

# Diffuse Parameters



In the parameters below, some of them can be overridden by a [PxrLayer](#) when connected to the **Input Material** or through a [PxrLayerMixer](#). [PxrLayerSurface](#) is designed to better illustrate which parameters are not able to be overridden in a layer by including only parameters that are global. We recommend this material when you know you will be layering. The results of these settings are unchanged.

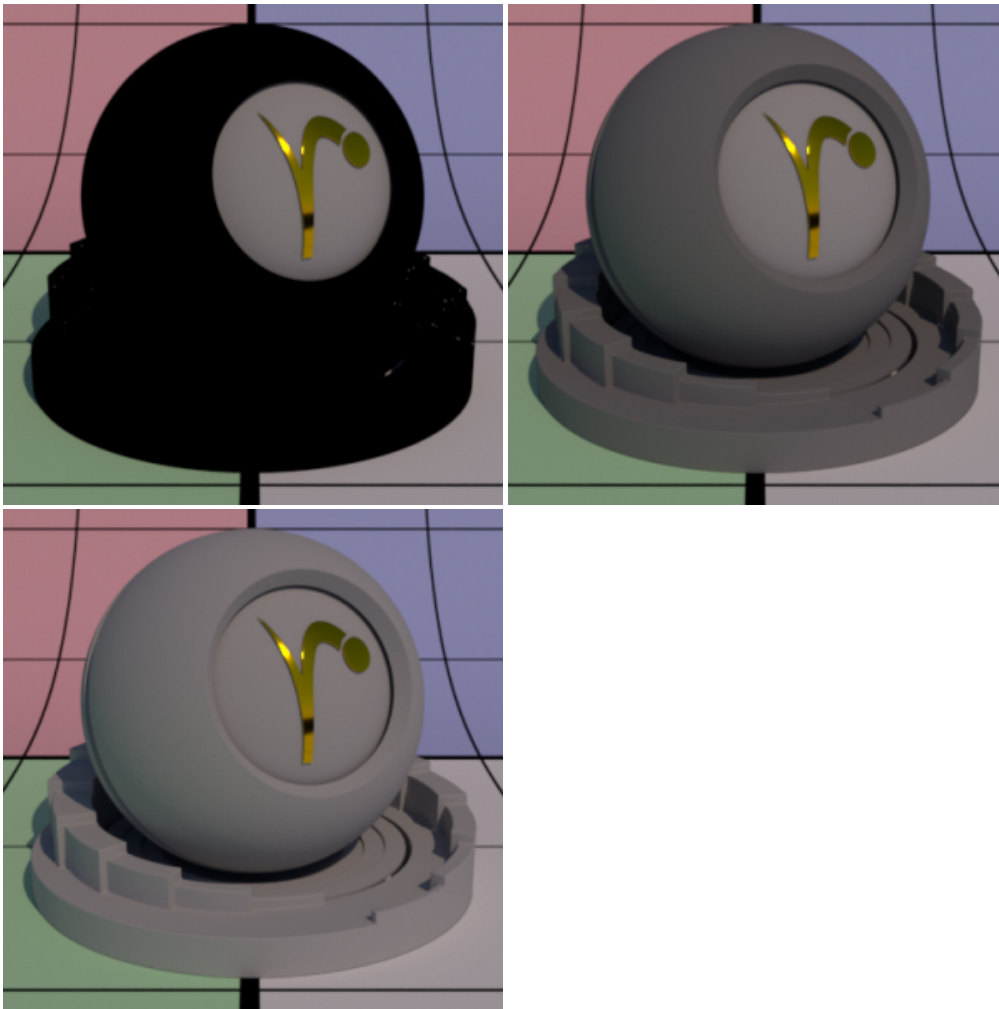
- **Bold Face** parameters are layerable, able to be overridden per layer.
- *Italicized* parameters are not able to be layered or overridden. These are globally obeyed for all layers. For example: Choosing GGX as a specular model will mean all layers will be GGX for that parameter.

## Diffuse Parameters

The diffuse parameters control the look of basic diffuse reflection. These are often used to define primary color attributes for opaque objects; wood textures, label text, polka dots, or more, you can find them all connected here. This lobe is on by default.

