

10.3.4.4 Render - Cycles Render Engine - Nodes - Shader Nodes

Shader Nodes.....1

Diffuse.....1

Translucent.....2

Glossy.....2

Anisotropic.....3

Toon.....4

Transparent.....4

Glass.....5

Refraction.....5

Velvet.....6

Subsurface Scattering.....6

Emission.....7

Background.....8

Holdout.....8

Ambient Occlusion.....9

Mix and Add.....9

Shader Nodes

Diffuse

Lambertian and Oren-Nayar diffuse reflection.

Color input

Color of the surface, or physically speaking, the probability that light is reflected or transmitted for each wavelength.

Roughness input

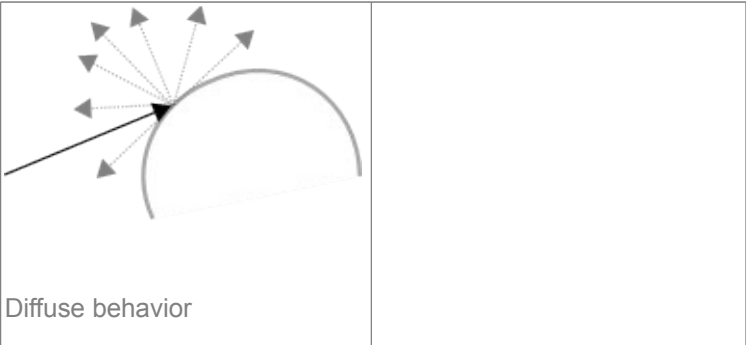
Surface roughness; 0.0 gives standard Lambertian reflection, higher values activate the Oren-Nayar BSDF.

Normal input

Normal used for shading; if nothing is connected the default shading normal is used.

BSDF output

Diffuse BSDF shader.





Translucent

Lambertian diffuse transmission.

Color input

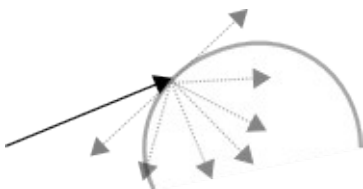
Color of the surface, or physically speaking, the probability that light is transmitted for each wavelength.

Normal input

Normal used for shading; if nothing is connected the default shading normal is used.

BSDF output

Translucent BSDF shader.



Translucent Shader

Glossy

Glossy reflection with microfacet distribution, used for materials such as metal or mirrors.

Distribution

Microfacet distribution to use. *Sharp* results in perfectly sharp reflections like a mirror, while *Beckmann*, *GGX* and *Ashikhmin-Shirley* can use the *Roughness* input for blurry reflections.

Color input

Color of the surface, or physically speaking, the probability that light is reflected for each wavelength.

Roughness input

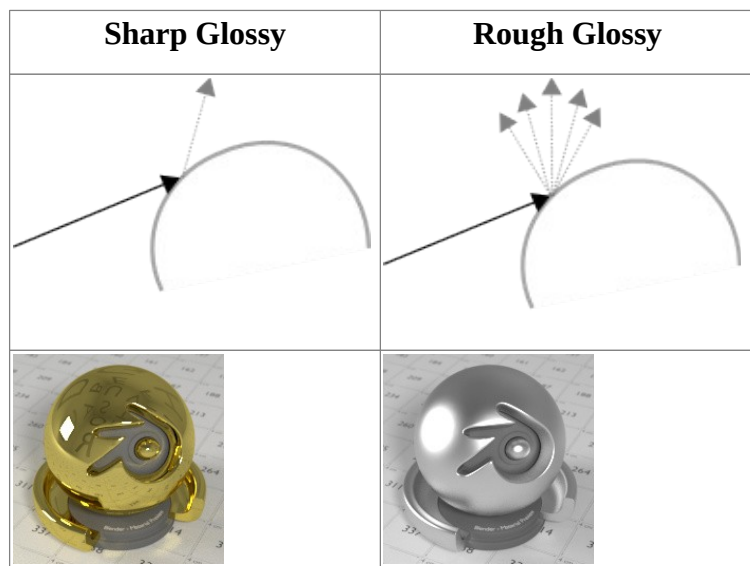
Influences sharpness of the reflection; perfectly sharp at 0.0 and smoother with higher values.

Normal input

Normal used for shading; if nothing is connected the default shading normal is used.

BSDF output

Glossy BSDF shader.



Anisotropic

Anisotropic glossy reflection, with separate control over U and V direction roughness. The tangents used for shading are derived from the active UV map. If no UV map is available, they are automatically generated using a sphere mapping based on the mesh bounding box.

