TurboCad TurboLux

Version 1.1

Rendering engine introduced in TurboCAD 2022

This guide is a compilation of information from the internet, Turbolux help file, Luxcore help file, and my experimentation.

It is not meant as a tutorial.

Whilst everything has been tested in TC 2022, I'm only human. So it is still possible errors may remain.

Note, as TurboLux evolves, it in inevitable, that some thing will break, and cease working as the programmers release new builds. Therefore, some items in the document, may not work with your Turbocad build. (TC 2022 onward).

This was mainly done, to save trawling the internet or various help files. When I forget what something does.

Andy H

Contents

Rendering points to remember 6
TurboLux engine (program settings). 8
TurboLux scene (display settings). 16
Camera properties 27
Material and Luminance palette. 29
Render manager 31

Luminance. 35
- General luminance properties. 36
- Constantinfinite. 40
- Infinite. 41
- Distant. 43
- SharpDistant. 45
- Fluorescent lights. 46
- Laser light. 47
- Projection Light. 49
- Point light. 51
- Mappoint light. 53
- Sphere light. 55
- Mapsphere light. 56
- Spot light. 58
- Sky22 light. 60
- Sun. 62

Materials. 65
- Common properties. 66
  - Internal Volume. 67
  - External Volume 73
  - Light Emission 74
  - Bump texture 76
  - Normal Texture 79
  - Bump sampling distance 82
  - Transparency 80
  - Samples 82
  - Indirect diffuse visibility 82
  - Indirect glossy visibility 82
  - Indirect specular visibility 82
  - Shadow catcher 83
  - Shadow catcher onlyinfinitelights 83
  - Shadow transparency 84
- Car Paint 85
- Cloth 87
- Disney. 89
- Glass Architectural 93
- Glass 95
- Glass Rough. 99
- Glossy2 100
- Glossy Translucent 104
- Glossy Coating 107
- Matte 109
- Matte transluceent 111
- Matte rough 113
- Metal 114
- Mirror 117
- Mix 119
- Null 122
- Velvet 124

Properties
Input
- Float (Const_float) 125
- Float 3 (Const_float3) 125
- Grey colour 125
- Grey value 125
- RGB colour 126
- RGB value 126

Light emission
- Blackbody 126
- Irregulardata 127
- Lampspectrum 127

Map
- Image Map 130
- Normal Map 133
- 2d uv mapping 133
- 3d uv mapping 134
- 3d uv global setting 134

Math and Operations
- Add 136
- Subtract 137
- Divide 137
- Scale 137
- Abs 138
- Power 138
- Less than 138
- Greater than 138

Math and Operations
Clamp 139
Mix 139
Makefoat3 140
Split float 3 140
Remap 140

Noise
Fbm 141
Marble 143
Wrinkled 144
Density grid 146

Procedural type
Brick 147
Dots 148
Checkerboard 2d 149
Checkerboard 3d 149

Procedural Blender
Blender maps
Blend 150
Clouds 151
Distorted noise 153
Magic 155
Marble 159
Musgrave 158
noise 160
Stucci 161
Wood 163
Voronoi 165

Optical
Fresnelapproxn 167
Fresnelapproxk 167
Fresnelcolour 167
Fresnelpreset 168
Fresnelsopra 168
Fresnelluxpop 168
Distort. 168

Other
Band 169
Bilerp 169
Colourdepth 169
Dot Product 170
HitpointAlpha 170
Hitpointcolor 170
Hitpointgrey 170
HSV 170
Objectidcolor 171
Objectidnormalised 171
Tryplaner 171
UV test 171
Windy 171

.
Appendix
Appendix 1. Vectors. 173
Appendix 2. Texture scale. 179
Appendix 3. IOR. 185
Appendix 4. Caustics. 187
Appendix 5. UV mapping tools. 189
Appendix 6. Colour codes. 191
Appendix 7. Seeding. 196
Appendix 8. Volumetric effects. 199

In the following pages. The following colour codes are used.

Bold Blue.	Main topic headings
Bold Red,	Sub topic headings.
Plain Blue.	Luxcore help file and Luxcore forum posts.
Plain Red.	TurboLux help file.
Bold black	property headings
Plain black	My notes, and experimentation.

Most items have been checked through experiments. However some stuff, I have no idea what it does. This is partly because I have found very little on the internet to show its use. Or , as it doesn't seem logical within TurboLux. Hitpointalpha for example.

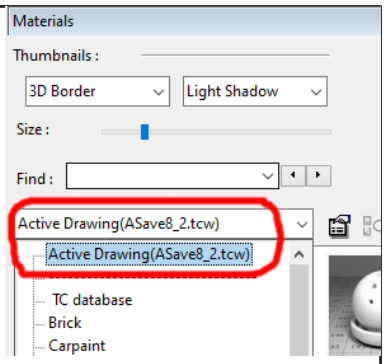
This document is NOT meant as a tutorial.

The most important point to remember, is that most parameters are selected via drop down boxes, allowing then allows access to individual properties, like imagemap, float, RGB colour etc.

On many occasions It is totally acceptable to change the default drop down selection. Often to float or const_float. This is because some properties (IOR for example), are simply a number input, and do not need a colour input, like grey colour or rgb colour confusing things..

NOTE. There will be a mixture of English and American spellings. Especially where Information from the Luxcore help file has been included, (the plain blue text). Luxcore and TurboCAD help is American, and has just been copy - pasted. I'm from the UK, so the black text is in UK English, except when reproducing TurboLux names.

Rendering Points to remember.



Active drawing

When a material or luminance is applied from the database. TC duplicates the material / luminance in the active drawing section.

Any alterations to material or luminance in a drawing should be done in this section. Changing the equivalent material in the TC database, will not be reflected in the drawing. And can affect future drawings.

Black screen.

TurboLux likes perspective view. Therefore, if you set up a standard view and then render with Lux in perspective (without presetting perspective view), the view will change. And often one will end up with the view inside an object. This can result in a black screen. Which may seem like a bug, but its just the perspective view needs setting before rendering.

This can be especially confusing, if one has chosen to tick 'do not show again' in the perspective warning message tick box.

Black screen can also happen instead of purple screen below.

Black screen or Purple picture frame instead of render.

This can occur in black or purple screen.. Usually because there is no turbolux scene lighting, or no manual luminance or HDRI set up.

However it can occur if one has no Lux scene set up. But instead set up luminance or emission from one or more objects. And then one accidentally selects a non-luminance/ emission object, whilst rendering the scene. In this instance Lux will attempt to render just the selected object/s. But with no light or emission attached, it cannot render properly, so displays the picture frame as a king of error message.

Black screen could be displayed instead of purple. But if Black, check the perspective view is not inside an object first.

White screen.

By default. TurboLux scene (options menu - drawing setup), may be too bright for the drawing. In this instance the screen can be white, simply because of the bright sky / sun. Check the scene lighting. The Sun/Sky gain may need turning right down, eg. 0.00002.

Environment maps will generally be between 0.4 and 0.8 gain. Much larger than Sun or Sky. But it does depend on the environment map used.
Also check any luminance power, especially if multiple luminance have been added.

Odd coloured background.

Can be due to auto-tone mapping. It can change the background when re calculating the overall tone.

Animation.

TurboLux may not be suitable for animation (at the time of writing). The first problem is CPU usage. TurboLux ignores what is set in options menu - preferences - multi threading. And can crank all cpu cores up to 100%. For a long animation, the CPU could get pretty hot without excellent cooling. Not as big a deal if using OCL mode. (Turbolux engine). However even with OCL, Turbolux can still use the CPU for some operations.

The second problem is time. TurboLux needs time to reduce the noise level to a reasonable level. Even when using the denoiser, some lighting setups requires quite a lot of time.
In Lightworks and RedSDK, Draft render is often all that is needed for animation. TurboLux draft can be noisy, as the max time allowed is 10 seconds, And thus producing way too much noise to be usable.

Not forgetting, an AVI movie can be 25 frames per second. So a longer time to render, can result in a very long time to animate.

Inserted luminance not working.

This can happen if one accidentally turns off Use Luminance', in the TurboCAD scene settings. (Options menu - drawing setup - turbolux scene.). Or one has unticked 'Enabled' in the luminance property.

Some luminance, required the correct direction to be set. Which aligns with the objects entity coordinate system. Setting the direction to the world coordinate system, can make the luminance shine the wrong direction.

Also if Position in set inside a solid colour object, it will not show any light. Some luminance require the position setting to at least the surface of the object (in metres)..

Orange render of an object. (when it should not be orange).

This can occur when a specific set or render parameters, cannot be rendered. For example, if one chooses imagemap as kd, from a material drop down menu. Applies the uv scale factor, but forgets to locate and apply an actual image from the hard drive.

Another example, if a material parameter, requires input within a specific range. And one inputs an incorrect figure. Many times this will be handled internally, and auto reset to within range. But occasionally, if it cannot be reset. It will render orange.

Small light or view alterations, produces a large change in lighting.

This can happen if Automatic Linear is selected in TurboLux engine. There is nothing wrong with using Auto Linear. But because it automatically adjusts the tone depending on the average tone of the current view. Changing the view, lighting or drawing, can change the tone quite a lot.

If this becomes a problem, switch to Linear, and alter the lighting and tone as necessary.

Poor draft render.

At present, TurboLux is poor in this respect, Especially if one is used to Luxcore in Blender. Lightworks or RedSDK. I do not know why. But it is possible Denoise has not been linked into draft render.

With other render engines, they appear to put the scene into memory, therefore moving the view simply has to address a different part of memory. TurboLux appears to have to redraw the scene, with every movement, making it a lot slower and poor quality.

If the render is particularly blocky, and one cannot make out objects when draft rendered, Pressing F5 (Redraw) will often clean up the render, it will still be grainy, but should be much cleared, and allow objects to show up better.

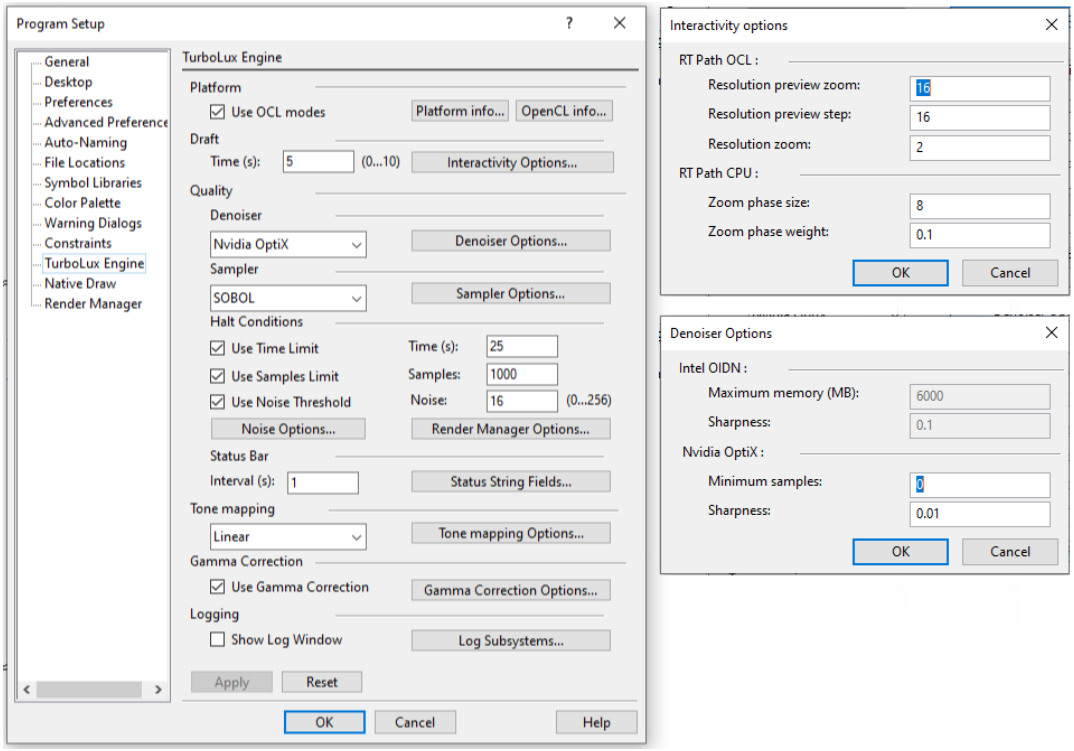
Setup.

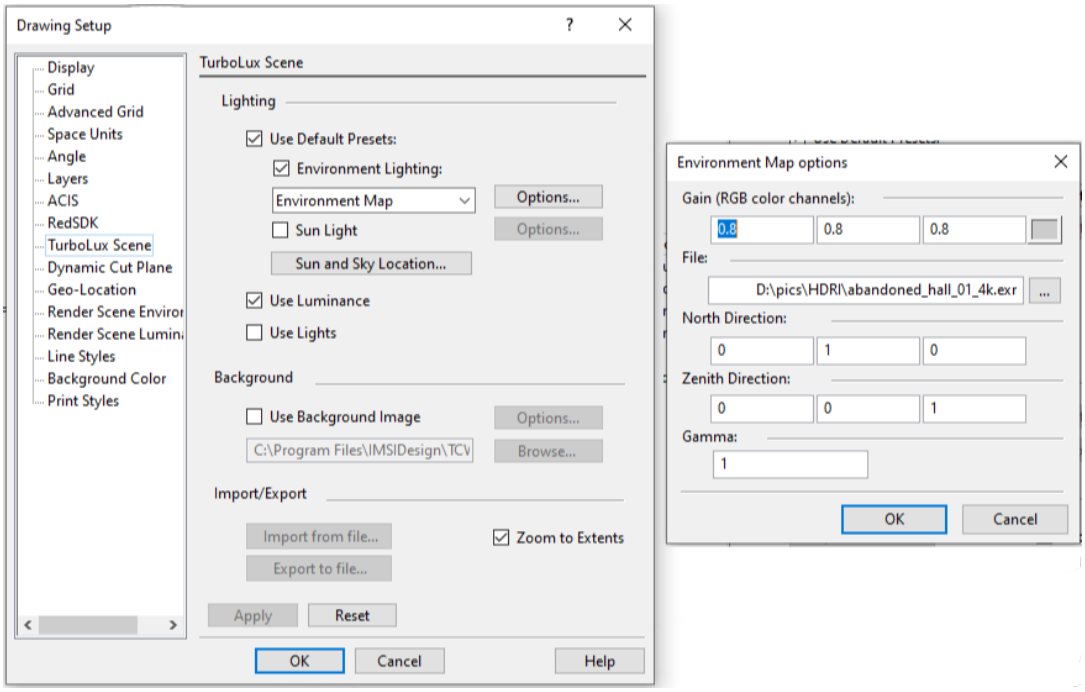
The following few pages show the Turbolux engine and scene settings.

Most settings are personal preference, apart from CPU / OCL which depends on ones computer system / graphics card.

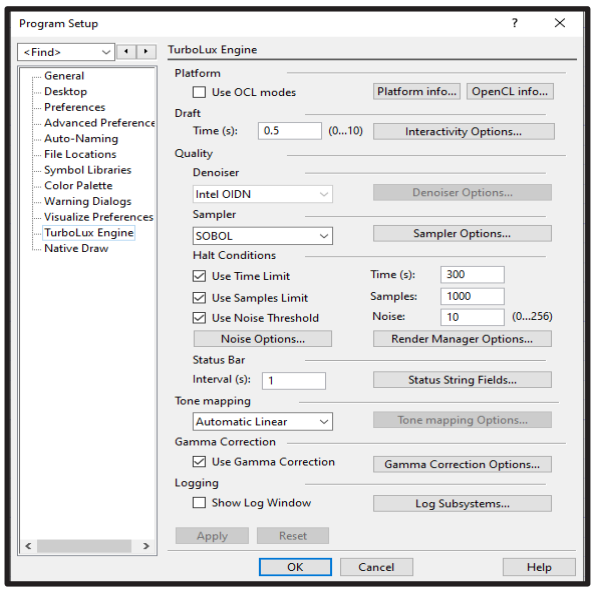
These are my normal test settings using OCL and an Nvidia card. For more quality rendering (less noise), I simply change the halt settings.

Different people will have their own favourite settings. The settings are shown here simple as an example of what I use. And what I consider a starting point for ones alterations.





TurboLux Engine.



The Turbolux engine requires setting up for one own personal preferences, and that of the computer it is used on. Incorrect setting may slow down the render, or not render at all.

Platform.

Use OCL mode. Tick box,

This tells the program whether you wish to use OpenCL (Open computing language), on the graphics card or to render on the CPU.

Whether to use OCL depends on ones computer system. It generally boils down to what works best on ones own system. Leaving it un-checked will tell the program to use CPU only.

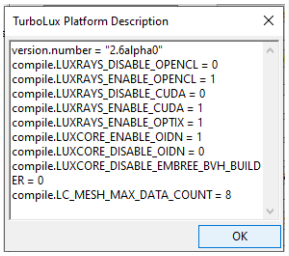
All rendering scenes will use the CPU even if set to OCL. At a minimum, it needs to format the rendering data in order to start the rendering process. If the scene is set to use caustics, these are generally rendered on the CPU.

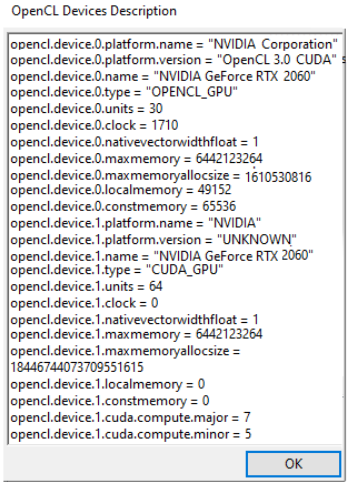
Do some test renders with it turned on and off, to determine the best for your computer. It is suggested by IMSI that the graphics card drivers are up to date.

Platform Info.
OpenCL Info.

Shows information about the current Lux configuration, and / or OpenCL and Cuda if applicable.

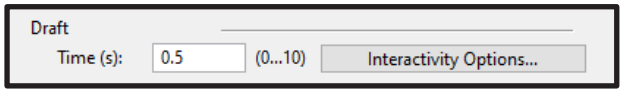
All this is for information only, and allows one to check the correct graphics card setup is being used.





Draft.

Time (s). The maximum length of time the render should spend on creating a draft render. However, no matter the time, the render will likely have noise in the scene.

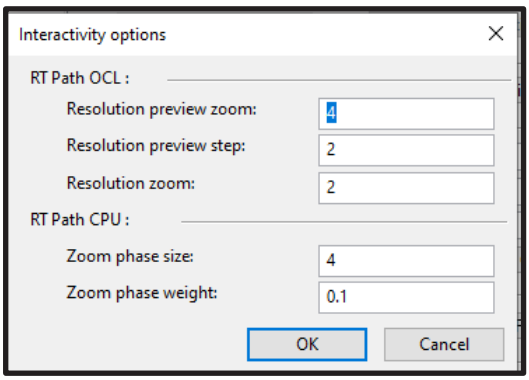


Increasing the time, will result in cleaner render, but a more jerky movement when changing the view. For example when moving the mouse whilst using the wheel click.

A quirk of some TC versions is the need to refresh the screen. If the render results in a very noisy view, where the objects are difficult to make out. Pressing F5 can clean the view up a little. It will still contain noise, but the objects will be more easily visible.

Draft render does not appear to use the denoiser, Therefore even at the max 10 seconds, the render will contain noise.

Draft Interactivity options button.
From the help file



'RT Path OCL'..

Resolution preview zoom Description: Render a sample every n x n pixels in the first passes. For instance 4x4 then 2x2 and then always 1x1. Recommended Range: [1 – 32] By default: 4. Higher numbers can aid view movement, by not fully rendering per frame. i.e. it renders the start with more noise, and moves the view without fully rendering the scene. It results in more noise while moving, but renders fully when the view stops moving.

Resolution preview step Description: Each preview step is rendered for n frames Recommended Range: [1 – 32] By default: 2. Higher numbers can make smoother movement, by not fully rendering each frame of movement.

