



COCOS CREATOR

3D ASSETS WORKFLOW

GETTING STARTED GUIDE

MODEL MATERIAL TEXTURE LIGHTING

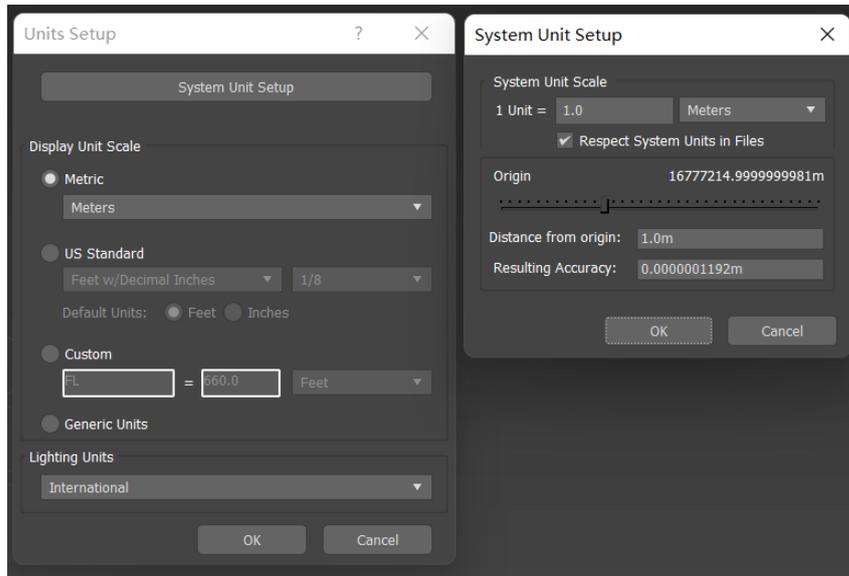
CONTENTS

Cocos Creator 3D assets workflow Getting Started Guide	2
Chapter 1: Exporting Models from 3ds Max	2
Chapter 2: Making textures	4
Chapter 3: Creating 3D Assets with Cocos.....	5
1. Create a new Cocos project on Cocos Dashboard	5
2. Create project	6
3. Import model assets to Cocos	7
4. Create a scene	8
5. Import model assets into the scene	8
6. Making PBR Material textures	10
7. Setting up the environment and shadows.....	17
8. Lighting	20
9. Set the shadow and bake properties of the models	25
10. Baking lightmaps.....	27

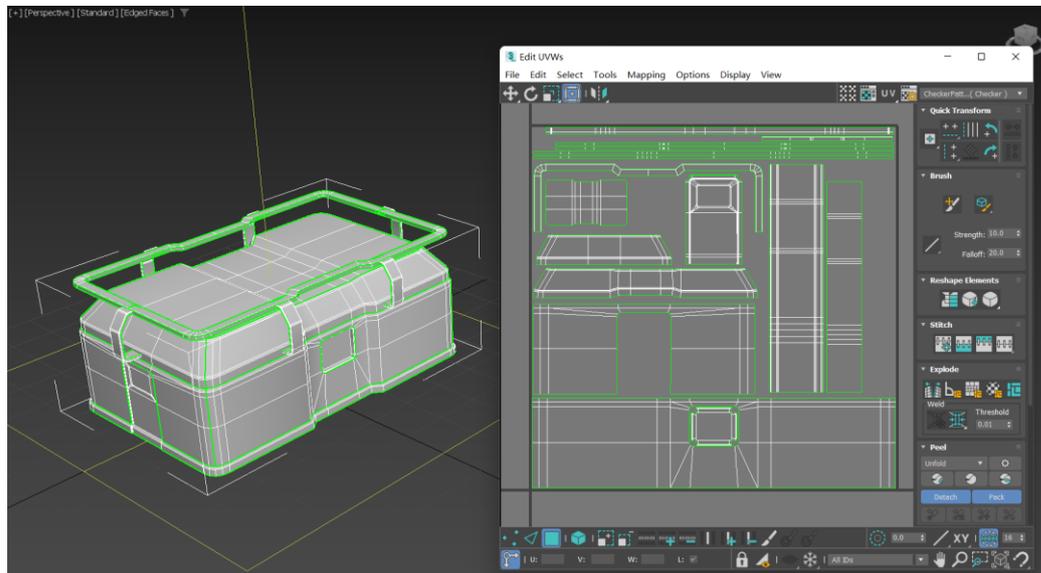
Cocos Creator 3D assets workflow Getting Started Guide

Chapter 1: Exporting Models from 3ds Max

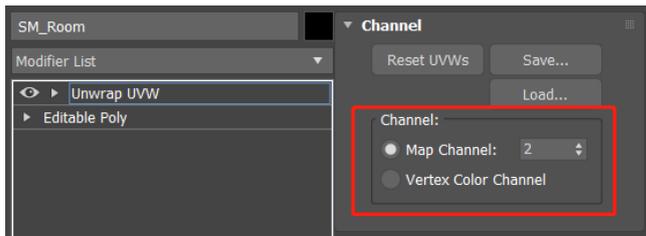
01. 3ds Max unit setup: Set the 3ds Max units to Meters. (Cocos default unit is Meters)



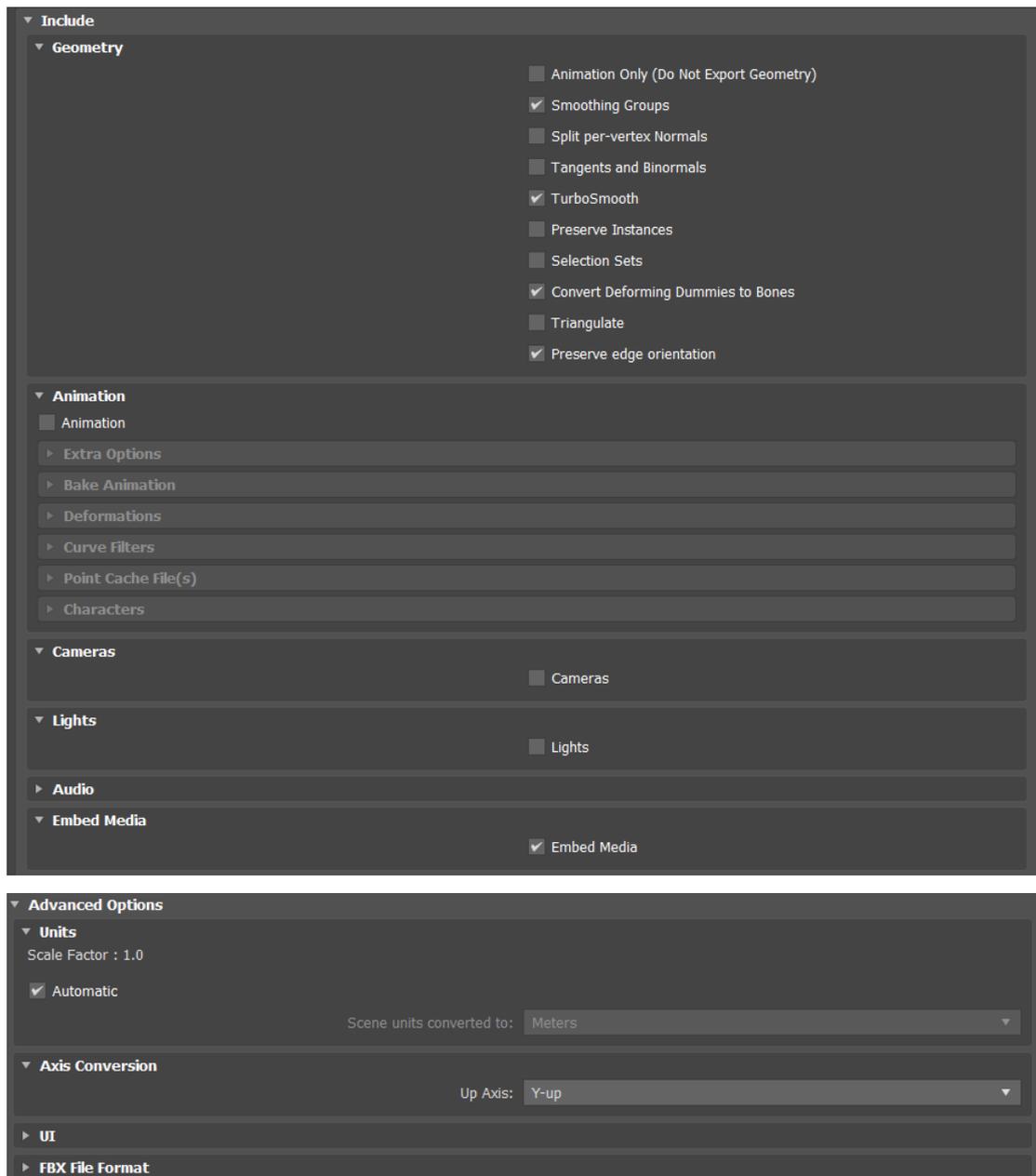
02. The model UV needs to be flattened. Maximize the use of UV space.



03. Make the second UV for light baking. Second UV cannot overlap.



04. 3ds Max export FBX model settings.



1. Enable Smoothing Groups.
2. If animation is included, enable Animation.
3. Cameras and lights are not enabled.

4. Enable Embedded Media.
5. The unit is selected: Automatic, Meters.
6. Axis Conversion: Y-up.

Chapter 2: Making textures

Cocos engine supports PBR materials and textures. Materials created manually in the **Assets** panel use the builtin-`standard.effect` shader by default, which we call the PBR material, which uses the Metal/Roughness workflow of the PBR process.

When rendering with PBR materials, at least these textures are required for proper rendering:

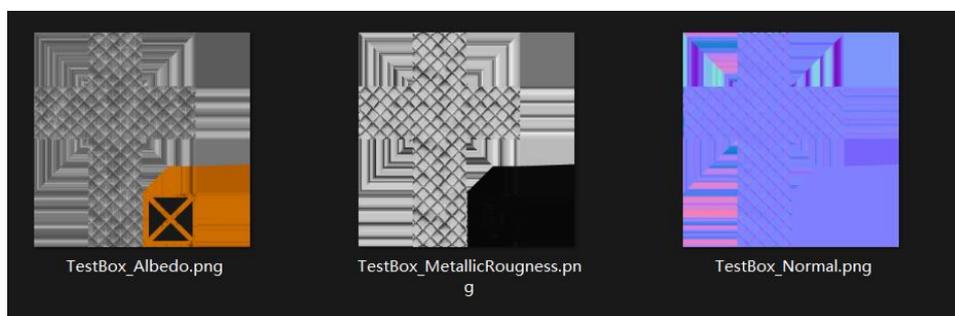
Albedo

Metallic Roughness

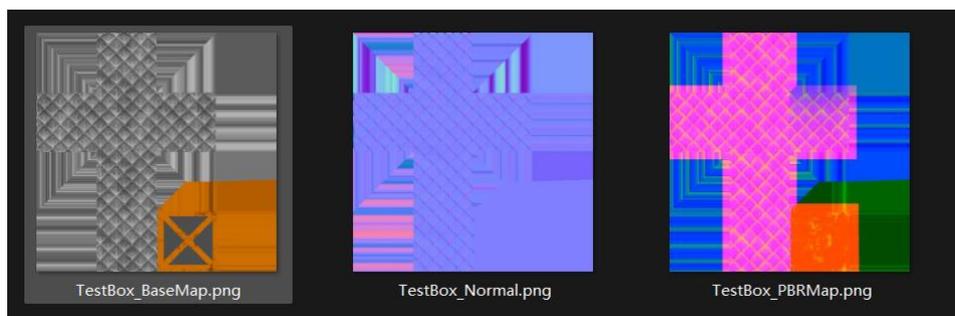
Normal

These textures can be made using Photoshop, Substance, or other image editing software. These textures can be set in the Material Resource Inspector panel. For details on PBR materials and textures, please refer to the documentation:

<https://docs.cocos.com/creator/manual/en/shader/effect-builtin-pbr.html>



PBR textures supported by Cocos can also be used. (recommended to use this texturing method)



A PBR texture is a hybrid texture.