Distribution

Microfacet distribution to use. *Sharp* results in perfectly sharp reflections like a mirror, while *Beckmann*, *GGX* and *Ashikhmin-Shirley* can use the *Roughness* input for blurry reflections.

Color input

Color of the surface, or physically speaking, the probability that light is reflected for each wavelength.

Roughness input

Sharpness of the reflection; perfectly sharp at 0.0 and smoother with higher values.

Anisotropy input

Amount of anisotropy in the reflection; 0.0 gives a round highlight. Higher values give elongated highlights orthogonal to the tangent direction; negative values give highlights shaped along the tangent direction.

Rotation input

Rotation of the anisotropic tangent direction. Value 0.0 equals 0- rotation, 0.25 equals 90- and 1.0 equals 360- = 0- . This can be used to texture the tangent direction.

Normal input

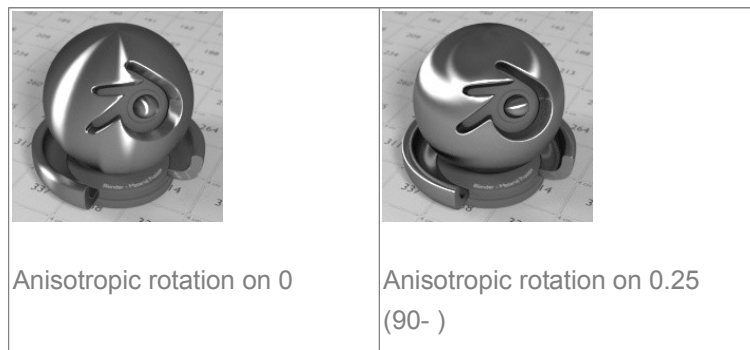
Normal used for shading; if nothing is connected the default shading normal is used.

Tangent input

Tangent used for shading; if nothing is connected the default shading tangent is used.

BSDF output

Anisotropic glossy BSDF shader.



Toon

Diffuse and Glossy Toon BSDF for creating cartoon light effects.

Color input

Color of the surface, or physically speaking, the probability that light is reflected for each wavelength.

Size input

Parameter between 0.0 and 1.0 that gives a angle of reflection between 0- and 90- .

Smooth input

This value specifies an angle over which a smooth transition from full to no reflection happens.

Normal input

Normal used for shading; if nothing is connected the default shading normal is used.

BSDF output

Toon BSDF shader.



Toon Shader

Transparent

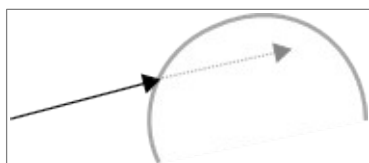
Transparent BSDF without refraction, passing straight through the surface, as if there were no geometry there. Useful with alpha maps, for example. This shader affects light paths somewhat differently than other BSDF s. Note that only pure white transparent shaders are completely transparent.

Color input

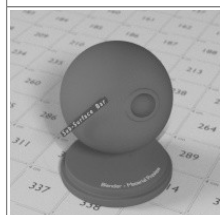
Color of the surface, or physically speaking, the probability for each wavelength that light is blocked or passes straight through the surface.

BSDF output

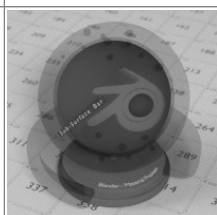
Transparent BSDF shader.



Transparent behaviour



Transparent Shader (pure white)



Transparent Shader (gray)

Glass

Glass-like shader mixing refraction and reflection at grazing angles. Like the transparent shader, only pure white will make it transparent. The glass shader tends to cause noise due to caustics. Since the Cycles path tracing integrator is not very good at rendering caustics, it helps to combine this with a transparent shader for shadows; for more details see [here](#)

Distribution

Microfacet distribution to use. *Sharp* results in perfectly sharp refractions like clear glass, while *Beckmann* and *GGX* can use the *Roughness* input for rough glass.

Color input

Color of the surface, or physically speaking, the probability that light is transmitted for each wavelength.

Roughness input

Influences sharpness of the refraction; perfectly sharp at 0.0 and smoother with higher values.

