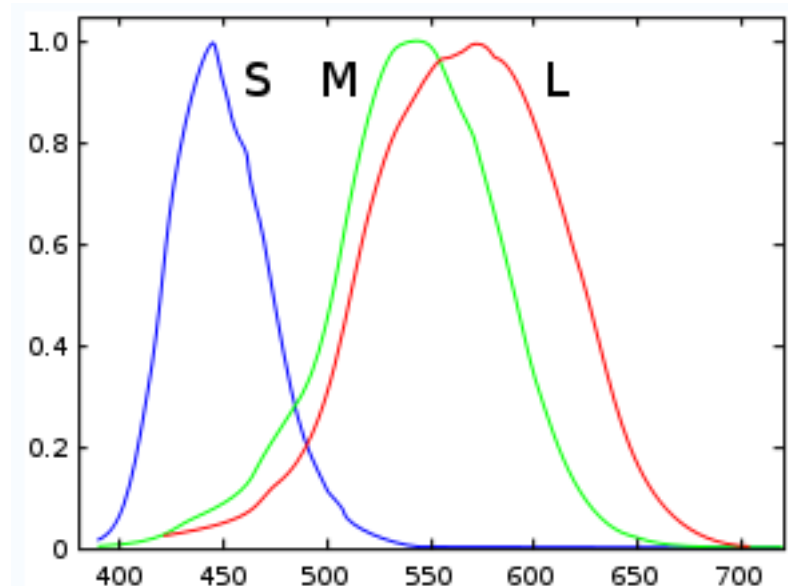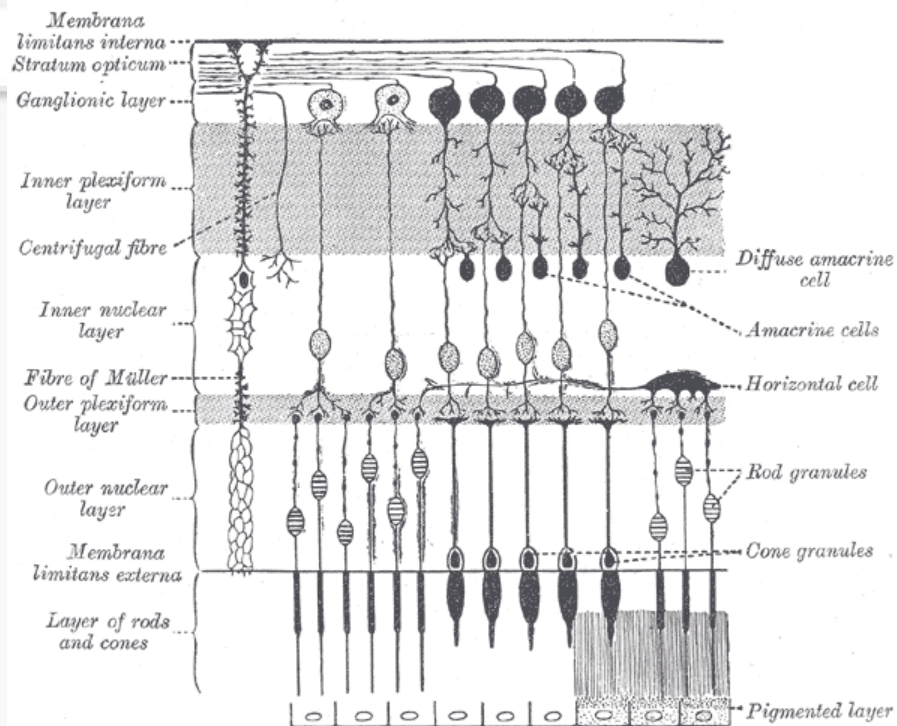
INTERACTIVE MEDIA

多媒體圖形
技術組

Outline

Color, Material, and Lighting
  – Color
  – ShadeModel
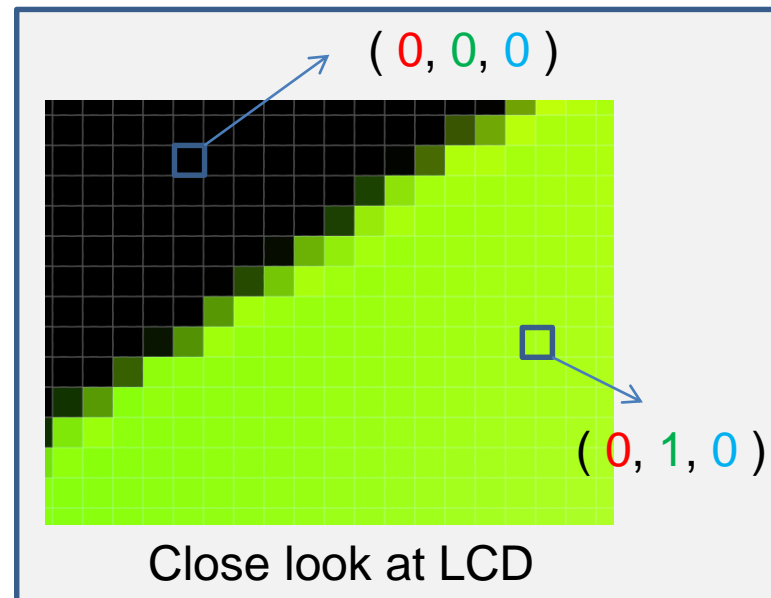Material in Unity
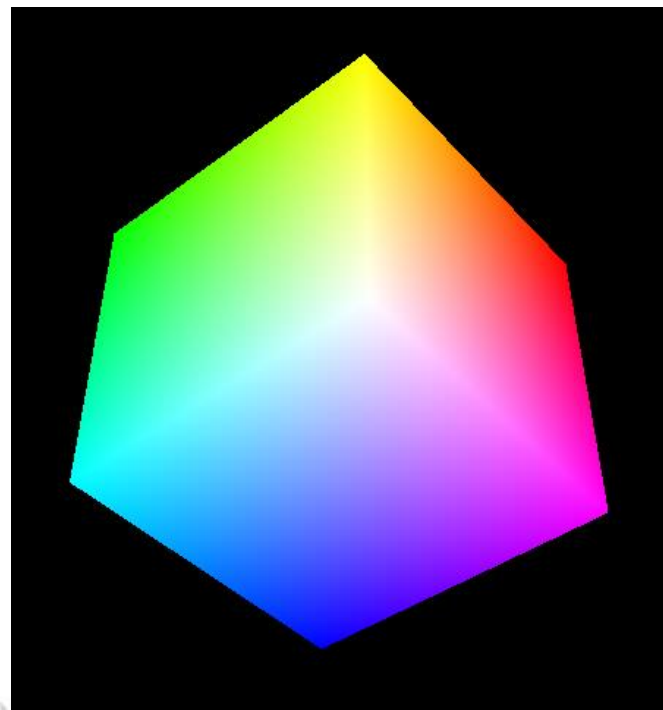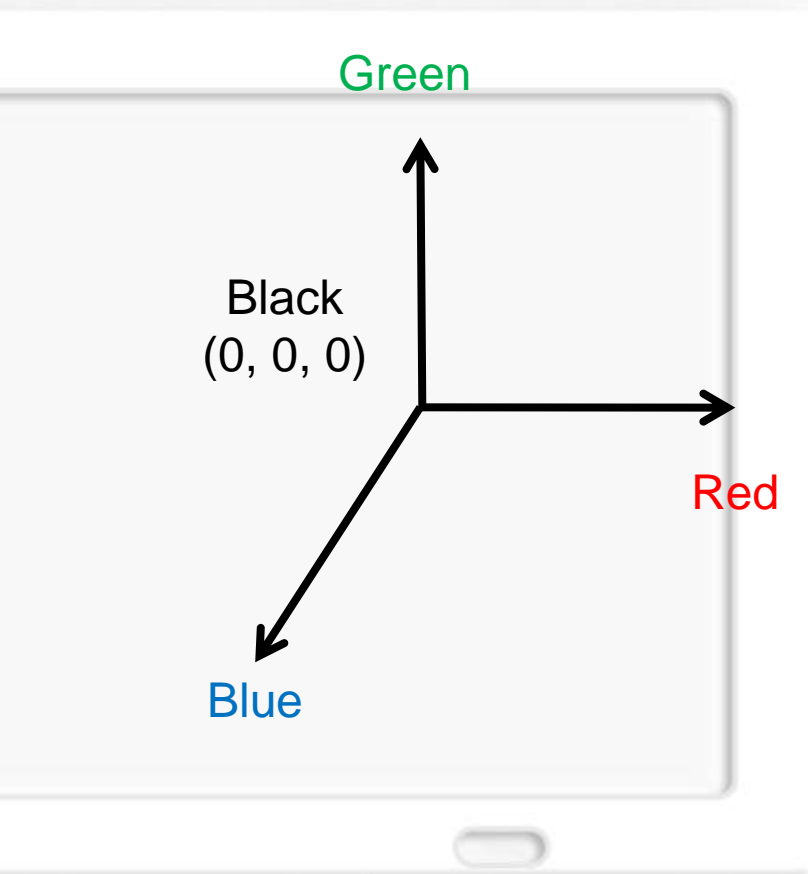Lighting in Unity

Your Personal Photon Detector

PC Display Modes

📎 Screen resolution

📎 Color Channel:
– RGBA

📎 Color Depth
– Each pixel can be represented by
  (red: 0~255, green: 0~255, blue: 0~255)

(0, 0, 0)

(0, 1, 0)

Close look at LCD

The Color Cube

Green

Black
(0, 0, 0)

Red

Blue

Set color

RGBA

e. g. : Red color

RGB 0-255

Unsigned byte(255, 0, 0);

RGB 0-1

Float (1.0f, 0.0f, 0.0f);

Hex FF0000FF

Color

RGB 0-255

R		255
G		244
B		214
A		255

Hexadecimal   FFF4D6

▼ Swatches

Click to add new preset
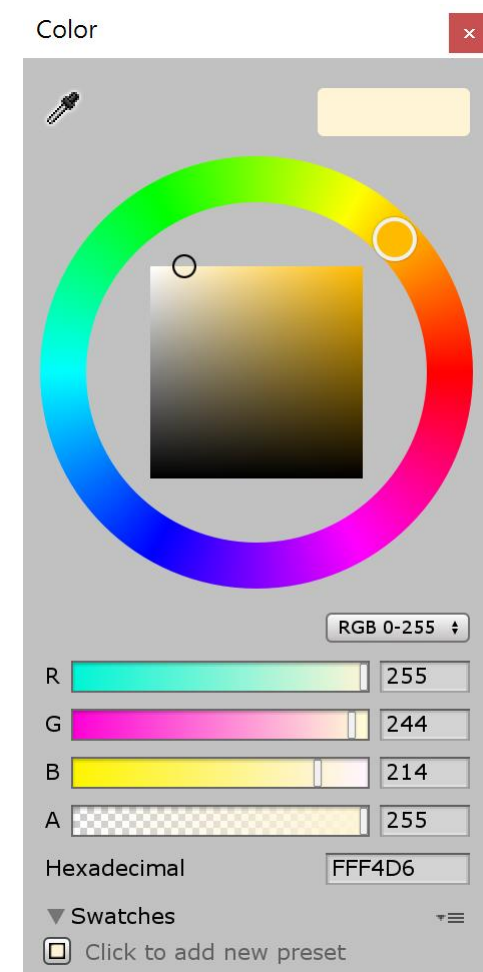
INTERACTIVE MEDIA

Shading

📎 The operation of determining the effect of a light on a material is known as shading

What is Shading

Light as a Particle
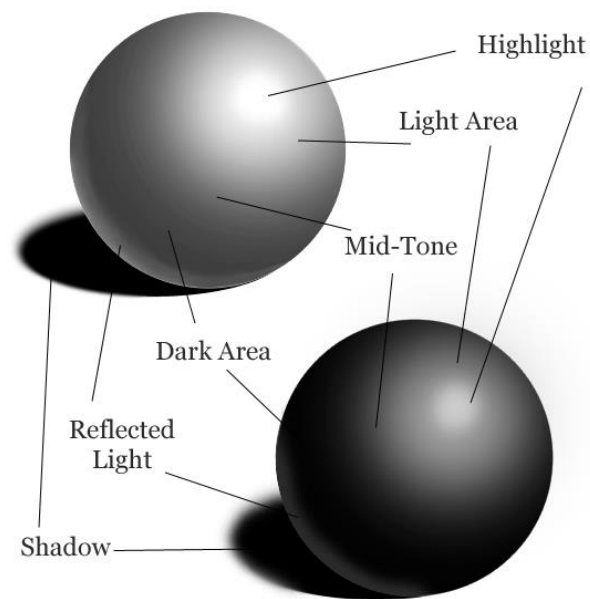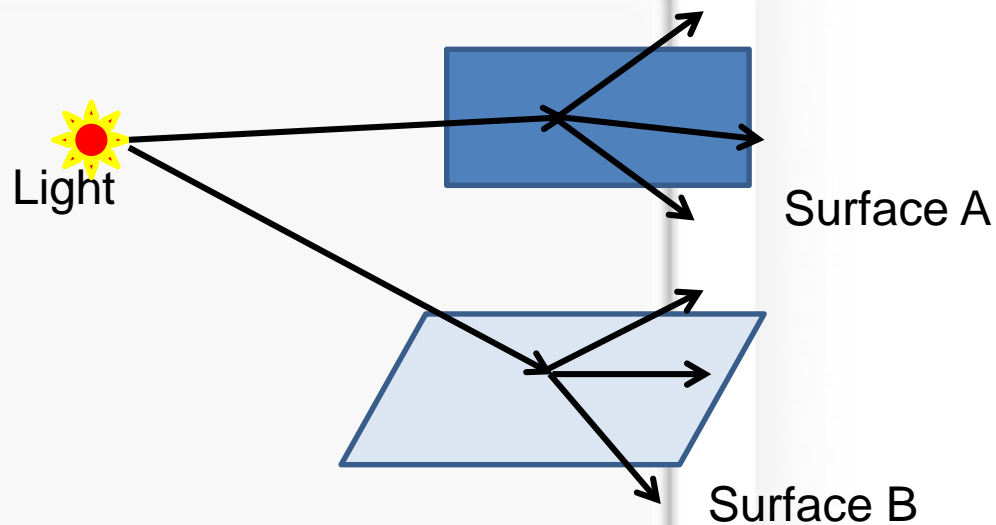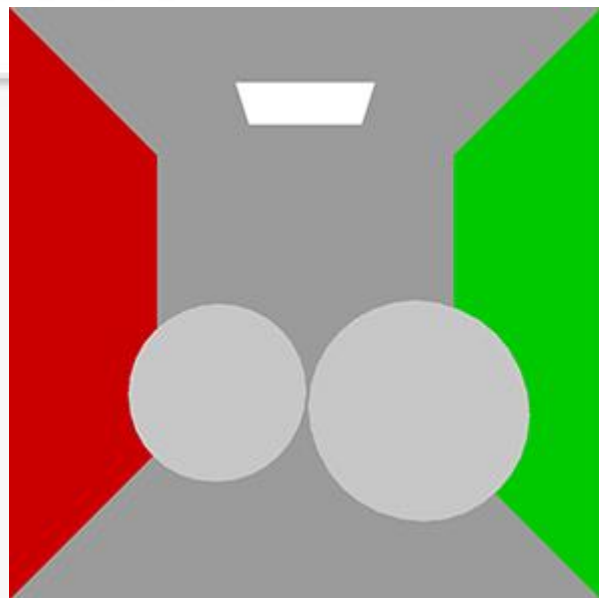