Each of the 3 channels in the PBR texture contains a texture.

R (AO) R channel: Ambient light masking
G (Roughness) G channel: Roughness
B (Metallic) B channel: Metallicity

Cocos currently supports texture formats: JPG, PNG, BMP, TGA, and HDR formats.
Texture size: it is recommended not to be larger than 2048 X 2048.

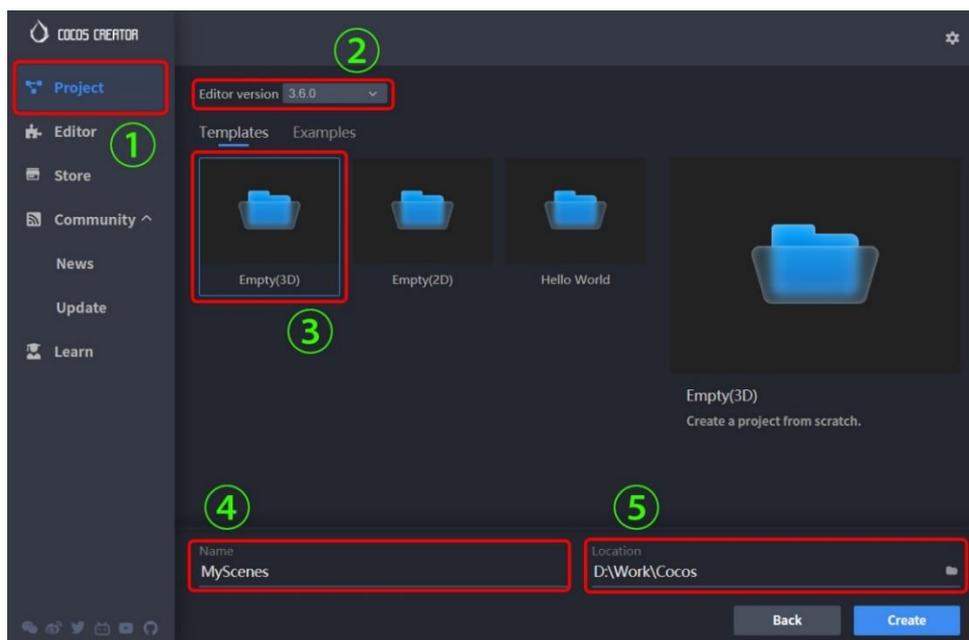
Chapter 3: Creating 3D Assets with Cocos

1. Create a new Cocos project on Cocos Dashboard

01. On the official website www.cocos.com/en/creator, download and install Cocos Dashboard. Cocos uses Cocos Dashboard to manage Cocos program installation and create projects.

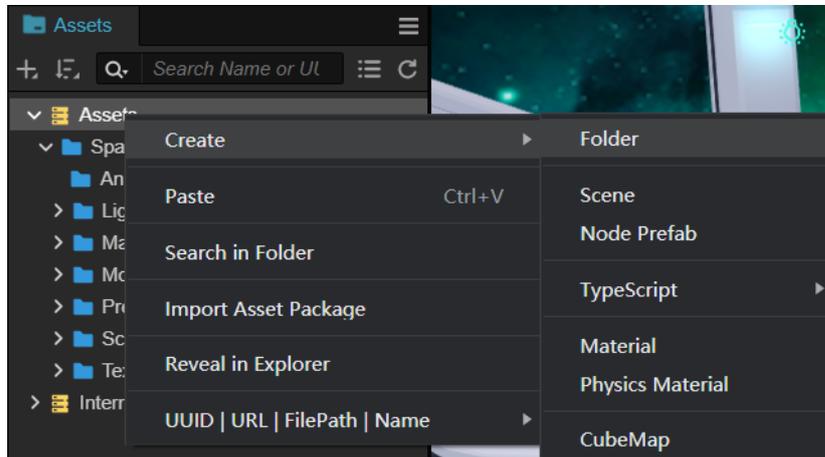
02. Create a new Cocos project in Cocos Dashboard.

1. Click the Project panel.
2. Select the Cocos version.
3. Select the Empty (3D) project template.
4. Enter a project name.
5. Select the location where the project is saved on your hard disk.



2. Create a project

01. Right-click create -> folder in assets. Create a new project folder and name the folder.



02. Create some resource folders to store model assets and scene files.

Animation: Store Animation files

Lightmap: Store baked lightmaps.

Materials: Store the materials.

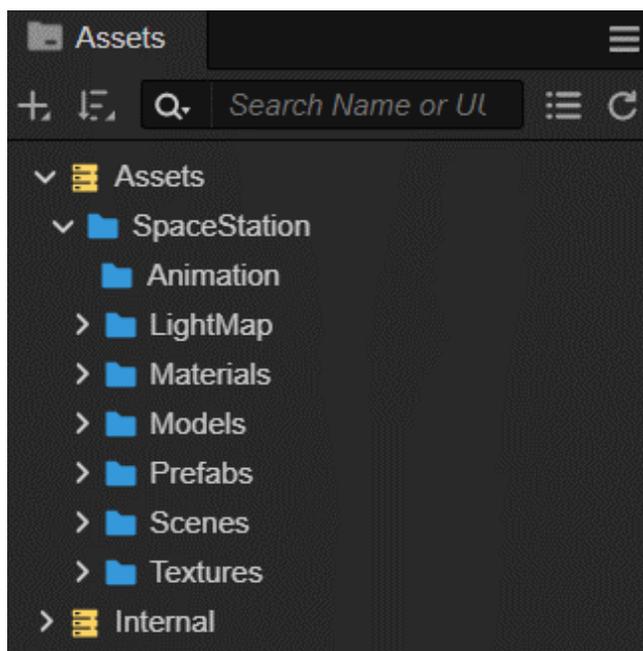
Models: Store Models.

Particles: Store particles.

Prefabs: Store Prefabs.

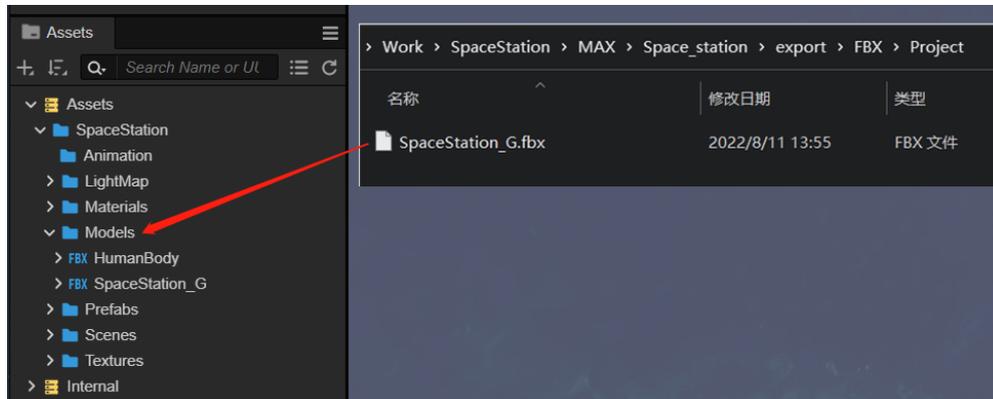
Scenes: Store Scenes.

Textures: Store Textures.

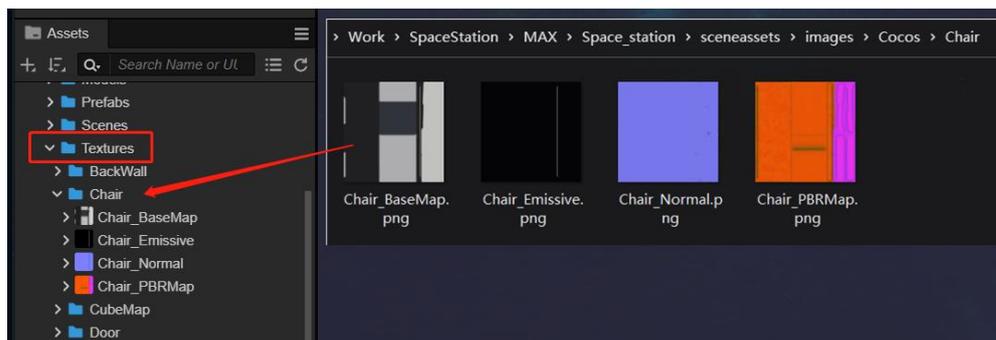


3. Import model assets to Cocos

01. Drag the completed model file into the Models folder under Project. Cocos will automatically load all assets.



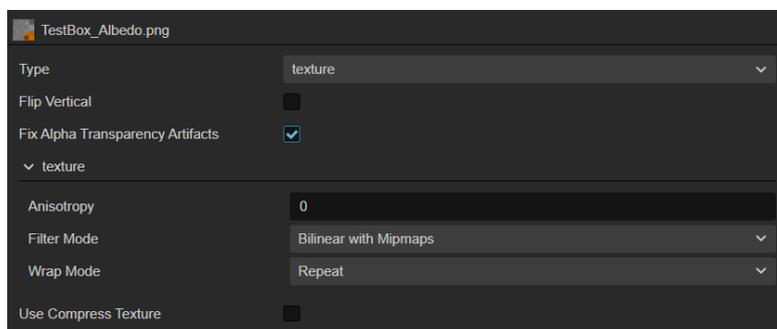
02. Drag the finished texture file into the Textures in the project folder.



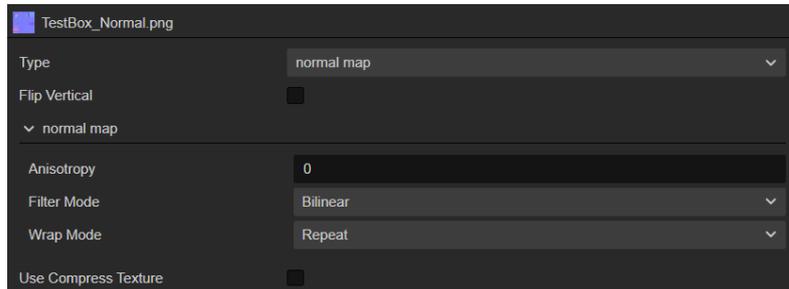
03. After the texture is imported into Cocos, set the type of the texture.

Albedo, Metallic Roughness textures. Type: texture.

Filter Mode: Bilinear with Mipmaps.

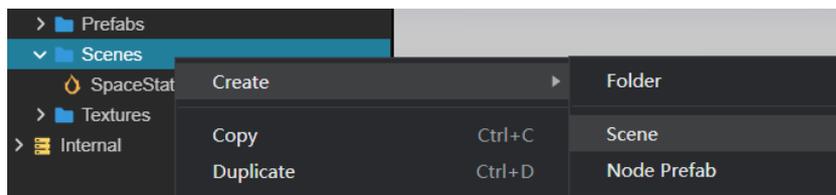


Normal texture, Type: normal map.
Filter Mode: Bilinear with Mipmaps.