IOR input

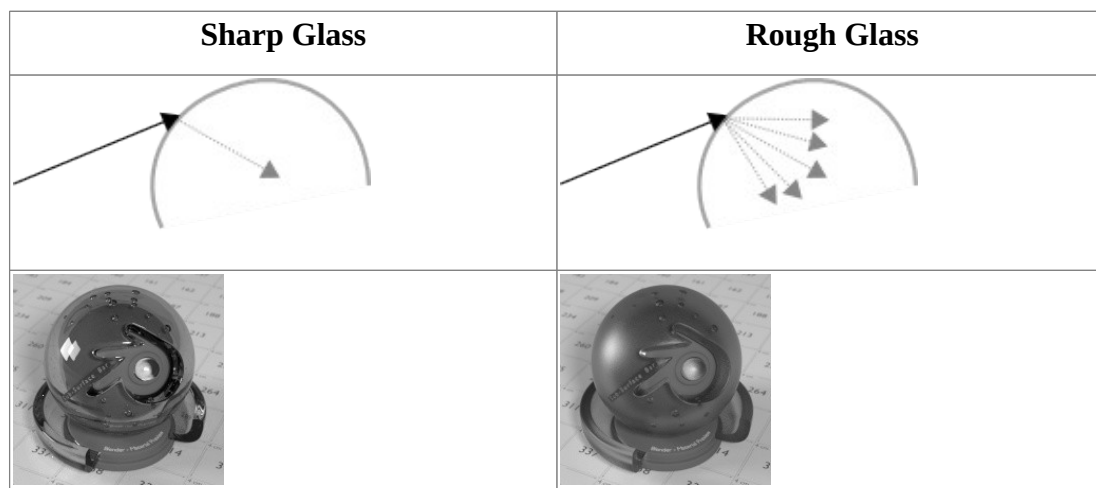
Index of refraction defining how much the ray changes direction. At 1.0 rays pass straight through like transparent; higher values give more refraction.

Normal input

Normal used for shading; if nothing is connected the default shading normal is used.

BSDF output

Glass BSDF shader.



Refraction

Glossy refraction with sharp or microfacet distribution, used for materials that transmit light. For best results this node should be considered as a building block and not be used on its own, but rather mixed with a glossy node using a fresnel factor. Otherwise it will give quite dark results at the edges for glossy refraction.

Distribution

Microfacet distribution to use. *Sharp* results in perfectly sharp refractions, while *Beckmann* and *GGX* can use the *Roughness* input for blurry refractions.

Color input

Color of the surface, or physically speaking, the probability that light is refracted for each wavelength.

Roughness input

Influences sharpness of the refraction; perfectly sharp at 0.0 and smoother with higher values.

Normal input

Normal used for shading; if nothing is connected the default shading normal is used.

BSDF output

Glossy BSDF shader.



Refraction Shader.

Velvet

Velvet reflection shader for materials such as cloth. It is meant to be used together with other shaders (such as a *Diffuse Shader*) and isn't particularly useful on it's own.

Color input

Color of the surface, or physically speaking, the probability that light is reflected for each wavelength.

Sigma input

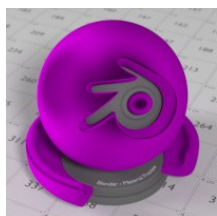
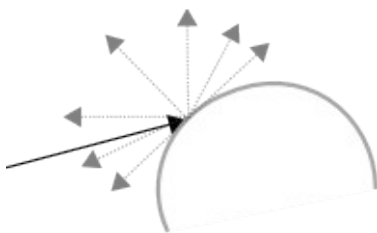
Variance of the normal distribution, controlling the sharpness of the peak - can be thought of as a kind of *roughness*.

Normal input

Normal used for shading; if nothing is connected the default shading normal is used.

BSDF output

Velvet BSDF shader.



The Velvet Shader

Subsurface Scattering

Simple subsurface multiple scattering, for materials such as skin, wax, marble, milk and others. For these materials, rather than light being reflect directly off the surface, it will penetrate the surface and bounce around internally before getting absorbed or leaving the surface at a nearby point.

How far the color scatters on average can be configured per RGB color channel. For example, for skin, red colors scatter further, which gives distinctive red-colored shadows, and a soft appearance.

Falloff

Lighting distance falloff function. **Cubic** is a sharp falloff useful for many simple materials. The function is $(\text{radius} - x)^3$ **Gaussian** gives a smoother falloff following a normal distribution, which is particularly useful for more advanced materials that use measured data that was fitted to one or more such Gaussian

functions. The function is $e^{-8x^2/\text{radius}^2}$, such that the radius roughly matches the maximum falloff distance. To match a given measured variance v , set $\text{radius} = \sqrt{16*v}$.