Light

Surface A

Surface B

Highlight
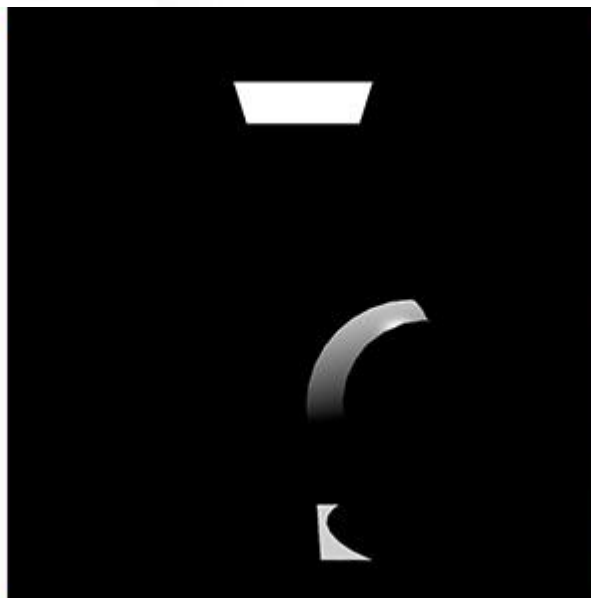
Light Area

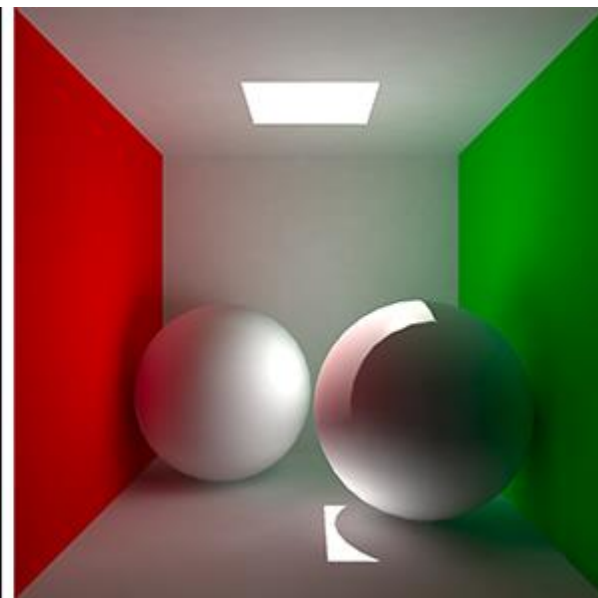Mid-Tone

Dark Area

Reflected Light

Shadow

Lighting and Rendering
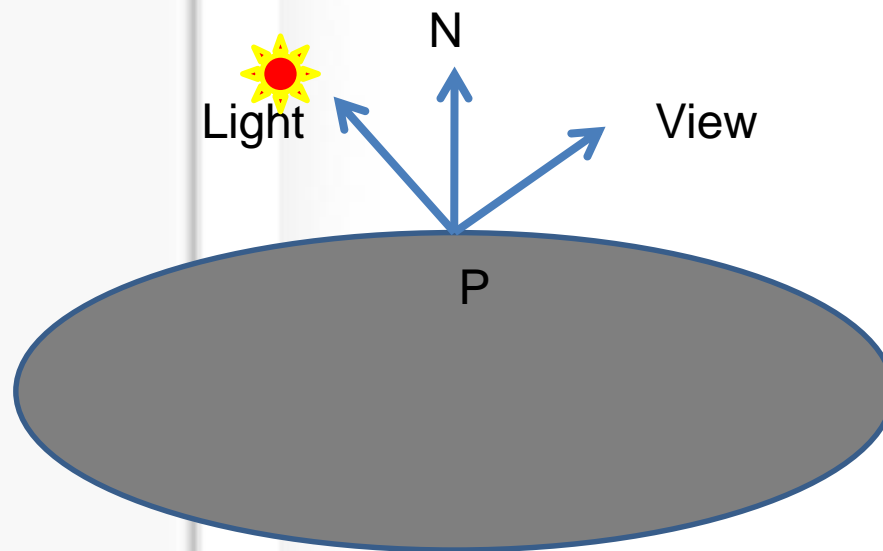
With no lighting

with 'direct light' only

with indirect 'global illumination'

Local illumination

📎 Only consider the light, the observer position, and the object material properties.

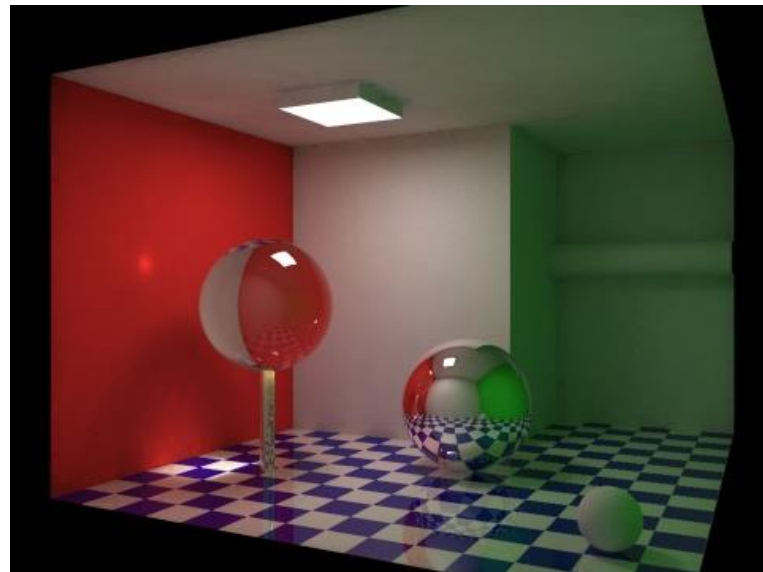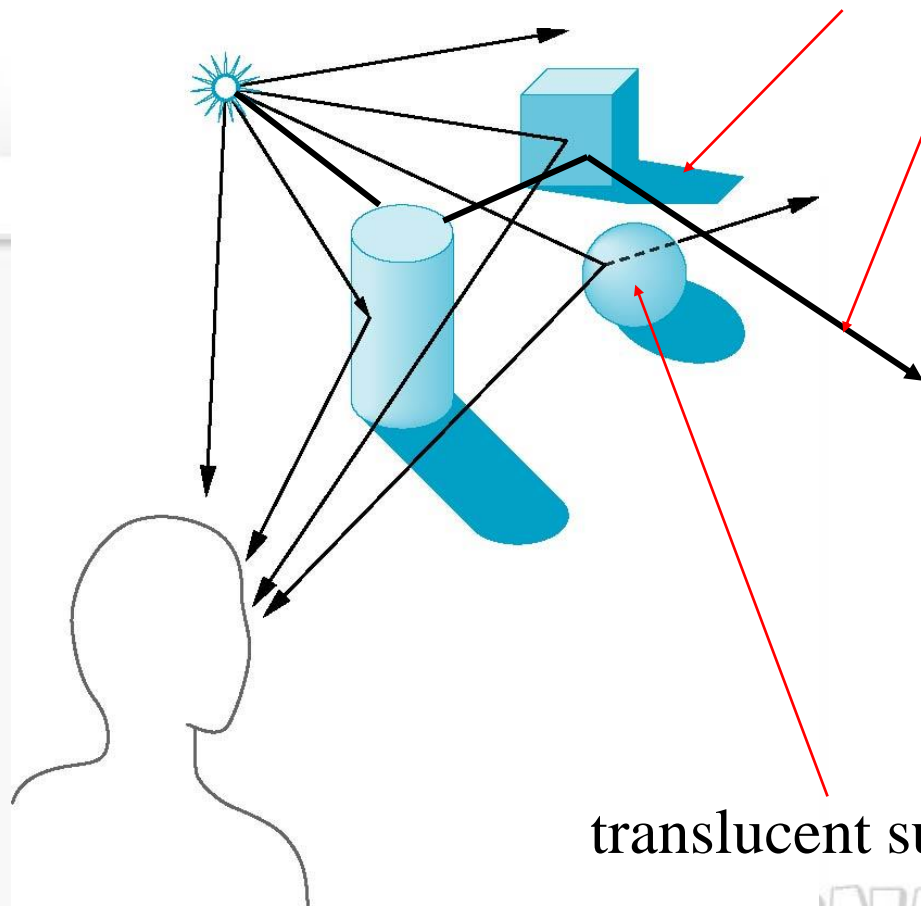# Global Effects

shadow
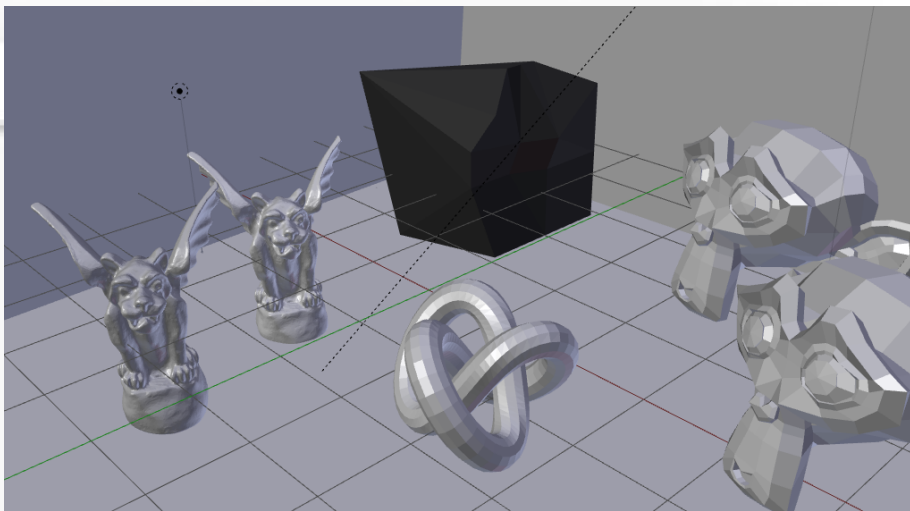
multiple reflection
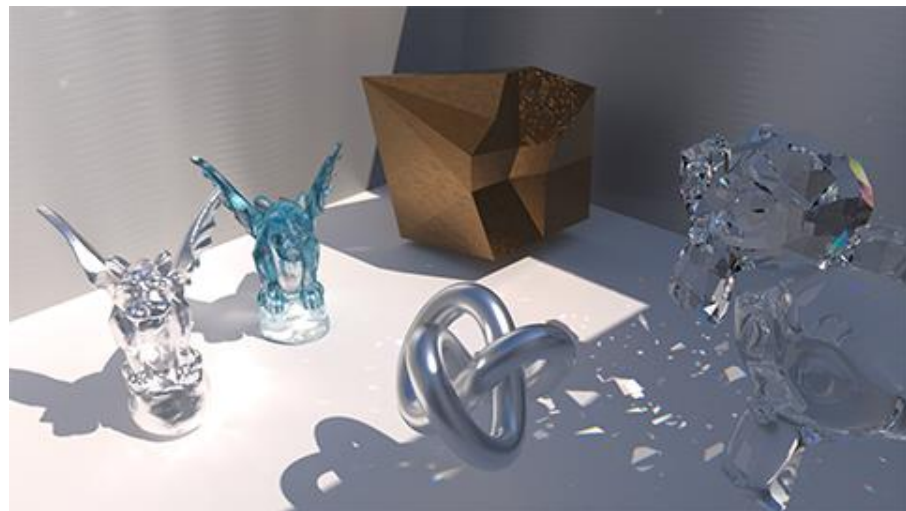
translucent surface

Local versus Global Illumination

Local illumination

global illumination

To understand shading properly, we need to review some basic notions of physics…

Material in the world

📎 Material Properties
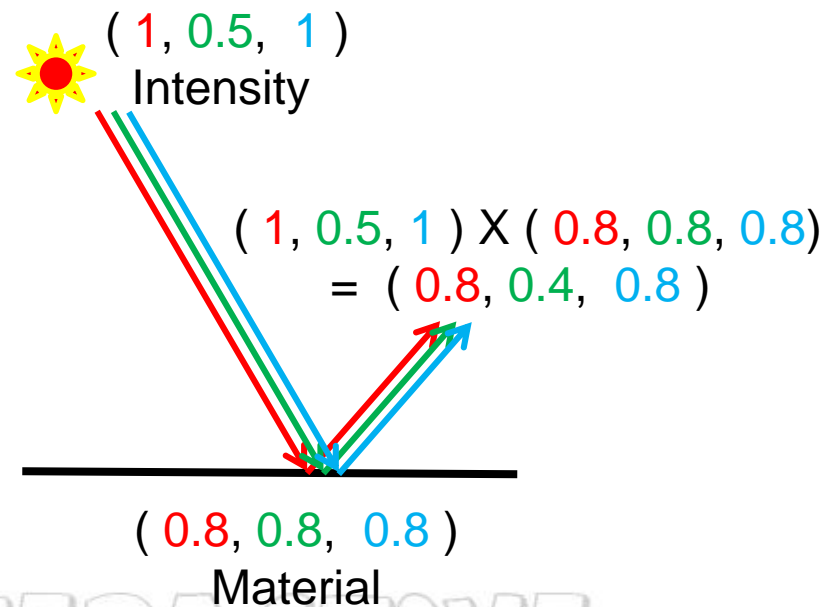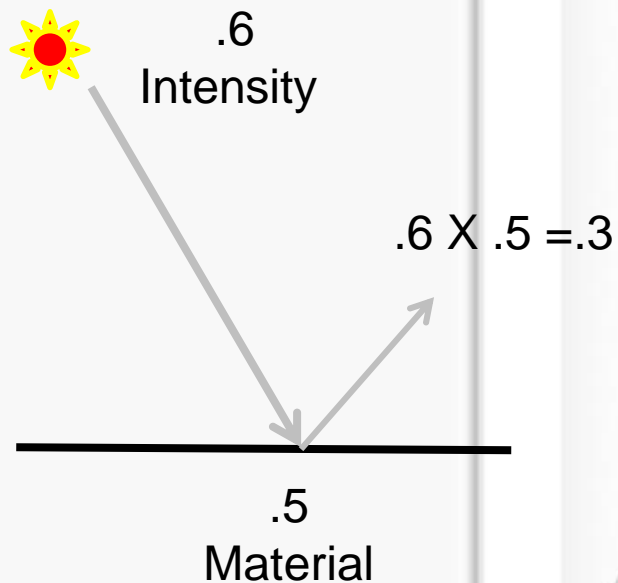
Why we need shading