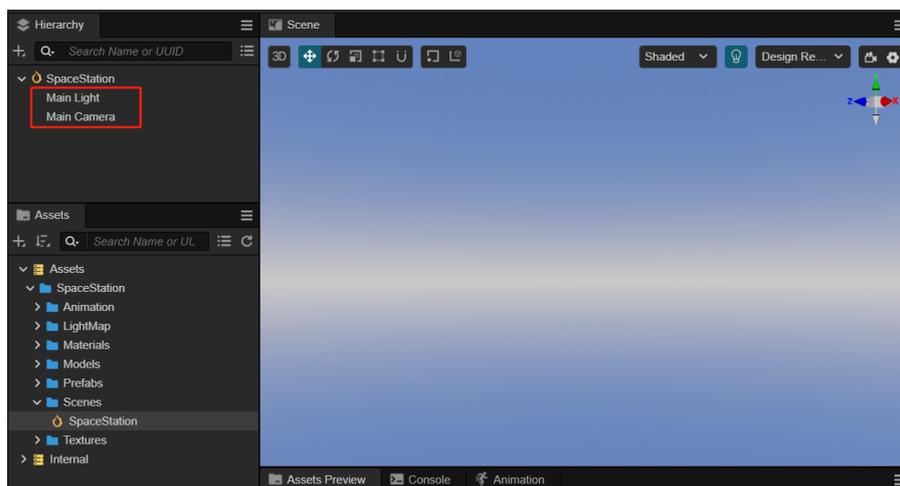


4. Create a scene

Right-click on the Scene folder to create a scene, name it 'SpaceStation', then double-click to open the 'SpaceStation' scene.



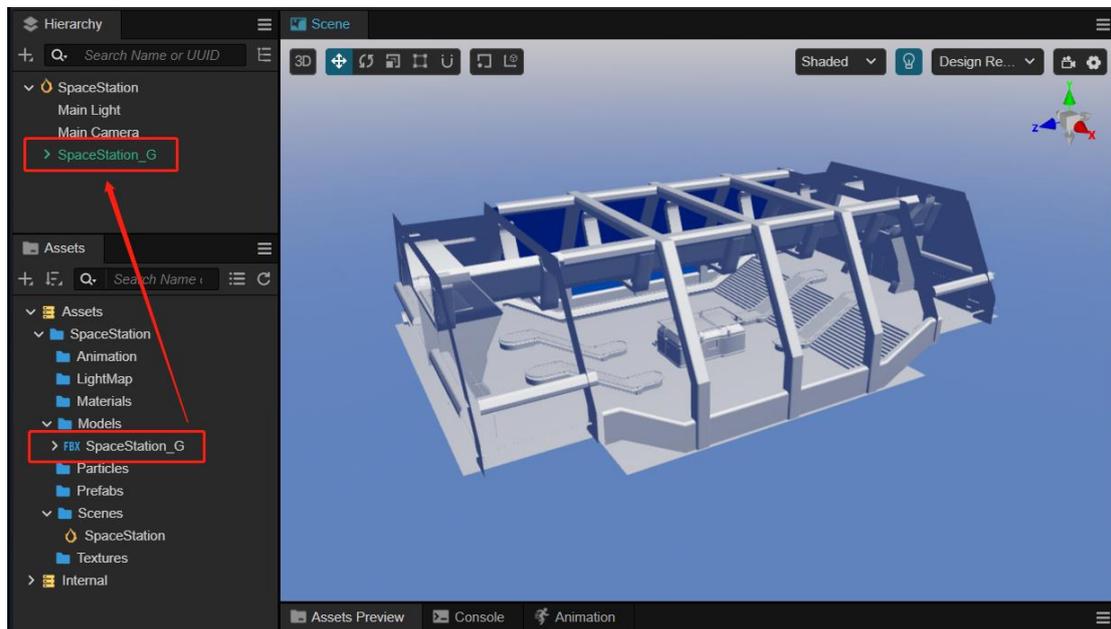
The scene is currently using the default settings. There is only one main light and main camera and the default skybox in the scene.



5. Import model assets into the scene

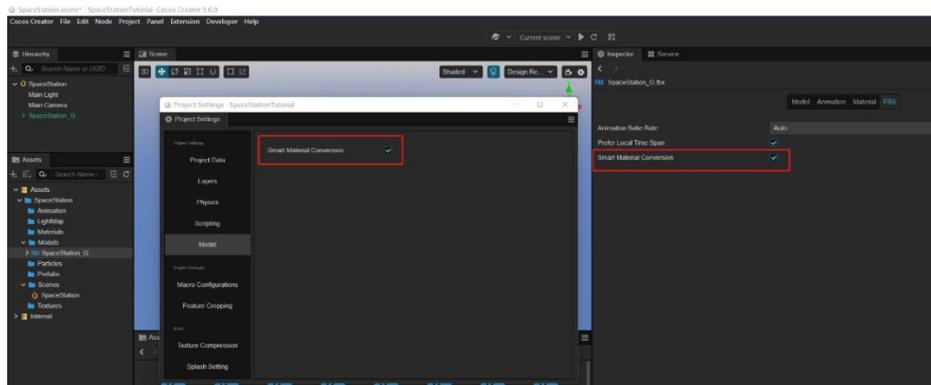
01. Import FBX model files without textures.

Drag the SpaceStation_G.FBX model file into the Hierarchy. Model assets are loaded into the scene.



02. Use the FBX Smart Material to import a textured FBX model file.

FBX Smart Material Conversion is a feature of the model importer. It can map some standard materials in DCC(Digital Content Creation) tools to Cocos Creator's built-in materials and restore the visual effects of materials in DCC tools.



FBX Smart Material supports the following DCC tools and materials.

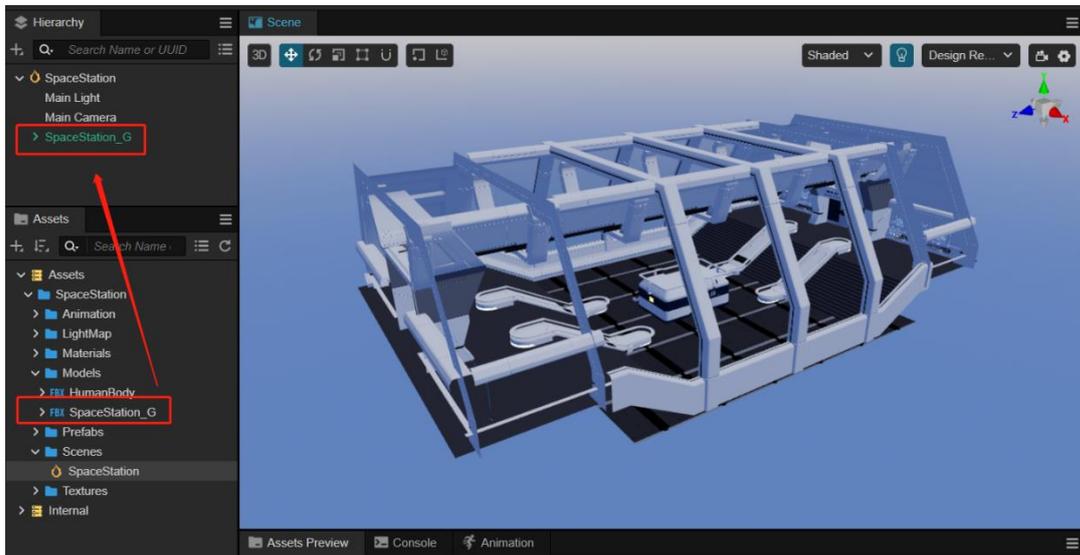
Software	Phong	PBR
3ds Max	Standard(legacy)	Physical Material
Blender	N/A	Principled BSDF

C4D	Standard	N/A
Maya	Lambert/Blinn/Phong	Standard Surface

Please refer to the documentation for more information on the FBX Smart Material Guide.

<https://docs.cocos.com/creator/manual/en/importer/materials/fbx-materials.html>

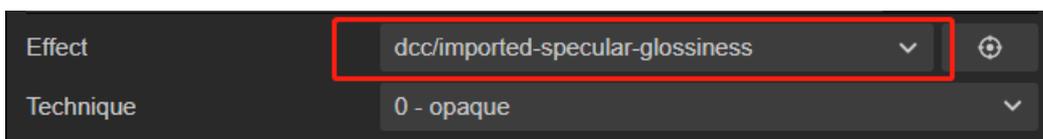
After importing FBX models and textures into Cocos, drag the SpaceStation_G, FBX model file into the Hierarchy. Model and texture assets are loaded into the scene



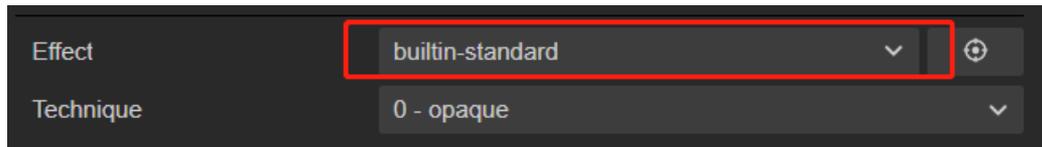
6. Making PBR Material textures

01. Making PBR Materials

In the default imported FBX model, the built-in material Effect is: dcc/imported-specular-glossiness.



We're going to change to the usual format: builtin-standard.



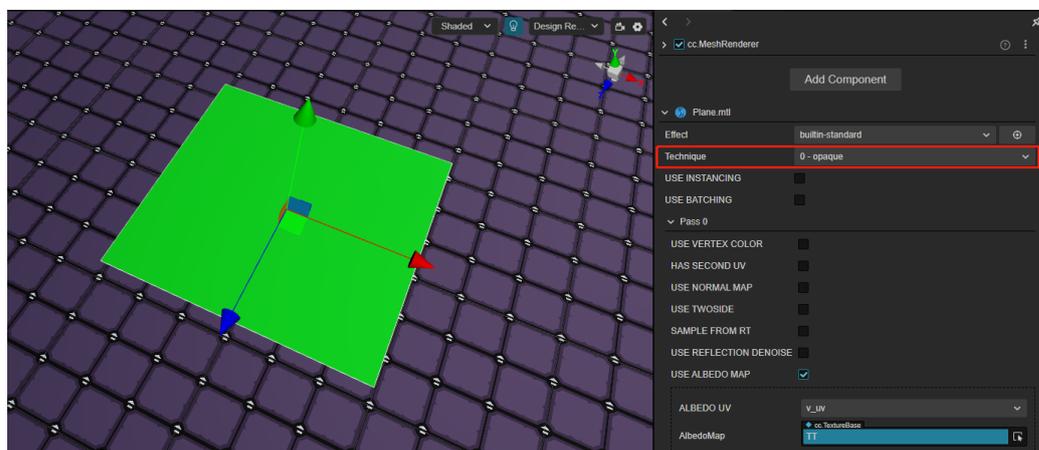
Material selection for opaque objects: 0-opaque.



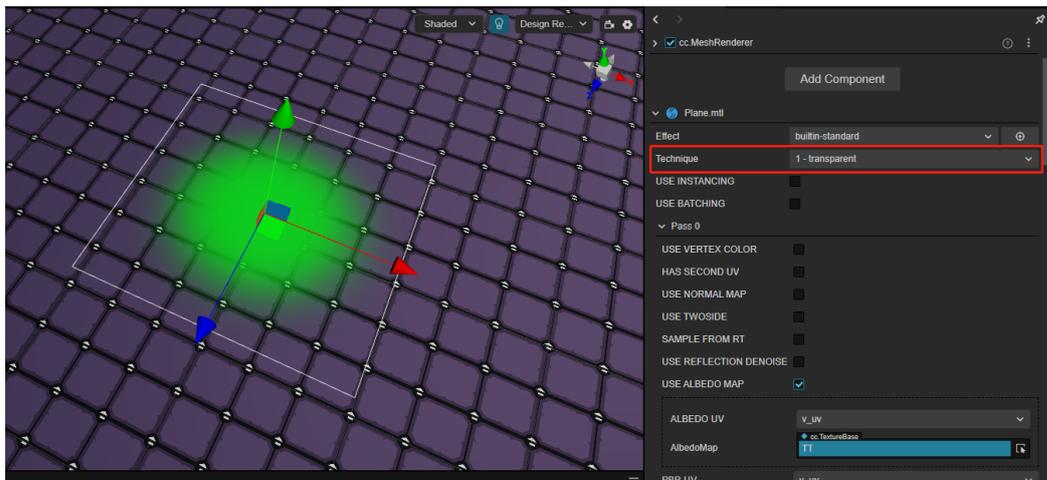
Material selection for transparent objects: 1-transparent.