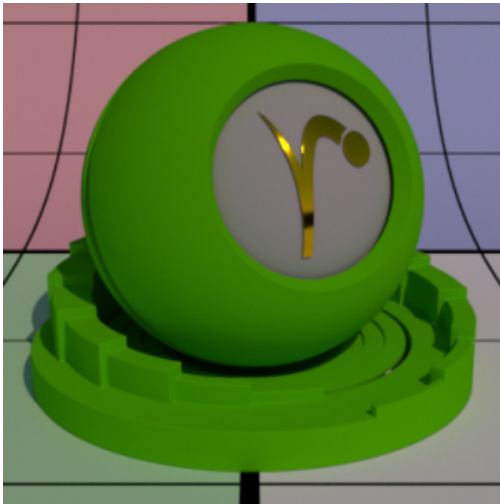
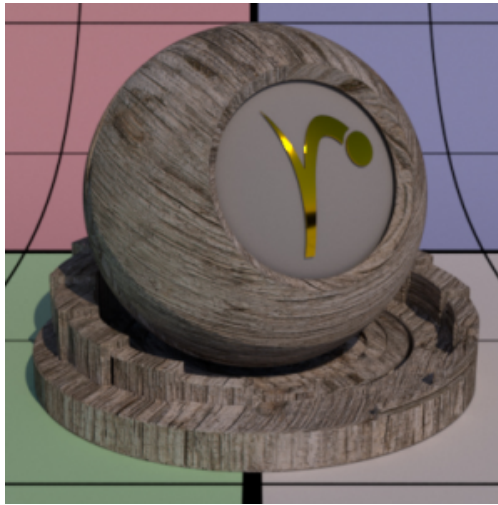
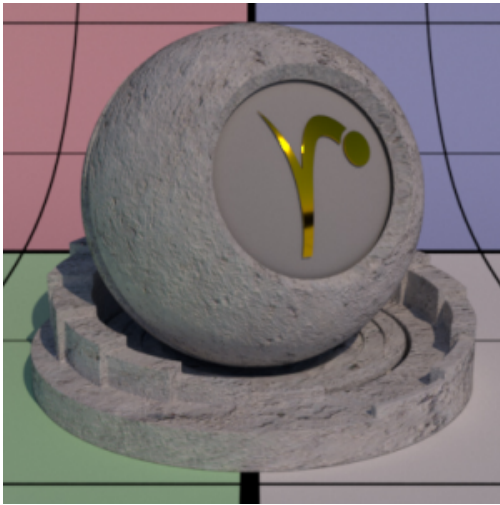
### Gain

Gain is the weight applied to the diffuse parameters. You may also drive this with another pattern to show things like fading or wetness (where liquid darkens a surface). Below are examples at 0.0, 0.5, and 1.0 gain for a 50% gray material.