Resolution zoom Description: Render a sample every n x n pixels, outside the preview phase, in order to reduce the per frame rendering time Recommended Range: [1 – 32] By default: 2. Higher numbers = more noise, with improved view movement, but the noise in the final render, is not worth any view advantages. Best left at 2 or set to 1

For a reasonable view movement / rotation, one could try 16, 16, 2. (The setting I use), This will be more noisy whilst rotating the view, but the view movement will be smoother. When one stops rotating the view, the draft will complete properly.

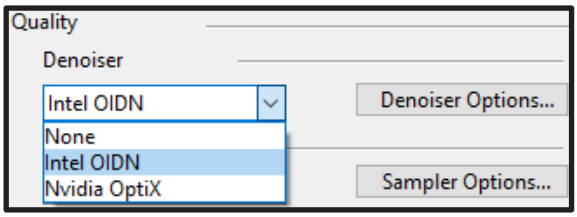
'RT Path CPU'..

Zoom phase size Description: render scaled down film for the first few frames after a scene edit (to make interactions more fluid) Recommended Range: [1 – 32] By default: 4

Zoom phase weight Description: TODO (description not provided by the LuxCore render developer) By default: 0.1

Increasing phase size to something like 8 or even 16 can produce smoother view movement, but it can also produce more noise.
Increasing weight can speed up view movement, but it will introduce more noise in the scene. Decreasing to 0.01 or lower can produce reasonable results. (I use 8 and 0.01)

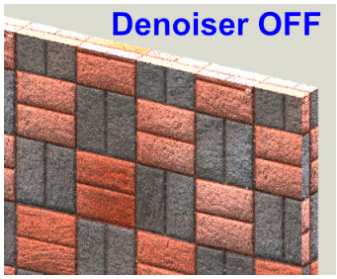
Quality - Denoiser.

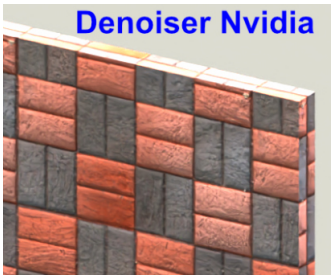


Allows one to set which denoiser Turbolux will use. Unless you have an Nvidia card with installed OptiX, then the only option will be Intel OIDN. Plus if rendering CPU only (OpenCL not ticked). Then it cannot use OptiX

As the name suggests, it attempts to remove excess noise in the rendered image, Whilst at the same time preserving detail. Noise is generally pixels which don't match surrounding pixels, and can be as simple as random white pixels on a black background.

The images right, show a difference between denoise at None (off) and Nvidia (ON)

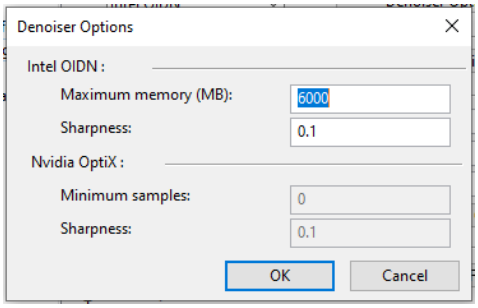




Intel OIDN (OpenImageDenoise). Although from Intel, it will work with Intel, Nvidia and AMD processors. **From the OIDN website** "The purpose of Intel Open Image Denoise is to provide an open, high-quality, efficient, and easy-to-use denoise library that allows one to significantly reduce rendering times in ray tracing based rendering applications. It filters out the Monte Carlo noise inherent to stochastic ray tracing methods like path tracing, reducing the amount of necessary samples per pixel by even multiple orders of magnitude (depending on the desired closeness to the ground truth)"

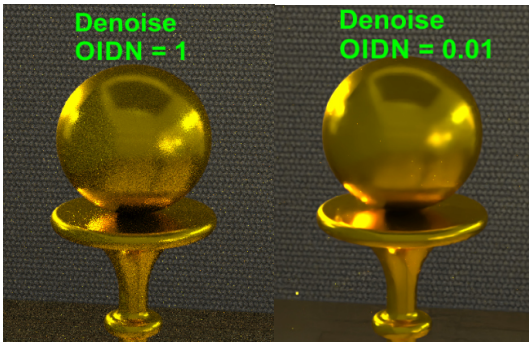
Nvidia OptiX. The obvious drawback is having to have an Nvidia graphics chipset, with OptiX installed. **From the Nvidia website** "An application framework for achieving optimal ray tracing performance on the GPU. It provides a simple, recursive, and flexible pipeline for accelerating ray tracing algorithms. Bring the power of NVIDIA GPUs to your ray tracing applications with programmable intersection, ray generation, and shading."

Denoiser options. **From the help file.**



Intel OIDN.
Maximum memory (MB) Description: Approximate maximum amount of memory to use in megabytes (actual memory usage may be higher). Limiting memory usage may cause slower denoise due to internally splitting the image into overlapping tiles. By default: 6000 Enabled: if Denoise = Intel OIDN

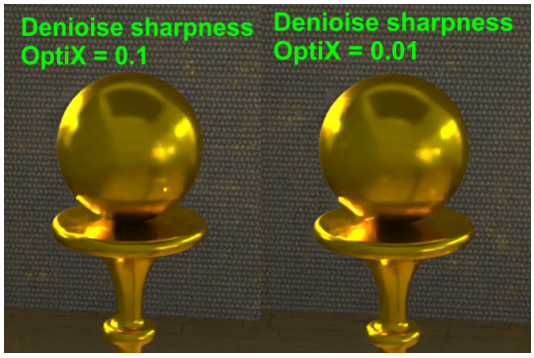
Sharpness Description: Linear interpolation between the denoise result (0.0) and the noisy input (1.0). By default: 0.1 Enabled: if Denoise = Intel OIDN



Lower values can produce better results.

On my computer, OptiX does a better job at removing noise, BUT. It can blend areas that should remain sharp.

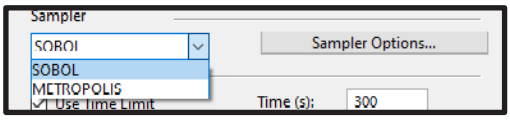
You need to do your own tests on what is best for your computer system.



Nvidia OptiX.
Minimum samples Description: Minimum amount of samples to be rendered before the denoise is run (at lower sample numbers, it does not modify the input). By default: 0 Enabled: if Denoise = NVidia OptiX

Sharpness Description: Linear interpolation between the denoise result (0.0) and the noisy input (1.0). By default: 0.1 Enabled: if Denoise = NVidia OptiX

Sampler.



Sets the TLux sampler one wishes to use.

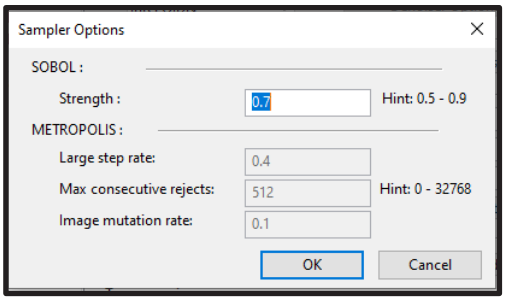
From the Luxcore manual….
Sobol*… Random sampler with an improved noise pattern. Supports adaptive sampling to spend more samples on noisy areas of the image.*

A progressive quasi-random "dumb" sampler. It distributes samples in an even but random fashion, using a sobol sequence.

Sobol is the best choice when you want a "dumb" sampler, such as simple scenes. It also offers excellent performance with GPU rendering, where metropolis can be slow. Sobol offers a continuous even distribution of pixels, regardless of how many samples per pixel are used, and can be stopped at any time. It is based on Blender Cycles' sobol sampler.

Metropolis*… Sampler that spends more samples on bright areas of the image, thus rendering caustics much better than the other samplers. The main disadvantage, especially when rendering on the GPU, is the higher RAM usage than the other samplers. It is not recommended to use this sampler for rendering on GPUs.*

Sampler Options. From the help files.
SOBOL..
Strength Description: it sets if the sampler should be more or less adaptive. By default: 0.7 Range: [0 ≤ x < 0.95] Hint: [0.5 - 0.9] Enabled: if Sampler = SOBOL

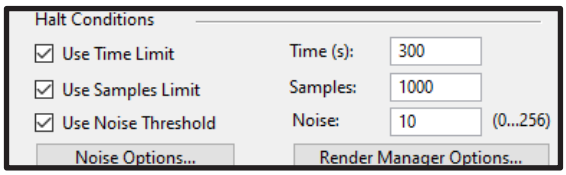


METROPOLIS..
Large step rate Description: Probability of generating a large sample mutation By default: 0.4 Range: [0 ≤ x ≤ 1] Hint: [0 - 1] Enabled: if Sampler = METROPOLIS

Max consecutive rejects Description: Number of consecutive rejects before a next mutation is forced. Low values can cause bias By default: 512 Range: [x ≥ 0] Hint: [0 - 32768] Enabled: if Sampler = METROPOLIS

Image mutation rate Description: Maximum distance over the image plane for a small mutation By default: 0.1 Range: [0 ≤ x ≤ 1] Hint: [0 - 1] Enabled: if Sampler = METROPOLIS

Halt condition.



TLux has three halt conditions to stop the render automatically. It is advisable to have at least one of them ticked.

Use time limit (tickbox). If ticked, TLux will use the time shown in **Time (s)** to stop the render. The time is in seconds and may be altered by the user. If just testing how it looks, the limit can be low. Down to 20 to 30 seconds. But for a high quality render, a high setting may be desirable, (maybe 600 or above). Stopping the render too soon, may leave too much noise in the scene. Although the denoise will help, increasing time, is likely to produce better quality finish.

For some renders, like a high quality fully furnished room, with multiple lighting. It may be preferable, to turn off the time limit, rather than guess at the amount needed. And use samples / noise limit, or manually escape the render when satisfied with the results.

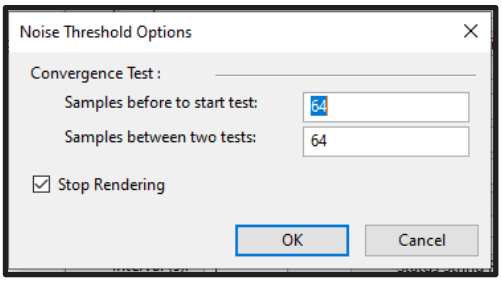
Use samples limit. If ticked, Tlux will use the **Sampling** input box. This is the threshold for sampling And is the average number of samples per pixel, Setting a low number for the sample limit, will likely produce a noisy render.

Use noise threshold. If ticked, Tlux will use the **Noise** input box. This tells the render, how much noise is acceptable in a render, and stops the render if the amount of noise in a scene is below this threshold. It decides this by comparing one set of samples to a previous set. Setting this too low, will increase render times unless a time or samples limit is also set. This could be set higher if Denoise is used.

If time and / or samples are turned on, they will likely reach their limit before noise threshold is reached, and stop the render.

If time and / or samples are turned off, only the noise limit will control the render. Setting this too low, for example 5 can produce a very long render time. Setting noise to 16 or 32 may be acceptable. At low noise levels, GPU render will generally give a faster render than CPU.

Noise Options button.

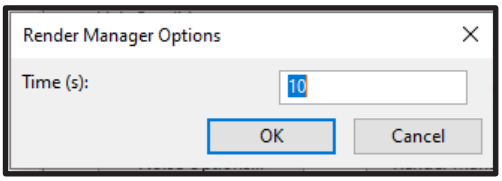


Noise threshold enables image convergence test, and stops when the error is under the defined value. Use a value like "n / 256.0" where n is the level of error you are looking for. It may be hard to achieve a level of error smaller than 3.0 / 256.0. -1.0 disables the halt condition.

Samples before.
Number of average samples per pixel before to start the convergence test

Samples between.
Number of average samples per pixel between two tests

Render manager Options button.

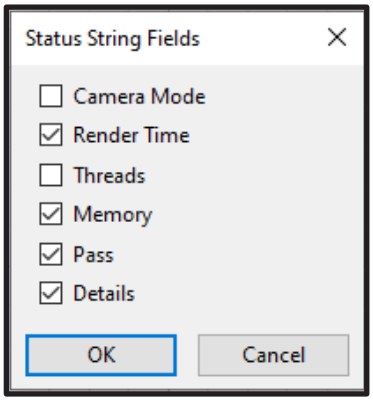


Time (s). Number of seconds before stopping the rendering in Render Manager. By default: 10 Enabled: Always.

Status Bar.

Information shown on the status bar (bottom left of TC) during the rendering process.

Status bar intervals (S).
Intervals. The refresh time of the fields in seconds.



Status string fields Button.
These are just information fields, they do not affect the render.
The dialog allows you to select status string fields.

Camera Mode Description: Show camera mode. By default: 'off' Enabled: Always

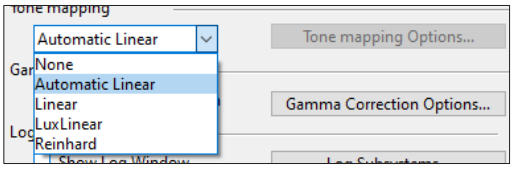
Render Time Description: Show rendering time. By default: 'on' Enabled: Always Threads Description: Show threads count. By default: 'off' Enabled: Always

Memory Description: Show used memory size. By default: 'on' Enabled: Always

Pass Description: Show passes count. By default: ‘on’ Enabled: Always

Details Description: Show additional fields (fields in square brackets). By default: ‘on’ Enabled: Always

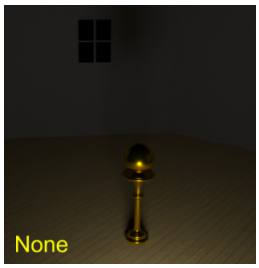
Tone mapping.



The drop down box allows for the setting of the desired tone mapping, or turn it off.

NOTE all the following tone map examples use the same drawing. With only a light emission object just out of view, at the bottom of the drawing..

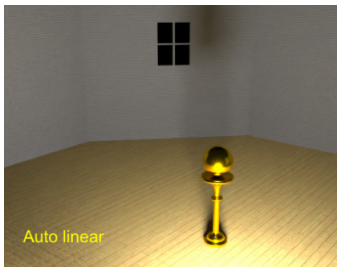
None



Turns off tone mapping, therefore the render simply relies on the user lighting setup.

This can be useful if one does not want any tone mapping to affect. and possibly alter ones own light setup. Especially if using a HDRI environment map.

Autolinear Tonemapper



Similar to the linear tone mapper, but estimates the average brightness in the scene to compute the scale value automatically. Automatic, adapts to the image brightness. (This can lead to confusing effects).

Can make darker scenes lighter, which may be undesirable in some situation. It can also sometimes affect the background light.

Linear Tonemapper

Simply multiplies all pixels in the image with the scale value.
Not automatic (does not adapt to the image brightness, the scale value has to be set manually).

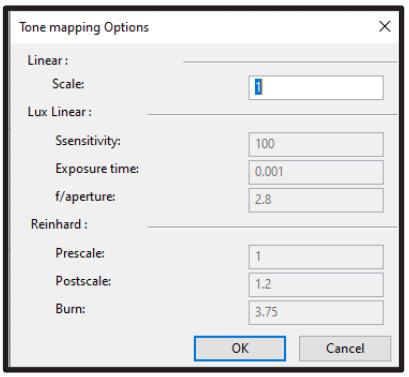
LuxLinear Tonemapper

Similar to the linear tone mapper, but the scale value is derived from ISO sensitivity, exposure time and aperture. Can be used to match footage taken with a real camera. **Not automatic**.

Reinhard02 Tonemapper

An implementation of the reinhard tone mapper: **Automatic**.

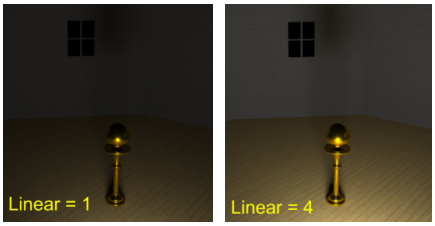
Tone mapping options.



Used to alter the properties of the tone map applied to the drawing.

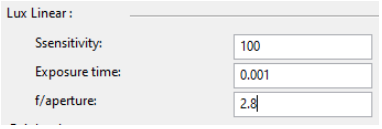
NOTE. Reminder. A scene was chosen, as it hopefully shows the effect better than a bright scene. It contains only one object emission light, no scene lights.

Linear.



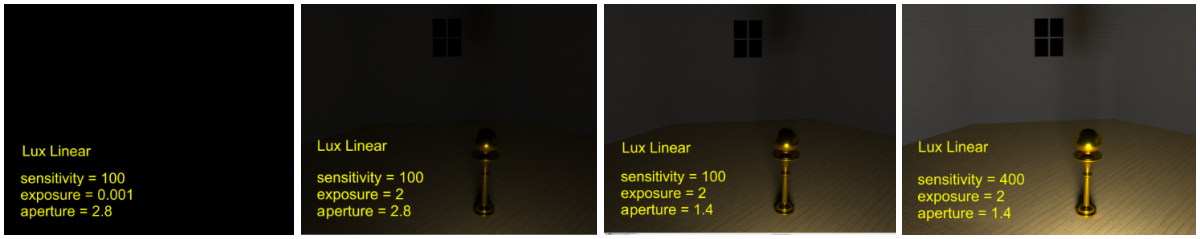
Scale. **Default 1**
Adjusts the multiplication factor for the linear tone mapping. Increasing the linear scale can increase the brightness and contract of the render.

Lux Linear.



Sensitivity. **Default 100 ISO sensitivity**
Exposure time. **Default 0.001**
f/aperture. **Default 2.8**

Altering just one of these setting may make only minor difference, However changing multiple settings, as one would with a normal camera, can completely change the scene from the default.

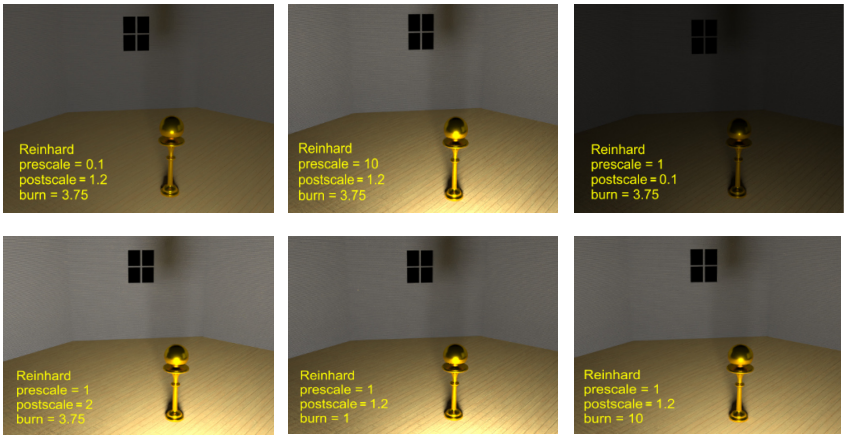


Reinhard.

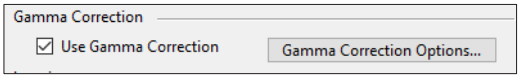
Prescale. **Default 1**
Postscale. **Default 1.2**
Burn. **Default 3.75**

Although Reinhard is listed as an automatic render. It still allows for some manual alteration.

This can be useful for setting dark scenes. If a test render is too dark. Rather than altering multiple luminance / lighting, tweaking the pre or post tone, may all that is required.

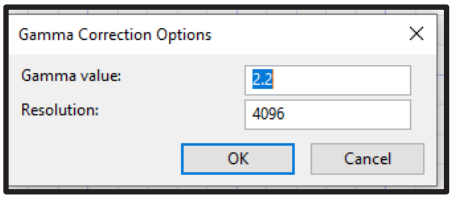


Gamma correction (Tickbox).



Tells the render whether to use gamma settings. Once ticked, the gamma correction options can be selected.

