

# Atmosphere

This component handles the rendering of planet or star atmospheres. The atmosphere is rendered using a separate material which can be applied on top of anything inside the atmosphere, so it's compatible with any shader that doesn't modify vertex positions. The atmosphere rendering is fully volumetric, so the camera can seamlessly travel from space to the ground with no loss of visual quality. The SgtTerrain component also has support for atmospheres.

## Color

This allows you to tint the atmosphere color.

## Brightness

The **Color**.RGB values get multiplied by this when written to the shader. This is useful in combination with HDR.

## Render Queue

This allows you to set which render queue group the atmosphere will be placed in.

## Render Queue Offset

This allows you to tweak the render queue position. For example, the Transparent render queue is 3000, so an offset of 5 will set the render queue to 3005.

## Height

This allows you to set how high above the surface you want the atmosphere to appear in local space.

## Inner Fog

This allows you to set how foggy the surface is. A value of 0 is the default.

## Outer Fog

This allows you to set how foggy the sky is. A value of 0 is the default.

## Sky

This allows you to set how quickly the sky turns opaque when the camera goes inside the atmosphere. A value of 1 is the default.

## Lit

If you enable this, then the atmosphere will be subject to ambient lighting, and allow lighting + shadows.

## Lighting Tex

This allows you to set the lookup table used to calculate the current brightness & color of the atmosphere.

The SgtAtmosphereLighting component can be used to generate this.

## Scattering

If you enable this, then lights will scatter as they shine on the atmosphere, giving a halo around them.

## Ground Scattering

If you enable this, then scattering will be applied to the ground (inner) material too.

## Scattering Tex

This allows you to set the lookup table used to calculate the scattering color based on the camera to fragment to light angle.

The SgtAtmosphereScattering component can be used to generate this.

## Scattering Strength

This allows you to set how strong the scattering effect is.

## Scattering Mie

This allows you to set the sharpness of the mie scattering term. This affects the shape of the forward scattered light, creating a halo around the light source.

## Scattering Rayleigh

This allows you to set the brightness of the rayleigh scattering term. This affects the brightness of the front and back scattered light.

## Lights

If you want a light to shine on this then add it here.

NOTE: Only the first active light will be used.

## Shadows

If you want a shadow to cast on this then add it here.

NOTE: Only the first two active shadows will be used.

## Inner Depth Tex

This allows you to set the lookup table used to calculate the optical depth of the surface.

The SgtAtmosphereDepth component can be used to generate this.

## Inner Mesh Radius

This allows you to set the actual radius of the meshes referenced in the **Inner Renderers** list. For example, the default Unity sphere primitive has a radius of 0.5 units.

## Inner Renderers

This allows you to set which renderers are being used for your planet's surface. For example, if your planet surface is currently only using one MeshRenderer, and simply drag and drop it here.

## Outer Depth Tex

This allows you to set the lookup table used to calculate the optical depth of the sky.

The SgtAtmosphereDepth component can be used to generate this.

## Outer Mesh Radius

This allows you to set the actual radius of the meshes you set in the **Outer Meshes** list. For example, the default Unity sphere primitive has a radius of 0.5 units.

## Outer Meshes

This allows you to set the meshes you want to use for the atmosphere. These should be standard spheres with outward facing normals.

## [CONTEXT] Update Materials

This will update all settings for the inner and outer material.

NOTE: This should get called automatically.

## [CONTEXT] Apply Inner Material

This will apply the inner material to all inner renderers.

NOTE: This should get called automatically.

## [CONTEXT] Remove Inner Material

This will remove the inner material from all inner renderers.

NOTE: This should get called automatically.

## [CONTEXT] Update Outers

This will update all outer GameObjects and their settings.

NOTE: This should get called automatically.