Suppose we build a model using many polygons and color it with `glColor`. We get something like

But we want
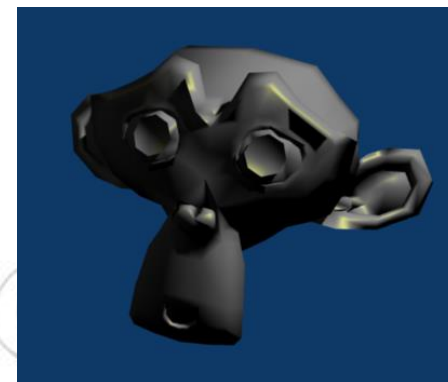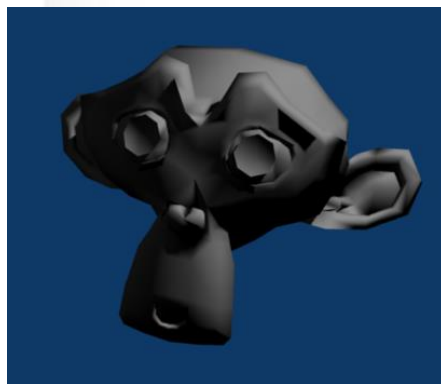
Phong shading (1973)

A simple model that can be computed rapidly
Has three components
- Diffuse
- Specular
- Ambient
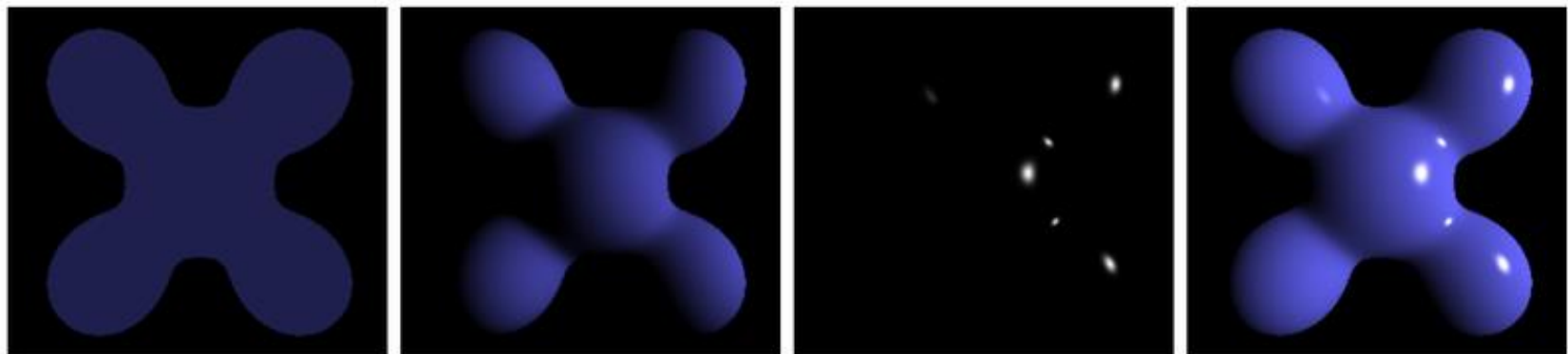
Ambient   +   Diffuse   +   Specular   =   Phong Reflection

Using a Light Source

📎 Surface Normal

Normal

Reflect vector

P
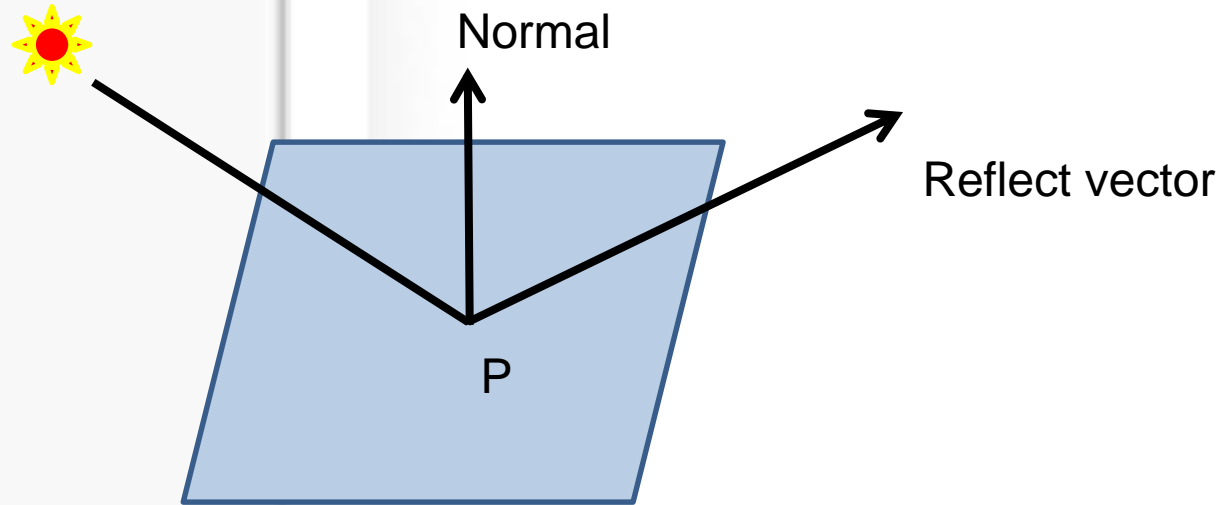
Phong shading (1973)

Uses four vectors
- To source
- To viewer
- Normal
- Perfect reflector

Normal

view vector

reflection vector
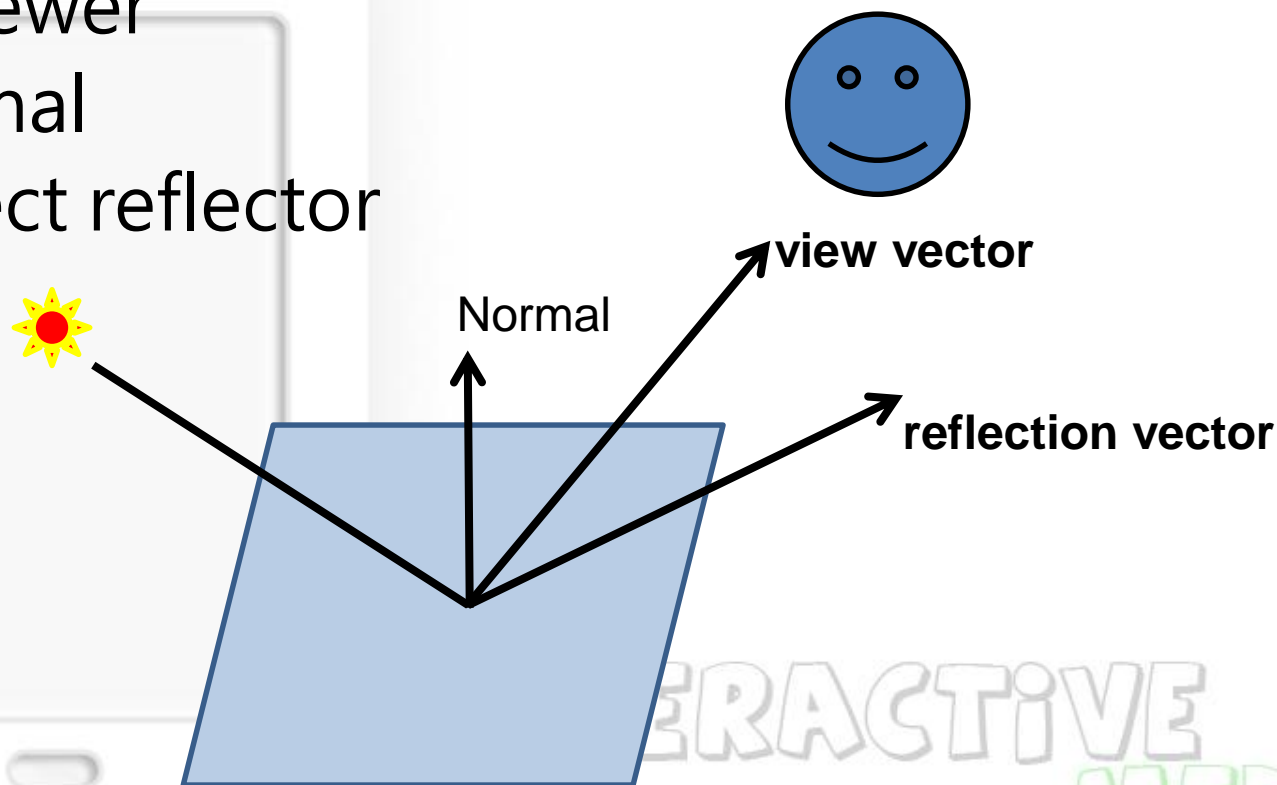
Phong shading

For each light source and each color component, the Phong model can be written (without the distance terms) as

$$I = k_d I_d \ \mathbf{l} \cdot \mathbf{n} + k_s I_s (\mathbf{v} \cdot \mathbf{r})^{\alpha} + k_a I_a \qquad \text{ambient}$$

diffuse   specular

For each color component
we add contributions from
all sources

Find normal

N

w

v

Normal = cross_product(v, w)
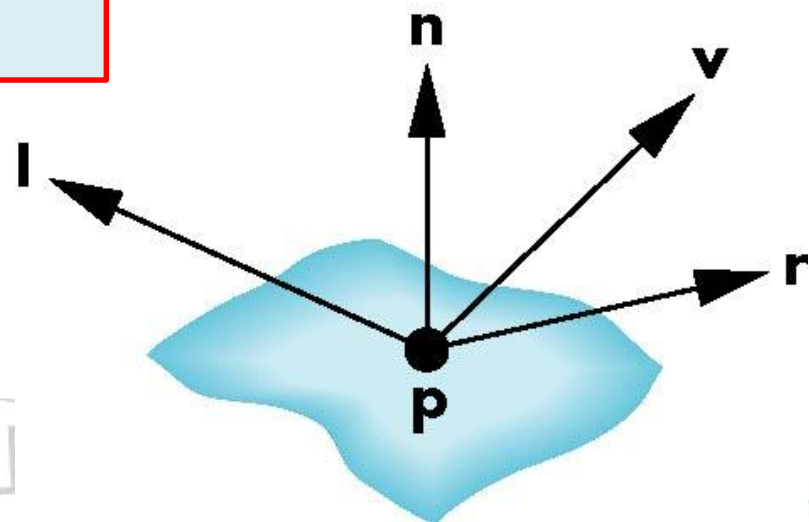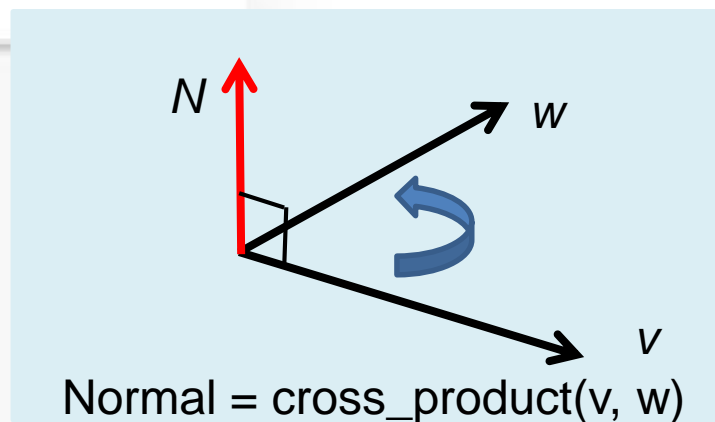
Ambient + Diffuse