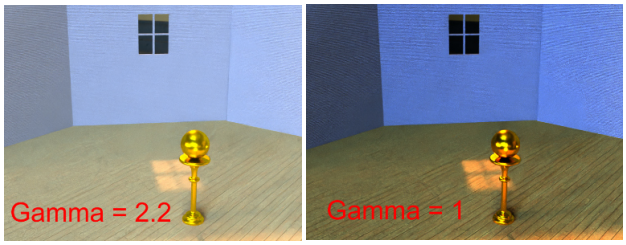
Note. This is a global setting for the render. Individual texture settings may have their own gamma setting.



Gamma correction Options.

Gamma Value. **Default 2.2**
Generally one would leave this as default.

Resolution. **Default 4096**
No idea.

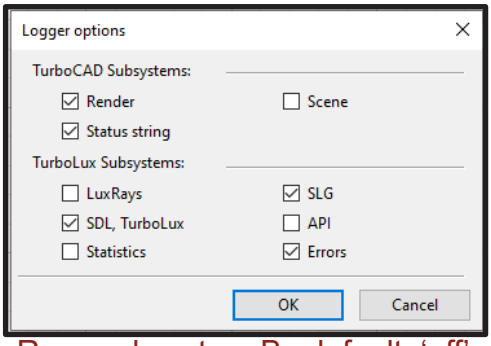


Logging. Log subsystems.

Log Subsystems The dialog allow you to enable/disable log messages of a single subsystem.

When rendering, a separate window will open, displaying in text form what is occurring per second.

Probably useful for testing and diagnostics. Not required for normal rendering activities in TurboCad.



Section 'TurboCAD Subsystems'
Render Description: enable/disable log messages of a TurboCAD render By default: 'on'

Scene Description: enable/disable log messages of a TurboCAD render scene By default: 'on'

Status string Description: enable/disable log messages of a TurboCAD render status string By default: 'off'

Section 'TurboLux Subsystems'
LuxRays Description: enable/disable log messages of LuxRays subsystem By default: 'off'

SDL, TurboLux Description: enable/disable log messages of SDL, LuxCore subsystems By default: 'on'

Statistics Description: enable/disable log messages of LuxCore Statistics subsystems By default: 'off'

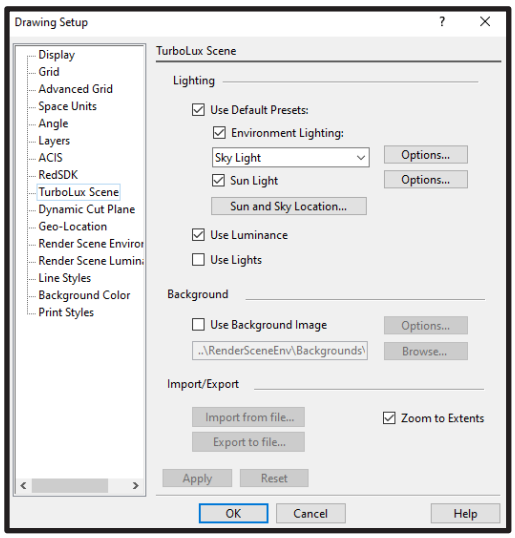
SLG Description: enable/disable log messages of LuxCore SLG subsystems By default: 'on'

API Description: enable/disable log messages of LuxCore API subsystems By default: 'off'

Errors Description: enable/disable log messages of LuxCore Errors By default: 'on'

Drawing Setup.

Options menu - TurboLux scene.



Here you can set the global scene lighting, which will be used for the render. It is possible to disable the settings and decide to use a completely manual approach, by adding ones own lighting / luminance.

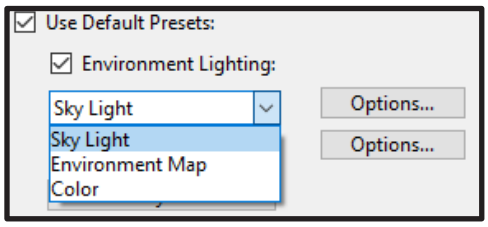
I read that for indoor scenes, (woodwork, metalwork, jewellery etc). It may be better to turn off sunlight.

Scene lighting - Use default presets.

Ticking this tells the program to use either Sky, environment and /or sun, which can be set up using the options buttons. The default 'out-of-the-box' scene, may not be to everyone's liking. (possibly not to anyone's liking). The scene is also affected by, TurboLux engine - tone mapping. Which can change how the background looks.

If not ticked, then manual luminance or emission must be used. If no luminance / emission. Then one will likely get either a black background, or purple picture error frame.

Environment Lighting.



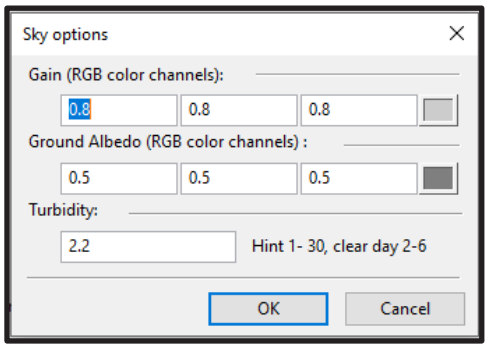
A drop down box allows this to be set at Sky, Environment color, or Environment map. If this is turned off (not ticked), then sun, a manual inserted luminance, TC lights or object material emission must be used,

Whether to use Environment (Global) lighting, depends on the scene. For a majority of occasions, some global lighting is an advantage, to give a balanced lighting for the whole scene. Then other lighting would be added for specific purposes. Including more control over shadows.

For some indoor scenes, mechanical components, and special effect. One can leave this off, and manually insert luminance or emission, for the desired results. Though using an environment map can be an advantage for reflective objects like chrome.

Sky.

The sky, is similar to the sky luminance, except that some settings are programmed in. After setting the Sky, using the drop down box. Click the options button to bring us the sky properties.



Gain.

This uses RGB colour values in the range of 0, 0, 0. (Black) To 1, 1, 1. (white). However Rarely would you want a pure white sky. As this can totally overpower a scene.

Keeping all the colours the same value is the same as using grey scale. The colour can be varied, but staying close to grey scale is often a good option.

One can think of sky gain as being the power of

the light, except instead of a single power setting, one supplies the power to each colour channel. In some scenes, small changes can make a big difference. So it is quite possible a scene may require a setting as low as 0.0002 or even 0.00002 for all three colour channels.

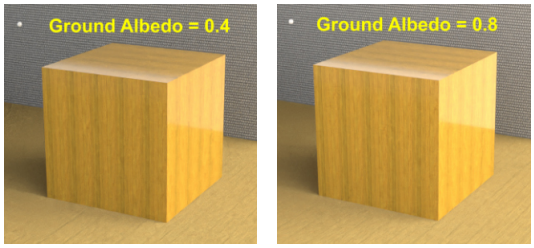
If the gain is set differently, then whole scene can change.

The three images below, show a normal sky, (first image). And two extremes, by setting the gain greater for the red and blue channels respectively.

Just like real life, there is no 'perfect setting'. A real sky can be bright blue, grey and overcast or night time.

Ground Albedo.

The light reflected from the ground. Again an RGB colour selection. For most needs this will probably be left with all values equal. As this is only ground reflection, the effect is not as dramatic as changing the gain. Being RGB, a setting of 1, 0 , 0 will give white objects a reddish tint.

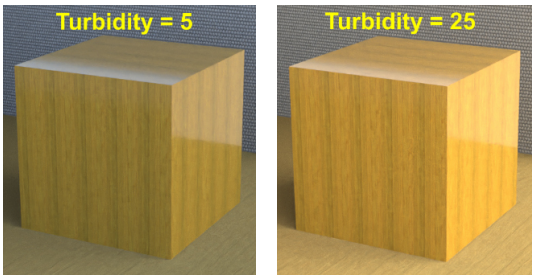


Turbidity.

This is the haziness on the sky. Or more precisely, particles of dust, moisture etc. present in the atmosphere. TLux will use the turbidity to determine how much light from the sky / sun hits the scene.

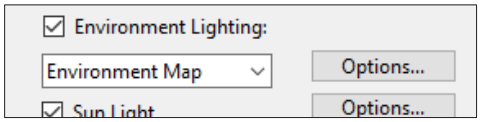
Normally. A lower turbidity means a clearer sky. and a higher turbidity means more haze. On my computer it appears to work the opposite. I suggest you do your own test, to determine which way round the turbidity works on your computer.

If setting the gain still produces a brighter scene than required, changing the turbidity may help.



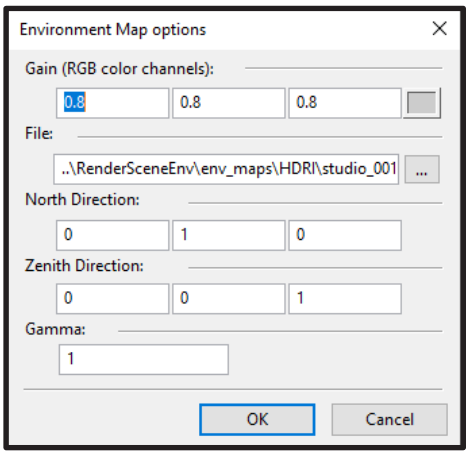
Environment Map

An environment map can be a a standard 2d image, or a HDRI (high dynamic range image). HDRI will generally give a much better quality. Depending on the HDRI used, some scenes with all round illumination from the HDRI, may be devoid of shadow. Therefore it is sometimes necessary to add sun or standard luminance in addition to an image or HDRI map, A Hi-res map will give the best quality. A 4k HDRI should be adequate. 8k may be overkill for most situations, A 2k image / HDRI would probably be the minimum resolution.



Although a HDRI is a light source. It can be used in conjunction with a sunlight. Or any other light, luminance or emission. Adding Sunlight may be especially useful for outdoor scenes, and light entering indoors through a window.

Environment Options

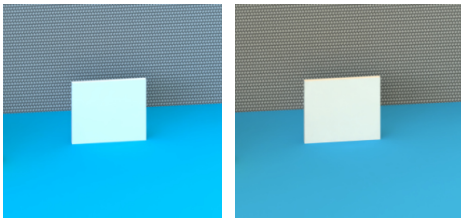


After setting Environment map in the drop down box, click the options button to set the map details.

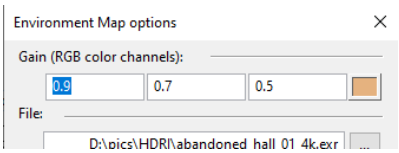
The Gain works similar to sky. larger numbers increase brightness of the image (standard or hdri) and thus the light power, lower numbers reduce the brightness and power.

Care should be taken not to make the values greatly different, unless one wishes a special effect.

The effects of changing the gain to mismatched colours, can be seen in the images on the right. The first image uses a standard gain of 0.8, 0.8, 0.8. This clearly shows the blue tint on the white box, and the grey back wall, caused because of a large blue base everything is say on.



To combat this tint, one can change the gain to affect the light given off from the HDRI. In this case we reduce the blue quite a bit, and tweak the red and green. i.e. 0.9, 0.7, 0.5.



The Blue material was altered slightly (not enough) to compensate for the reduced blue in the environment. The white and grey show up much better.

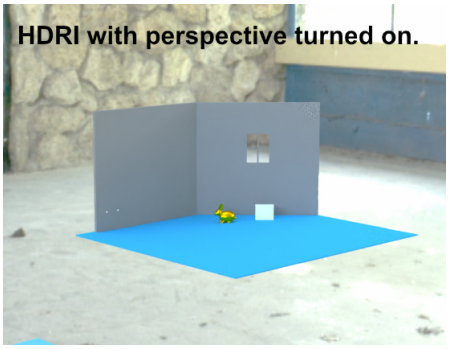
As stated previously, One would normally leave the figures all the same. i.e. 0.8, 0.8 0.8. Or 0.4, 0.4, 0.4

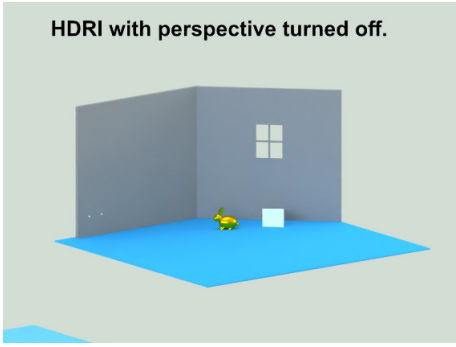
File. Click the three dots button, and navigate to the file being used. These can be hdr, exr, bmp, gif, jpg, jpeg, png, tif, tiff, pcx, tga.

The preferred format is to use HDR or EXR. (both are hi-dynamic range). In tests, I did not notice any visible difference in quality of the render between HDR and EXR.

If using a standard format (jpg, png etc.), it may be preferable to use a hi-res panorama. Low res jpg, can result in a blurry background, and seams. It may also be necessary to adjust the gamma.

Note.. When rendering, If perspective is turned off, the image will be used for lighting, but will not show up in the background, If perspective is turned on. The image will show up in the background. Though this can be overridden if 'Background' is ticked.

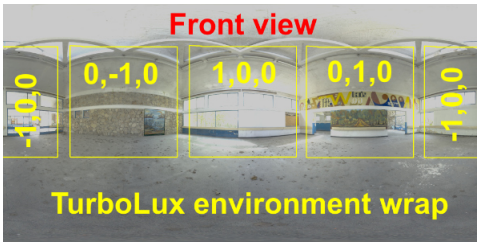




North direction. The north vector in x, y, z. Note these are vectors not degrees, And as such the standard range is -1, to 1. See appendix 1 for vectors.

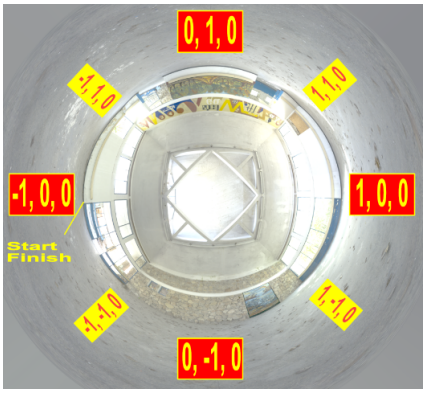
This allows one to change the centre location of the HDRI (Or start location, I don't know which). Allowing one to rotate the HDRI or image, and thus alters the way the light hits the scene. It also allows one to rotate the background, so a particular part of the background is in view of the camera. Providing TurboLux scene Background option is not used. Thus having a different view, without having to rotate the whole drawing.

TurboLux appears to wrap the environment sphere from the outside. Therefore when viewed from inside the sphere where the drawing is, the image is reversed.



The environment map will look different depending on the current view, and the north direction chosen. Although there are three input boxes, do not use the last 'Z' box. Leave it at '0', and use the first two, from -1 to 1, depending on what one requires.

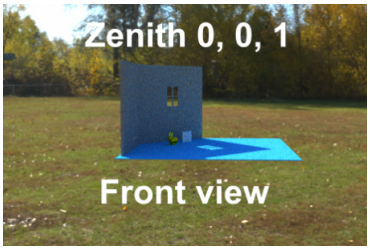
For a 45 degree view of the map, one could use, for example 1,1,0.



The image (right), shows the spherical wrap used for the environment map. And the vectors required for a particular part of the HDRI to show up in TC's Front view.

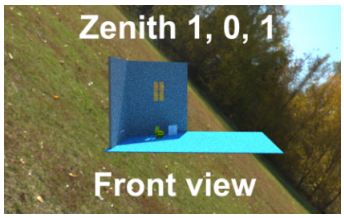
If the HDRI display being reversed is unacceptable, it may be necessary to flip the hdri in a paint program, prior to use in TurboLux.

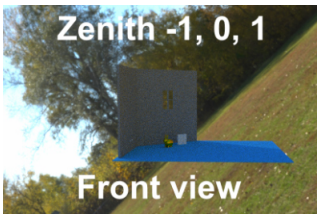
Zenith direction. (upwards).
Whilst North direction can be seen as horizontal rotation around the horizon, or the earths equator. Zenith sets the vector rotation from pole to pole,

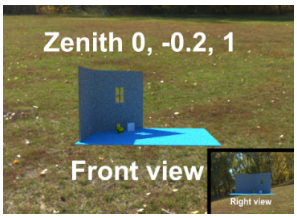


Again it is in the range -1 to 1. In general one would leave this is default, which is 0, 0, 1, this is essentially +Z straight up. One would not normally use minus figures for the Z vector, i.e. 0, 0, -1 as this would flip the hdri about the horizon (upside down)..

If for example, one wanted to to tilt the map downwards, to show more sky. One could use vectors of 0, 0.1, 1.



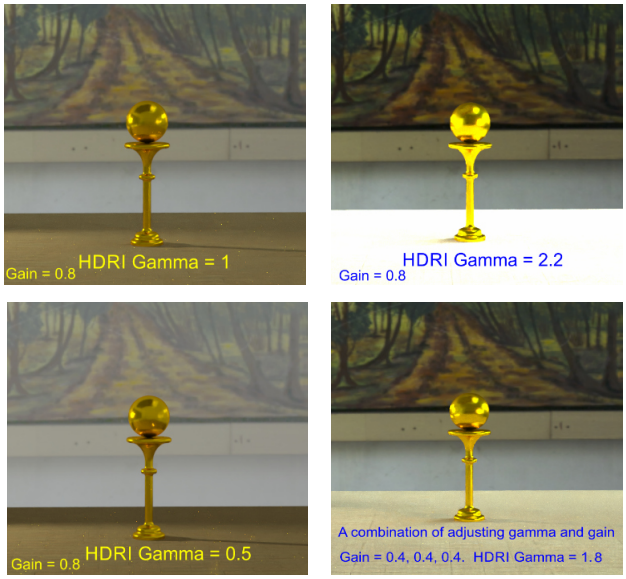




Gamma.
Adjusts the gamma of the image or HDRI. In general HDRI would have a gamma of 1. Whereas JPG's could have a gamma of 2.2 .

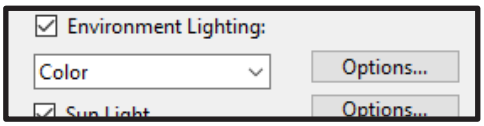
In the four images right, different gamma settings were used, The first three use the same gain. Adjusting the gamma alone, can produce unsatisfactory results. But by altering both the gamma and the gain, one can set the HDRI lighting to suit ones needs.

Like always, sun or additional luminance may be required, especially for producing any desired shadows.

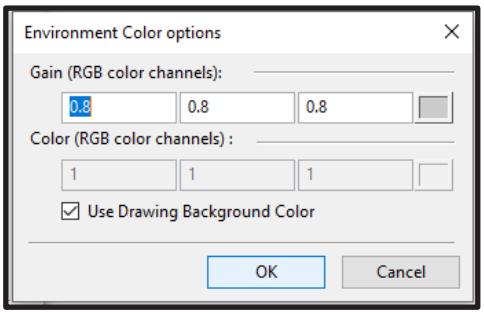


Environment Color

The environment colour Sets a plain colour as the background map. It has much simpler options, and includes an option to use drawing background colour.



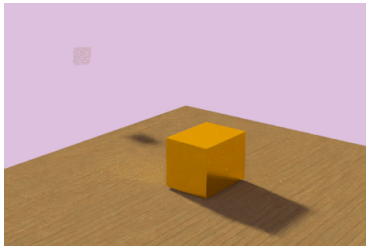
One sets the **gain**, just as with sky and environment map. And then sets the appropriate colour. Reduce the gain if the colour makes the scene too light.



Colour. These are RGB colour, in the range of 0 to 1. Greyed out is 'use drawing background' is ticked.

Generally one would avoid strong colours unless for a specific reason.

On its own, the scene can be devoid of shadow. A colour along with sun light, can give a scene the required shadows. Not ideal for outdoor scenes.