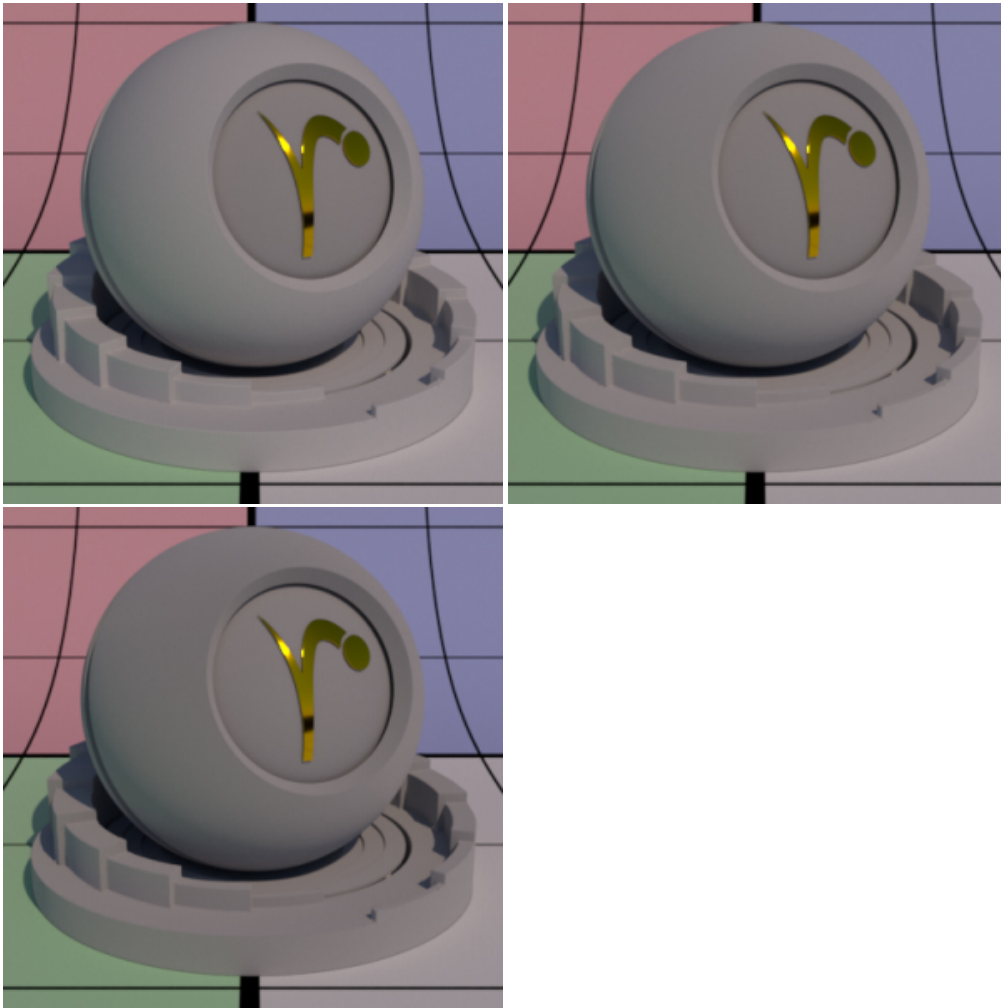
### Color

Color is typically where textures or patterns are connected to create color for opaque objects. This is where a wood color texture would go, for example.