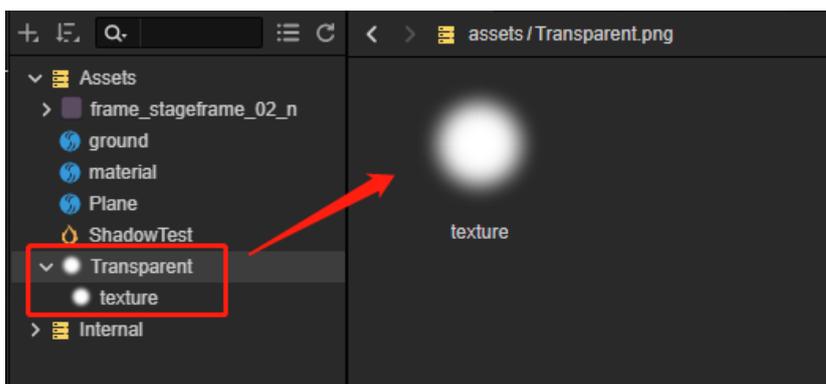
If you want to use a transparent material, Albedo texture needs to contain an alpha channel
Material of opaque objects: 0-opaque.



Material of transparent object: 1-transparent.

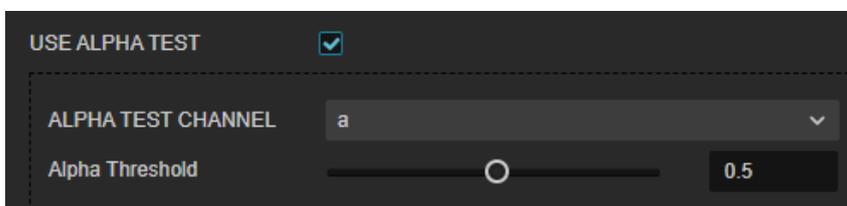


Albedo contains an alpha transparency channel.



Another way to make a transparent material:

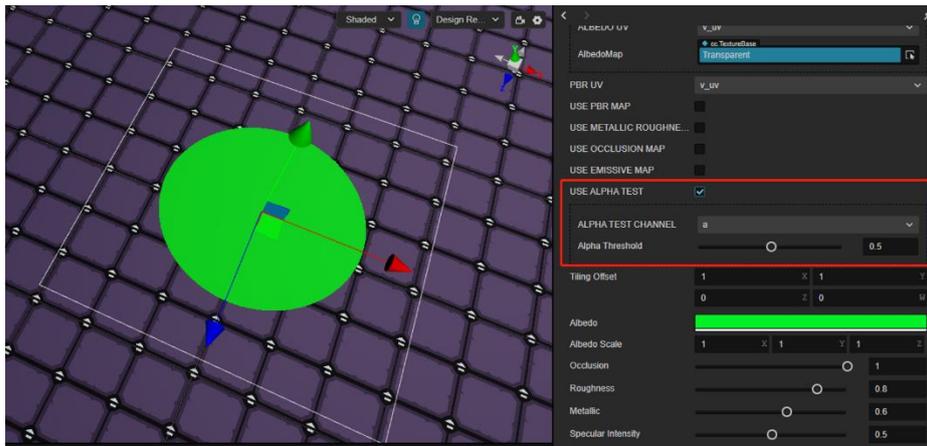
USE ALPHA TEST.



Select the channel (Alpha channel or Red channel) where the template mask (Stencil) is located in the **ALPHA TEST CHANNEL** parameter. Adjusting the threshold for discarding the meta-brightness of the slice via the **Alpha Threshold** property.

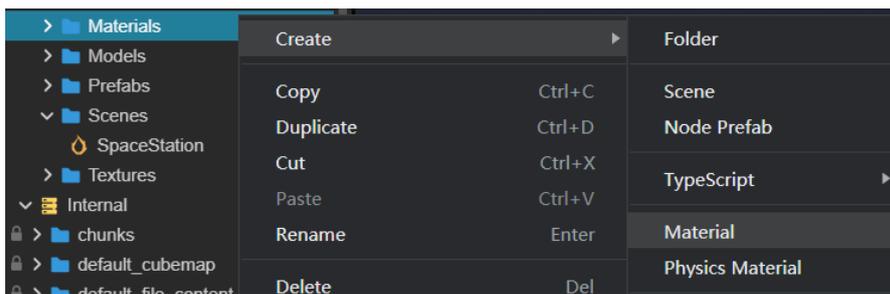
USE ALPHA TEST will ignore the gradient of the transparent map, and the edges will appear as hard edges. Therefore, choose different transparent methods according to the

needs of the project.



02. Create a new material

Right-click in the materials folder Create – Material, Create a new material and name it.



03. Add textures to materials.

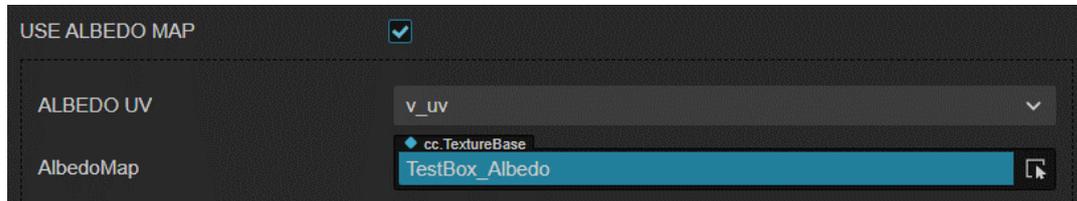


There are two ways to add PBR textures.

1. Use separate textures

Import the created Albedo, Metallic Roughness, and Normal textures into the specified position of the materials.

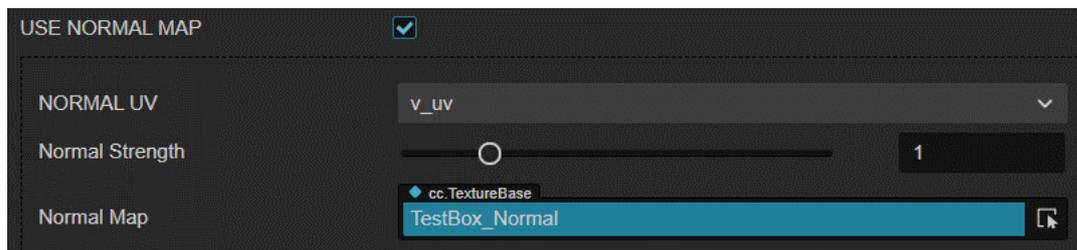
Albedo textures:



Metallic Roughness textures:

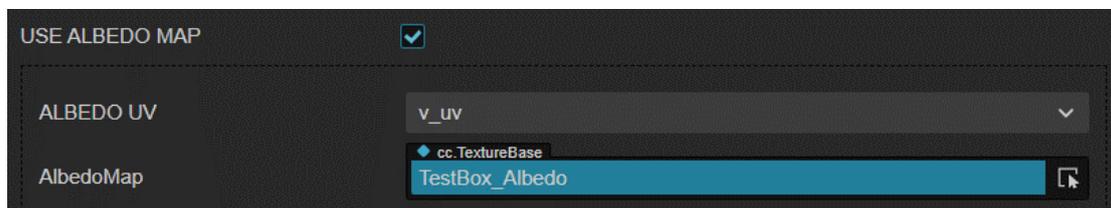


Normal textures. Normal Strength: Adjust normal texture strength.

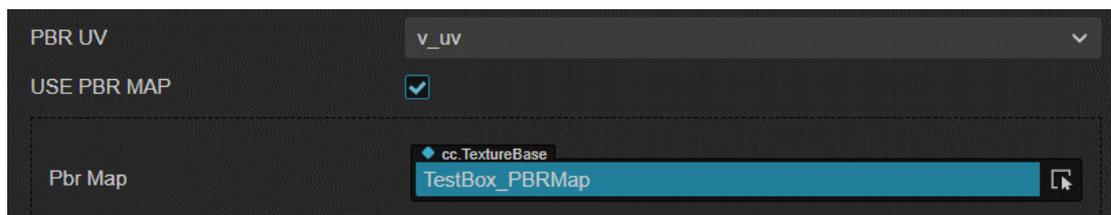


2. Use PBR textures:

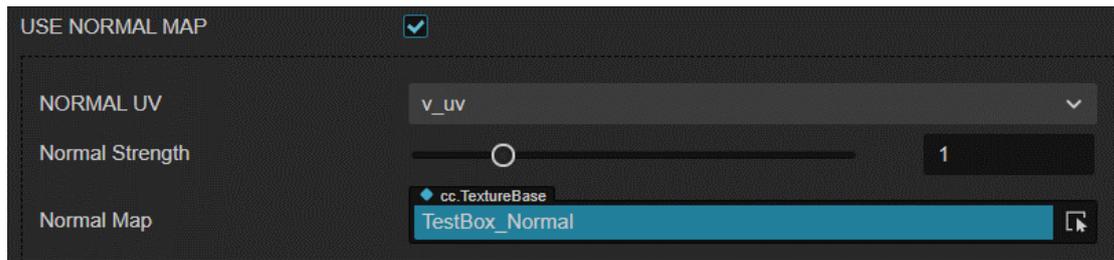
Albedo textures:



PBR textures:



Normal textures. Normal Strength: Adjust normal texture strength.



Please refer to the documentation for more information on the PBR material guide.

<https://docs.cocos.com/creator/manual/en/shader/effect-builtin-pbr.html>

04. PBR material Parameter settings:

Tiling Offset: Scaling and offset of the model UV.

Albedo Scale: Albedo intensity of the model, used to control the weight of the albedo color influence on the final color.

Occlusion: Ambient light occlusion factor.

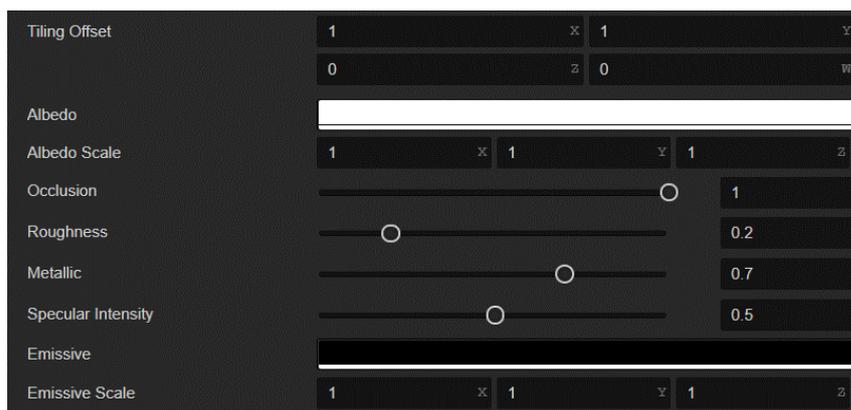
Roughness: Roughness coefficient.

Metallic: Metallicity coefficient.

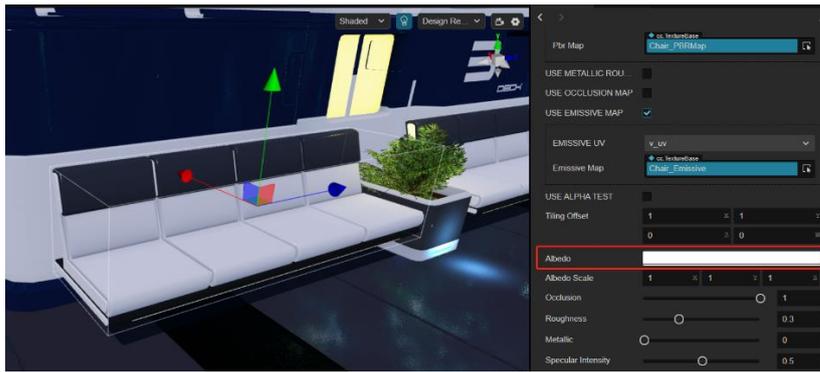
Specular Intensity: Specular Intensity coefficient.