Phong Reflection

small   Diffuse Kd   large

Specular Ks

large
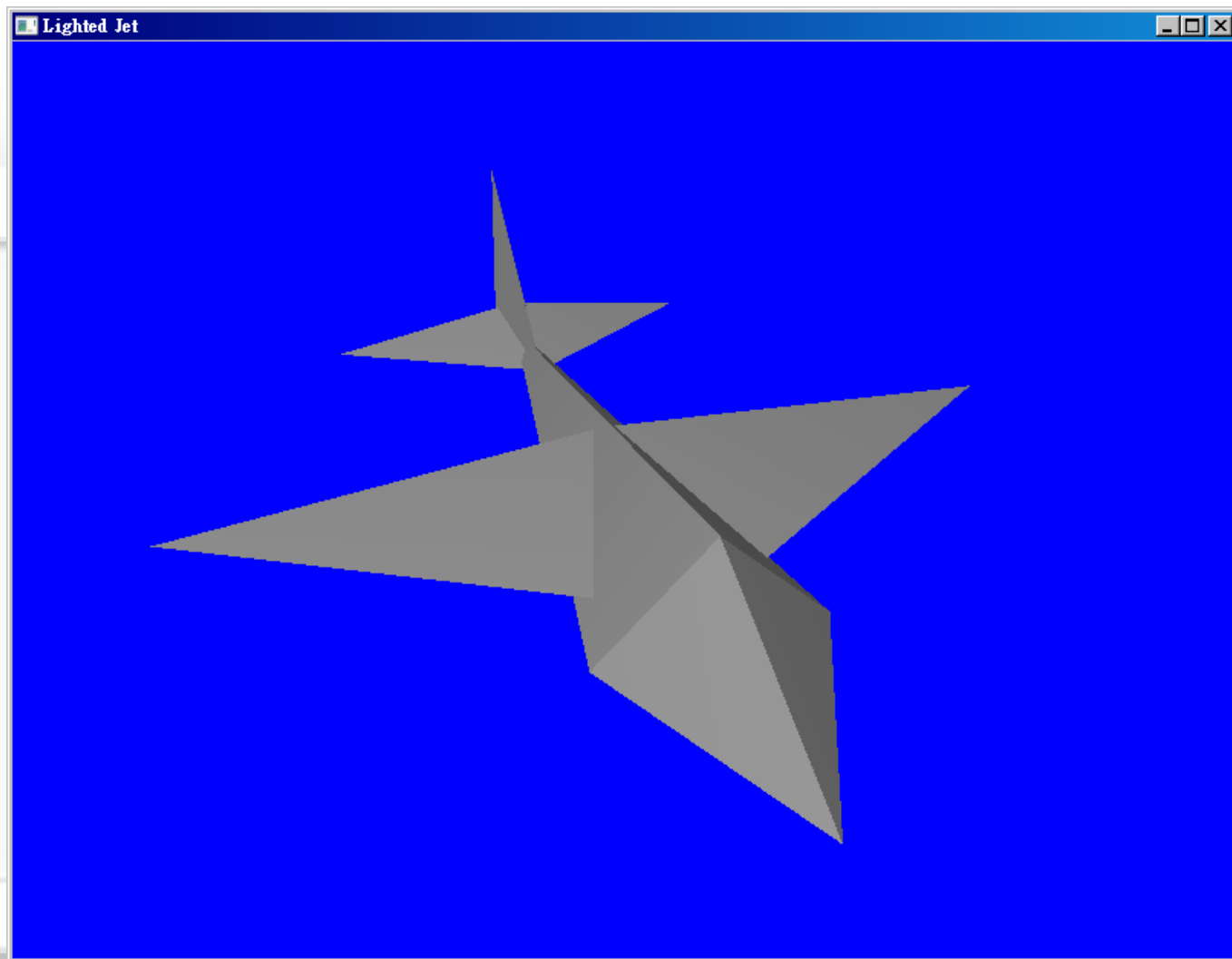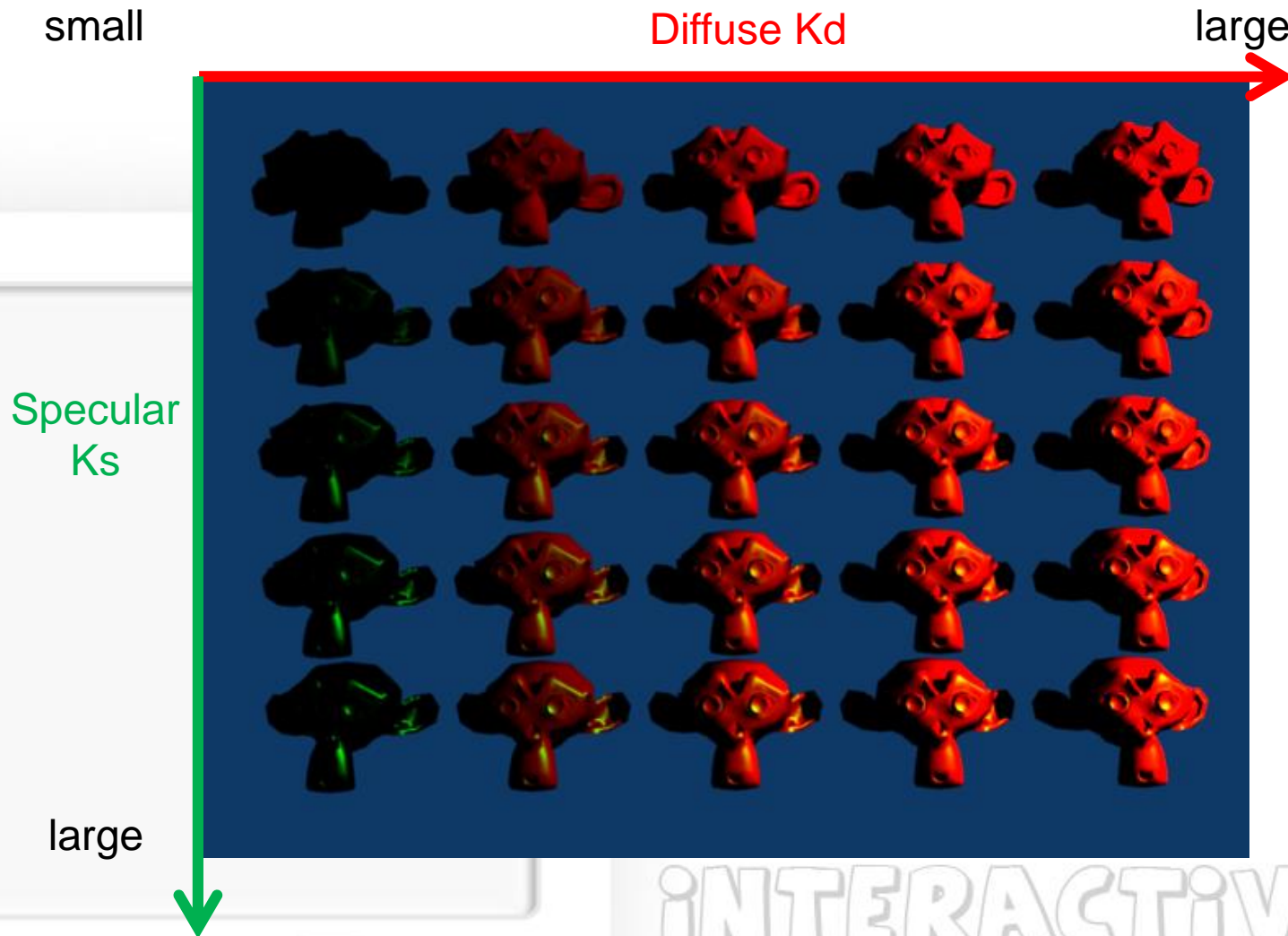
Add specular
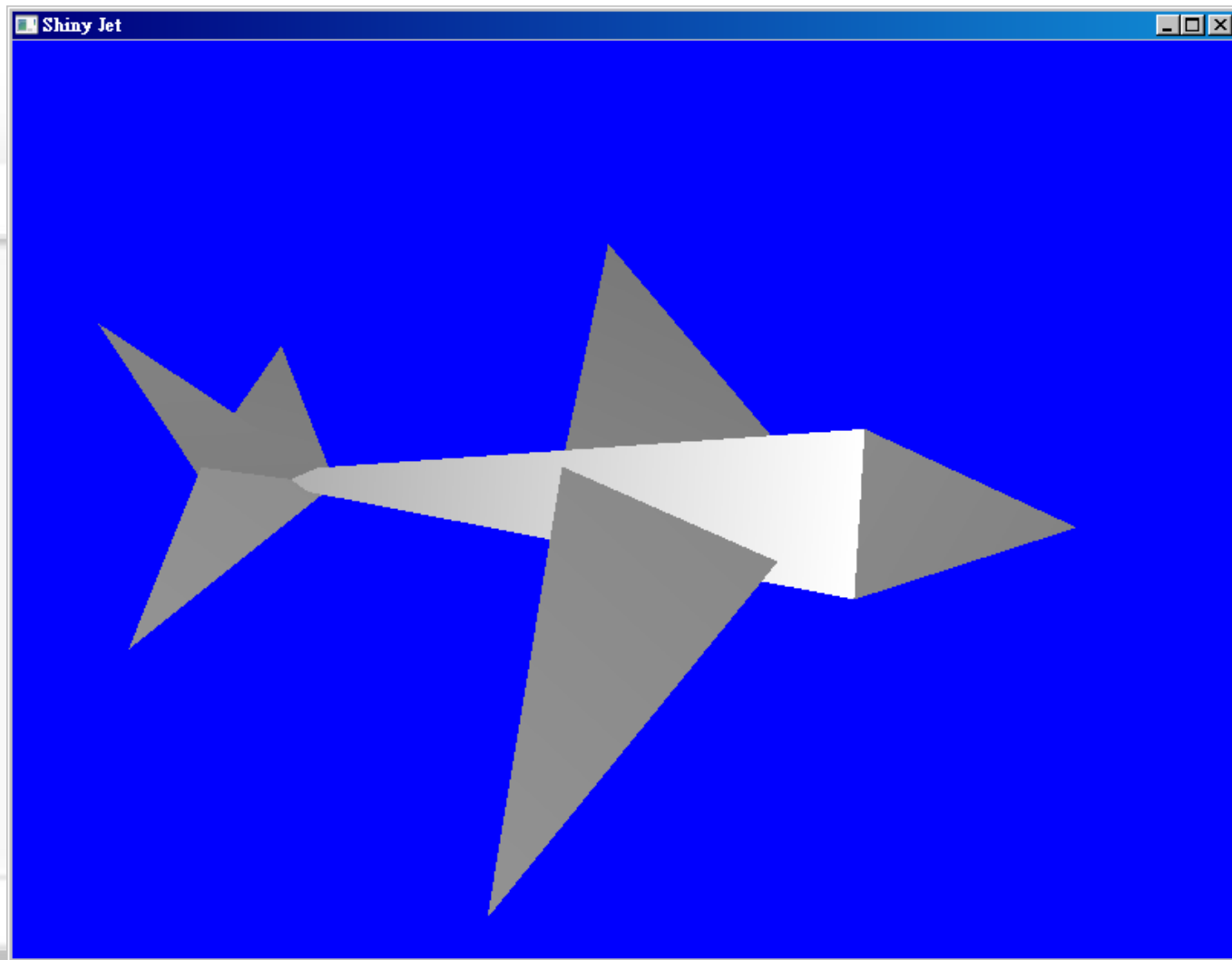
Normal Average
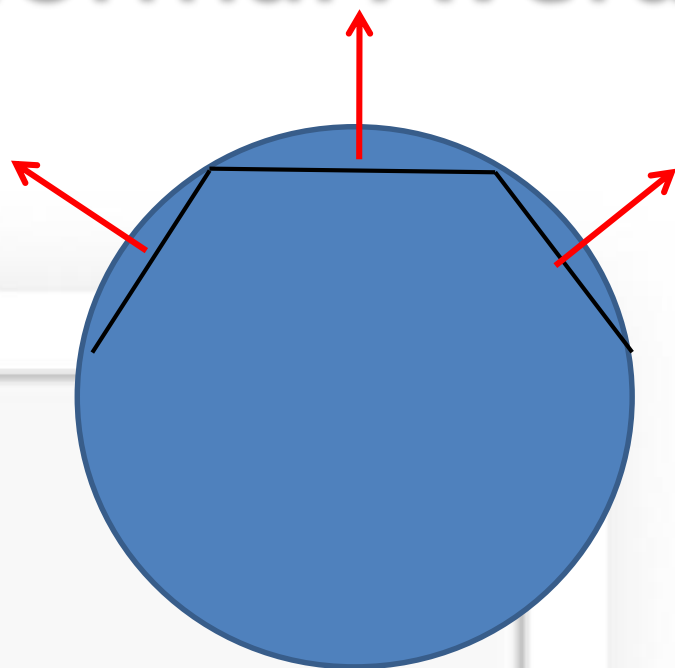
Face normal
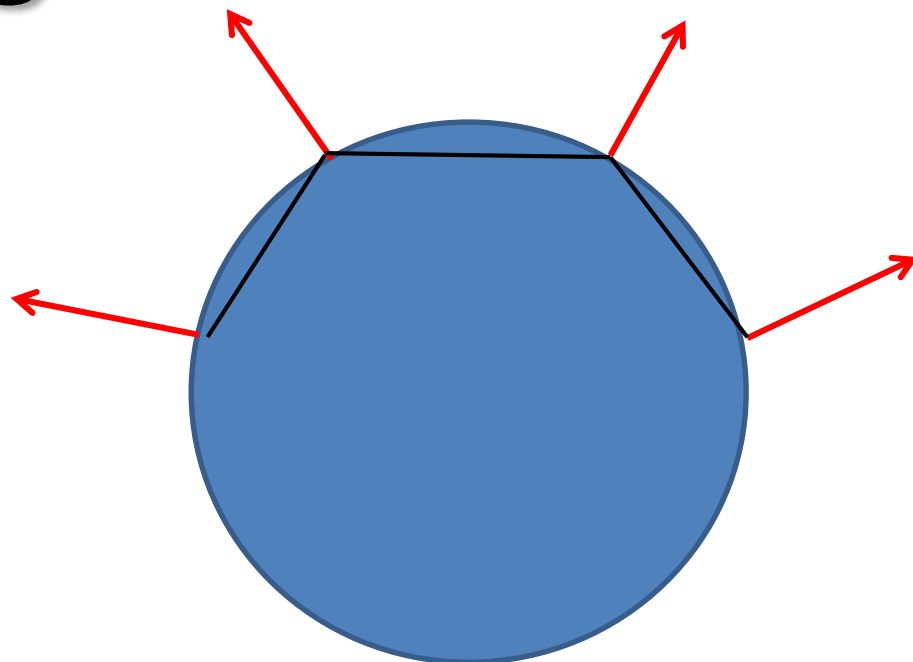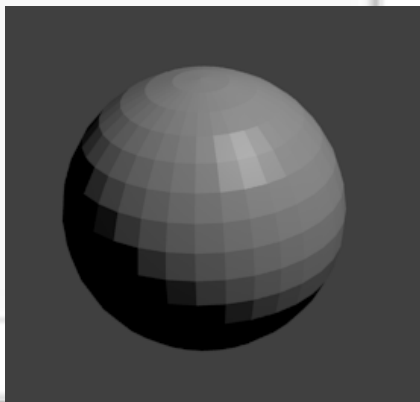
Vertex normal
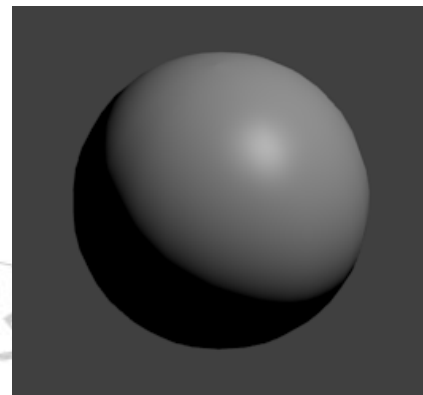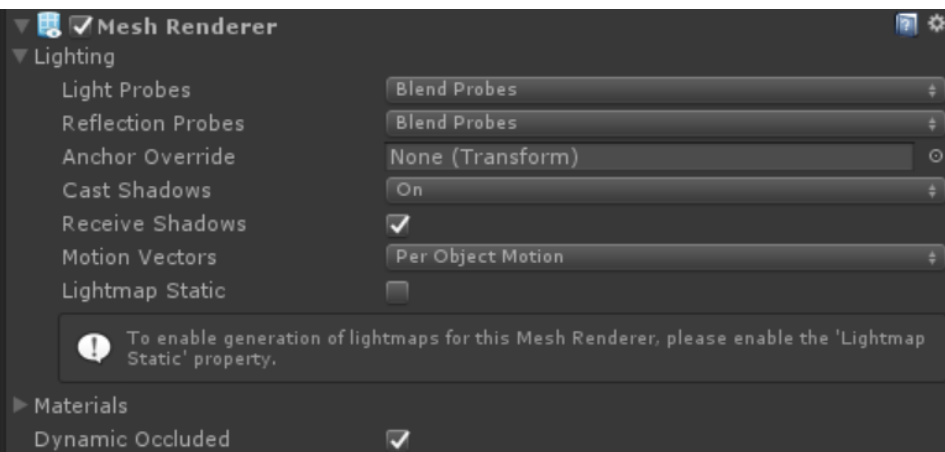
Mesh Renderer