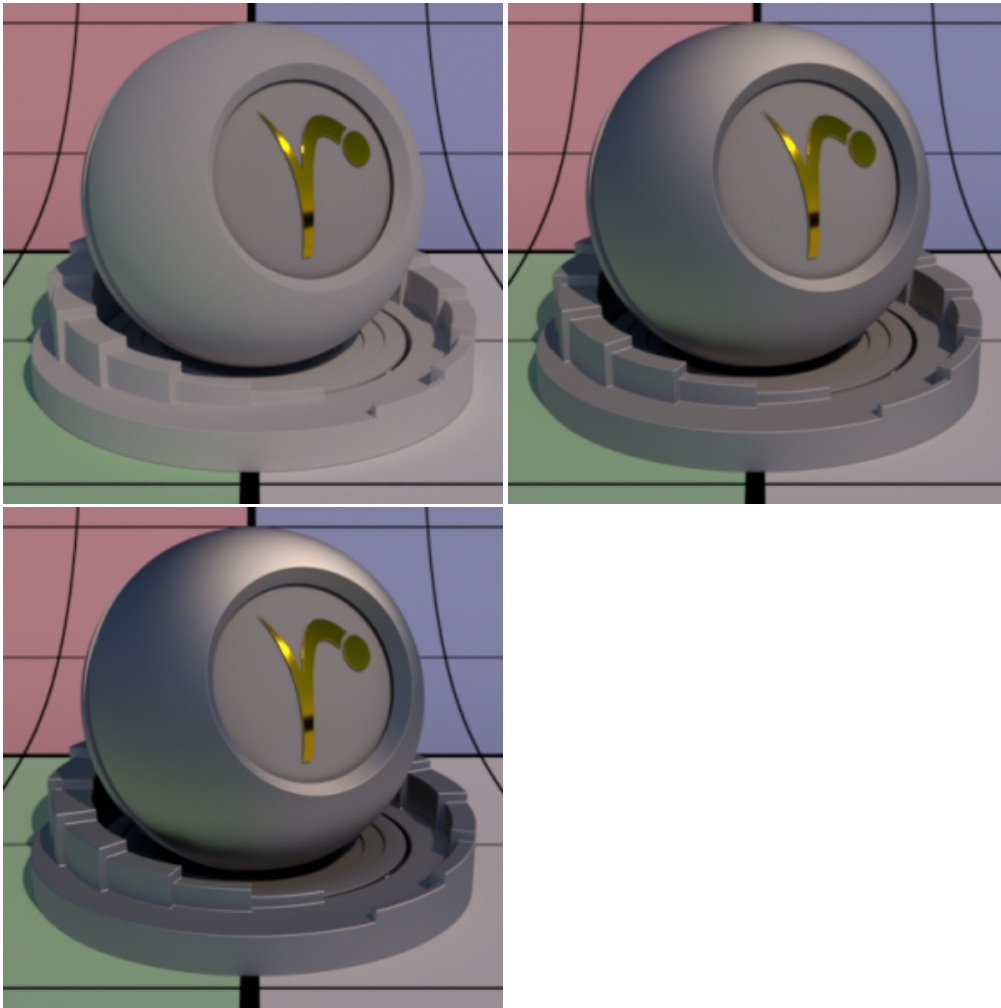
### **Roughness**

Diffuse roughness is how you would simulate a powdery surface like dried clay or dust. When roughness is 0.0, PxrSurface uses Lambertian model to calculate the diffuse response. When roughness is  $> 0.0$ , Oren Nayar model is used instead.



### Exponent

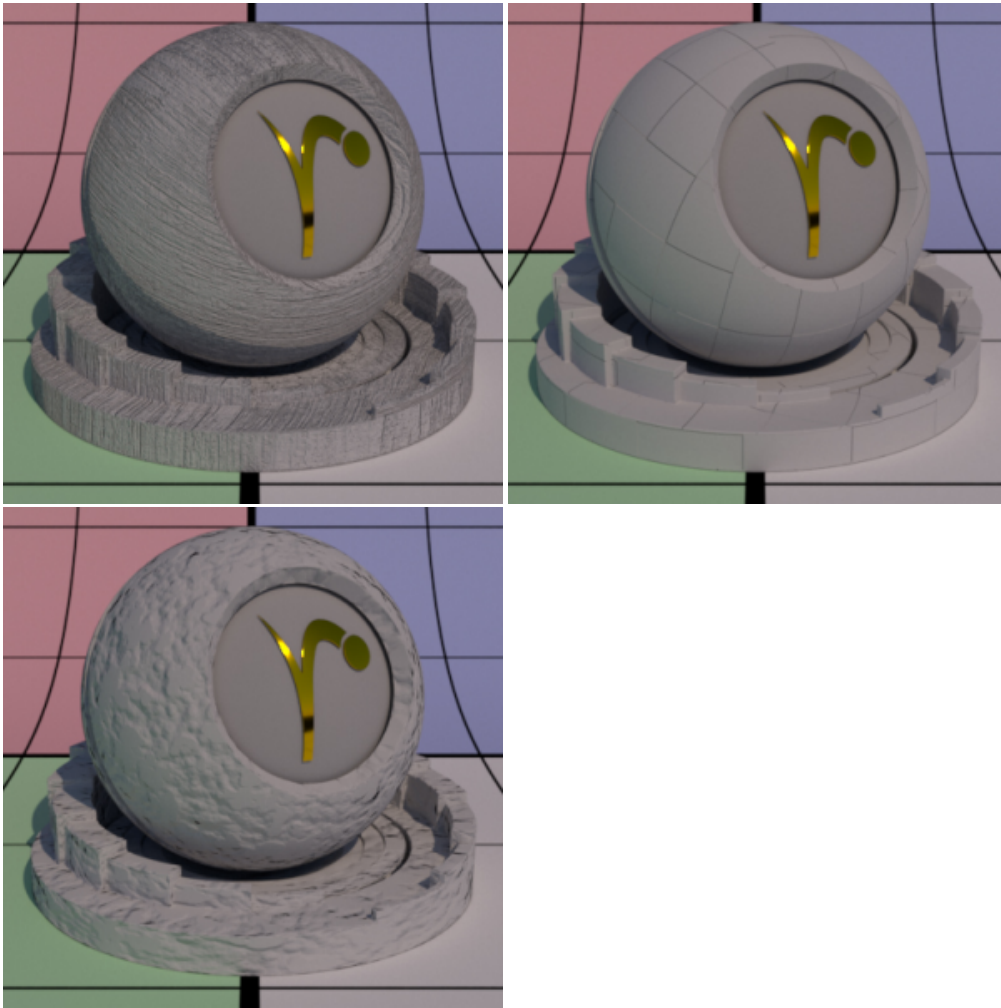
Diffuse exponent controls the diffuse falloff. It is a power exponent. Higher number gives us sharper falloff. For example, on the moon surface, you can set a high exponent to produce a sharper falloff.