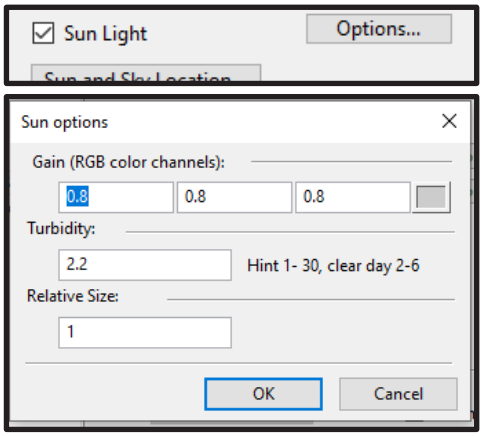
Note. One would probably need to turn off Auto Linear (or set it to something like linear), in Turbocad engine. If turned on, auto tone can change the background colour, often darkening it.

Use Drawing background colour (tickbox).

When ticked it uses the same colour as set for the wire frame background colour. (options menu - drawing setup - background).

This tries to ensures that, if one changes the drawing background, it will be reflected in the rendering. This can save altering the colour in multiple places. However. The colour is still affected by the gain. Therefore the colour may not match exactly, without some additional fiddling with the gain.

Sun.

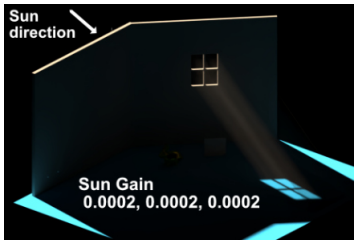


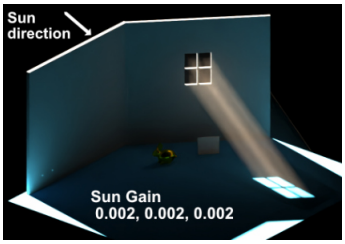
Ticking the sun. Allows it to be used as the scene illumination. The options box allow for adjustment of the sun parameters.

The controls are gain, turbidity and relative size. The sun and sky location vectors are set in a separate dialog box,

Sun is generally used with other illumination, like sky, or environment map. Rather than on its own.

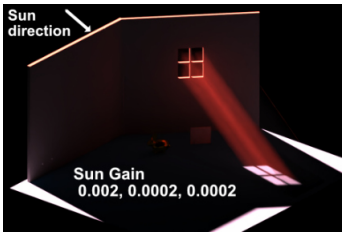
Gain.





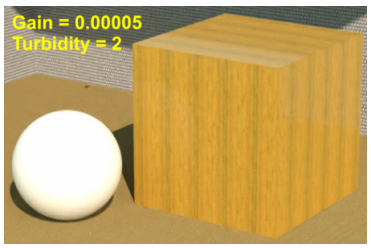
Like other gains, it increases or decreases the power of the sun, and is input per RGB channel, generally one would keep the channels the same figure, unless a specific colour tint is required.

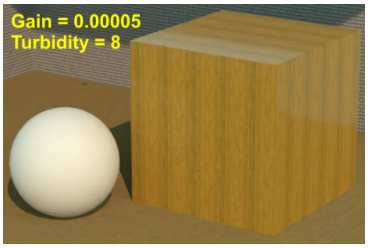
Gain for sun can sometimes be too powerful, and may require numbers as low as 0.0001.



Remember. These are RGB channels. Therefore setting different values, for example 0.01, 0,001, 0,001. Will give the sun a reddish tint, which may, or may not be desirable.

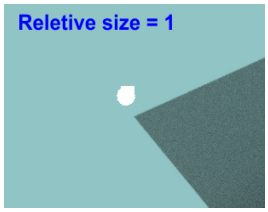
Turbidity.

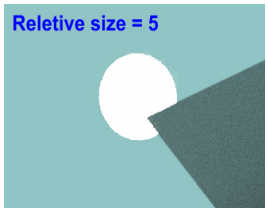




Increasing the turbidity, increases the haziness of the sun light, or more precisely, particles of dust, moisture etc. present in the atmosphere. This can lighten or darken the scene. (Unlike sky, this works the correct way round. Higher numbers = more turbidity).

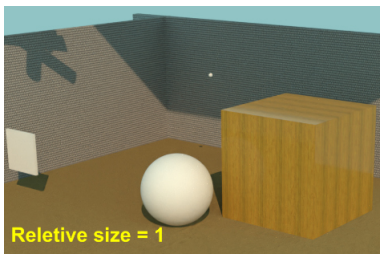
Relative size.

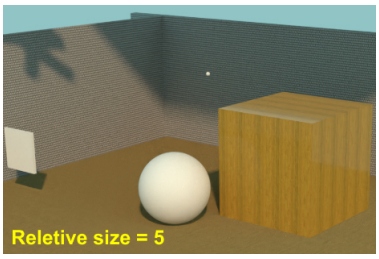


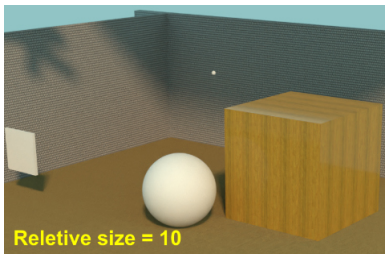


The sun appears as a white circle in the background, whether HDRI or colour. If it is not visible, rotating the view (not the objects) will often find it.

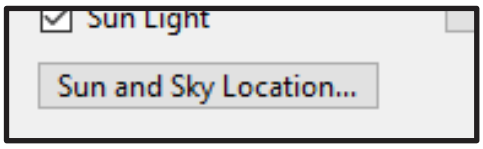
Increasing the relative size, makes the circle larger. Size does not affect the power of the sun light. However. This can affect the shadows created by the sun. A larger relative size will often produce softer shadow edges.



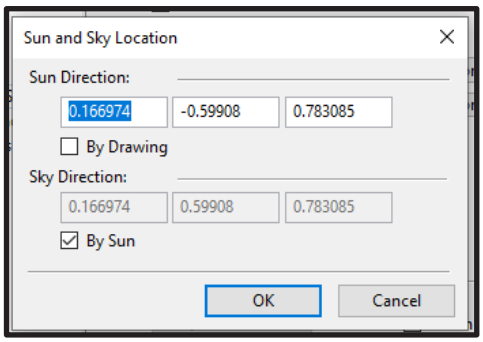




Sun and Sky Location



The Sun and Sky location button, brings up a box where one can enter the vector directions.

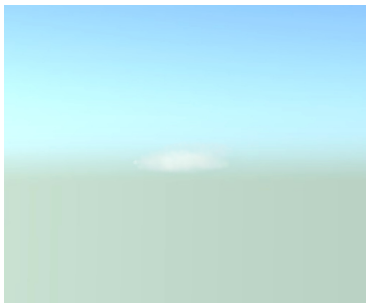


This is where one can change the direction of the light to suit ones drawing, rather than rotate the drawing to suit the sun position.

Sun direction

These are vector directions not degrees. Essentially they are the x, y, z direction, with 0, 0, 0 of the drawing extents. From -1, -1, -1. To 1, 1, 1. See appendix 1 for vectors.

Ticking the **'By Drawing'** tick box, allows the sun to use the information set in the drawings geo-location. Ensure the correct data has been set in geo location, and the drawing is at the correct orientation (north direction), if using this option.



Sky Direction.

The sky is similar to the image on the right, The 'whitish' part I added to show the bright part depicted in the Turbolux scene, and is the vector needed in sky location.

How much the white part affects the scene is controlled via the gain in sky options. A high gain will brighten the image so much that the whole scene becomes white.

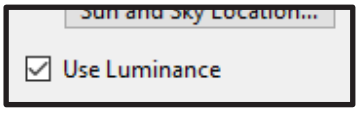
The default direction is x = -1, y = 0. Which in Default TC terms, means the light (and white part if the image) is coming from the left of the screen, If one change it to x = 1, y = 0, it would be coming from the right of the screen.

Although the sky will produce light and dark areas depending on the direction vector, it will not often create proper shadows, Therefore a sun, or luminance will likely be needed for shadow creation, if required.

The **'By Sun'** tick box ensures the sky and sun are aligned.

Although one can enter -0.5 for example in the the z box. This will place the sun below the drawing, which is not usually desirable.

Use Luminance tickbox.

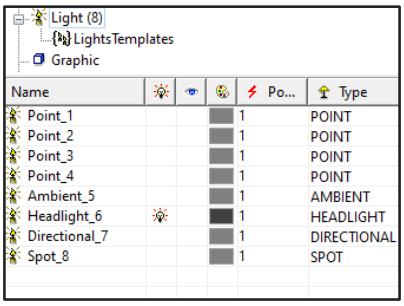


This enables or disables any luminance added to the drawing. Ticking it, tells the render that it is OK to use manually inserted luminance,. Such as Spot, Point, Sphere etc.

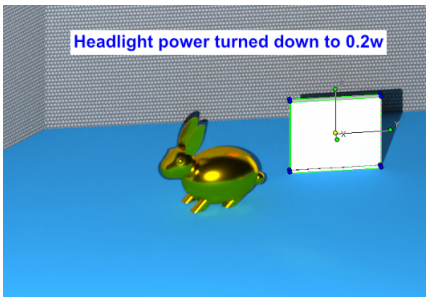
Un-ticking use luminance, turns off any manually inserted luminance. This can be handy if one has inserted multiple different types of luminance, One can turn off the luminance globally for testing and drawing / rendering different elements, and turn it on when needed to check whole scene.

Use lights. (from TC2022 - build 35.1)

This tells Turbolux to use standard TC lights. i.e. Point, Spot, Headlight Directional etc. However at the standard defaults, normal lights can be way too powerful, I suggest turning to almost black and work up from there. Headlight may need adjusting to something like 0.2 power

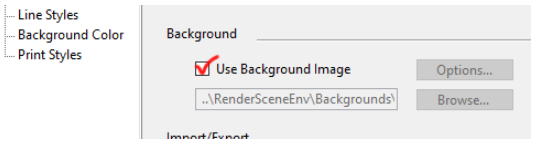


Even if the colour is near black, at default 1w, the headlight is still way too powerful. Turning down to 0.2w solved the problem.



Generally, Luxcore advises not to use standard lights, as they are not realistic. but, they are a lot easier to set up than luminance. And can help with any awkward dark spots.

Background. (from TC2022 - build 48.1)



TurboLux Scene, background tick box.

One may wonder. Why it can be beneficial to have a separate background button, when one would normally use a HDRI environment.

HDRI are indeed a great way to add background luminance. However, for mechanical drawing, One may wish the advantage of HDRI lighting, but requires a plainer background or a gradient colour.

For plain backgrounds one could use an environment colour, But there is one drawback. And that is reflection of the background. This can tint or in extreme cases, wash out any object colour.

Another method would be to use a plain colour object, or an object with a material attached as a 'wall', This works well, but may need some fiddling to avoid unwanted shadows on the wall.

And there is still a possibility of reflecting the 'wall' in the object render.

This is where the background option comes in. This allows for a non-reflective image to be applied as a background, The image could be a picture, patterned, or simply a plain coloured image.

The drawback being that unlike an object or an environment image, the background image cannot be moved or rotated, therefore of you wish a specific part of the background to be behind the object, it would have to be set up in an image editor prior to use.

If moving or rotating a background is not a requirement, Which especially in plain backgrounds, gradients, or simple textures it will not pose a problem. Then applying a background using turbolux scene is a good option.

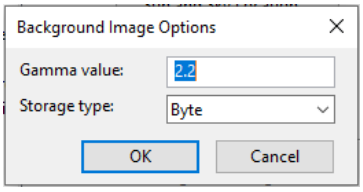
Application of the background can be done before or after setting the HDRI environment.

An advantage of setting the HDRI environment first. Is when surfaces will reflect the HDRI lighting. It is easier to see the effect of rotating the HDRI, if done when the background is not showing.

The two images right, differ only in the North direction of the HDRI environment.

And can be seen the reflections on the watch, differ depending on the orientation of the HDRI.

Background options. These are pretty simple.



First the storage type. At the present time. I have no clue what difference it makes, so I will leave it at byte.

Gamma value. For a png, jpg, tiff etc. standards image formats. The normal value is 2.2. However, when first rendered, if the background is too light, increase the gamma. If the background is too dark. Try reducing the gamma.

HDRI / exr images can also be used as a background as well as an Environment. However the HDRI applied as a background image, does not affect luminance. It is treated solely as an image. And in this instance, the gamma could be set at 2.2 or 1 depending on the HDRI.

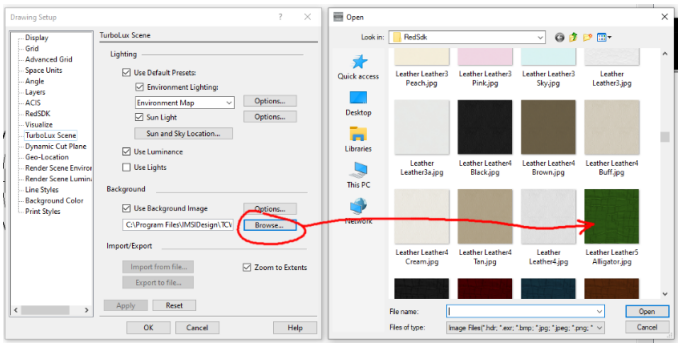
As can be see on the right. The environment was turned off, and the HDRI background used. The object is black because the background HDRI is not emitting light. It requires the environment for that.

Example usage.

In the C:\Program Files\IMSIDesign\TCW2022\Materials\Textures\RedSdk folder is an image Leather5 Alligator.jpg. And this one will be used here.

First we browse to the image file and select it in the dialog,

Enter background options and check the gamma setting, initial setting 2.2 for standard image format, or set to 1 if it is in HDRI type format.



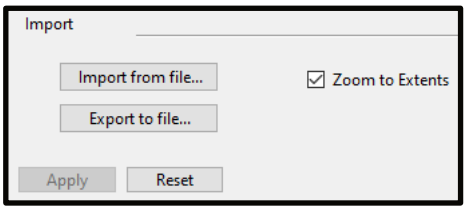
Set the desired environment as normal.

Draft render to check if the background gamma setting needs adjusting.

Quality render for final image. The image right shows the ring is totally unaffected by the background Image, and relies solely on the environment which has been set. Or any manually added luminance, if that is ones preference, instead of using an environment.

Final note. The above applies even if no environment is used. And instead one has manually set up the luminance. Like point, spot, sun etc.

Import/Export From the TC help files



The button "Import from File" is enabled if LuxCoreRender mode is set in the active window. Formats LXS, CFG and BCF are available for import now.

Basically draft render before import can be selected.

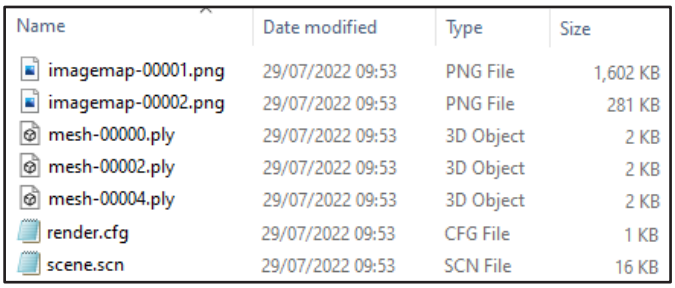
Please note: imported objects are **not** added to the drawing they are cosmetic only..

BCF is Bim collaboration format. CFG which is used by a lot of programs, and is simply a configuration file for the TurboLux setup LXS and SCNn is a scene file.

This allows one to import a previously saved scene into memory, So one can check a new objects placement, material, shadows etc, against a previously saved scene.

However, at the time of writing. The objects imported are for viewing only, they cannot be selected, nor edited. They are simply held in memory. And will vanish from memory when the work space is set back to wire frame, or hidden line. It is a bit like a temporary x-ref.

Export to file.



Name	Date modified	Type	Size
imagemap-00001.png	29/07/2022 09:53	PNG File	1,602 KB
imagemap-00002.png	29/07/2022 09:53	PNG File	281 KB
mesh-00000.ply	29/07/2022 09:53	3D Object	2 KB
mesh-00002.ply	29/07/2022 09:53	3D Object	2 KB
mesh-00004.ply	29/07/2022 09:53	3D Object	2 KB
render.cfg	29/07/2022 09:53	CFG File	1 KB
scene.scn	29/07/2022 09:53	SCN File	16 KB

Exporting a turbolux scene is automatic, one simply selects the desired folder (or create a new one) to save the scene into.

A scene is saved as multiple files. Including any image maps used, ply mesh data, and render configuration.

Imagemap, is any texture wrapping like wallpaper, floorboards etc. which were present when the file was saved.

Mesh .ply is a polygon mesh format of a 3d objects. These cannot be imported as a 3d object into TC using the normal file open dialog. But are imported into a scene file for viewing. As ply 3d objects, they can be opening in various programs, like Microsoft 3d builder, MS 3d viewer, Blender etc. (just not as a TC object).

Render.cfg. This is a text file of the turbolux engine settings, along with the scene file name to be used for importing.

Scene.scn. This is a text file of the scene settings. Including camera setup, lighting, textures / materials used, and which .ply objects to import.

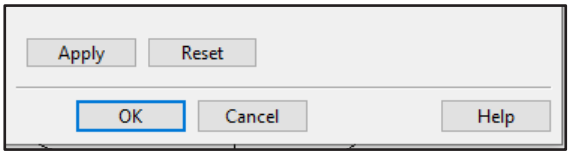
When importing one uses the cfg file, this loads the .scn file along with images and meshes for viewing purposes.

Zoom to Extents.

Description: if the option is enabled, the camera in the active window adjusts to the extents of imported objects. By default: On Enabled: Always

Apply/Reset

Apply Description: Apply user changes without having to close the property sheet dialog box. Enabled: If the values of the dialog parameters have been changed



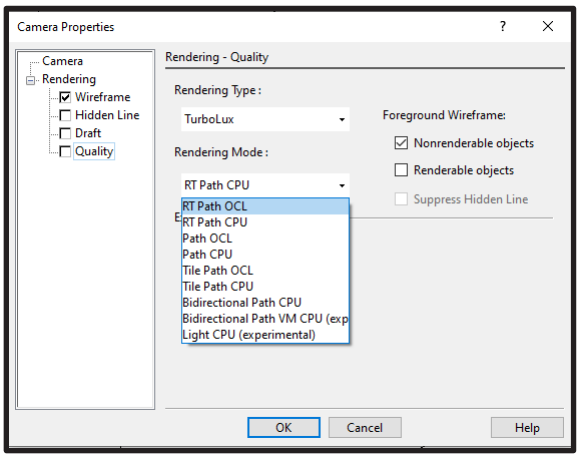
Apply is not required if changing properties whilst in wire frame mode.

It is useful in draft render, where clicking apply, will re-render the scene. And leave the scene dialog open, for further adjustments.

This saves having to keep open and closing the scene dialog box, when making minor adjustments.

Reset Description: Restore default settings. Enabled: If the values of the dialog parameters differ from the default values

Camera, rendering, properties.



Selecting camera properties, or right clicking the rendering teacup, allows one to access the rendering type.

After selecting TurboLux as the rendering type. The desired mode can be selected depending on ones personal preference. Note. The options available depend on whether one has set up CPU or GPU (openCL) as the rendering process, in the Options menu - TurboLux engine settings.

Draft render has only RT path, either CPU or GPU. Quality render has far more options.

RT path (CPU/OpenCL). is optimised for faster rendering, In some circumstances, it can allow the scene to regenerate faster than standard path rendering. It may produce a slightly reduced quality than the normal path render, but due to the denoiser, any reduced quality may be unnoticeable. .

Path (CPU/OpenCL). Is the main Luxcore path tracing engine. A Unidirectional path tracer that casts rays from the camera. Samples the whole film progressively. Supports all AOVs. (Arbitrary output values, google it, but you won't be manually creating any in TC)

Tiled Path (CPU/OpenCL). Unidirectional path tracer that is almost the same as Path, but uses a special sampler which iterates over the image in tiles (this leads to a slightly lower RAM usage). The OpenCL version adapts the number of tiles that are rendered at once, to the performance of the compute devices (GPUs/CPUs), so the tile size does not matter a lot (if rendering performance is bad due to small tiles, the number of rendered tiles is increased automatically). When the last tile of a pass is reached, it is split among the compute devices. Supports all AOVs.

Tile path, renders the screen in blocks. Note, if the time limit is too small, it may not render the whole screen in some builds of TC 2022.