2017.4
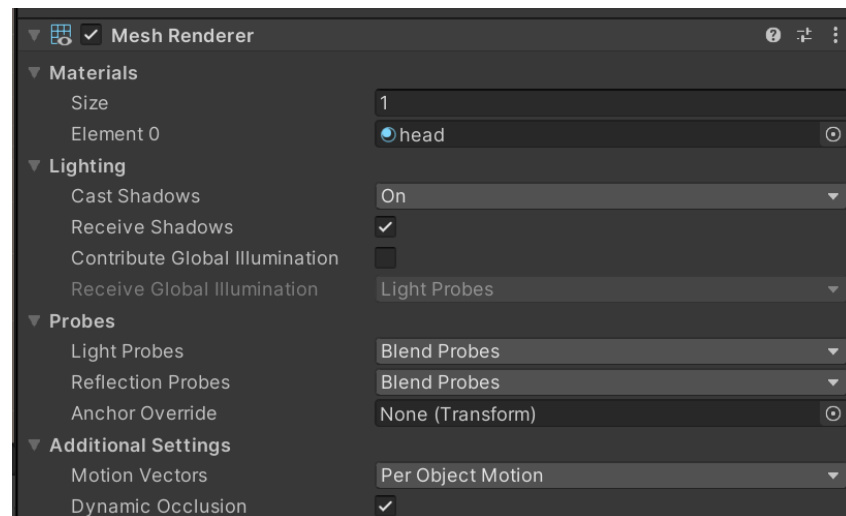
2019.4
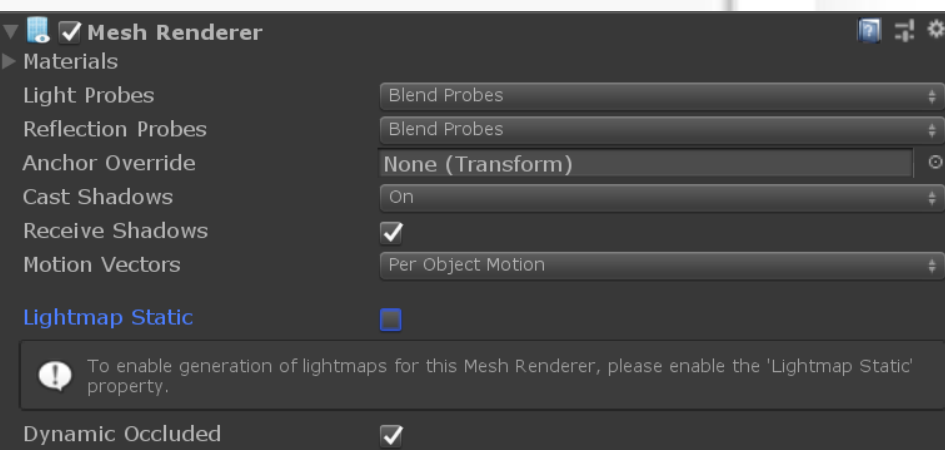
2018.4
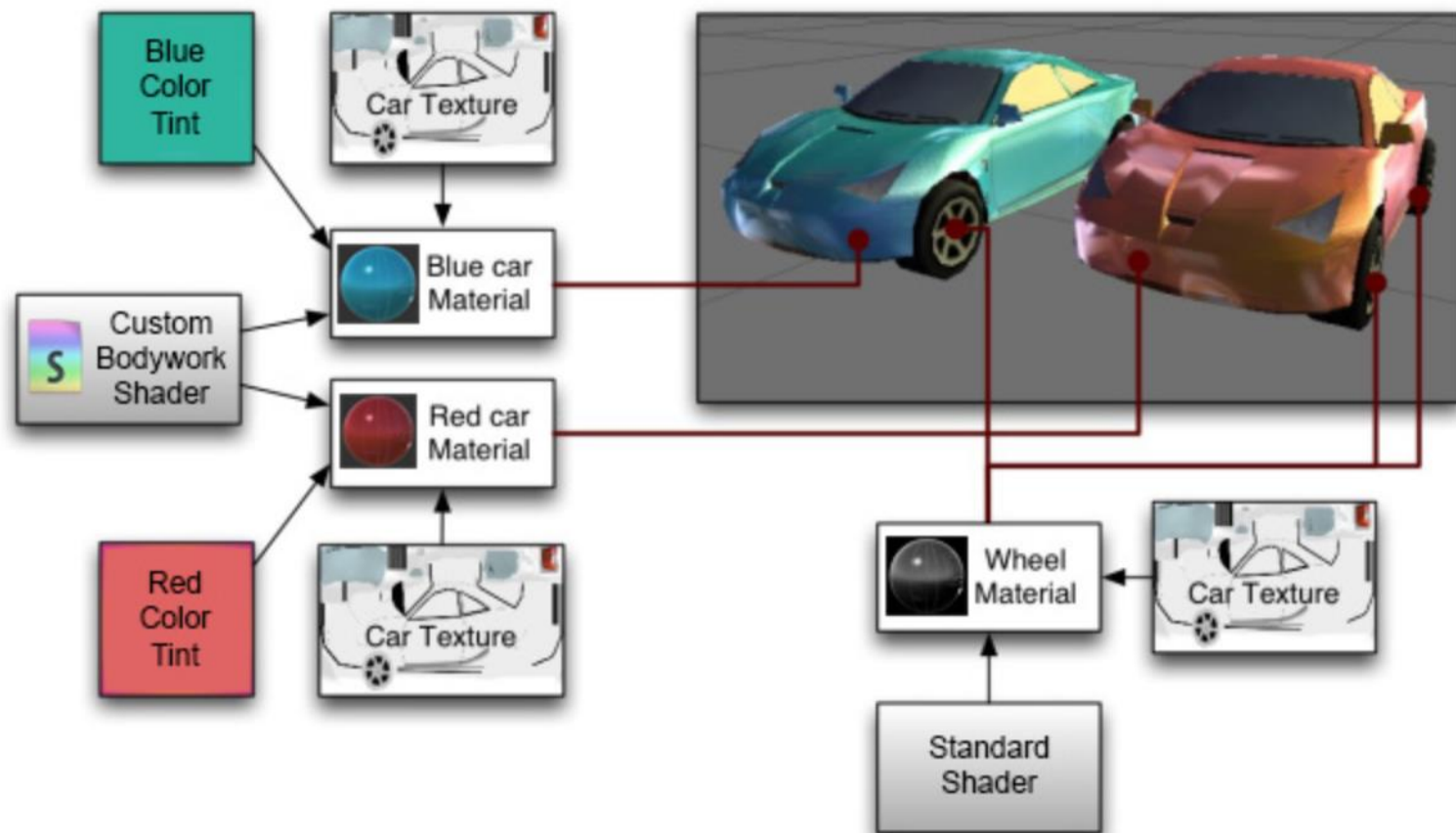
Material in Unity

Material in Unity

- 新增資料夾存放材質以方便管理
  - 在Project畫面空白處[右鍵]→[Create]→[Folder]
  - 可取名為Material
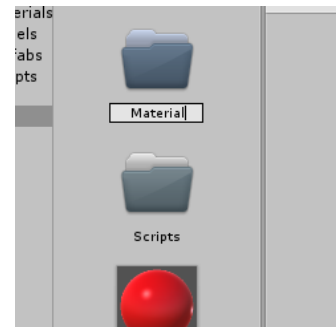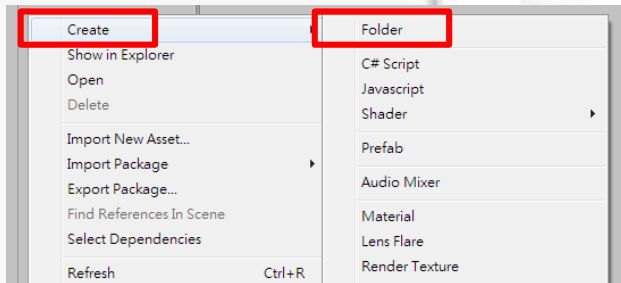
Material in Unity

建立新材質，準備套用到物件上
- 在Project畫面中剛剛新增的資料夾上按[右鍵]→[Create]→[Material]
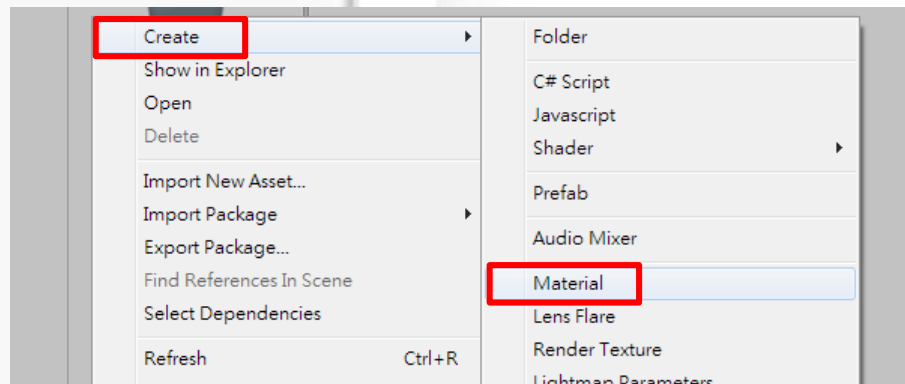
Material in Unity

📎 可在Inspector畫面中調整材質類型及顏色
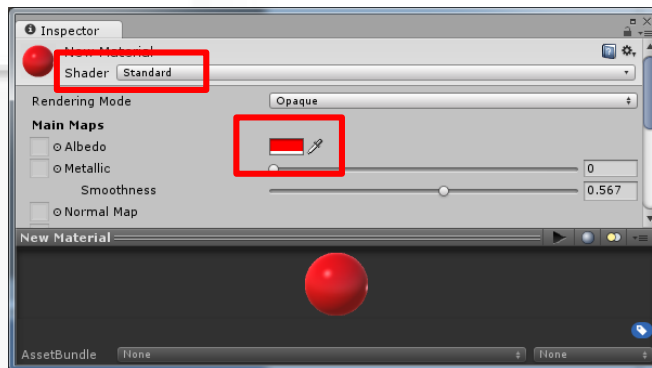
📎 之後把材質拖到Hierarchy畫面的物件中，或直接拖到畫面中的物件，可直接套用材質

Standard

Account ▾ | Layers ▾ | Layout ▾

ℹ Inspector

New Material 2

Shader Standard ▾

| Rendering Mode | Opaque | ‡ |

Main Maps

| ⊙ Albedo | ⬜ 🖉 | |
⊙ Metallic	○──────────────	0
Smoothness	──────────●─────	0.5
Source	Metallic Alpha	‡
⊙ Normal Map		
⊙ Height Map		
⊙ Occlusion		
⊙ Emission	⬛ 🖉	0
Global Illumination	Realtime	‡
⊙ Detail Mask		
Tiling	X 1	Y 1
Offset	X 0	Y 0

Secondary Maps

⊙ Detail Albedo x2		
⊙ Normal Map		1
Tiling	X 1	Y 1
Offset	X 0	Y 0
UV Set	UV0	‡

Forward Rendering Options

| Specular Highlights | ✔ |
| Reflections | ✔ |

New Material 2 ▶ ● ⦿ ▾≡

AssetBundle   None ‡   None

Metallic

0.0   1.0

Specular

Inspector

New Material 2

Shader | Standard (Specular setup) |

| Rendering Mode | Opaque | ⊕ |

Main Maps

⊙ Albedo	☐ 🖉
⊙ Specular	☐ 🖉
Smoothness	━━━━━○━━━━━ 0.5
Source	Specular Alpha ⊕
⊙ Normal Map	
⊙ Height Map	
⊙ Occlusion	
⊙ Emission	☐ 🖉 0
Global Illumination	Realtime ⊕
⊙ Detail Mask	
Tiling	X 1 Y 1
Offset	X 0 Y 0

Secondary Maps

⊙ Detail Albedo x2	
⊙ Normal Map	1
Tiling	X 1 Y 1
Offset	X 0 Y 0
UV Set	UV0 ⊕

Forward Rendering Options

| Specular Highlights | ✔ |
| Reflections | ✔ |

specular

0.0   1.0

METALLIC R

METALLIC DEFINES WHETHER A SURFACE APPEARS TO BE **METAL** OR **NON-METAL**
WHILST PURE SURFACES WILL BE EITHER 0.0 **OR** 1.0, BEAR IN MIND FEW PURE, CLEAN, UNWEATHERED MATERIALS EXIST IN REAL LIFE
WHEN **TEXTURING** A METALLIC MAP. THIS VALUE WILL ALWAYS BE **GREYSCALE** AND IS STORED IN THE **R CHANNEL** OF AN RGB FILE