Diffuse exponent only applies to the Lambertian diffuse, that is, when roughness is 0.0. Combining diffuse exponent with Oren Nayar roughness does not make sense so it will be ignored when roughness is  $> 0.0$ .

### Bump

Bump mapping is a great way to fake the appearance of physical detail using shading instead. If this is not set, it will use the global bump normal specified in the Properties section near the bottom of this page.



#### *Double Sided*

If on, illuminate both sides of the surface for this diffuse lobe, that is, this will illuminate the surface whose normal is pointing away from the camera (2-dimensional objects) as well. This is so the backface will be shaded instead of black. This is off by default.

#### *Use Diffuse Color*

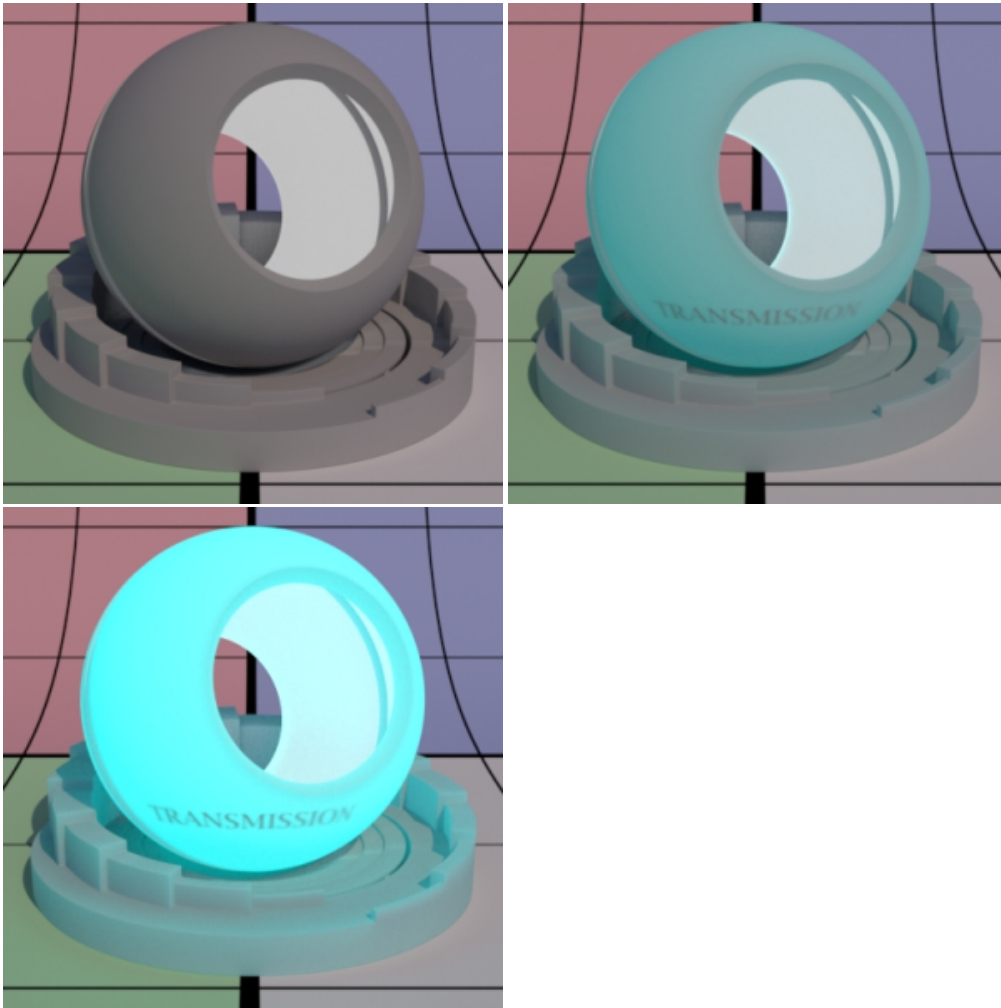
This only applies when *Double Sided* is on. By default, this is on to use the **Diffuse Color** for the back color.