Bidir (CPU only). Bidirectional path tracer that casts rays from both camera and light sources. Samples the whole film progressively. Supports only a subset of AOVs: RGB, RGBA, ALPHA, DEPTH and SAMPLECOUNT. It is recommended to combine this engine with the Metropolis sampler.

Bidir is recommended for better caustics. Which render on the CPU.

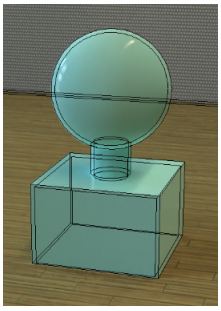
Bidir Path VM CPU (experimental)

VM (vertex merging). Couldn't find much out about this experimental render.

Light CPU (experimental)

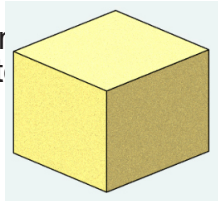
According to the developer it is a light tracing only rendering engine - for testing purposes. A simplified Bidir. I didn't really find out much more about it on the internet.

Non-renderable objects.



Turbolux may caution sometimes, regarding non renderable object (2d) when rendering. 2D will work in render view, though only displays the 2d when the render is complete.

When tested using v2022 and build 35.1. selecting renderable objects, and suppress hidden line, did not work. It rendered, then switched to hidden line, with no render showing. This worked ok in v2022. Build 48.1.



As stated. Non-renderable and renderable, will not be seen during the render It shows the 2D and wire frame once the render is complete. And works well t definition to the edges of a 3d object.

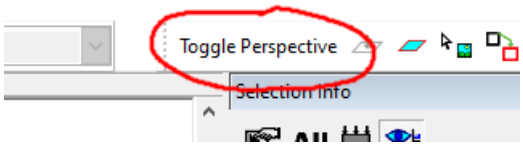
Perspective.

TurboLux will normally post a caution if rendering is selected, whilst camera is not in perspective mode. This warning can be turned off (do not show again ticked). However, until one is used to turning perspective on / off. It may be better to leave the warning active.

When switching back to wire frame from draft or quality, **it does not** turn perspective back off. This can cause problems if one forgets and starts adding further drawing. Perspective can make aligning objects, inaccurate, and if objects are drawn in wire frame, like a 2d line. It can be positions out in space, away from the drawing.

One would need to get in the habit of ensuring perspective is turned off when working in wire frame. And just turn it on prior to rendering.

I created a new toolbar and put the perspective switch in it, (along with some other tools). This just makes switching perspective very quick, and with no need to try and remember any shortcut.



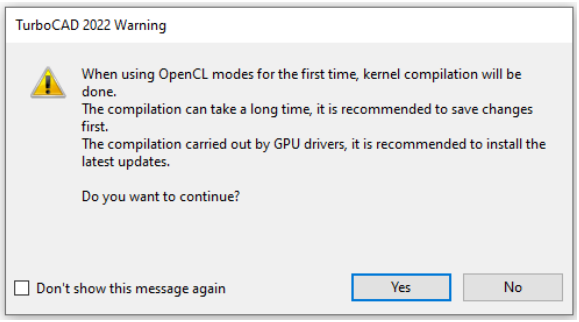
Graphic card Kernel.

When TurboLux is rendered for the first time, Using OCL mode. a message pops up, wanting to set up the graphics card. This will take a long time.

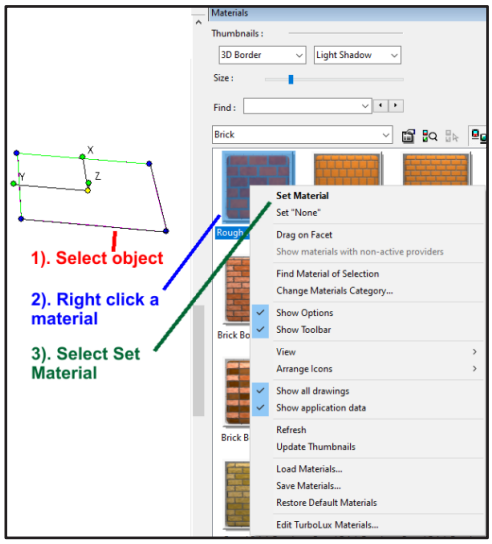
In theory one would think this is only needed once, unless one changes the graphics drivers.

However, For some reason. On my machine, TC will carry out this process at random intervals. I've no idea why.

It is a pain when it pops up, but you just have to live with it.



Material & Luminance palettes.



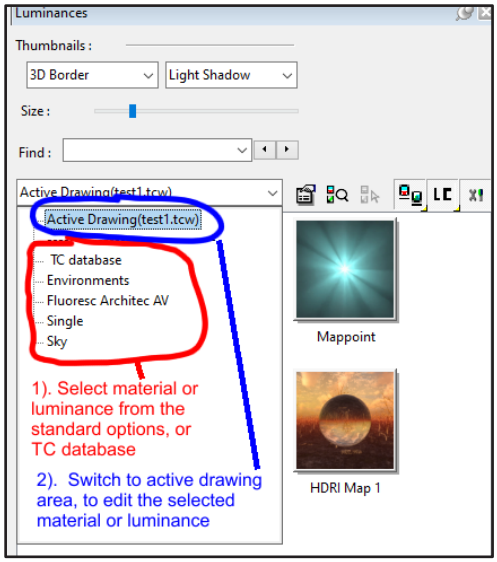
Materials and luminance can be applied in a number of ways. One way is the material or luminance palettes.

Open material or luminance palette.
Select an object.
Right click a material or luminance.
Select 'Set Material' from the right click menu.

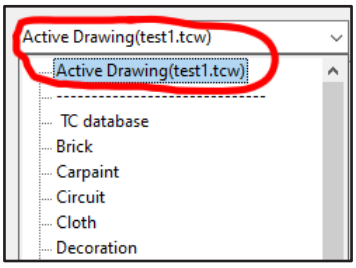
On Setting a material from the TC database area of the palette. This transfers the material from the TC selection database, into the Active drawing area of the palette.

The active drawing area is saved in a temp area on the hard drive, and then saved within the Tcw file if the file is saved. Here it can be safely altered without affecting the master materials

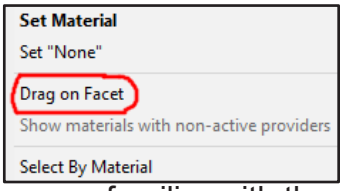
NOTE, Once a material or luminance is applied, altering it in the TC database will make no difference in the drawing.



I suggest. Always alter properties in the Active drawing section. Unless setting up a new material for using in multiple drawings.



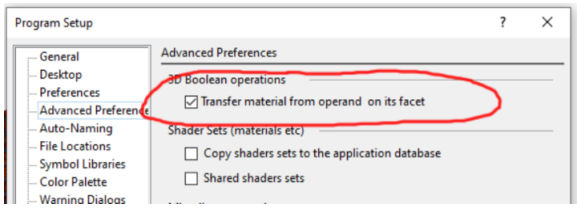
One can, simply click a material and drag it onto an object. **But be careful** if you have 'Drag on facet' ticked. As this will apply the material to a facet, and not the whole object. Which is not desirable in most circumstances. This can also cause problems further down the line, if you later wish to apply a new material to the whole object.



I suggest only using drag on facet when absolutely necessary. And you are familiar with the pro's and con's of using the option.

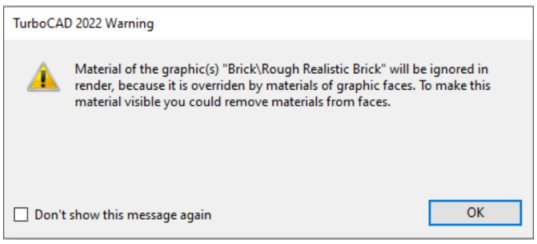
Whilst on the point of facets, a quick look at a common facet material problem.

Under Options menu - program setup - advanced preferences, is a setting called transfer materials from operand on its facet. What this does is to retain any materials on two or more object during a boolean operation.

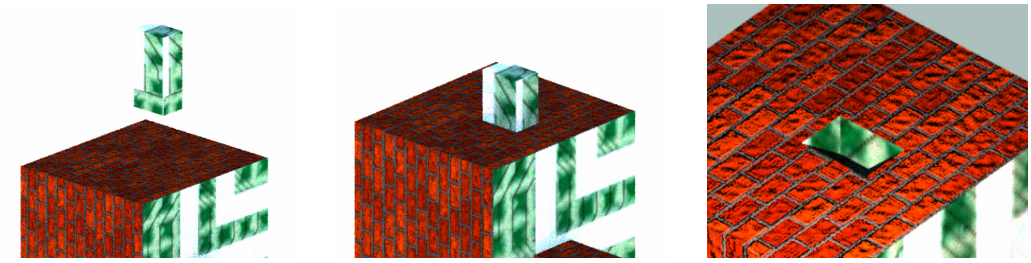


This may sound good, but can cause confusion if a drawing is opened, and one does not realise there are facet materials.

When rendering such objects, TC will (assuming the warning has not been disabled. Often pop up a warning saying it will override the material applied to the object as a whole.



As can be seen in the three images below. When the circuit box is moved down into the brick box, and subtracted, the facets inside the brick are changed to circuit.

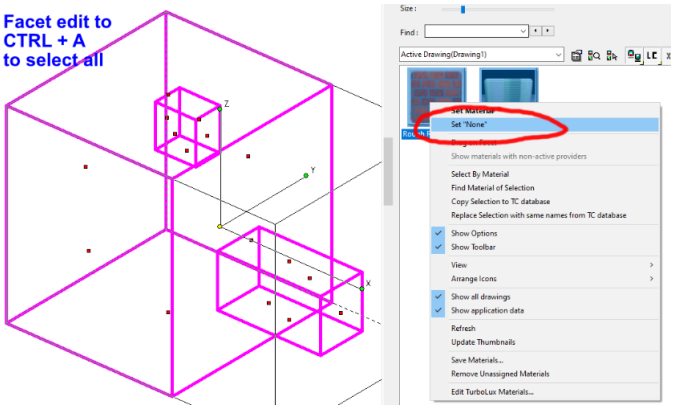


If this is done deliberately, one knows what to look for, but often it is not required. And undesirable. Personally, I turn off the option in advanced preferences, to stop this behaviour. And just apply to facets separately when required.

If the facet material is unwanted. How do we get rid of it.

TC Platinum. Here we can use the facet edit tool (works on solids or surfaces).

Select the tool. Click on a face. Press ctrl key + A (to select all the facets).

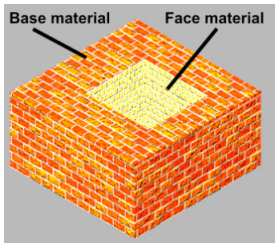


With them all selected, go to the material palette, active drawing section. You will probably notice more than one material has bee selected.

Right click (right mouse button) to bring up the menu, and select 'Set None'. This will set the whole object to base colour.

However you won't always know what the base colour is. Switch back to select tool (icon or press space bar). Select object, and only one material should now be highlighted. If it is the wrong one, simply change it to whatever required, just like normal.

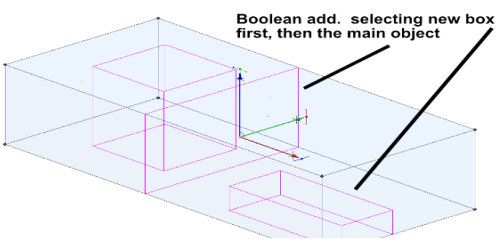
TC Deluxe (and possibly Professional).

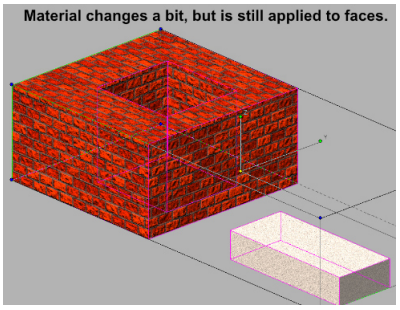


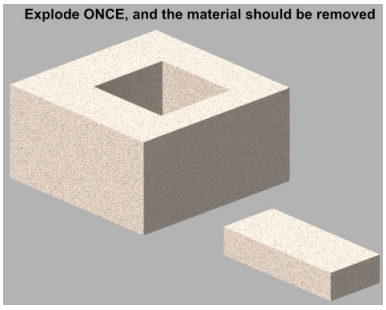
Deluxe does not have facet edit tool. Therefore a novel approach may be needed. If one has accidentally left advance preferences - transfer materials ticked, when doing a subtract. Some faces may now have a different material. One way to restore the object, is.

Draw a box AWAY from the main object, so there is a clear gap between them. Boolean add the two, selecting the new box first and the object second. Apart from adding, Nothing else will appear to have happened.

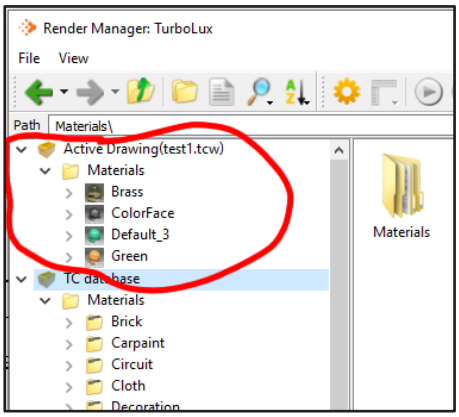
However, now explode the object once, to separate the new box from the original object. The facet material has been removed, ready to accept any new material one wishes.







Render Manager.



Material editing is done through the Render manager. Where one can add or alter materials and luminance. both in the database or the active drawing area.

If one is creating materials for use in multiple files, use the render manager to add categories, and materials / luminance into the TC database. Once added and used in a file, they again will be auto transferred to the active drawing area.

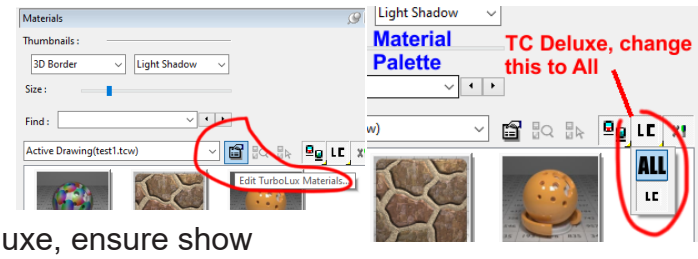
Again. Always alter materials in the active drawing area, unless you have a specific need to alter the database. Like, for example, adding a new global material. Or luminance.

Accessing render manager.

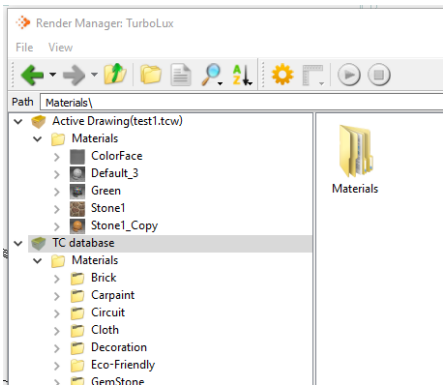
There are a few ways to access render manager. However, some early versions of 2022 deluxe have problems. Some methods simply had not been programmed in (so didn't work). And two things can cause TC deluxe to crash.

1. Selecting an object before selecting render manager, can eventually crash TC when materials are edited.
2. Selecting a material in the material palette then clicking render manager, can eventually crash TC when materials are edited. NOTE these may be repaired in a patch.

OK. Back to Render manager. Assuming all bugs have been repaired..



With the material palette open, ensure **no** object is selected. And **no** material is selected in the materials palette. Click the render manager icon at the top of the materials palette. Which is called 'Edit TurboLux materials'. (Note in early 2022 deluxe, ensure show provider = 'All' and not 'LC').



This will open the full render manager. The dialog contains both active drawing and TC database materials.

From here one can add, edit, copy materials. And add folders to store the materials.

Here we will add a new category and default materials.

Under where it says TC Database. Right click on the word materials, and choose Add Category,

In the name dialog. Type in a new unique name for the category. Note. This is the name for the folder that will contain your materials.

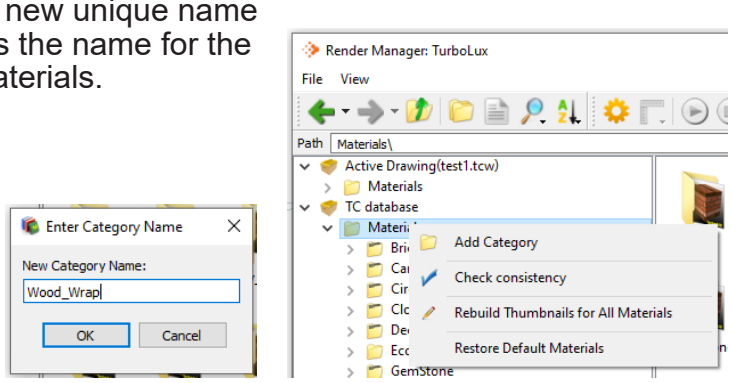
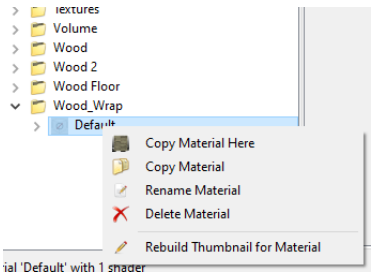


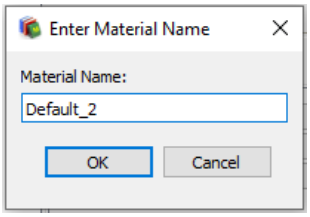
Fig 84

The new category will be created with a default material. The default material will likely be Matte. (unless the programmers change things). This is one of the simplest material.

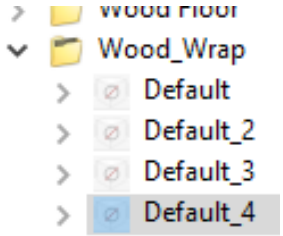
Next. Expand the new material category and select the default material. Right click on default, and choose copy material here.



Enter a new name. As these are just for an initial setup, I simply used Default_2.

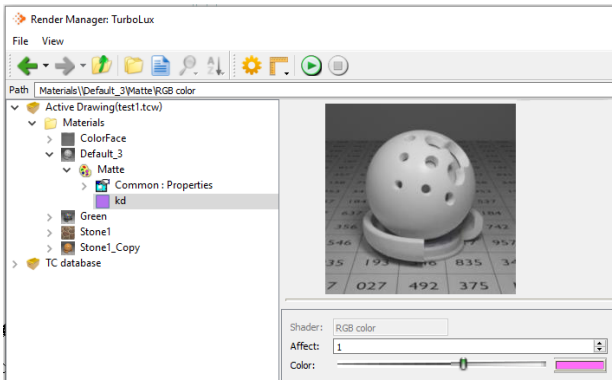


Add a couple more materials. This gives a good starting point for creating a variety of one own materials.



When one has decided on what the material will become, One can right click the material in tne render manager, and rename it to whatever is appropriate.

For editing a material, expand the headings in the left pane, and carry out the necessary adjustments in the right pane.



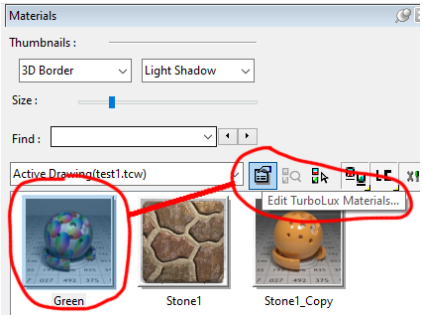
Note, as we are still in the 'full' render manager, the left pane of the display, still allows access to the TC material database display.This is not shown when editing a single material,

For displaying and editing a single material.