Emissive: Emissive colors, independent of light calculation, directly emitted by the model itself

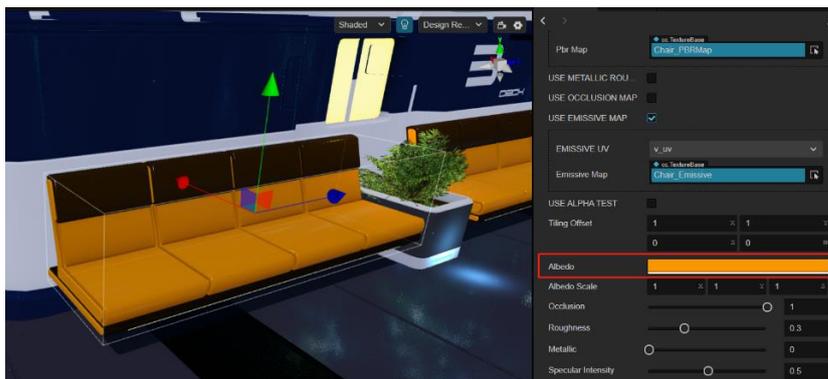
Emissive Scale: Emissive intensity. It is used to control the weighting of the emissive color on the final color.



Albedo Color can change the color of the texture. It will be multiplied with the Albedo map color. When the Albedo color is the default white, it will keep the texture default color.

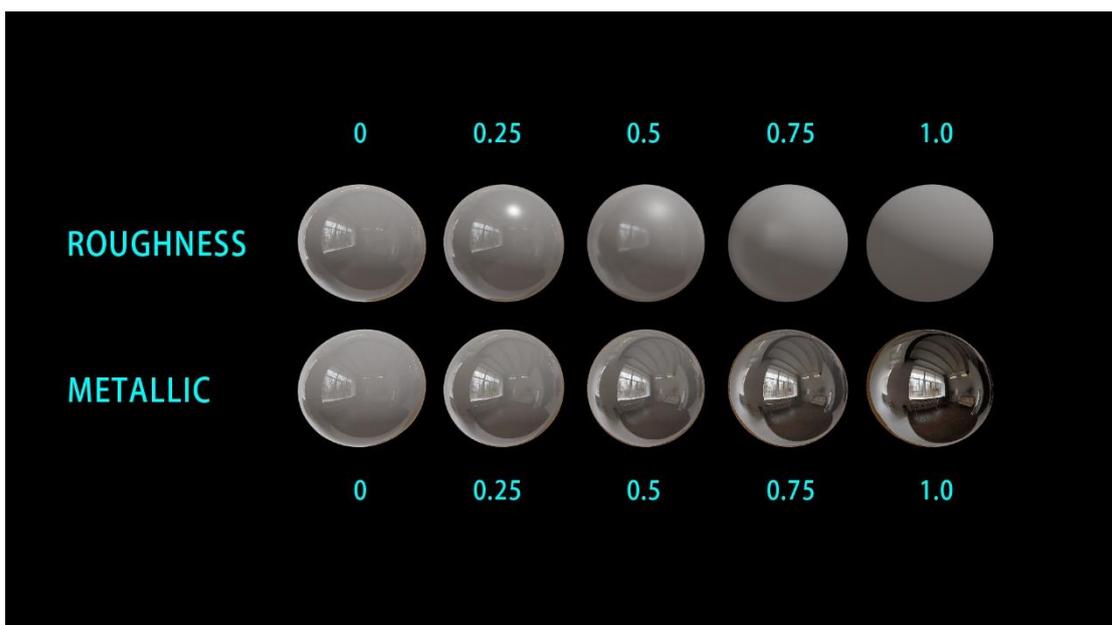


Adjust the Albedo color. It will directly affect the texture color.



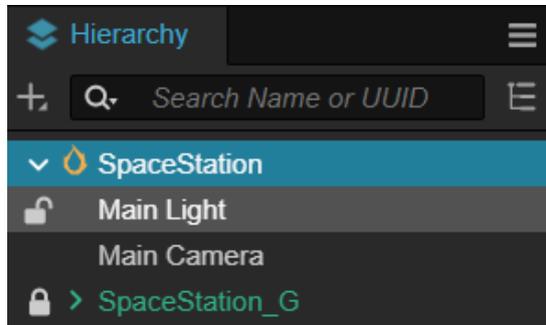
Roughness Controls the roughness of the material. The larger the value, the rougher it is, and the smaller the value, the smoother it is.

Metallic controls the metallicity of the material. 1 is metal. 0 is non-metal.



7. Setting up the environment and shadows

01. Select the SpaceStation scene within Hierarchy.



02. In the Inspector panel, set the required parameters.

Sky Lighting Color: Sets the sky color.

Sky Illum: Adjusts the sky brightness.

Ground Lighting Color: Sets the color of the reflected light from the ground.

Env Lighting Type: Select the type suitable for the scene. For the introduction of the three types, please refer to the relevant documents of the sky box.

<https://docs.cocos.com/creator/manual/en/concepts/scene/skybox.html>

Use HDR: If this option is checked, HDR (High Dynamic Range) will be turned on. If not checked, LDR (Low Dynamic Range) will be used.

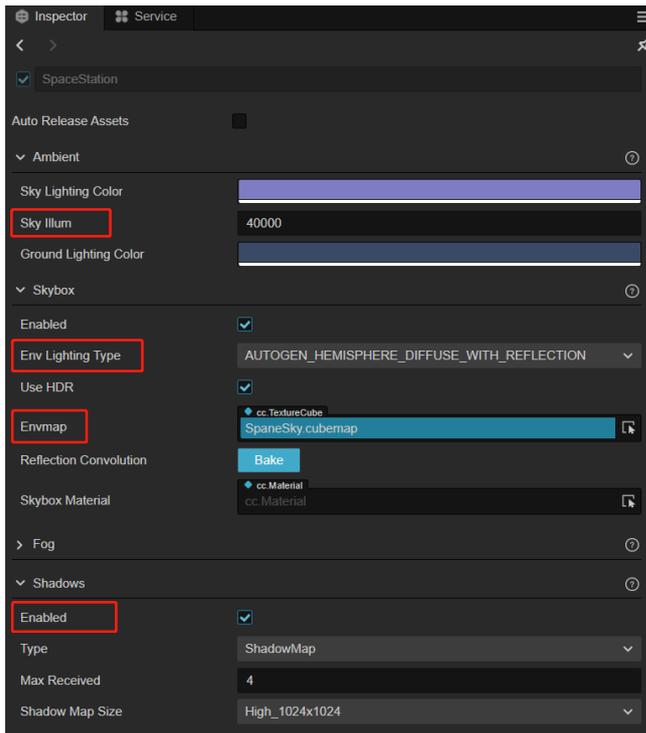
Envmap: Choose a skybox texture or use an HDRI environment texture.

Reflection Convolution: Click the bake button will generate a low-resolution environment map and perform convolution calculation on this map. A convolution map will be used for environment reflection.

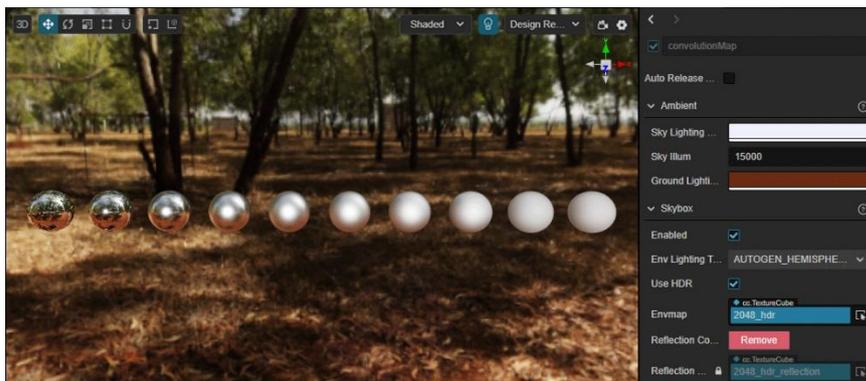
The generated environment reflection convolution map will fill the mipmaps for TextureCube, Sampling mipmap in the shader based on material roughness, thus providing a more realistic IBL effect

Skybox Material: Add custom material for the skybox.

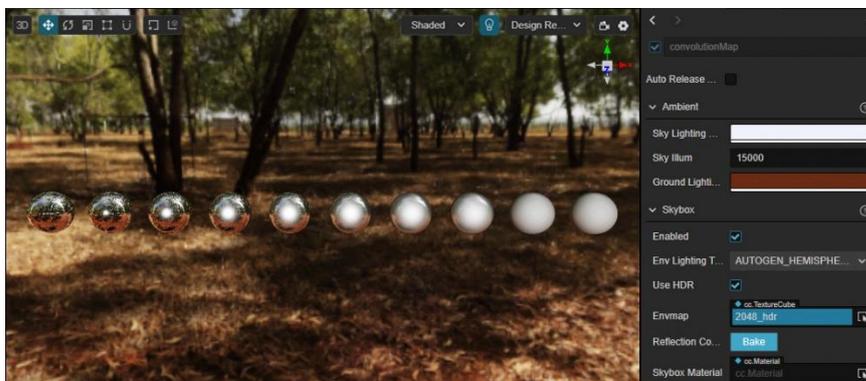
Turn on the shadows, and use the Planar shadow type by default.



Reflection Convolution: Bake.



Reflection Convolution: Remove bake



03. Light Shadow Properties

The Planar shadow type is generally used for simpler scenes.

Enabled: Whether to enable shadow effect

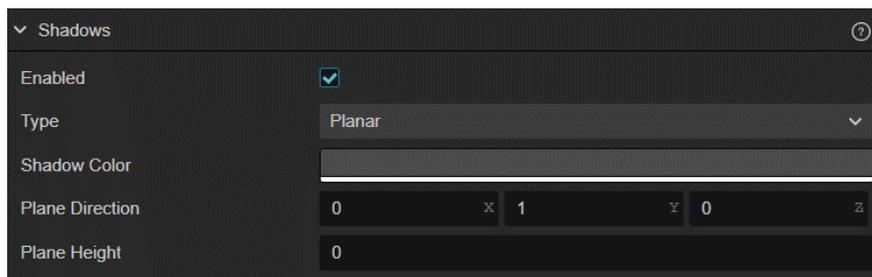
Type: Shadow type

Shadow Color: Shadow color.

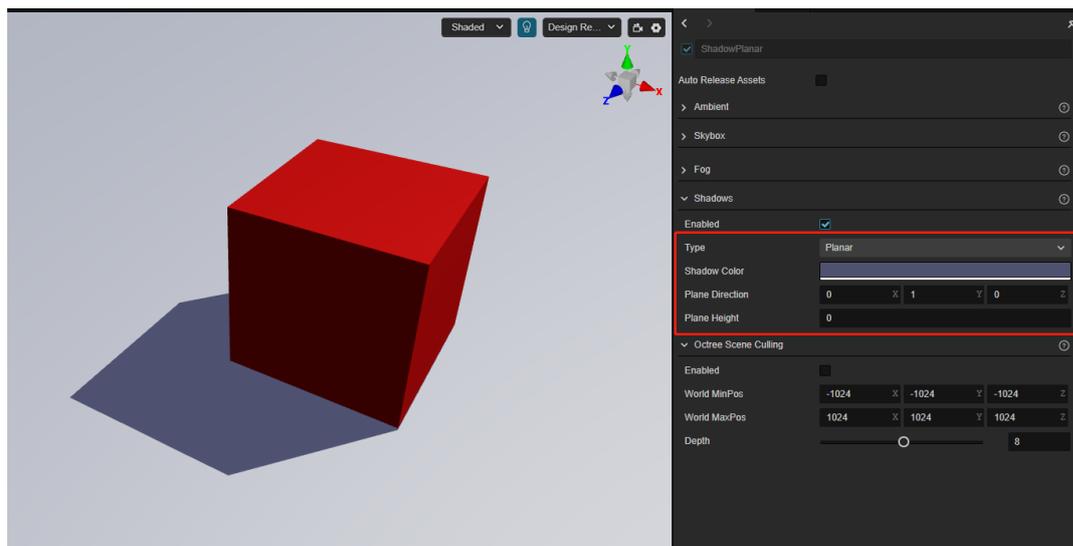
Plane Direction: The normal vector of the plane which receives shadow.