GREYSCALE

| 0.0 | 0.1 | 0.2 | 0.3 | 0.4 | 0.5 | 0.6 | 0.7 | 0.8 | 0.9 | 1.0 |

ALMOST ALL **NON-METALS** = 0.0   ALMOST ALL **METALS** = 1.0

SMOOTHNESS A
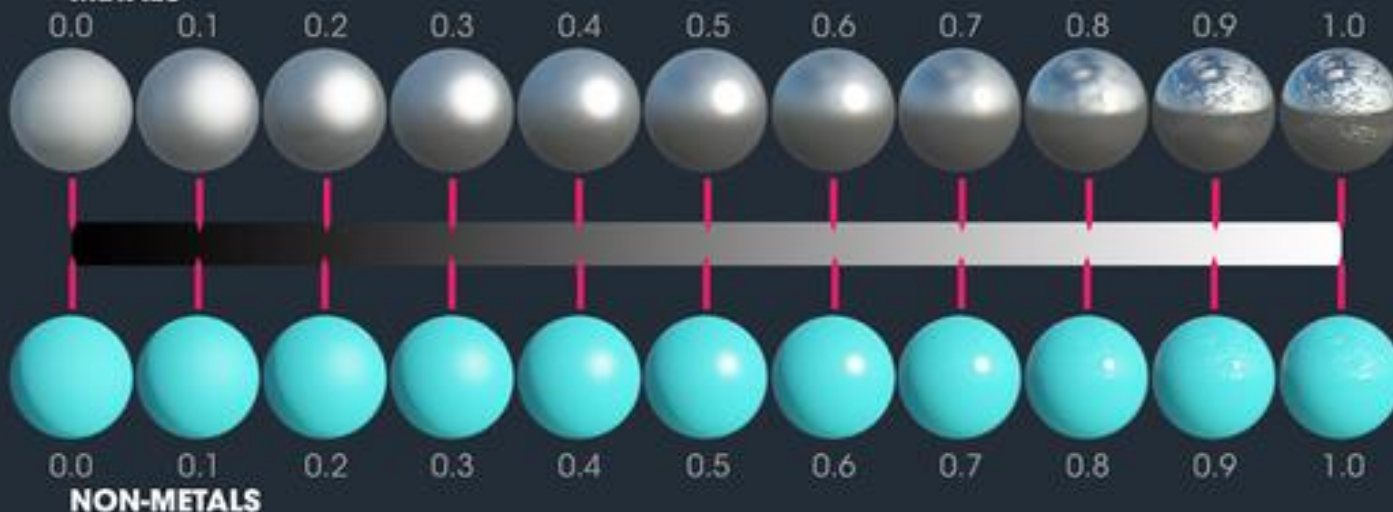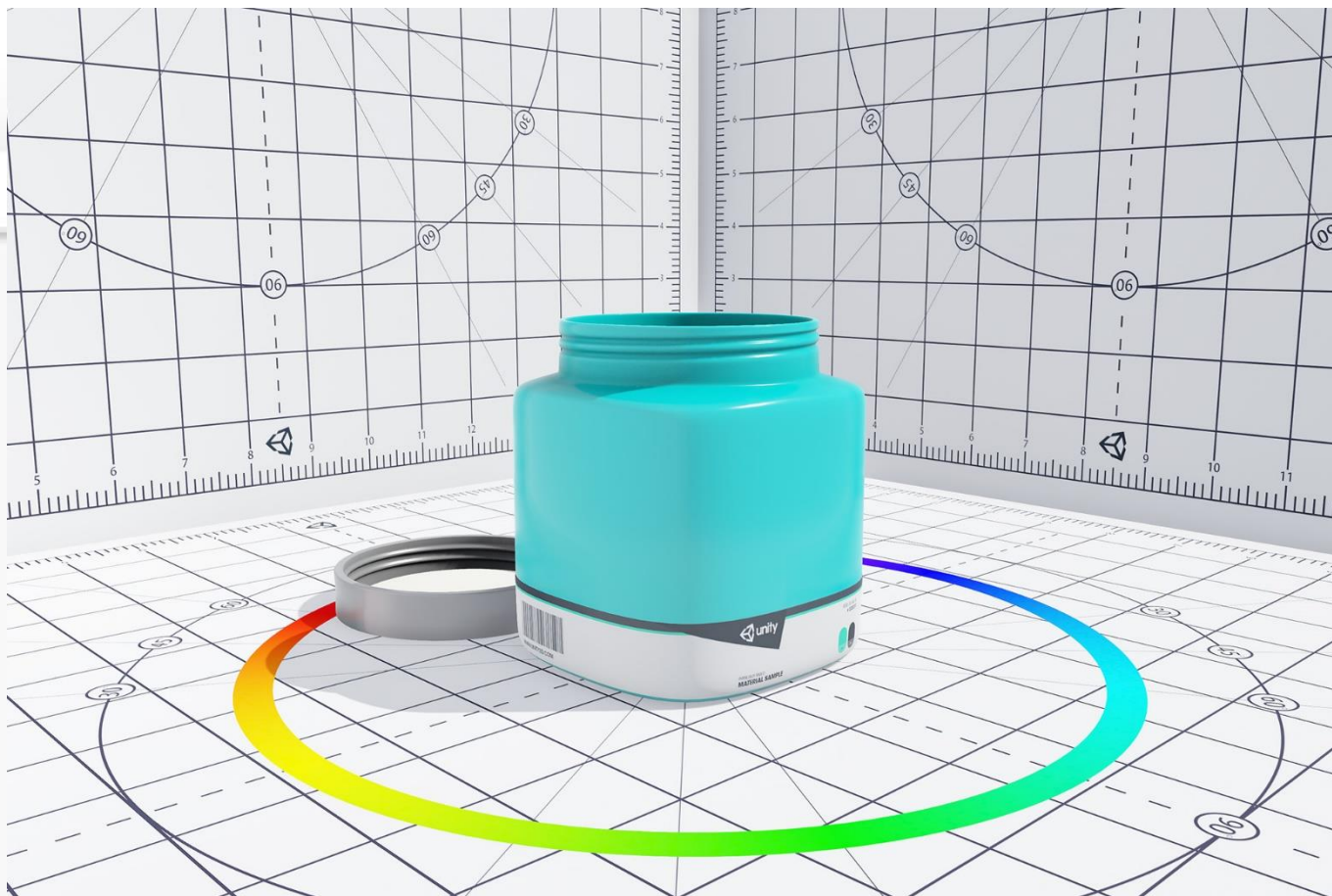
SMOOTHNESS DEFINES THE PERCEIVED **GLOSSINESS** OR **ROUGHNESS** OF A SURFACE
FOR TEXTURES, THIS IS STORED AS THE ALPHA CHANNEL OF THE **METALLIC MAP**

METALS

| 0.0 | 0.1 | 0.2 | 0.3 | 0.4 | 0.5 | 0.6 | 0.7 | 0.8 | 0.9 | 1.0 |

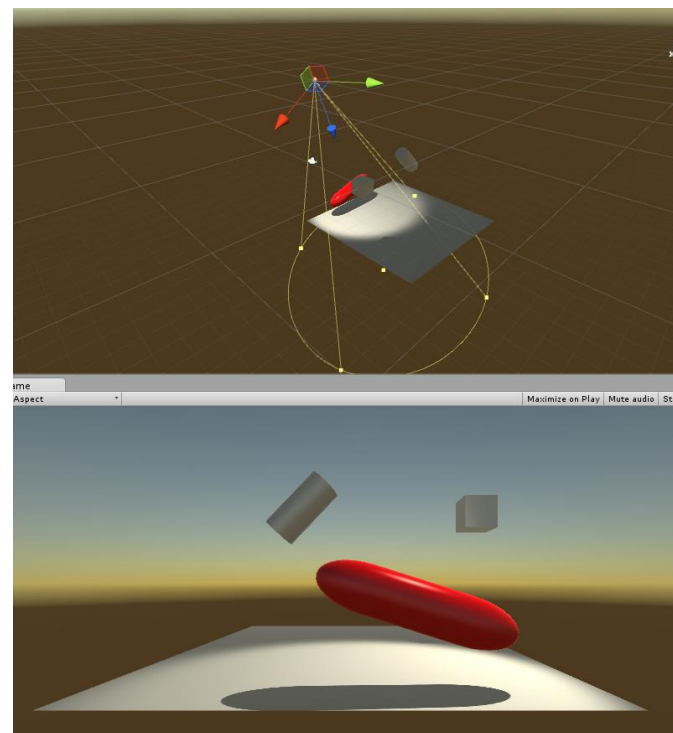| 0.0 | 0.1 | 0.2 | 0.3 | 0.4 | 0.5 | 0.6 | 0.7 | 0.8 | 0.9 | 1.0 |

NON-METALS

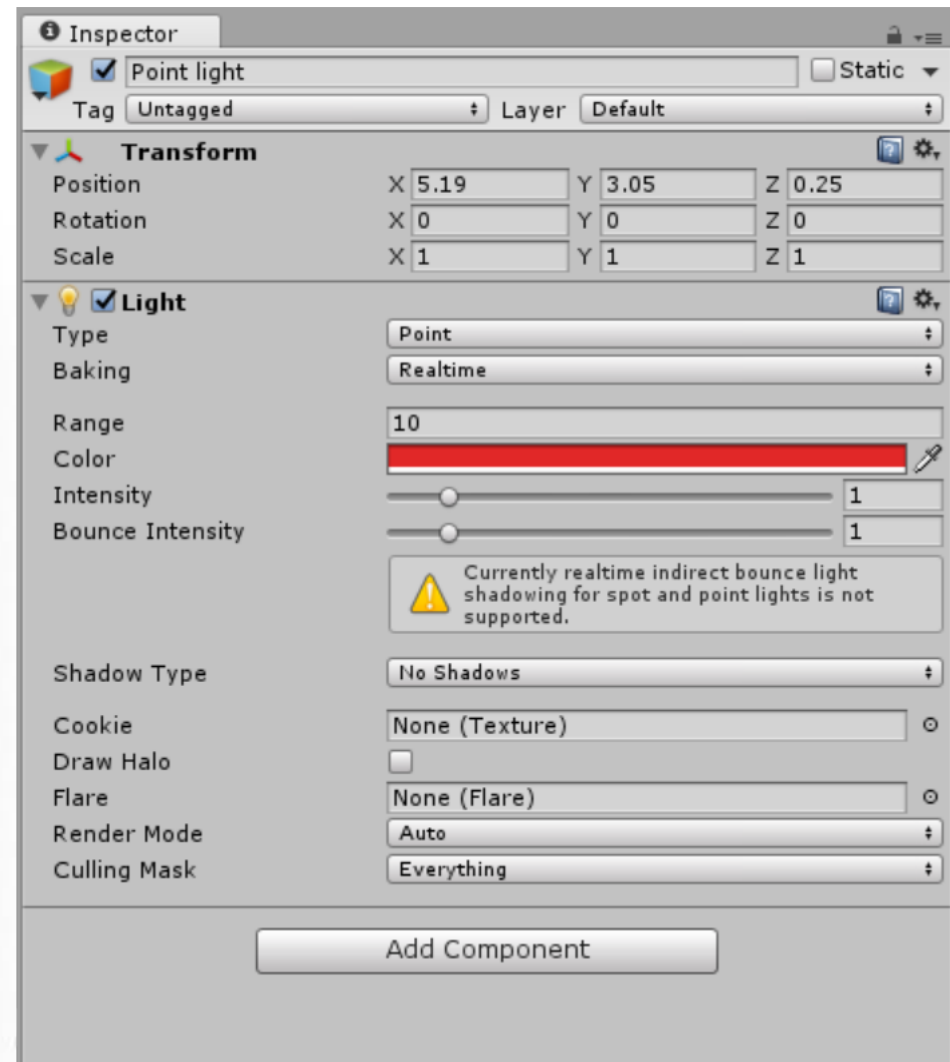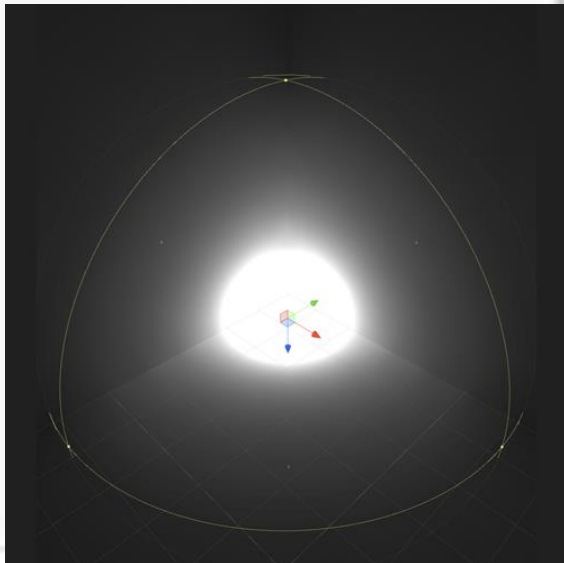2014 UNITY TECHNOLOGIES

Shader Calibration Scene

Lighting a Scene

Lighting in Unity

- Type: 光源種類
- Color: 光源顏色
- Intensity:
  - 光源強度(亮度)
- Shadow Type:
  - 產生的影子類型
- [Point/Spot]Range:
  - 光線可照到的最大範圍(黃圈)
- [Spot]Spot Angle:
  - 聚光燈的展開角度(黃圈)

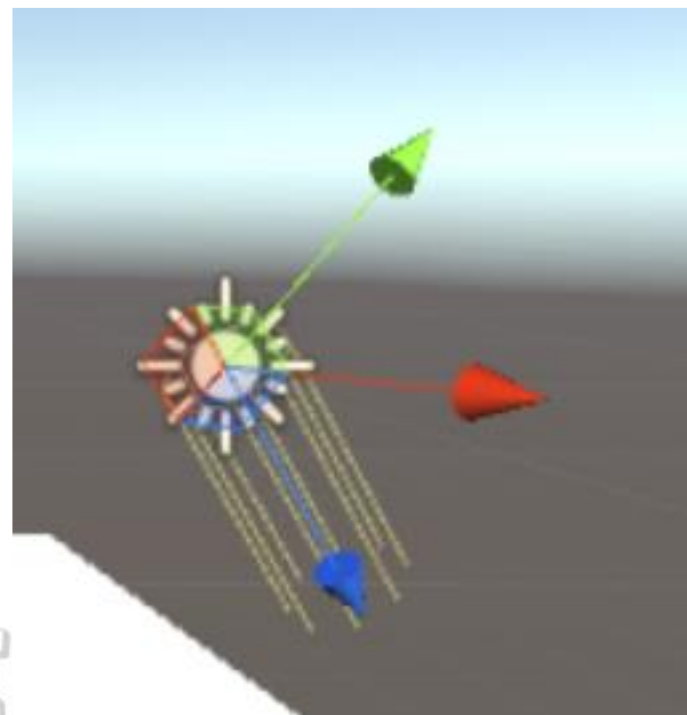# Point light

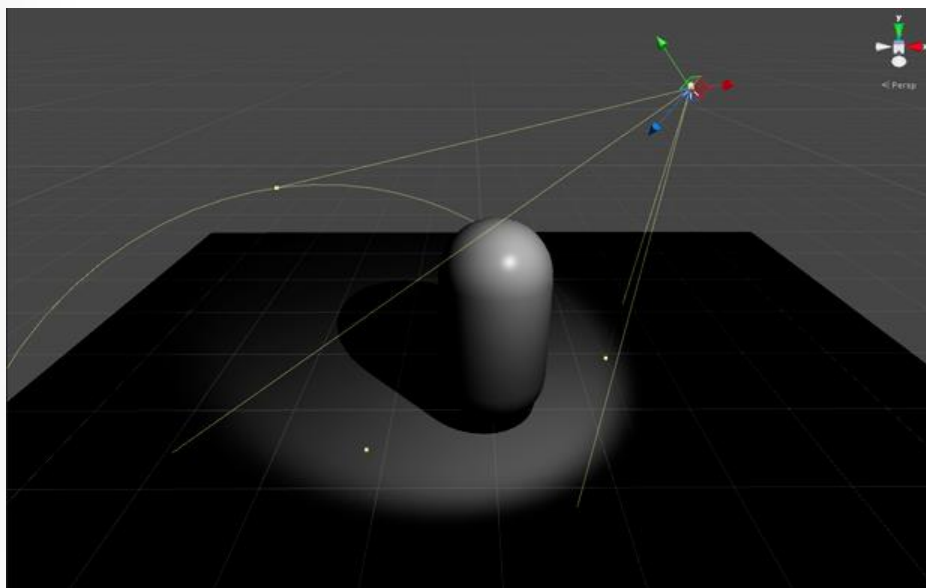- Ex: light bulb
- Position
- Range
- Intensity

Inspector

☑ Point light   ☐ Static ▾

Tag Untagged   ▾   Layer Default   ▾

Transform

Position	X 5.19	Y 3.05	Z 0.25
Rotation	X 0	Y 0	Z 0
Scale	X 1	Y 1	Z 1

☑ **Light**

Type	Point
Baking	Realtime
Range	10
Color	
Intensity	1
Bounce Intensity	1

⚠ Currently realtime indirect bounce light shadowing for spot and point lights is not supported.