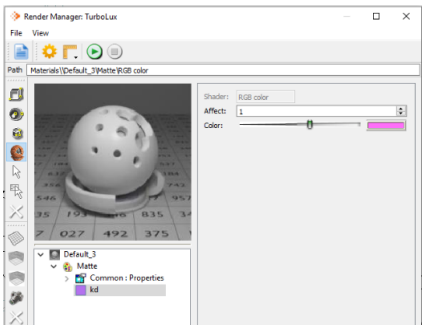
In TC 2022, I advise not having an object selected when editing the material. Ne only needs the object selecting when applying materials to an object, not editing.

Select the material in the active drawing section of the material palette.

Click the edit material button.



This brings up a render manager dialog, with only the chosen material visible. Ready for editing.

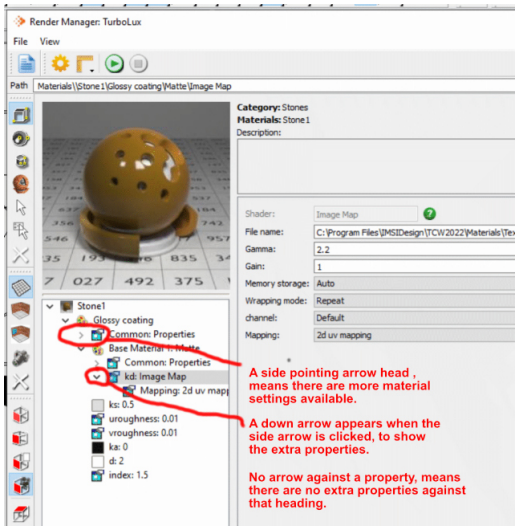


When using the render manager, unlike lightworks or RedSDK, material properties are not on tabs nor on a single page. The properties are contained on several pages.

These are controlled by side and down arrows. Clicking a side arrow reveals extra properties,

Simply click the required arrows until on gets to the required property, for example Image 2D mapping scale.

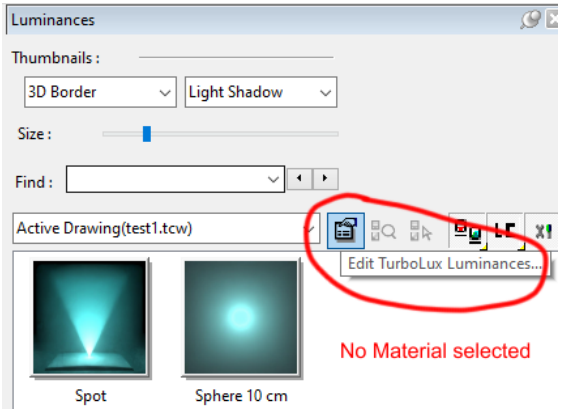
All setting are explained in the materials section of this documents, and in the TC help files.



Render manager Luminance.

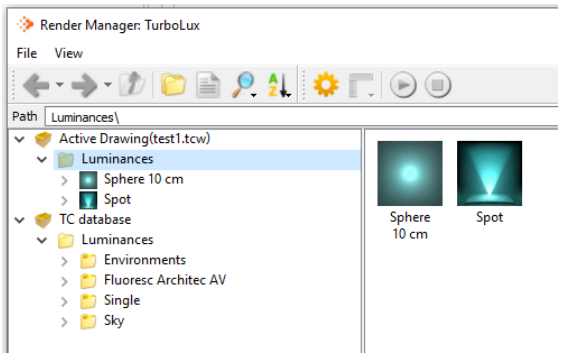
The same procedure applies to luminance, with regards the render manager.

Clicking on the 'Edit Turbolux Luminance' with no Luminance or object selected, will bring up the full render manager dialog.

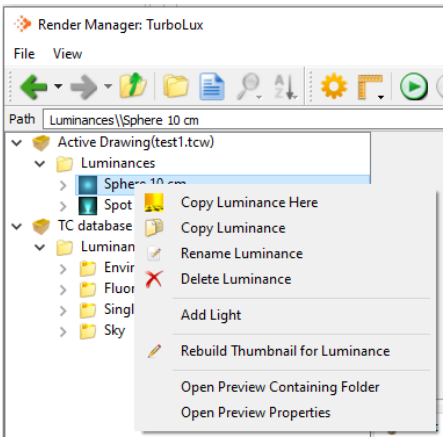


From there one can add, edit etc. Luminance. Just the same as for materials.

If luminance is just for that one drawing, always add or edit in the active drawing area, not the TC database area.

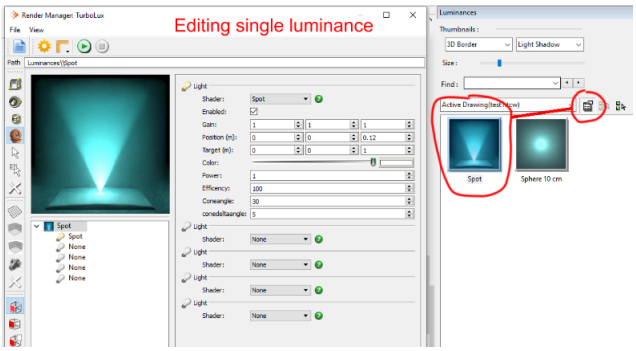


Luminance can be copied in the same was as maatertials are, simply copy luminance, and give it a unique name.



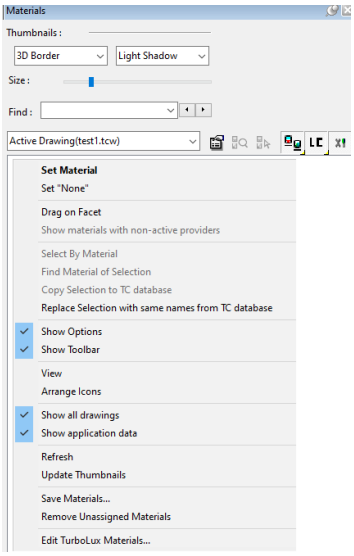
Clicking on a luminance in the luminance palette, before clicking the edit luminance button. Will bring of the single luminance render manager.

As with materials, one used the side and down arrows in the left pane, to expand or contract the heading to reveal the required properties for editing.

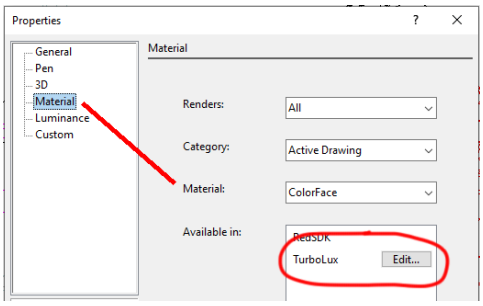


The luminance dialog allows for multiple luminance to be applied to a single object., Normally one wouldn't need to do this, except maybe when applying sky and sun together.

Another method which works in 2022 Platinum version, to access / edit materials, is to either, right click a blank area of the material palette, or right click a material. In the list at the bottom is Edit TurboLux material.



Alternatively. Right click an object. In object properties, material tab. Click the edit button against TurboLux.



Luminance

Luxcore provides a variety of lighting, but the Luxcore help files are confusing.

Seven standard lights are provided, Spot, Point, Mappoint, Projector, Sphere, Mapsphere and Laser. However Luxcore advises against using them. Instead you are advised to use Area lights, Mesh lights, Environment, Sun and Sky.

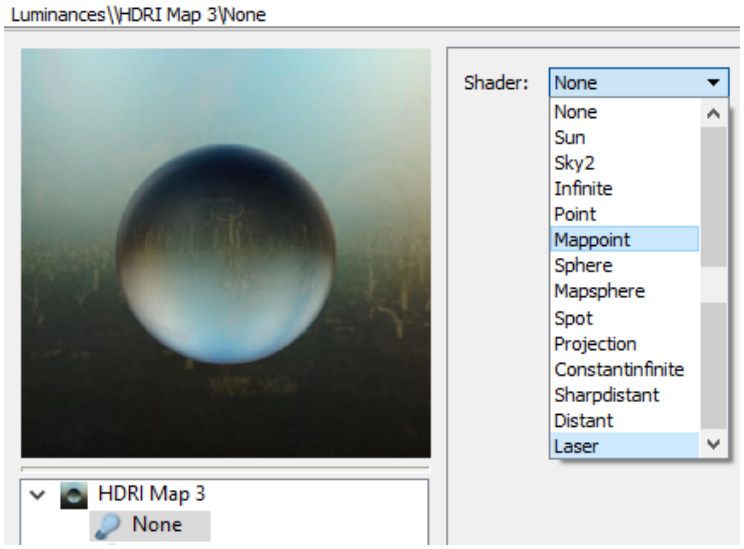
Quote *" Its recommended to use only mesh/area lights and environmental lighting. This is because the point and spot lights have no real world equivalent, as the light source is an infinitely small point. "* They do not mention the other 5 standard lights at all.

Whether you take that recommendation or not is a personal decision. Personally, I will still use them if I think it is appropriate.

In case anyone is wondering, **Mesh lights** are lights emitting from the normal's of a 3d object. They can be produced via the emission control of a material. Often a matte material, but others can be used. For example emission from the volume of a glass object. Using mesh lights, one can use IES lighting data. Either from the manufactures or an IES library.

At its most basic, Draw a bulb (Luxcore recommends as few faces as possible). Assign matte material. Set emission, and put in the IES data. You have a realistic light (hopefully).

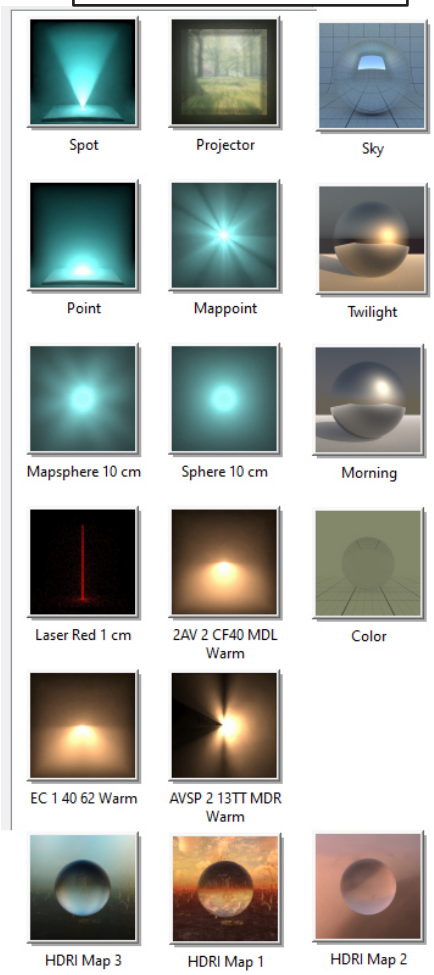
TurboLux supplies a number of luminance, which can be altered by the user to suit any given situation. Some are just minor variations of a standard light.



Other lights available.

Infinite
Constant Infinite
Distant
Sharp distant

Standard luminance



General Luminance properties.

Many luminance share the same initial properties, For example Gain, Position, Power, Efficacy (efficiency). In the same way Materials have 'Common' properties. Therefore to save a lot of duplication, these have been discussed below.

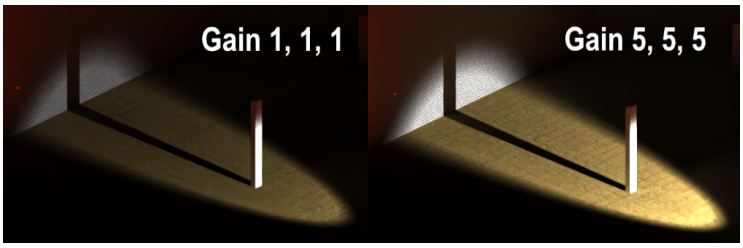
Enabled. Turns the light on / off. It allows one to test out several lights, and the different effects with them turned on or off.

Gain.

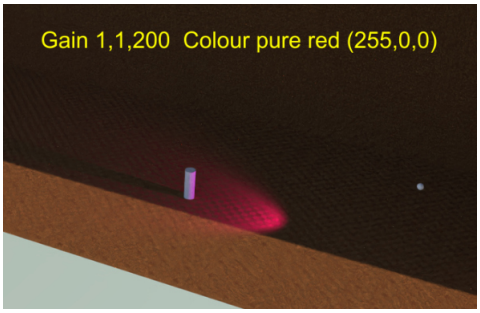
Gain is like a multiplier for the power of the light, but it acts on the RGB colour spectrum.

Setting a gain to '0' will turn off that colour channel. Setting gain to 0,0,0. With not show any luminance. Always having gain above zero.

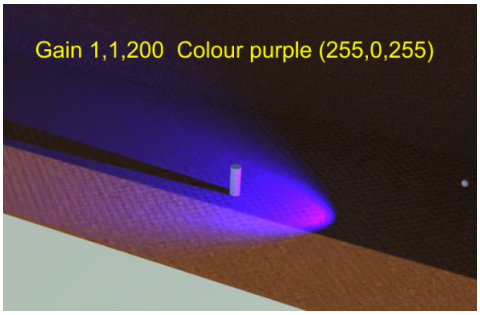
In many cases one could just leave the gain at 1,1,1. And use the power setting to alter the light. If all the values are the same, it is acting equally on all three colour channels. Effectively acting like a power level on grey scale colours, like the standard TC lights.



Incorrect settings, can make it seem like it doesn't work. For example. The dialog also includes a colour picker. Setting a pure red colour using the colour picker, is equal to RGB 255,0,0. Therefore if one set the gain as 1, 1, 255 to influence the blue spectrum. nothing can happen as pure red contains no blue.



However, changing the colour picker to purple (RGB 255,0,255). We can influence the blue spectrum, by setting the gain for example to 1, 1, 200. We will get the brighter part of the spot cone purple (red and blue), The colour then changes to reflect the increased gain on just the blue channel. As can be seen, the gain has increased the power of the light in the fade off areas.



Note. Some luminance do not have a power setting, for example infinite and distant. Therefore altering the gain is the only way to control the power of the light. Every scene is different, so one cannot say what gain may be required.

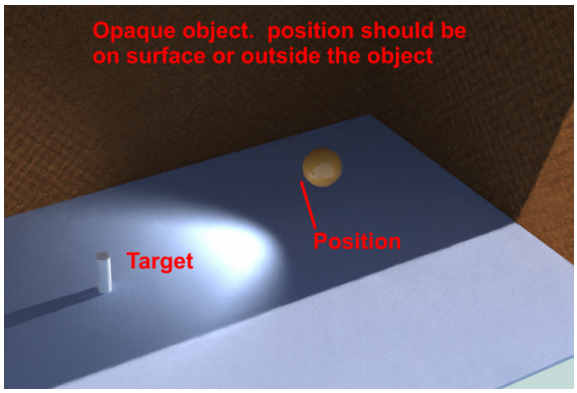
Position.

Note in the following figures m = metres

At the time of writing. The position level is in Metres, irrespective of what space units are chosen, or if units are metric or imperial. This may change in the future. The position is from the centre of the object to which the luminance is attached.

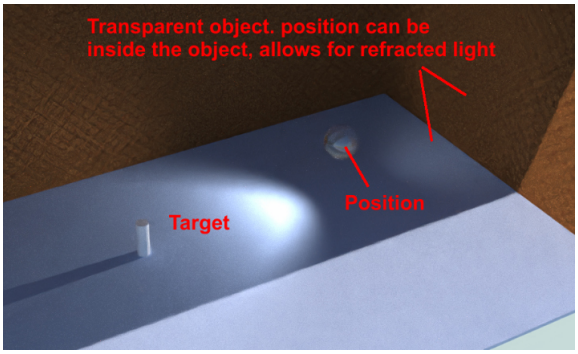
The object material and size is taken into account when producing the luminance. The three boxes are x, y and z. And depict the axes of the source object (entity), not Turbocad world axes.

For Opaque objects, brick, metal, most wood, etc. This should always be set to a positive or negative number. At least on the edge of the source object. Setting it to 0,0,0, will not allow any luminance to shine.

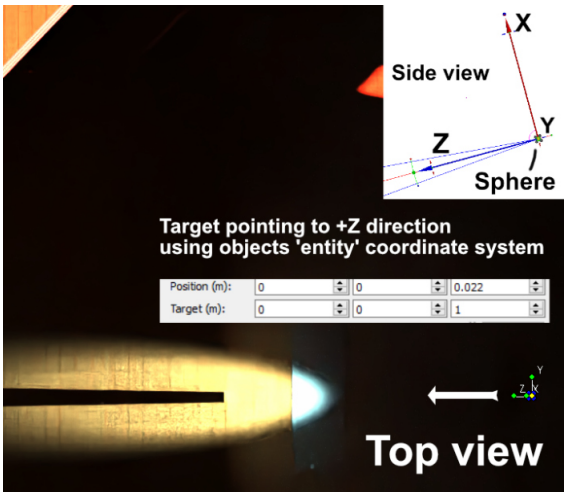


Also The object size is taken into account. For example, a 0.02m (20mm) diam sphere, obviously has a 0.01m (10mm) radius. We want the light to shine in the objects +Z direction (third column). Setting the position to 0, 0, 0.08 (8mm) will normally not allow the light to shine. Setting it to 0, 0, 0.1 (10mm) will allow it to shine.

For transparent objects, like glass. This can be set inside the objects, where the light will be affected by refraction, and probably absorption. For example in our 0.02m sphere, setting the light position as 0,0,0 will shine the light in the direction set in the target boxes, but also get refracted / reflected back in the minus target direction.



Note. It has been observed that some luminance are 'Backwards' i.e. to shine in the objects +Z direction one has to enter a negative number. This is mentioned under the appropriate luminance where know.

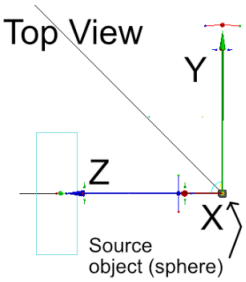


Target.

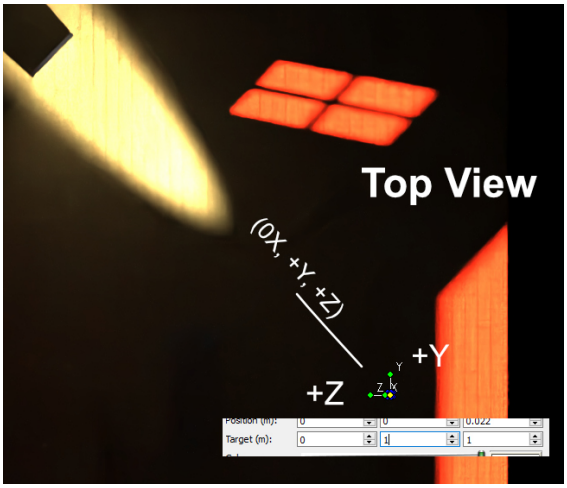
Target is straightforward. It is the x, y, z direction vector that the light will shine towards. However entering high numbers doesn't really make any difference.

For this, it is a matter of entering between -1 and 1 into the desired x, y, or z box. On most occasions simply enter 1 or -1 in the appropriate box, One variation from this would be if one wanted the target to be at an angle other than orthogonal x, y, z.

While one could use proper vector calculations, for most occasions you will likely just guess. For a 45 degree angle just enter 1 in two of the boxes.



If we look at the top view, we can see the X, Y and Z entity directions the source object is orientated.

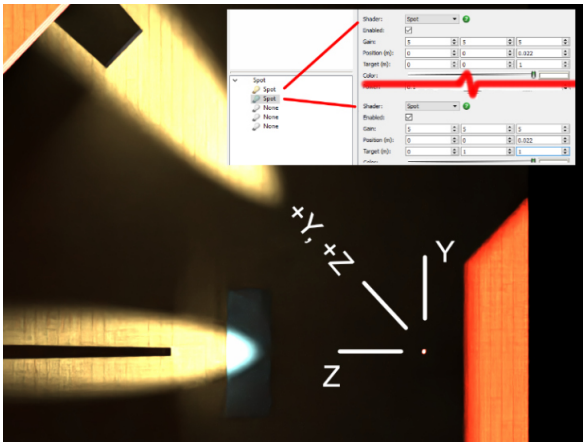


Therefore, to get it to shine 45 degrees, we can simply enter 1 in the Z box, and 1 in the Y box. As shown.

As previously stated, One can have more than one luminance attached to a single object. And these can have different targets.