Plane Height: The height from the plane's origin receives shadow.

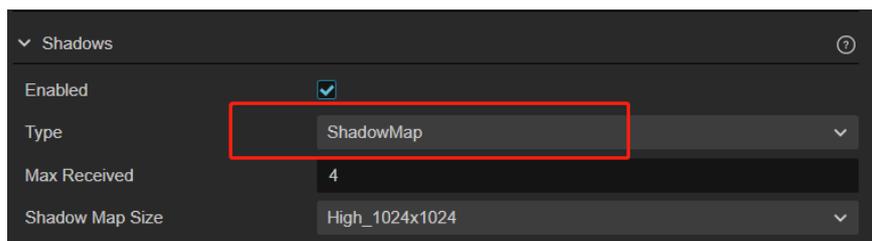
Note: planar shadows are only cast on planar surfaces, not on objects, which means that the ReceiveShadow property in the MeshRenderer component is invalid.



Planar shadow effect:



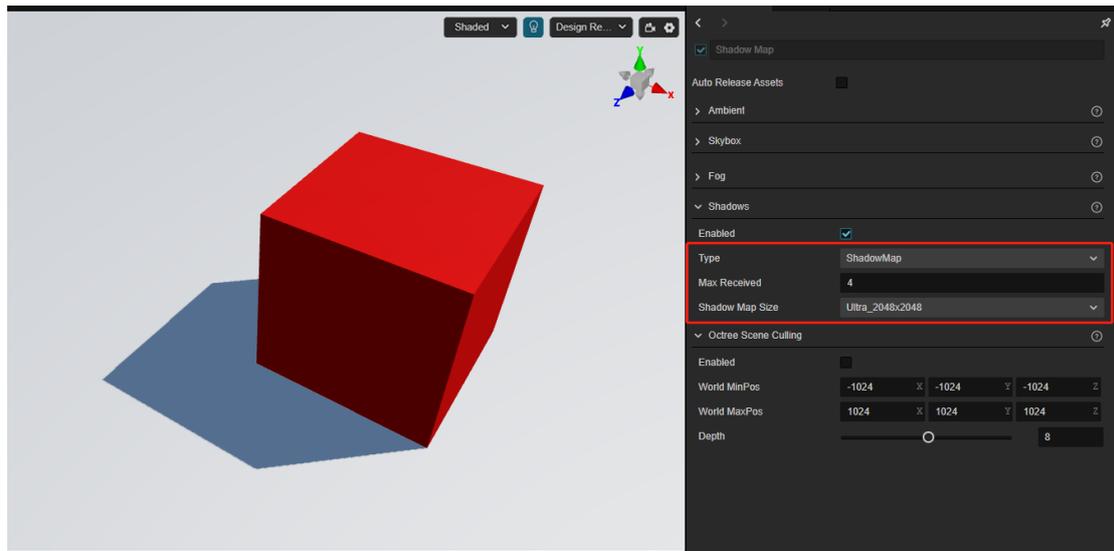
When the scene shadow type is: Shadow Map



Max Received: The maximum number of lights supported for shadow generation, default is 4, can be adjusted as needed.

Shadow Map Size: Set the texture size of the shadow. It currently supports Low_256x256, Medium_512x512, High_1024x1024, Ultra_2048x2048 four kinds of precision textures.

Shadow Map effect:



8. Lighting

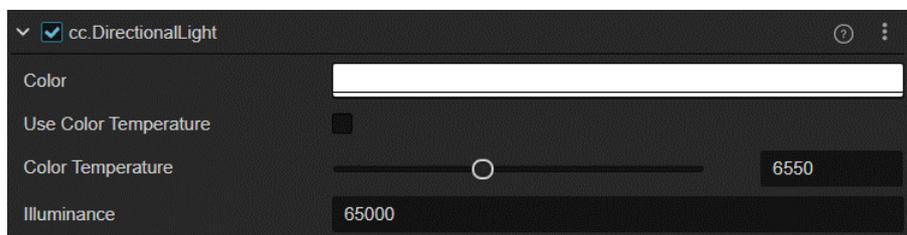
01. Main Light uses Directional Light to simulate sunlight.

Color: Sets the light color.

Use Color Temperature: Whether to enable the color temperature.

Color Temperature: Adjusts the color temperature.

Illumination: Illumination in lux (lx).



When the scene shadow type is: Shadow Map, you can turn on the shadow property of Directional Light.

Shadow Enabled: Whether to enable the shadow effect.

Shadow Pcf: Set the anti-aliasing level of the shadow edge, currently including HARD, SOFT, SOFT_2X.

Shadow Bias: Set the shadow offset value to prevent z-fitting.

Shadow Normal Bias: Set the normal offset value.

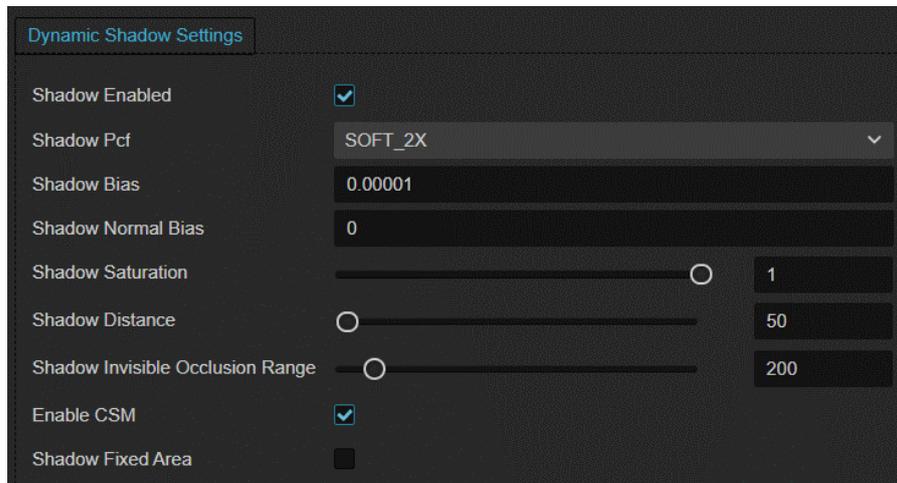
Shadow Saturation: Set the shadow saturation. It is recommended to set it as 1. If it is necessary to reduce the saturation of the directional light shadows, it is recommended to increase the ambient light instead of adjusting this value.

Shadow Distance: Set the range of shadow effects displayed within the visible range of the Camera, with the shadow quality inversely proportional to the size of this value.

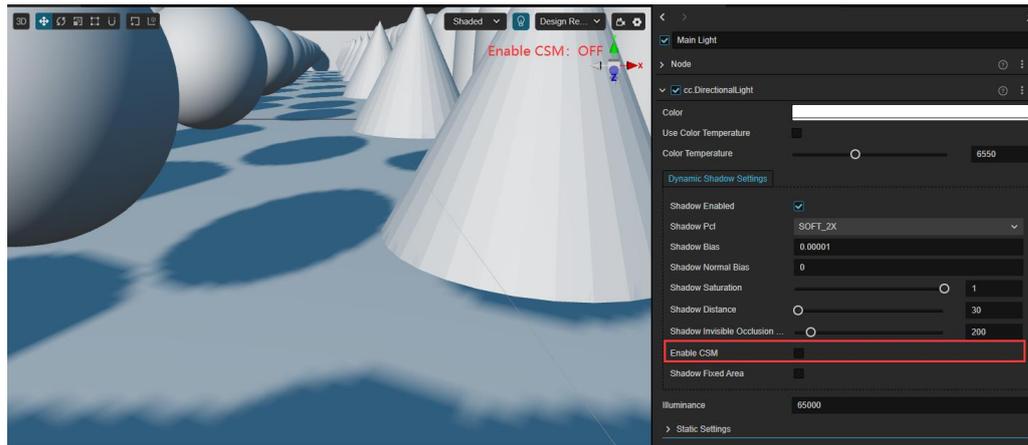
Shadow Invisible Occlusion Range: Set whether shadows from objects outside of the Camera's visible range are cast into the visible range, and if so, turns up the value.

Enabled CSM: Cascaded Shadow Maps.

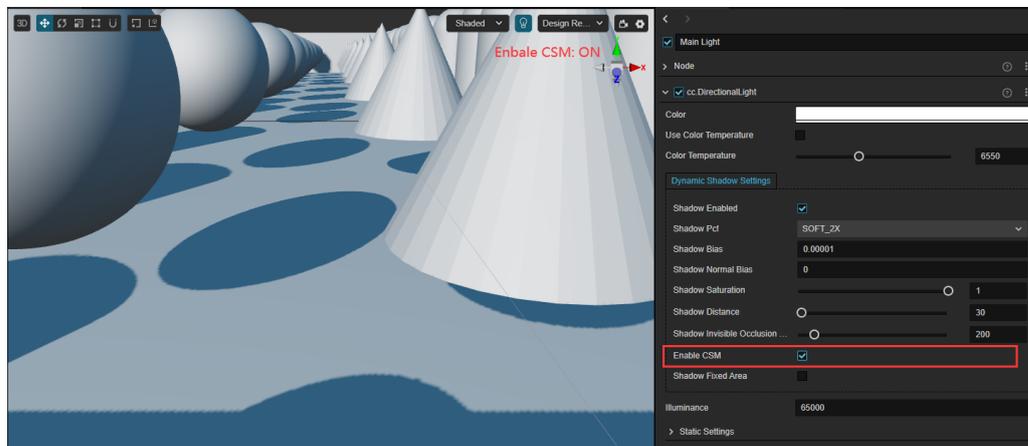
A higher quality shadow map is used for near scenes, and a lower quality shadow map is used for far scenes, with one of the two shadow maps chosen for use where they transition. This ensures that the observer sees higher quality shadows in the immediate area and lower quality shadows in the outlying area because the distant objects occupy only a small fraction of the screen. In contrast, the near objects occupy a significant fraction of the screen.



CSM OFF:



CSM ON:



Shadow Fixed Area: Enable Shadow Fixed Area.

Shadow Near: Set the near-clipping plane of the main lights shadow camera.

Shadow Far: Set the far-clipping plane of the main lights shadow camera.

Shadow Ortho Size: Set the ortho viewport size of the main lights shadow camera, with the shadow quality inversely proportional to the size of this value.



02. Using Sphere Lights

Term: Sets the type of light intensity unit, including LUMINOUS_POWER and LUMINANCE.

Luminous Flux: Luminous Flux.

Color: Sets the light color.