#### **Back Color**

This only applies when *Double Sided* is on. When Use Diffuse Color if off, this sets the back color (the color for the back side). By default, it uses the Diffuse Color but choosing a different color provides you with a way to make the backside of 2D objects appear differently.

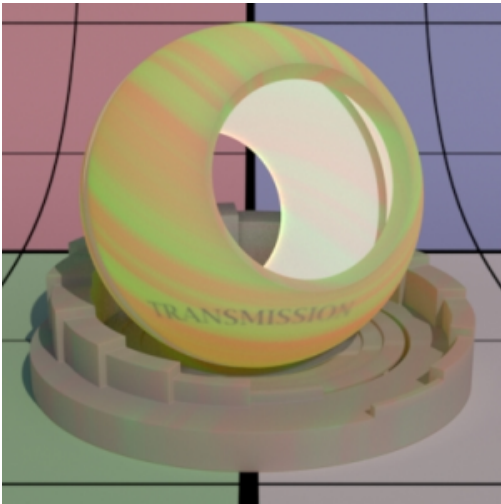
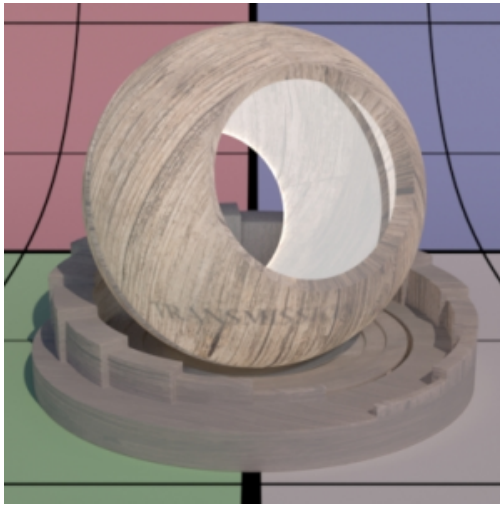
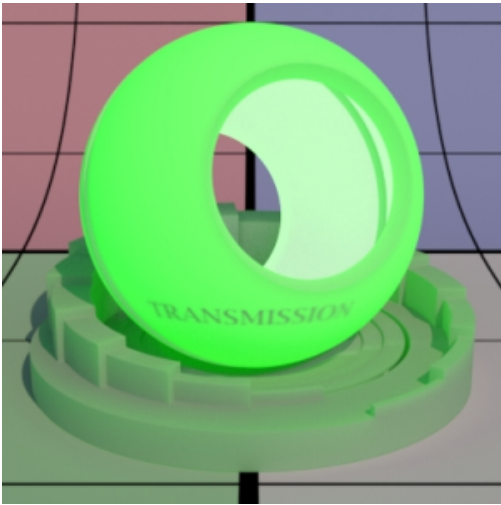
#### **Transmit Gain**

This only applies when *Double Sided* is on. Instead of using the **Diffuse Gain** for controlling the intensity of the diffuse transmission, we use the **Transmit Gain**. If it's 0.0, then the effect is off. Below the **Transmit Color** is bright blue.



### Transmit Color

This only applies when *Double Sided* is on. This sets the transmit color which could be different than the diffuse or back color. This is ignored if **Transmit Gain** is zero. This effect is useful for thin objects like leaves or paper. Below there's a light placed in the interior of the object and some interior text can be seen as light transmits through the surface.



[Back to the top](#)