Shown in the image on the right. We have two spot lights set up, one heading +z, and a second heading at 45 degrees, between +z and +y.

No idea in what situation one would want to do this. But just pointing out the possibility, if ever the situation presents itself.

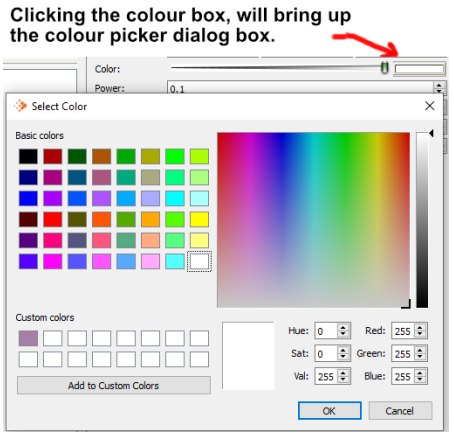


Color.

Color simply sets the base colour of the light. Assuming gain is set to 1,1,1 this will be the colour given off by the light. White (RGB equivalent 255,255,255) allows for the full spectrum available for the gain control. Setting it to Full Red (RGB 255,0,0) will restrict the light and gain to only the red channel.

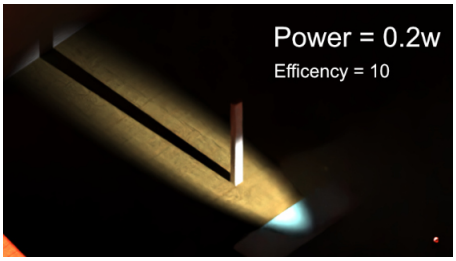
On a vast majority of occasions this would probably be left at white. Other colours could be used for special effects, red / blue police lights, amber flashing lights etc.

It may sometimes be necessary to restrict the colour by also adjusting the gain. For example, set the colour to red, and set the gain to 1, 0, 0.

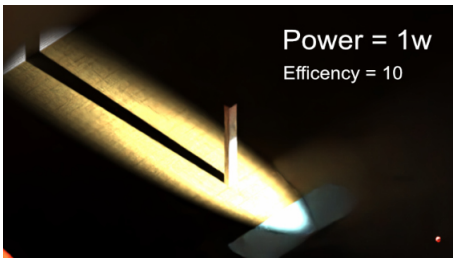


Power.

Power output in watts. The help document states *"Setting 0 for both power and efficiency bypasses this feature and uses only the gain."* Using a power is much simpler than using just the gain, which requires entries in the three RGB boxes. Use the gain to multiply the power setting, remember to also set the efficiency. .



The Power setting, is just ones own personal preference for any given situation. If you can find manufacturers data stating the power and efficiency, one can use those figures.

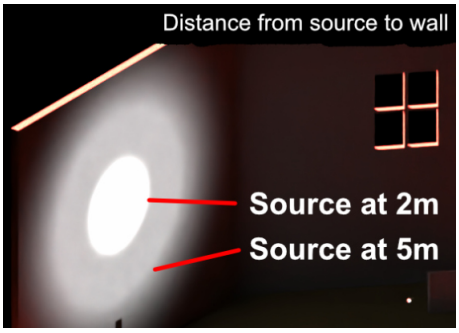


Distance from source will make a difference to the light and how much power is required.

Different light will not react the same to distance from the target, and will be trial and error to get it right.

Often the source cannot be moved, so altering the power, efficency, and other properties will be required.

The image (right) uses a spot light. And as can be seen, there is a difference in brightness when simply moving the source object further from the wall. When using the same power setting.



Efficency. (as spelt in the light properties).

This is the efficacy or efficiency of the power in Lumen per watt. Efficiency is normally a percentage, whereas efficacy is normally a value. An efficacy of 0 means the power is inefficient to the point of not shining. Lumen efficacy varies widely.

But a rough calculation could be, A UK incandescent bulb was stated as 610 lumens, and 60 watt. In theory it should be 10.16 efficacy (lumen / watt). However, is anyone going to try render a scene that accurately. I doubt it. **Typical values (from Wikipedia) are :**

Source	Lumens per watt	Source	Lumens per watt
Candle (13 lumens, 80 watt)	0.16	Arc mercury xenon	50 - 55
Gas Mantle	1 - 2	Arc mercury vapour (for projectors)	30 - 50
Incandescent 40w 230v	10.4	Arc mercury vapour (free standing)	58 - 78
Incandescent 100w 230v	13.8	Fluorescent 32w	60
Incandescent 40w 120v	12.6	Fluorescent 9 to 32	46 - 75
Incandescent 100w 120v	17.5	Fluorescent T8	80 - 100
Halogen incandescent 120v	17.7 - 24.5	Fluorescent T5	70 - 104.2
Halogen tungsten quartz 12-24v	24	Fluorescent 70 - 150w	71 - 84
Halogen photographic lamp	35	Tungsten (2800k)	15
LED Screw 120v	102	Paraffin lamp (Kerosene)	15
LED screw 230v, 5 to 16w	75 - 210	Sun (3.828 x 10 ^26 w)	98
LED 230v 21.5w tube	172		
Arc carbon	2 - 7		
Arc xenon	30 - 90		

The next three apply to material and lighting. However I cannot find much information about them. All say "This is an hint and can be optional supported by the render engine. It can be used to remove fireflies."

Indirect diffuse. Default On

Help file states *"if the light source is visible by indirect rays on diffuse surfaces.* Appears to affect the diffuse property of a material. As can be seen, it affects the matt floorboards.

Indirect glossy. Default On

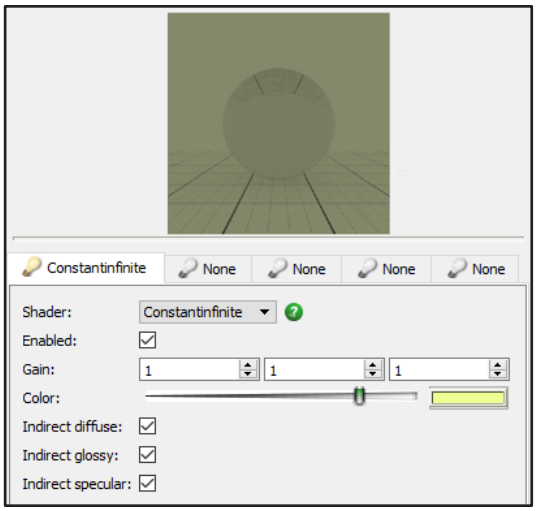
Help file states. "if the light source is visible by indirect rays on glossy surfaces. Appears to affect the glossy property of a material. As can be seen, it affects the glossy coating which is applied to the cylinder

Indirect specular. Default On

Help file states "if the light source is visible by indirect rays on specular surfaces.

Luminance properties

ConstantInfinite (Colour).



ConstantInfinite is a simple environment colour. It adds a plain colour luminance into the scene.

If one requires good shadows, then either sun, or another luminance may be required. Though on its own it can produce subtle shadow.

This can be useful for mechanical drawings, gears etc. there harsh shadows may obscure components.

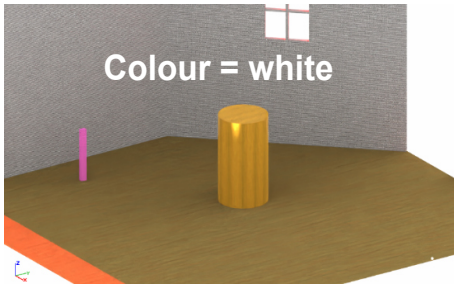
Even though it can be used to add a colour background, It works well with white.

Enabled. **Default On**
Turns it on of off.

Gain. **See Page 28.** **Default 1, 1, 1**

Sets the Red, Blue, Green colour spectrum channel. All figures being equal, results in grey scale adjustment. The gain normally acts to intensify the power setting. However Constant infinite has no power setting. Therefore the gain alone is used to increase the light power.

Color. **See Page 29.** **Default White**



Sets the primary background colour of the light. It can be used with gain to isolate a particular colour

This is a plain colour like adding a standard background colour in TC. However using constantinfinite adds luminance into the drawing. And thus will light the scene,

Normally one would leave it at a grey scale colour (white to black). Adding a rgb colour can make the scene look odd. Though in some situation it may be desirable (like a scene on planet Mars).

Indirect diffuse. **Default On**

Help file states *"if the light source is visible by indirect rays on diffuse surfaces. This is an hint and can be optional supported by the render engine. It can be used to remove fireflies"*.

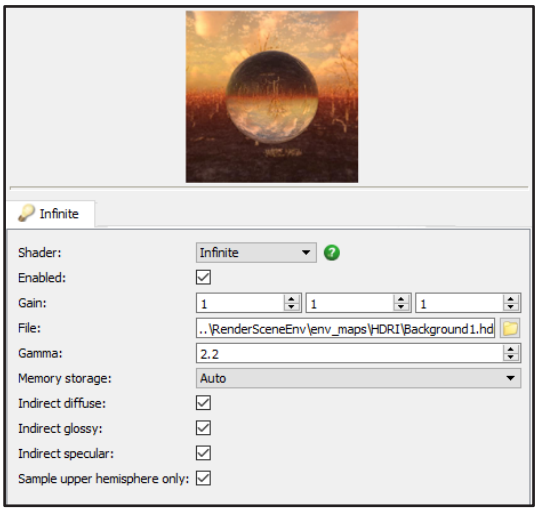
Indirect glossy. **Default On**

Help file states. "if the light source is visible by indirect rays on glossy surfaces. This is an hint and can be optional supported by the render engine. It can be used to remove fireflies."

Indirect specular. **Default On**

Help file states "if the light source is visible by indirect rays on specular surfaces. This is an hint and can be optional supported by the render engine. It can be used to remove fireflies."

Infinite.



Infinite luminance is used to add an environment map, in the same way T-Lux scene can add a map. The main drawback is the lack of additional settings like turbidity. But having a gamma setting, the tone of the background can be altered.

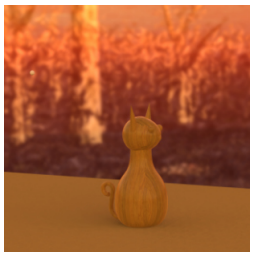
This can be applied as a global luminance, or attached to an object as a luminance.

Enabled. **Default ON**
Turns it on of off.

Gain. **See Page 28.** **Default 1, 1, 1**

Sets the Red, Blue, Green colour spectrum channel. All figures being equal, results in grey scale adjustment. The gain normally acts to intensify the power setting. However infinite has no power setting. Therefore the gain alone is used to increase the light power.

File. **Default “”**



Filename of the environment map. Although a standard image can be used. This is best used for inclusion of a HDRI image. Simply click the folder button, and navigate to the correct file.

If the HDRI / image is applied to an object, then the x, y, z orientation of the object will affect the rotation of the HDRI. If the background is not orientated correctly, simply rotate the object to correct the rotation.

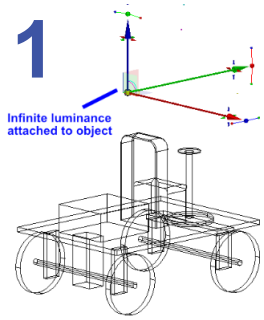
This can make it really easy to set the correct part of the image or HDRI as the background for a specific view. No need for any vector input. Example shown in the 4 images below.

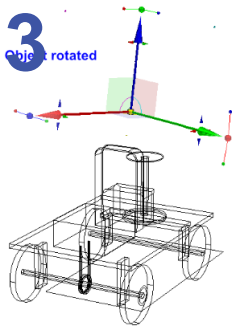
Number 1. The HDRI is attached to a sphere object, with default orientation.

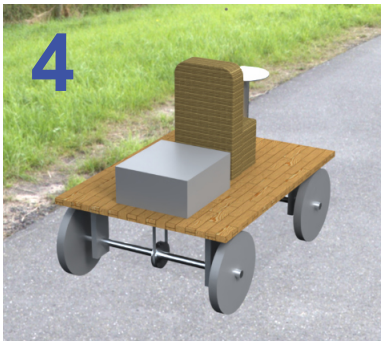
Number 2. The render clearly shows the new electric car is riding on the grass.

Number 3. The sphere is rotated to get the desired part of the HDRI to show in the render.

Number 4. The finished render, showing the HDRI has rotated. The car has stayed in the same orientation. As can be seen. Changing the orientation of the HDRI is far simpler tan calculating the correct vector. (or guessing at the vector of a particular place).







Gamma. **Default 2.2**

Gamma correction used by the image map. Increasing the gamma, has the effect of darkening the image or HDRI. Reducing gamma lightens the image.

The two images on the right, are of an HDRI, applied in the infinite luminance,

Memory storage. **Default Auto**

Environment map storage type ("auto", "byte", "half" or "float")

Indirect diffuse. **Default On**

Help file states *"if the light source is visible by indirect rays on diffuse surfaces. This is an hint and can be optional supported by the render engine. It can be used to remove fireflies"*.

Indirect glossy. **Default On**

Help file states. "if the light source is visible by indirect rays on glossy surfaces. This is an hint and can be optional supported by the render engine. It can be used to remove fireflies."

Indirect specular. **Default On**

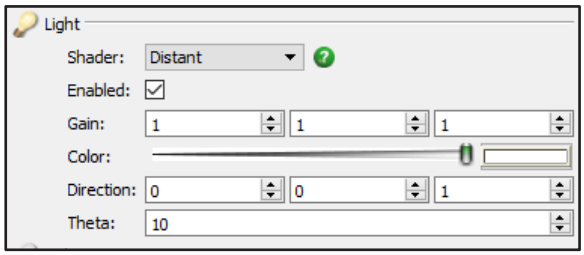
Help file states "if the light source is visible by indirect rays on specular surfaces. This is an hint and can be optional supported by the render engine. It can be used to remove fireflies."

Sample upper hemisphere only. **Default Off**

An option set to black the lower hemisphere. Mostly useful with shadow catchers

Distant.

From Luxcore… Distant luminance simulates a light source that is infinitely far a way, but does not simulate interaction with an atmosphere. Can be used to simulate a near star in space when rendering science-fiction scenes.



This can be applied as a global luminance, or attached to an object as a luminance, but the effect is the same. Distant uses vector coordinates. In a similar way the Sun works.

Enabled. **Default On**
Turns it on of off.

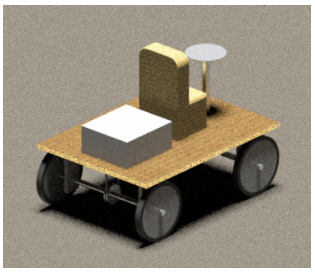
Gain. **See Page 28.** **Default 1, 1, 1**

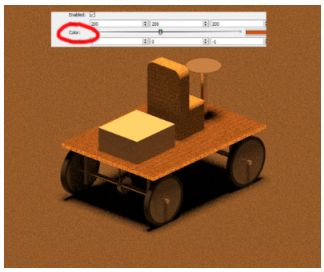
Sets the Red, Blue, Green colour spectrum channel. All figures being equal, results in grey scale adjustment (white to black). Distant has no power setting. Therefore the gain alone is used to increase the light power. Because of this, one may need to set this quite high. Maybe something like 200, 200, 500 or even higher.

Color. **See Page 29.** **Default White**

Sets the primary colour of the light. Can be used with gain to isolate a particular colour.

The images right, show the difference between white (first image). And colour changed to Red (far right).



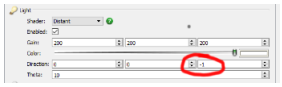


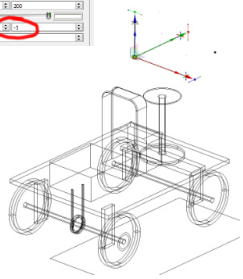
Direction. **Default 0, 0, 1**

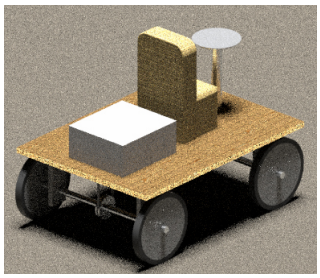
Sets the vector direction of the light. This is aimed at the drawing centre.

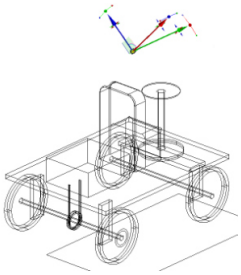
However, at the time of writing. Instead of the vector pointing to the light, which is the normal way. It appears to be the vector of which the light is pointing. Thus to have the light above the horizon, point at the centre of the drawing. One would set the z direction as minus. For example -0.5 (normally one would set the vector as +0.5).

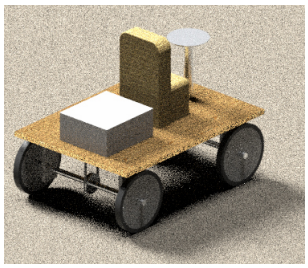
The images below show the changes that can happen, by just altering the sphere which the luminance is attached. As opposed to changing the vector direction of the light.





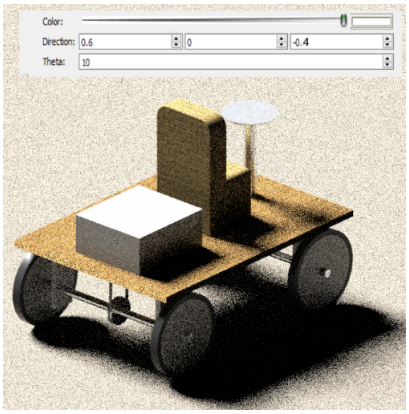






To obtain a similar direction using the vector input, for example if the luminance is applied as a scene luminance, instead of object. The vector would be similar to the image on the right.

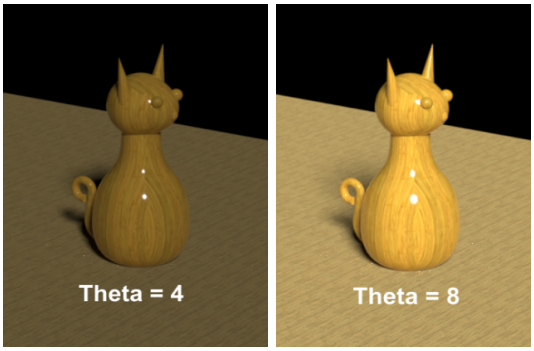
In this case , we need the light to point downwards (-Z), and also to the right (+X).



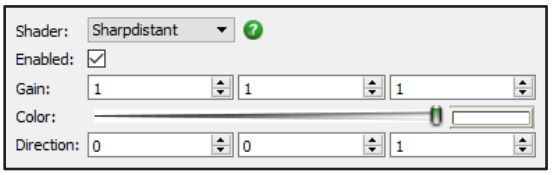
Theta. Default 10

The half angle of the light source in degree. Must be > 0 for soft shadows.

Increasing the angle, increases the brightness of the light.



SharpDistant.



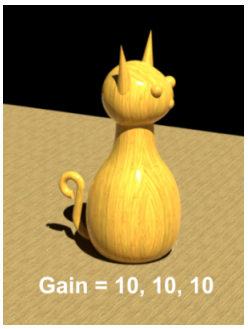
Sharp distant luminance simulates a light source that is infinitely far a way, but does not simulate interaction with an atmosphere. Can be used to simulate a near star in space when rendering science-fiction scenes.

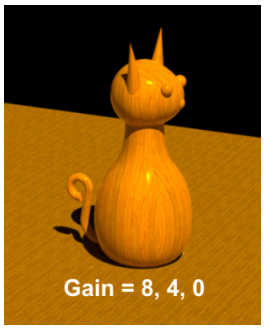
This can be applied as a global luminance, or attached to an object as a luminance, but the effect is the same. Distant uses vector coordinates from the drawing centre. In the same way the Sun works. But without turbidity etc.

Enabled. **Default On**

Turns it on of off.