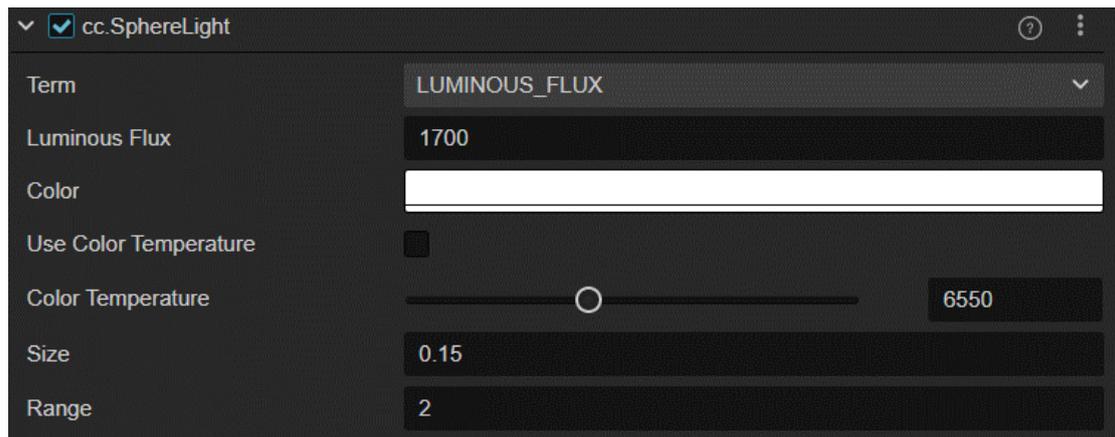
Use Color Temperature: Whether to enable the color temperature.

Color Temperature: Adjusts the color temperature.

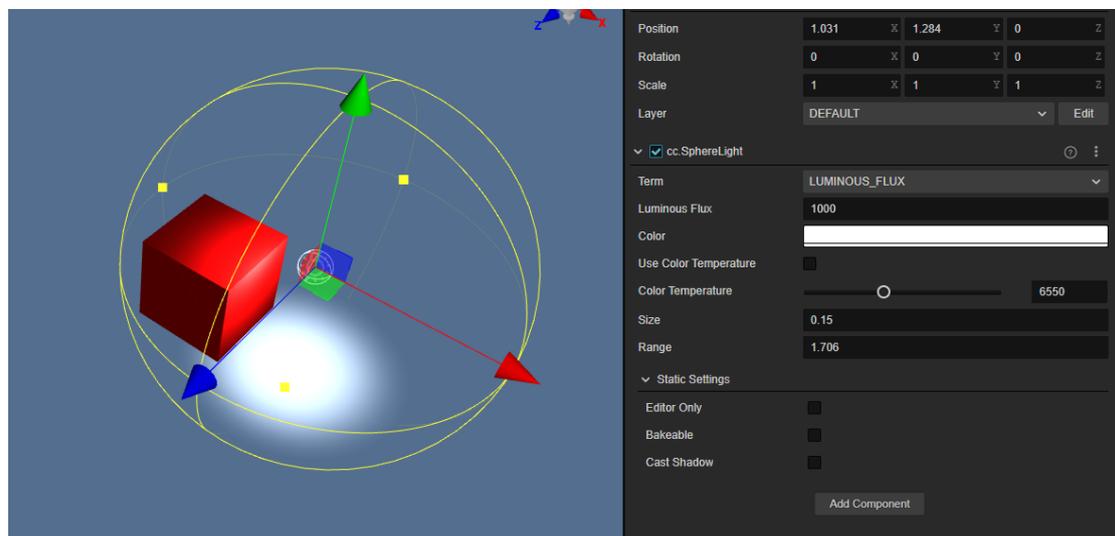
Size: Sets the light size.

Range: Sets the range of light effect.

Note: currently, the Size property of the spherical light does not take effect in actual operation, and the display of shadows is not supported. For now, we will optimize it in subsequent versions. Please pay attention to the update announcement.



Sphere Lights



03. Using Spot Light.

Color: Sets the color of the light.

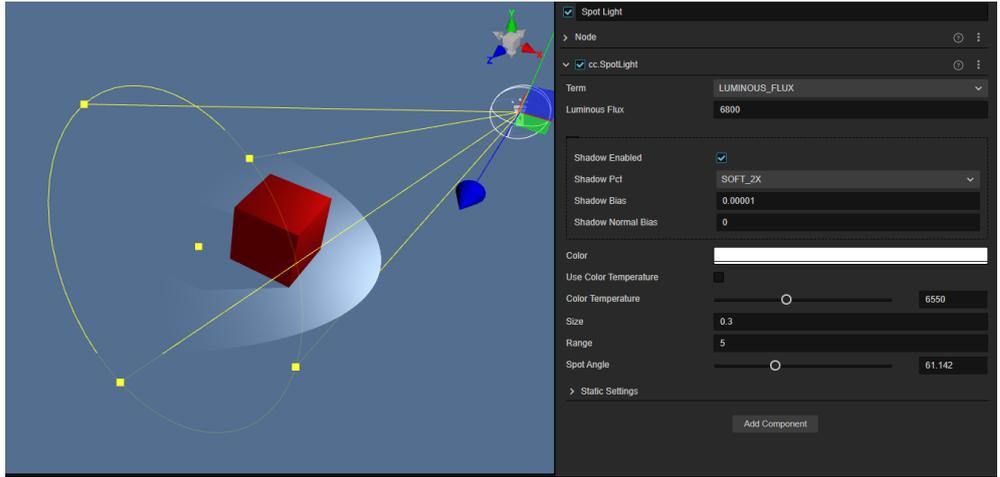
Use Color Temperature: Whether to enable the color temperature.

Color Temperature: Adjusts the color temperature.

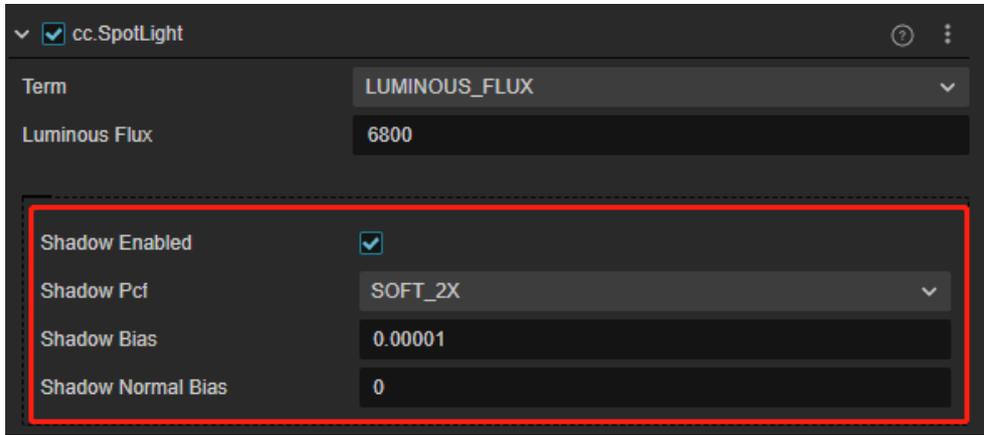
Size: Sets the light size.

Range: Sets the range of light effect.

Spot Angle: Adjusts the spot angle to control the light range.



Spotlight Shadows: Spotlight Shadows are separated from the Scene Settings panel and are no longer affected by the Global Shadows parameter.)



- Shadow Enabled: Whether to enable the shadow effect.
- Shadow Pcf: Set the anti-aliasing level of the shadow edge, currently including HARD, SOFT, SOFT_2X.
- Shadow Bias: Set the shadow offset value to prevent z-fitting.
- Shadow Normal Bias: Set the normal offset value.

04. PCF Soft Shadow

Percentage Closer Filtering (PCF) is a simple, common technique used to achieve shadow

edge desampling by smoothing shadow edges to eliminate jaggedness in shadow mapping. The principle is to sample around the current pixel (also called a fragment), then calculate the ratio of the sample closer to the lights compared to the fragment, use this ratio to scale the scattered light and specular light, and then color the fragment to blur the shadow edges.

Cocos Creator currently supports a hard sampler (HARD mode), 4x sampler (SOFT mode), and 9x sampler (SOFT_2X mode). The larger the magnification, the larger the sampling area and the more softer the shadow edges.

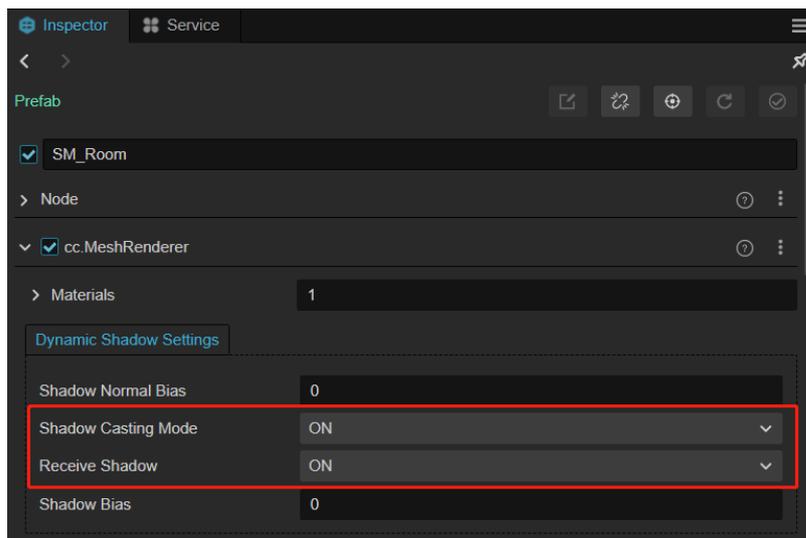
9. Set the shadow and bake properties of the models

01. Shadow settings for models

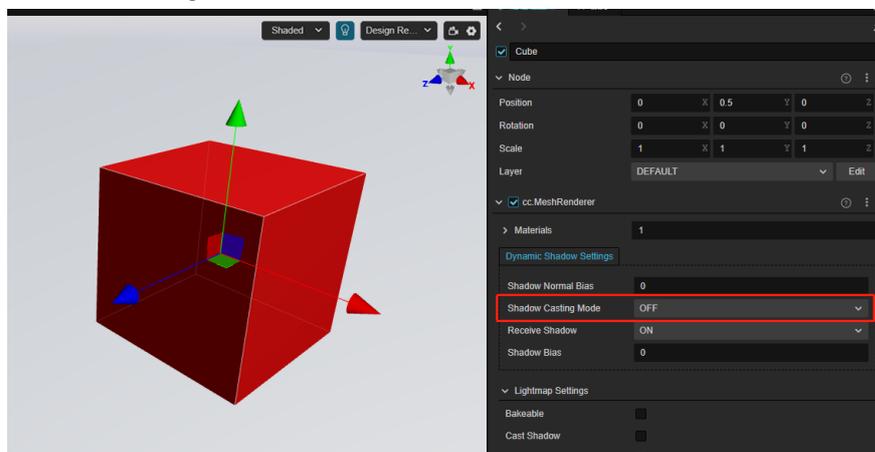
Set relevant content in the MeshRenderer component of the model.

Shadow Casting Mode: on

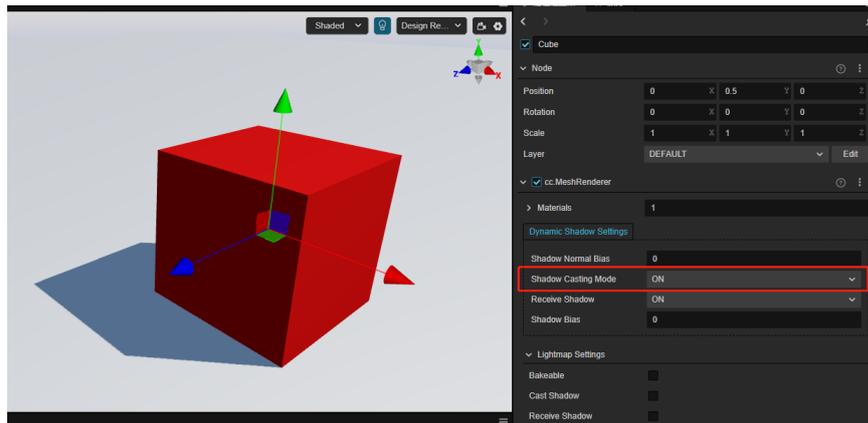
Receive Shadow: on



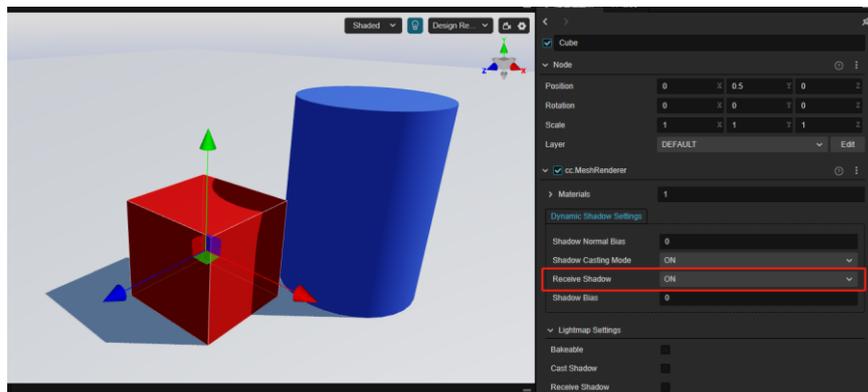
Shadow Casting Mode: OFF.