Shadow Type	No Shadows
Cookie	None (Texture)
Draw Halo	☐
Flare	None (Flare)
Render Mode	Auto
Culling Mask	Everything

Add Component

Directional Light

– E.g. Sun
– Direction (forward (+Z) direction)

Spot light

Light source
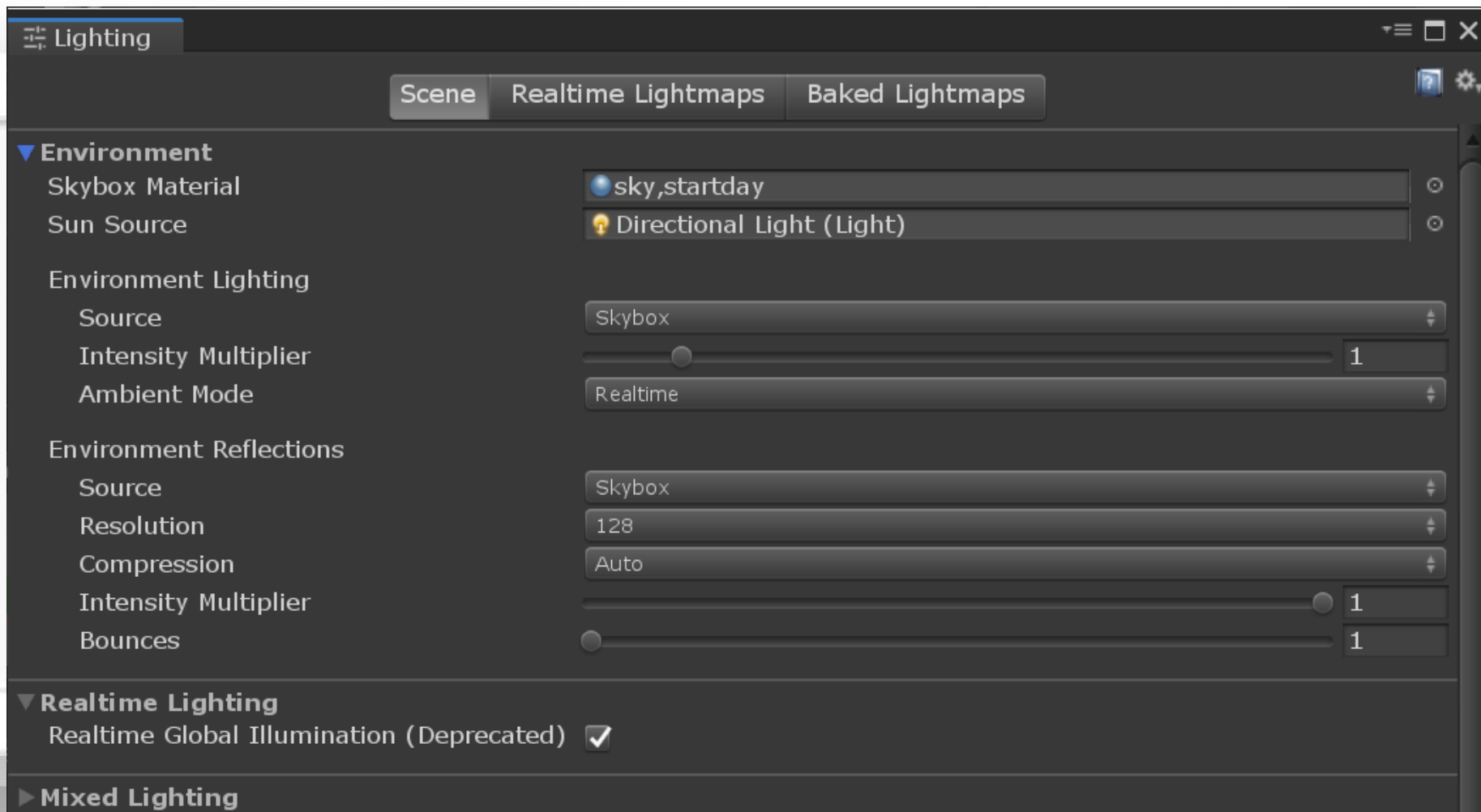
angle

forward (+Z) direction
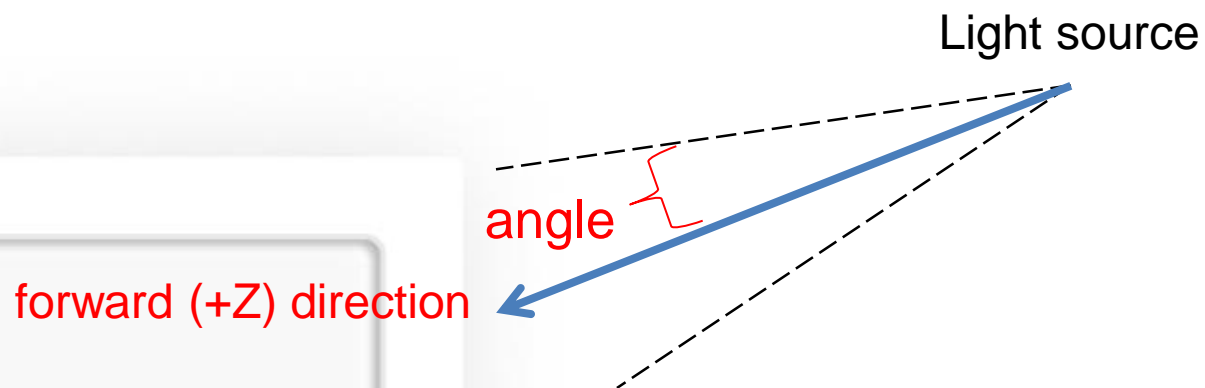
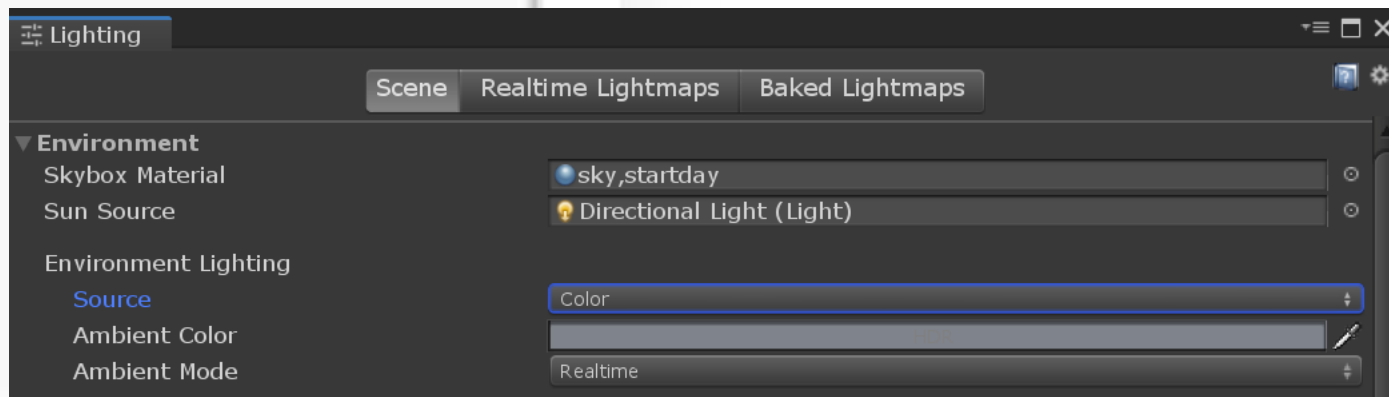# Emissive Materials
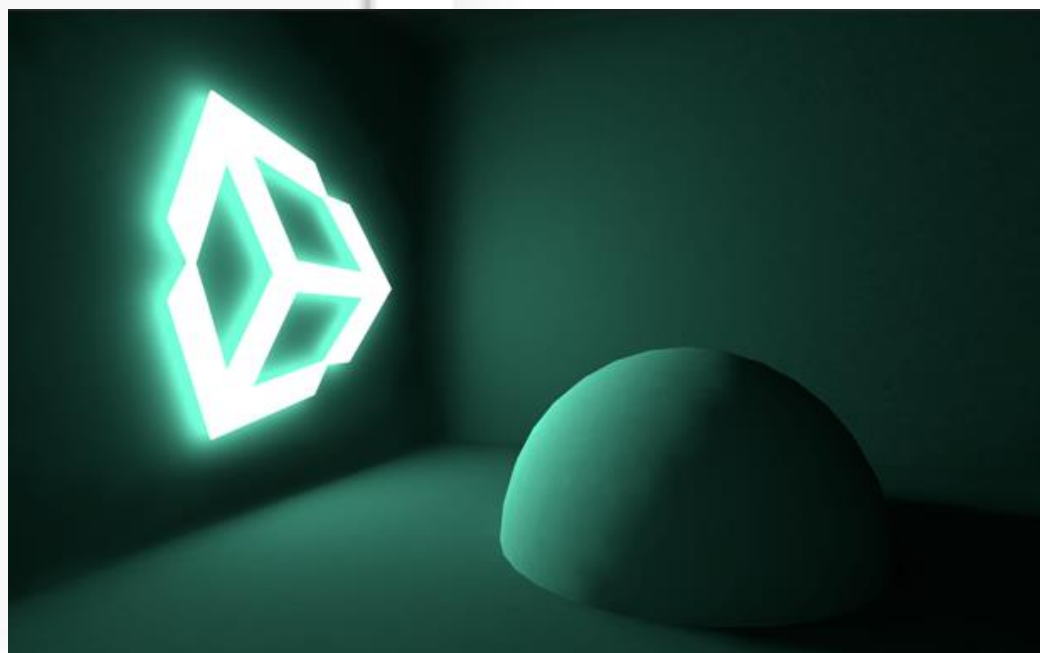
Lighting Windows

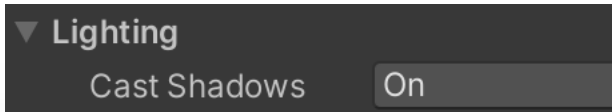📎 menu: **Window** > **Rendering** > **Lighting Settings**

Ambient light

📎 Ambient light is light that is present all around the scene and doesn't come from any specific source object
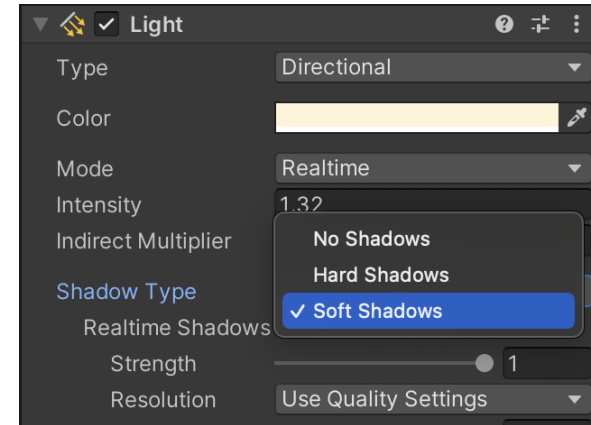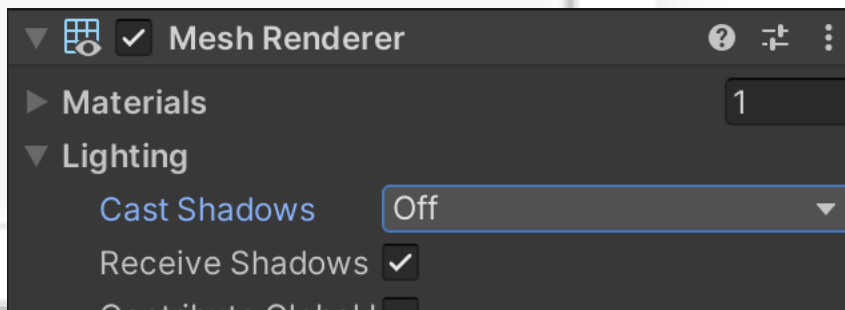
📎 Set in Lighting window

shadow

Lighting

Cast Shadows   On

Cast shadows

Light

Type	Directional
Color	
Mode	Realtime
Intensity	1.32
Indirect Multiplier	

No Shadows
Hard Shadows
✓ Soft Shadows

Shadow Type
   Realtime Shadows
    Strength   1
    Resolution   Use Quality Settings