Color input

Color of the surface, or physically speaking, the probability that light is reflected for each wavelength.

Scale input

Global scale factor for the scattering radius.

Radius input

Scattering radius for each RGB color channel, the maximum distance that light can scatter.

Sharpness input

Used only with **Cubic** falloff. Values increasing from 0 to 1 prevents softening of sharp edges and reduces unwanted darkening.

Normal input

Normal used for shading; if nothing is connected the default shading normal is used.

Texture Blur input

How much of the texture will be blurred along with the lighting, mixing the texture at the incoming and outgoing points on the surface. Note that the right choice depends on the texture. Consider for example a texture created from a photograph of skin, in this cases the colors will already be pre-blurred and texture blur could be set to 0. Even for hand painted textures no or minimal blurring might be appropriate, as a texture artist would likely paint in softening already, one would usually not even know what an unblurred skin texture looks like, we always see it blurred. For a procedural texture on the other hand this option would likely have a higher value.

BSSRDF output

BSSRDF shader.



A skin-toned SSS shader with color radius: 1.0, 0.8, 0.5.

Emission

Lambertian emission, to be used for material and lamp surface outputs.

Color input

Color of the emitted light.

Strength input

Strength of the emitted light. For point and area lamps, the unit is Watts. For materials, a value of 1.0 will ensure that the object in the image has the exact same color as the Color input, i.e. make it 'shadeless'.

Emission output

Emission shader.



Emission shader, with strength at 1.0	Emission shader, with strength at 3.0
---------------------------------------	---------------------------------------

Cycles uses a physically correct light falloff by default, whereas Blender Internal uses a smoothed falloff with a Distance parameter. A similar effect can be found by using the Light Falloff node with the Smooth parameter.

Lamp strength for point, spot and area lamps is specified in Watts. This means you typically need higher values than Blender Internal, as you couldn't use a 1W lamp to light a room; you need something stronger like a 100W lamp.

Sun lamps are specified in Watts/m², which require much smaller values like 1 W/m². This can be confusing, but specifying strength in Watts wouldn't have been convenient; the real sun for example has strength 384600000000000000000000000W. Emission shaders on meshes are also in Watts/m².

Background

Background light emission. This node should only be used for the world surface output; it is ignored in other cases.

Color input

Color of the emitted light.

Strength input

Strength of the emitted light.

Background output

Background shader.

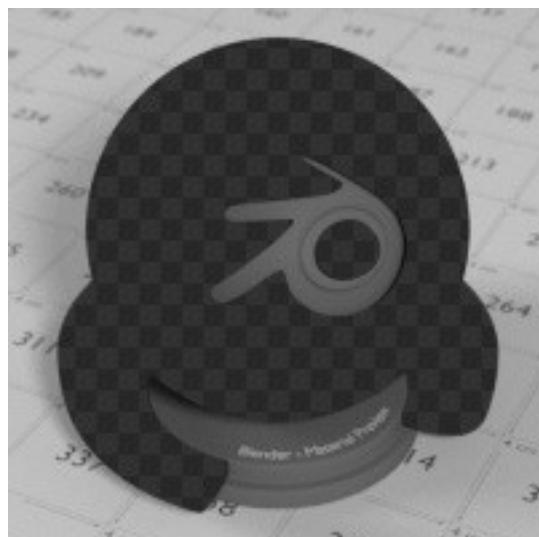
Holdout

The holdout shader creates a “hole” in the image with zero alpha transparency, which is useful for compositing (see alpha channel).

Note that the holdout shader can only create alpha when Properties ▸ Render ▸ Film ▸ Transparent is enabled. If it's disabled, the holdout shader will be black.

Holdout output

Holdout shader.



The checkered area is a region with zero alpha.

Ambient Occlusion

The ambient occlusion node gives per-material control for the amount of AO. When AO is enabled in the world, it affects all diffuse BSDFs in the scene. With this option it's possible to let only some materials be affected by AO, or to let it influence some materials more or less than others.

Color input

surface reflection color.

AO output

Ambient Occlusion shader.



White AO shader.

Mix and Add

Mix or add shaders together. Mixing can be used for material layering, where the *Fac* input may, for example, be connected to a Blend Weight node.

Shader inputs

Shaders to mix, such that incoming rays hit either with the specified probability in the *Fac* socket.

Fac input

Blend weight to use for mixing two shaders; at zero it uses the first shader entirely and at one the second shader.

Shader output

Mixed shader.



A mix of a glossy and a diffuse shader makes a nice ceramic material.