Shadow Casting Mode: ON.



Receive Shadow: ON. The model will receive and display shadows generated by other objects (this property only takes effect when the shadow type is Shadow Map)



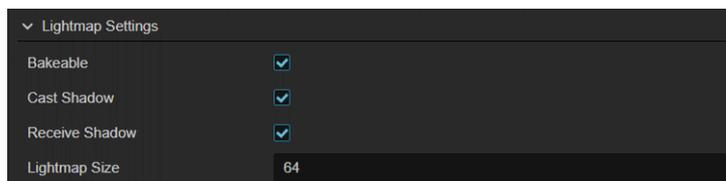
02. Lightmap settings for the model.

Bakeable: whether to bake static lighting.

Cast Shadow: whether to cast static shadows.

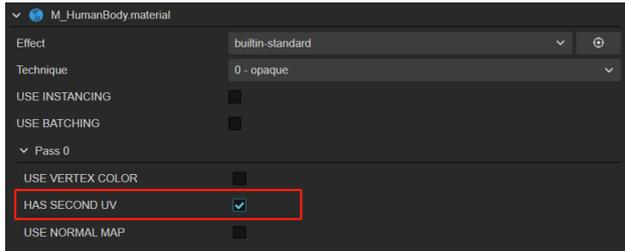
Receive Shadow: whether to receive static shadows.

Lightmap Size: the size of the model lightmap



Notes: to generate lightmaps for a model, there are two requirements:

1. When the artist creates a model resource, in addition to the UVs of the model itself, another set of UVs for lightmapping needs to be included.
2. The model's Materials need to have the HAS SECOND UV rendering option turned on.



10. Baking lightmaps

01. Select the light source node in the Hierarchy, set the Static Settings of the light source component in the Inspector, and enable the Bakeable parameter.

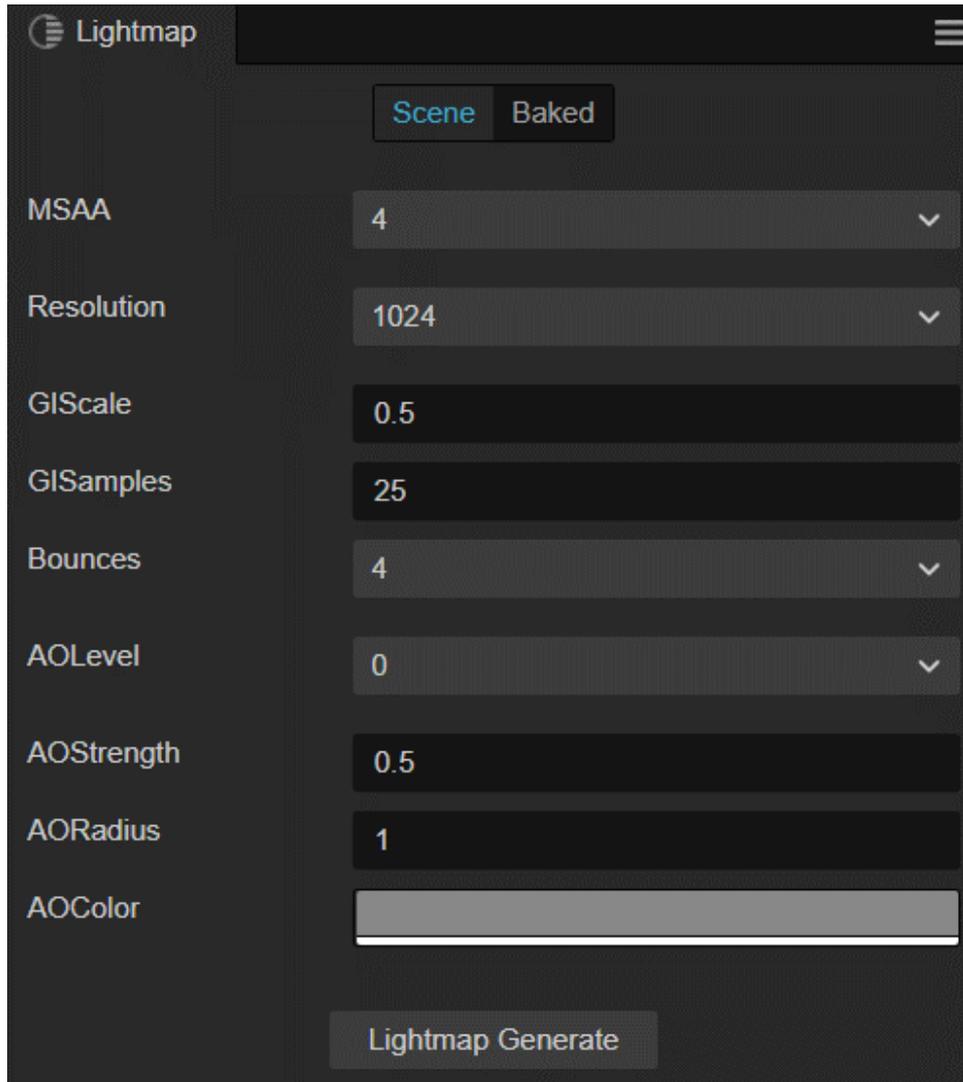


Editor Only: whether to take effect only in the editor.

Bakeable: whether to bake static lighting.

Cast Shadow: whether to cast static shadows.

02. Open the Lightmapping panel. Click on the editor menu bar, Project—Lightmap, and open the lightmap panel. The lightmap panel consists of two pages, scene and baked.



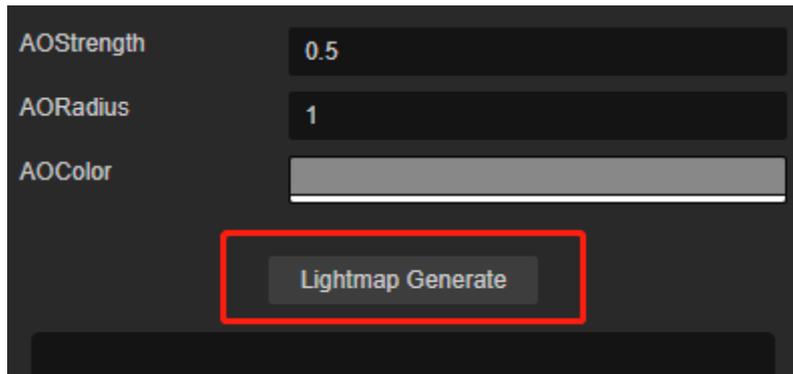
Scene: mainly used to configure the parameters related to generating the lightmap.
Baked: mainly used to display the generated lightmap and its related information.

The description of each property on the Scene page is as follows:

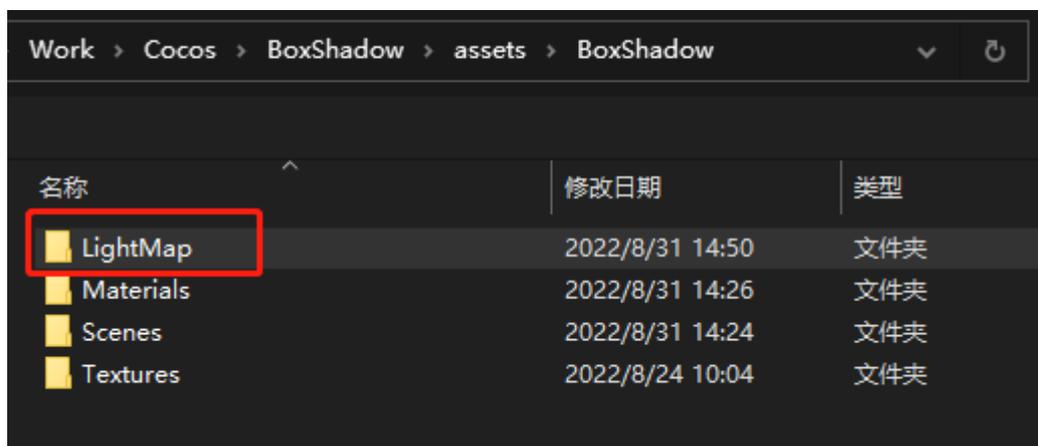
Property	Description
MSAA	Multiple Sampling, optional values include: 1, 2, 4, 8
Resolution	The resolution of the generated lightmap, optional values are 128, 256, 512, 1024, 2048
GIScale	Global lighting scaling factor
GISamples	Global illumination sampling factor
Bounces	Number of light bounces

AOLevel	AO (Ambient Occlusion) level
AOStrength	AO intensity
AORadius	AO radius
AOColor	AO color

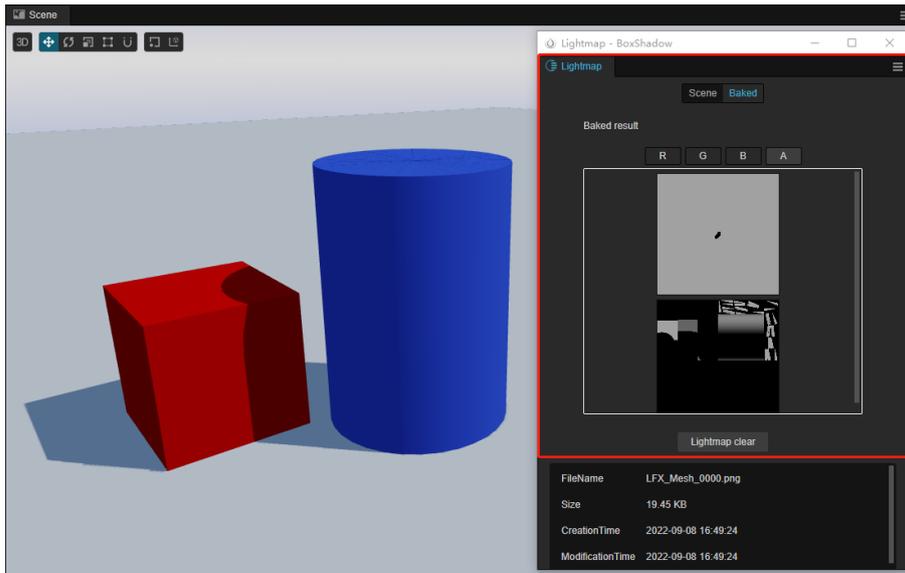
Lightmap Generate



The engine needs to specify a lightmap storage location first.



After baking, the generated lightmap, the file name, size, and other related information can be viewed on the Baked page of the Lightmapping panel. The generated lightmaps are automatically processed by the engine and do not need to be manipulated by the developer.



1. Bake result: shows the generated lightmap after baking, in format RGBE8, with R/G/B options to view the corresponding channels of the lightmap as required.
2. Lightmap clear: used to delete the generated lightmap and related information.
3. Information output panel: shows the generated lightmap file name, size, and other related information.