Receiveve shadows

Mesh Renderer

Materials	1

Lighting

Cast Shadows   Off

Receive Shadows ✓

Render Pipeline

```
                              ┌──────────────────────────┐
                              │     Build-In Render      │
                              └──────────────────────────┘
┌──────────────────────────┐
│  Unity Render Pipline    │
└──────────────────────────┘
                              ┌──────────────────────────┐   High Definition Render Pipeline
                              │ Scriptable Render Pipline │
                              └──────────────────────────┘   Universal Render Pipeline（URP前身为
                                                             Lightweight Render Pipeline）
```
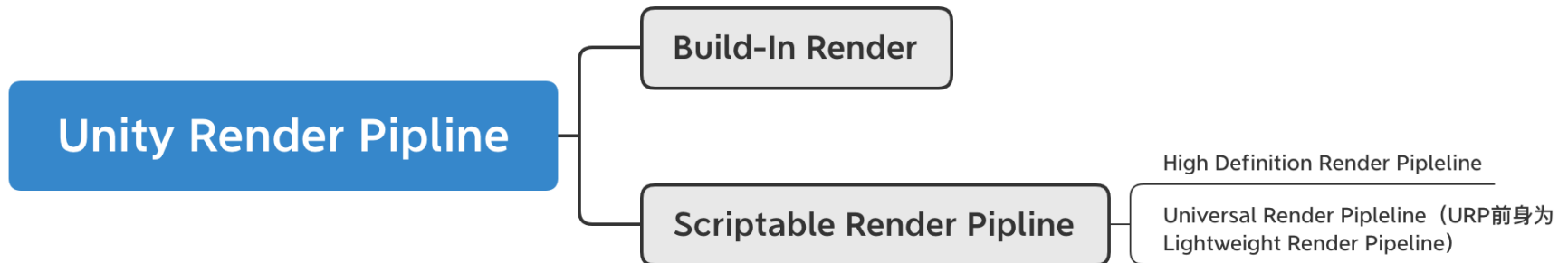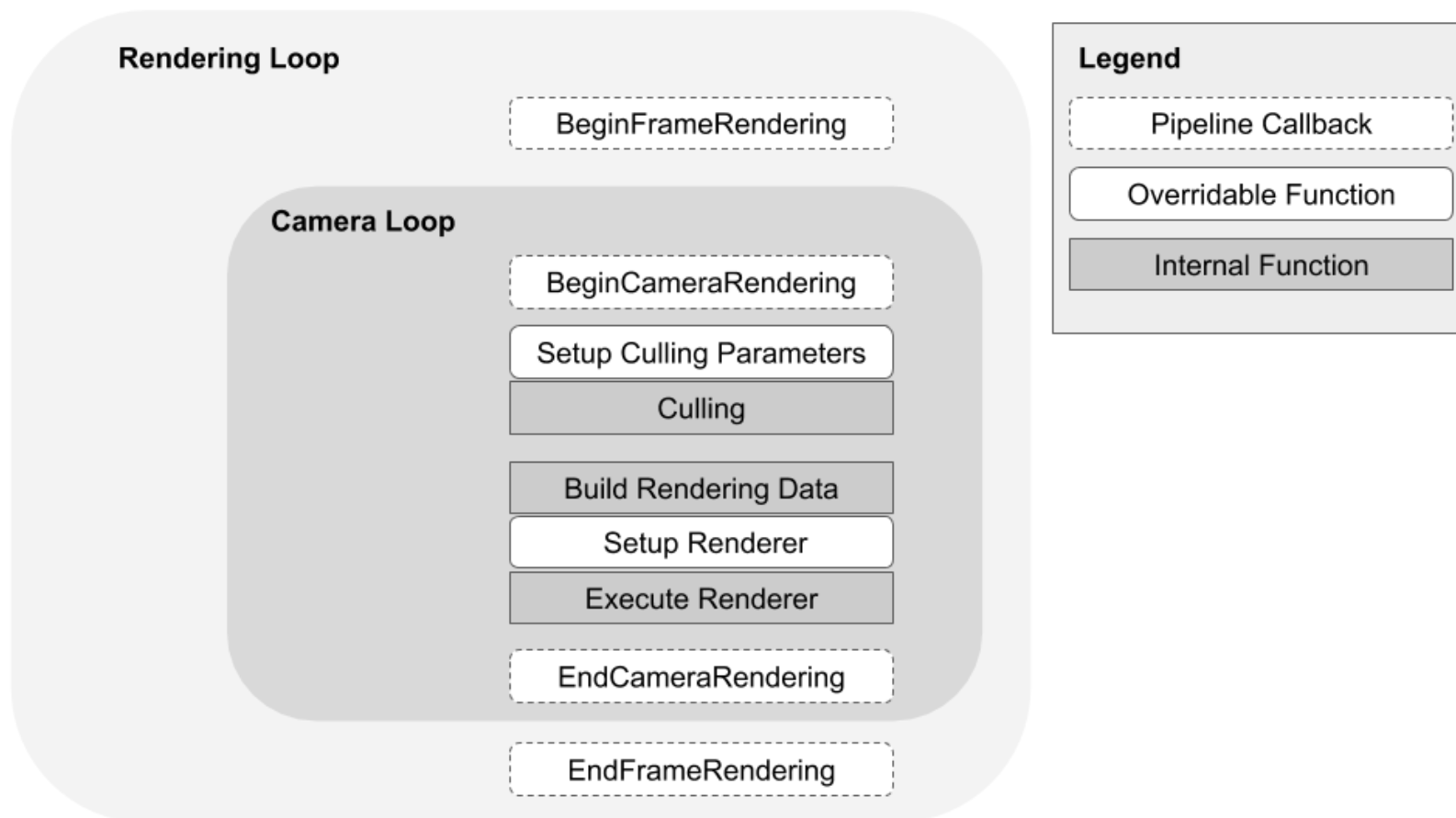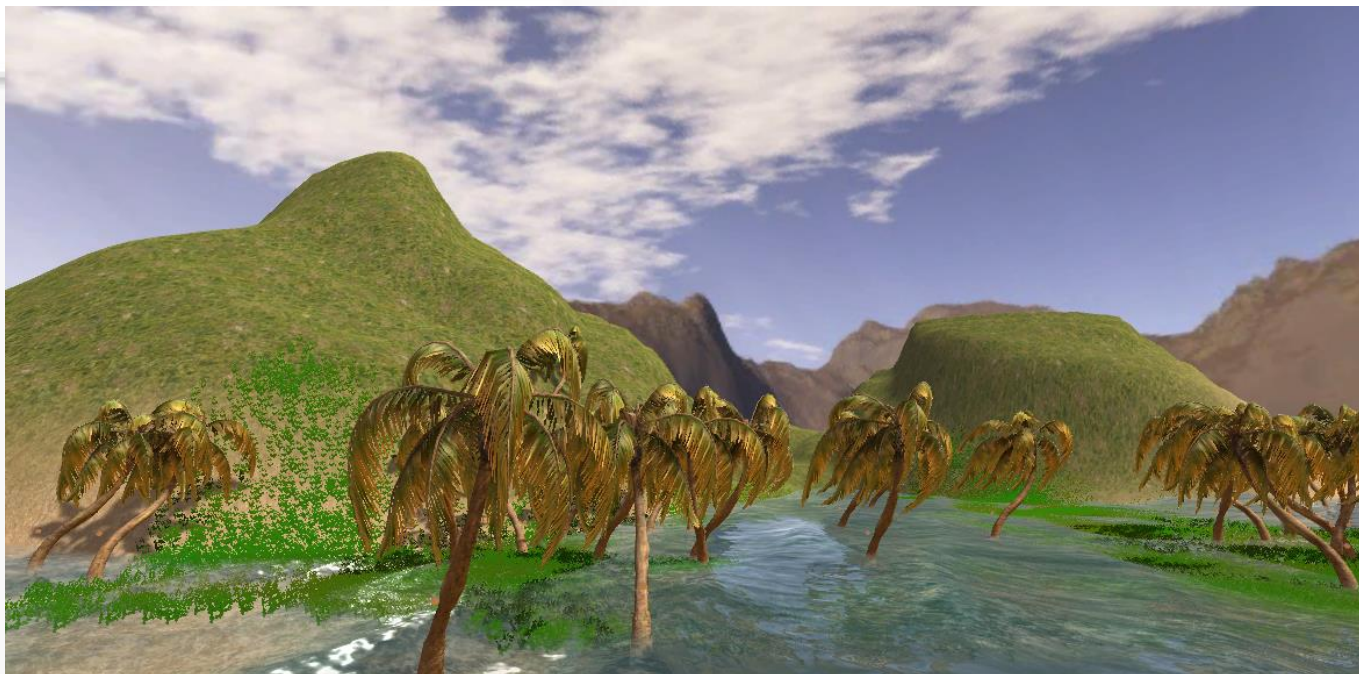
URP pipeline

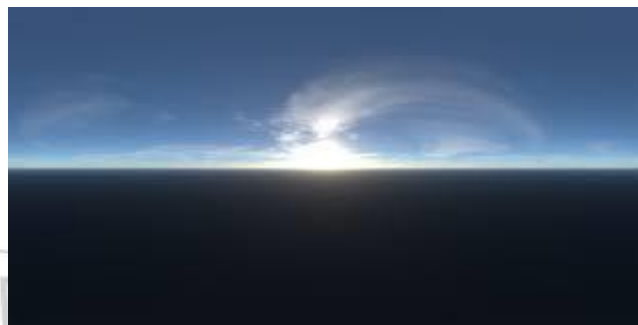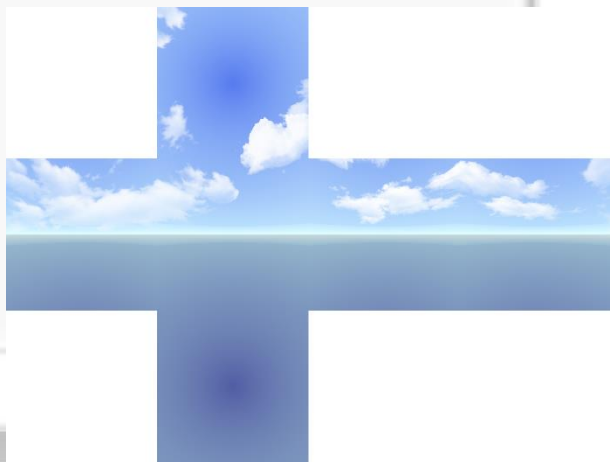The following illustration shows the frame rendering loop of the URP Universal Renderer.
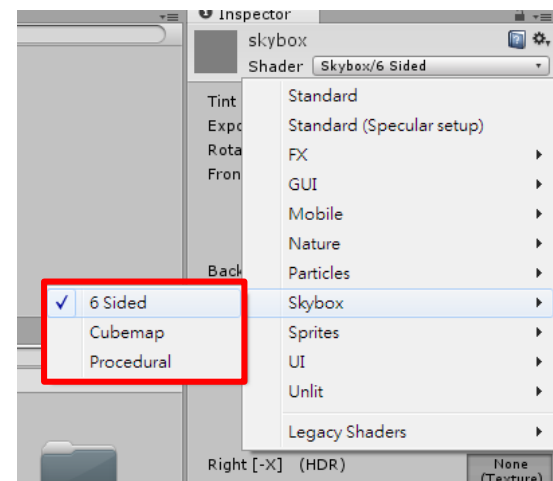
Skybox

INTERACTIVE
MEDIA

Skybox

- 一個包覆場景的巨大靜態圖片
- 沒有打光（沒有光源仍然看的到）
- 用靜態圖片展示遠景
- Cube使用6張圖片，分別是上、下、左、右、前、後

Skybox

📎 創立Skybox建立一個material

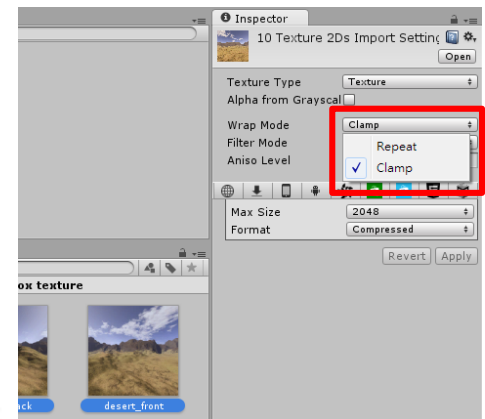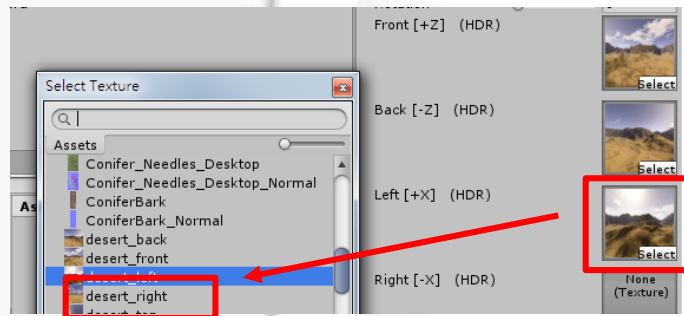- Shader → Skybox → 6 Sided

Skybox

- 將天空圖片加進Unity專案內
- 圖片的Warp Mode要改成Clamp
  - 避免天空盒在邊界時會產生顏色不match
- 設定Skybox material六面的圖片

Skybox

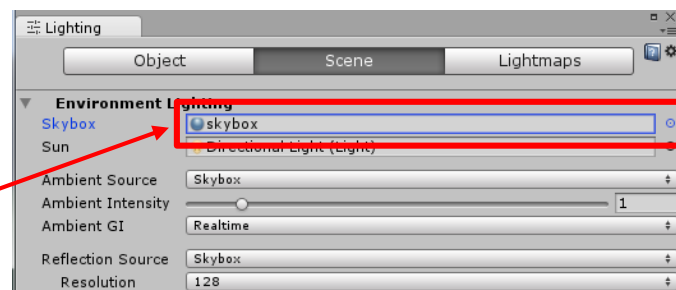## 設定Skybox
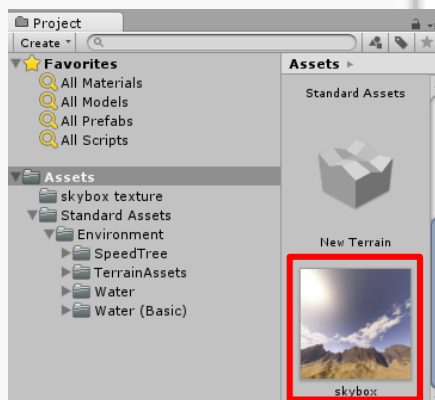
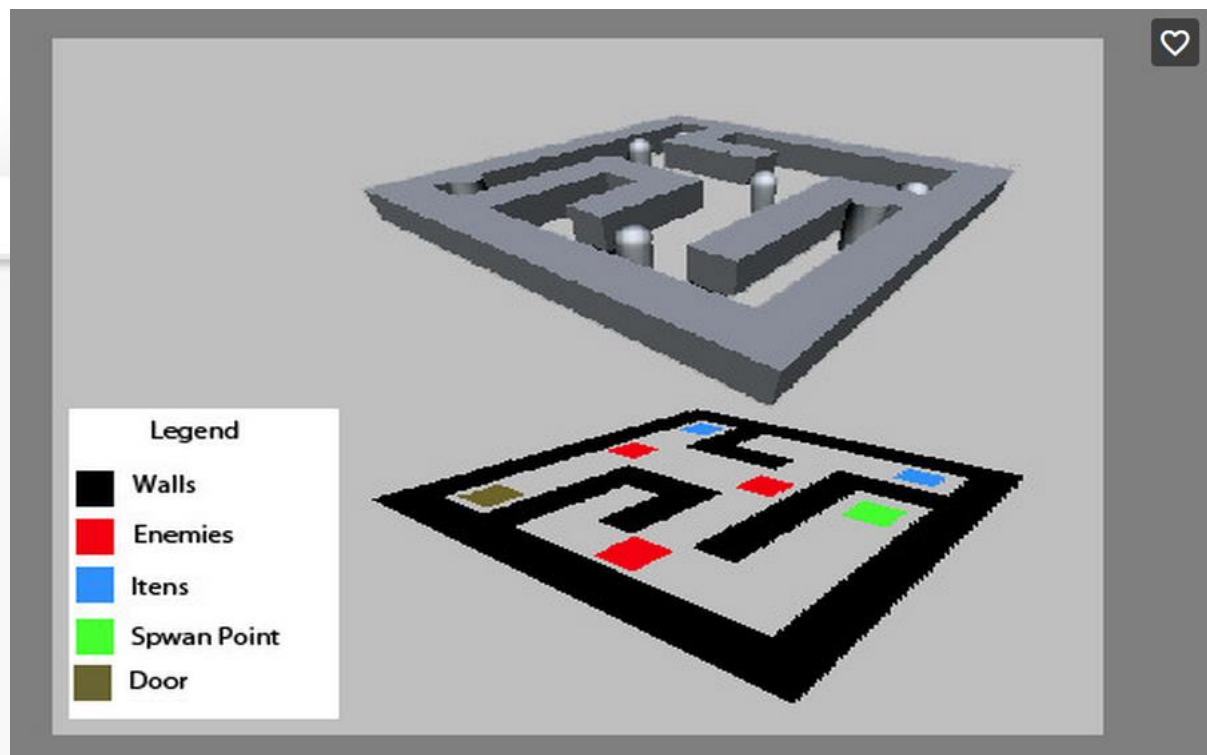- 功能表列 Window → Lighting
- 將剛剛建好的material套用進Skybox

iNTERACTiVE
MEDiA

Level Design

Legend
- ■ Walls
- ■ Enemies
- ■ Itens
- ■ Spwan Point
- ■ Door

GUILHERME ALBARRANS LEITE

Create Level Tool

★★★★☆ ▾   3 user reviews

Popular Tags
Add a new tag right now?

Add tags

The Level Generator Tool is a simple tool
easily and quickly, from an image source