Gain. **See Page 28.** **Default 1, 1, 1**



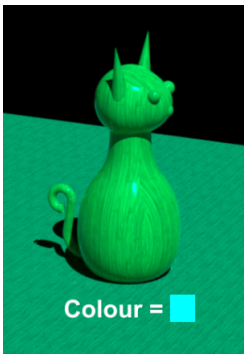


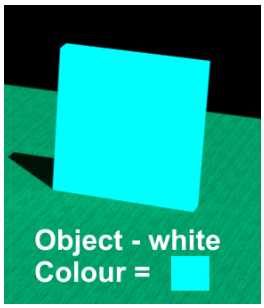
Sets the Red, Blue, Green colour spectrum channel. All figures being equal, results in grey scale adjustment. The gain normally acts to intensify the power setting. However distant has no power setting. Therefore the gain alone is used to increase the light power.

Altering the RGB channel of the gain affects to colour of the light.

The gain may need setting larger than normal, for some scenes. For example 60, 60, 60.

Color. **See Page 29.** **Default White**





Sets the primary colour of the light. Can be used with gain to isolate a particular colour.

Setting the correct colour can be trial and error. As the colour will interact with material colours. If the object is white, the correct colour will be displayed.

Direction. **Default 0, 0, 1**

Sets the vector direction of the light. This is aimed at the drawing centre.

Instead of the vector pointing to the light, It is the vector of which the light is pointing. Thus to have the light above the horizon, point at the centre of the drawing. One would set the z direction as minus. For example -0.5 (normally one would set the vector as +0.5)

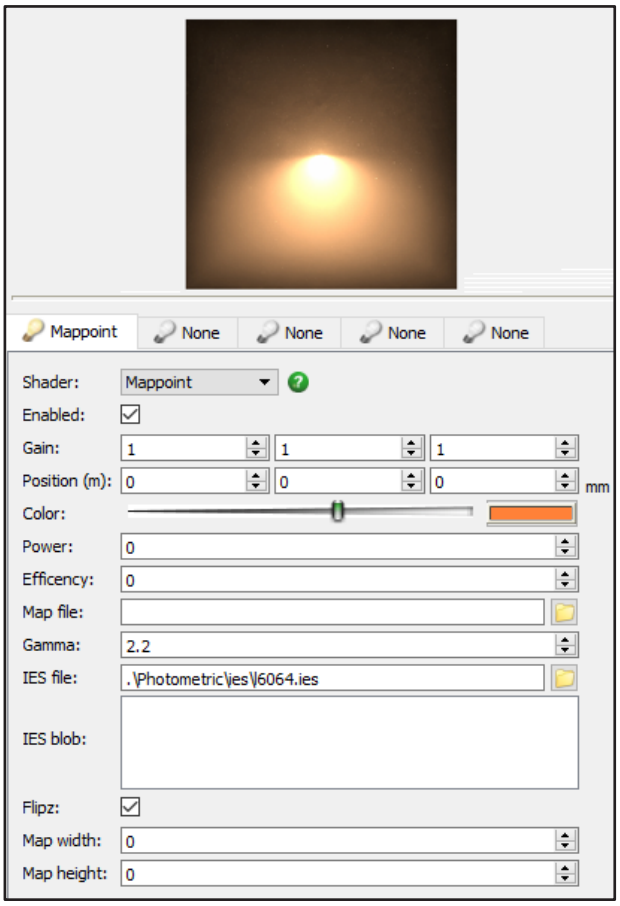
Se the previous luminance (Distant). For examples of altering the object, if the luminance is applied to it. Or altering the vector, if the luminance is applied as a global luminance.

Fluorescent light
24V 2 CF40 MDL Warm

Fluorescent luminance files are simply Mappoint luminance with added IES data.

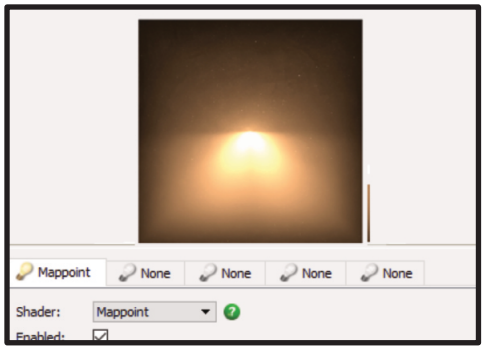
There are three different flourescent lights, and the difference is in the IES files they use.

See mappoint light on page 41 for the settings.



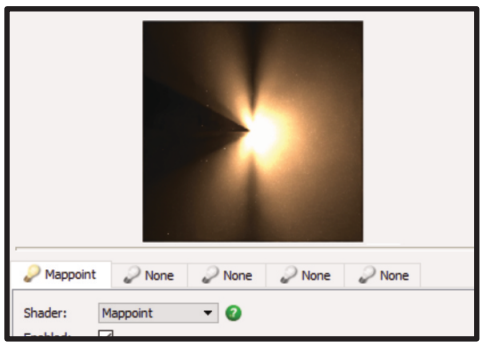
IESNA91
[TEST] L6064
[MANUFAC] Lithonia Lighting, Lithonia Fluorescent
[LUMCAT] 2AV 2 CF40 MDL
[LUMINAIRE] AVANTE SMALL OFFICE 2' X 2'
[MORE] 2 LAMP TT5RS SLOTTED BASKET
[MORE] WITH OVERLAY
[LAMP] 3150 LM LAMP

Fluorescent light
EC 1 40 62 Warm



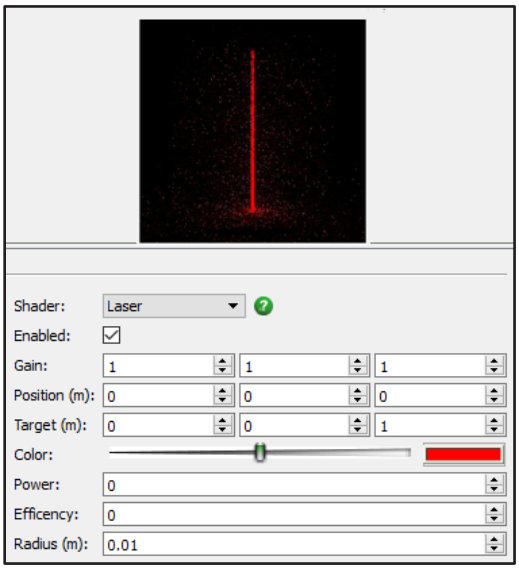
IESNA91
[TEST] I22075
[MANUFAC] LITHONIA LIGHTING FLUORESCENT GROUP
[LUMCAT] EC 1 40 62
[LUMINAIRE] COFFERED CEILING SYSTEM 4'X4' 1 LAMP T12RS
[MORE] HOLOPHANE PERCEPTA LENS
[LAMPCAT] F40/T12CW
[LAMP] 3200 LUMEN LP

Fluorescent light
AVSP 2 13TT MDR Warm



IESNA91
[TEST] LTL6744
[MANUFAC] LITHONIA LIGHTING - FLUORESCENT
[LUMCAT] AVSP 2 13TT MDR
[LUMINAIRE] AVANTE SHIELD SCONCE 10IN. X 12IN. 2 LAMP TT5 13-WATT
[MORE] PERFORATED METAL OVER 1/8" ACRYLIC DIFFUSER,
[MORE] HELD IN PLACE BY PRONG DETAIL
[LAMP] (2) PL-S 13W/27 RATED 900 LUMENS EA.

Laser.



Turns this lamp into a laser that emits parallel light rays that start at its location (as opposed to the sun/distant lamps, which also emit parallel light rays but are always infinitely far away). The area lamp was chose to represent this LuxCore light because it can visualize the laser light radius. However the laser light source will emit light in a circle shape, not quadratic like the area lamp shows in the OpenGL viewport.

This is equivalent to the Beam light in RedSDK.

Enabled. **Default On**
Turns it on of off.

Gain, Position, Colour, Power and Efficiency, are the same as for Point light. Therefore only a brief description is stated here.

Gain. **See Page 28.** **Default 1, 1, 1**

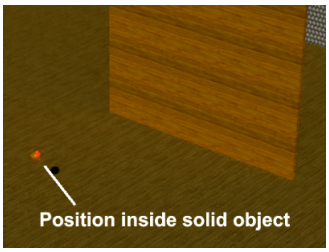
Increases the power of the light, specific to the RGB channel chosen. Setting all channels to zero may prevent the light from showing. Setting just one channel, restricts the light to that one channel, For example, 1, 0, 0 restricts it to the red channel. See page 28

Most often one would keep all the values equal to affect all the colours the same amount (grey scale).

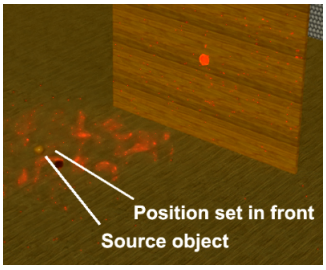
Position. **See Page 28.** **Default 0, 0, 0**

Sets the position of the light relative to the source object. Normally lights need to be set to outside a solid object. However because laser light has a radius, it just need the radius high enough so it is outside of solid objects. Though the centre is not lit. (see radius).

A



B



C

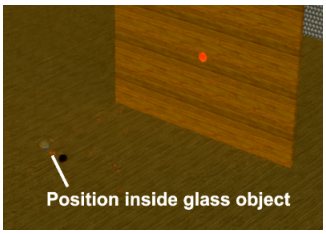


Fig (A) shows the position inside the object, with a laser radius smaller than the object radius. Nothing will show up.

Fig (B) shows the position moved towards the target. Using the same small laser radius.

Fig (C) shows position back inside the object, but the object material changed to glass, allowing the laser to shine.

See also radius setting.

Target. **Default 0, 0, 1**

This is the primary direction one wishes the light to shine, and is in x, y and z coordinates of the source object. Example. In our solid sphere above, the Z axes is pointing where we want the light to shine. And the position is set to 0, 0, 0.01m. Therefore a target position of 0, 0, 1 will ensure success.

The target can be positive or negative numbers. To reflect a positive or negative object entity axes

A couple of things to keep in mind. If the target is set the same as the position, usually no light will shine. Therefore always set the target different to the position.

A second thing is. If you have a position of say 0, 0, 0.1m and wishes the light to shine in the positive z direction. One would set 0, 0, 1. However If one sets the target to 0, 0, 0.01m . i.e. the target is closer to the source than the position. The laser will shine backwards towards the source. Therefore in most cases, you should set the target greater than the position.

Always remembering, the boxes are for object entity coordinates. Personally I just use either 0 or 1. There is often no need to enter decimal numbers for target.

Color. **See Page 29.** **Default White**

Sets the primary colour of the light. Can be used with gain to isolate a particular colour

Power. **See Page 29.** **Default 0**

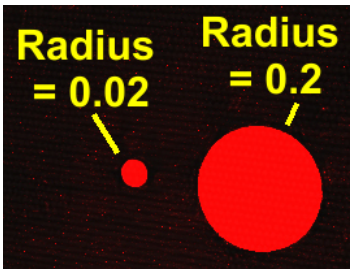
Sets the power of the light in watts. A setting of zero will prevent the light from showing

Efficency. **See Page 29.** **Default 0**

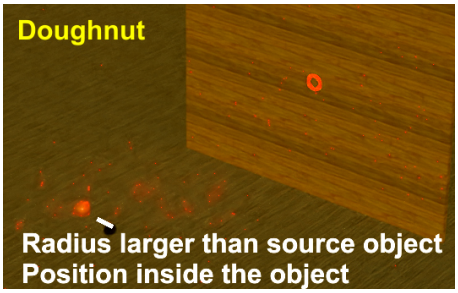
Sets the efficiency of the power level. Lumen per watt.

Radius. **Default 0.01 Metre**

Radius is in Metres. This Sets the radius of the laser. It does not affect any associated object size. With the 'position' set in front of a solid object, or inside a transparent object. the beam will be a solid circle with this radius.

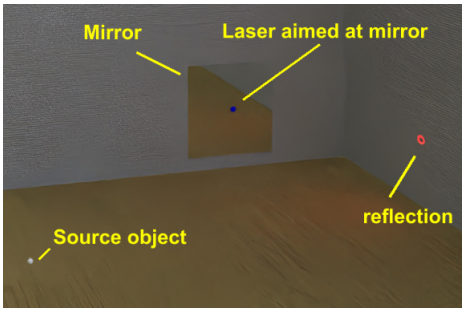


With the position inside a solid object, and a larger radius than the object. It is possible the laser will be a doughnut. Due to the centre of the laser being obscured by the source object.

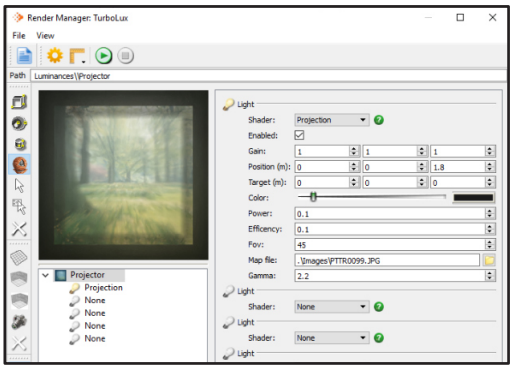


Laser will pass through glass, (Set render to bi-directional), but are fiddly to set up, as one has to take into account reflection, refraction, IOR, and any volume absorption applied.

Laser reflects very well. Aimed at a mirror it will reflect onto other surfaces



Projection.



Turns this lamp into a projector that emits parallel light rays that start at its location

Source object can be any 3d object. Sphere or flat surface makes good objects.

Enabled. **Default On**
Turns it on of off.

Gain, Position, Colour, Power and Efficiency, are the same as for Point light. Therefore only a brief description is stated here.

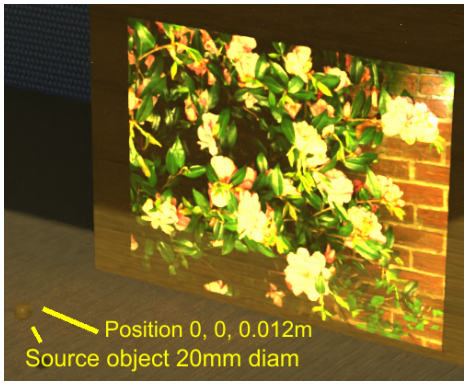
Gain. **See Page 28.** **Default 1, 1, 1**

Increases the power of the light, specific to the RGB channel chosen. Setting all channels to zero may prevent the light from showing. Setting just one channel, restricts the light to that one channel, For example, 1, 0, 0 restricts it to the red channel.

Most often one would keep all the values equal to affect all the colours the same amount (grey scale).

Position. **See Page 28.** **Default 0, 0, 0**

Sets the position of the light relative to the source object. Normally lights need to be set to outside a solid object and can be set inside on transparent objects.



Target. **Default 0, 0, 1**

This is the primary direction one wishes the light to shine, and is in x, y and z coordinates of the source object. Example. In our solid sphere above, the Z axes is pointing where we want the light to shine. And the position is set to 0, 0, 0.012m. Therefore a target position of 0, 0, 1 will ensure success.

The target can be positive or negative numbers. To reflect a positive or negative object entity axes

A couple of things to keep in mind. If the target is set the same as the position, usually no light will shine. Therefore always set the target different to the position.

A second thing is. If one sets the target closer to the source than the position. The light may shine backwards towards the source. Therefore in most cases, you should set the target greater than the position.

Always remembering, the boxes are for object entity coordinates. Personally I just use either 0 or 1. There is often no need to enter decimal numbers for target.

Color. **See Page 29.** **Default White**

Sets the primary colour of the light. Can be used with gain to isolate a particular colour . However unlike other lights, we are projecting a 2d image, therefore colour is less relevant.

Note applying a colour, but NOT including a 2d map file, will likely crash TC. If you wish to project a colour onto a wall, use a 2d image of the colour as a map file.

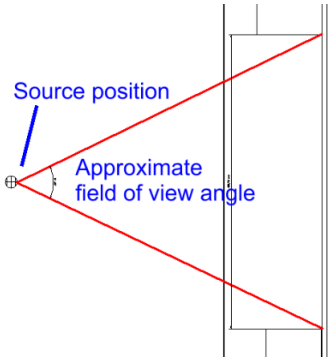
Power. **See Page 29.** **Default 0**

Sets the power of the light in watts. A setting of zero will prevent the light from showing

Efficency. **See Page 29.** **Default 0**

Sets the efficiency of the power level.

FOV. **Default 45**



Field of view in degrees, (plus or minus 10 degrees). Values approaching 180 degrees won't display correctly, as this is projection not wrapping.

If using a rectangular image, the angle will obviously be different in x direction to y direction. The correct axis and angle, depends on the source objects orientation. Square images, will be correct whether measured using the images x or y dimensions.

Map file. **Default ""**

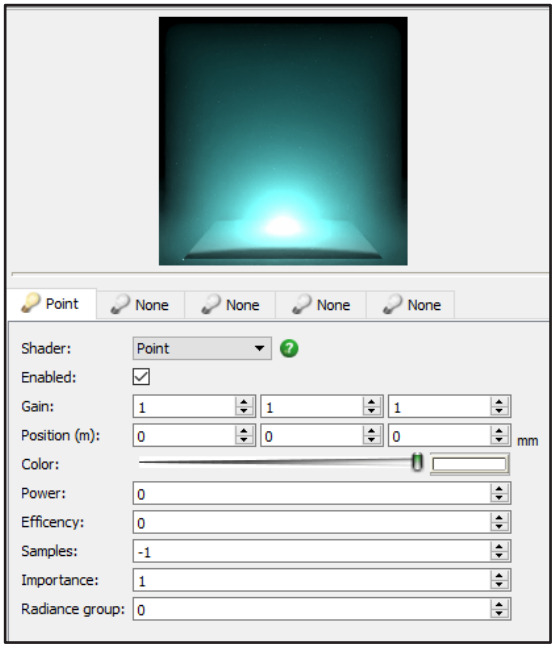
IMPORTANT. At the time of writing, if you render a scene with projection, without a file name as the map file. TC will crash.

This is simply the name of the 2d image used for the projection. Images can be bmp. jpg, tif or png.

Gamma. **Default 2.2**

Adjusts the gamma of the image being used as the projection, The standard gamma is 2.2. Increasing gamma will darken the image, lower values brighten the image.

Point light.



The Lux point light emits light in multiple directions. Similar to the standard TurboCAD point light. And as such, does not have a target field like the spot light.

The thumbnail suggests that point light is a surface light, emitting from an object. This is only partly true.

As a point light is normally associated with an object, (as opposed to being a global luminance like HDRI), the darker area behind the source object is shadow. Caused because it is shining in all directions including back at the source object.

The point light contains fall off, which is likely to be quadratic. What this essentially means is that, If a point light is put inside a box the corners will look darker, simply because they are further away from the source. Inside a sphere, the point will illuminate all areas equally.

Enabled. **Default On**
Turns it on of off.

Gain. **See Page 28.** **Default 1, 1, 1**

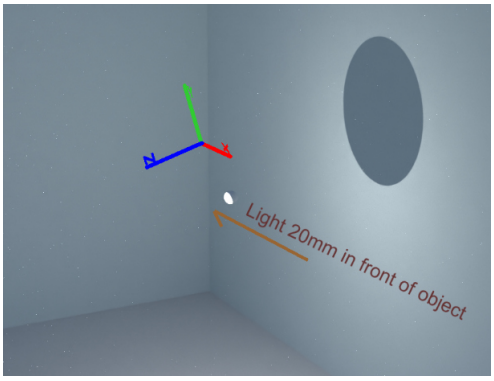
Increases the power of the light, specific to the RGB channel chosen.

Most often one would keep all the values equal to affect all the colours the same amount (grey scale).

Position. **See Page 28.** **Default 0, 0, 0**

Sets the position of the light relative to the source object.

Because a point light shines is all directions, This will differ from lights like spot. In that it will shine back at the source object. If the position is outside the source object, then a shadow will be cast behind the source.



In fig right, the light is attached to a 10mm radius sphere. The position was set in the z axes at 0.03m. This puts the position outside the sphere and 20mm in front. As can be seen, because point shines in all directions, a shadow is cast where the light shines back onto the sphere.

For transparent objects, like glass. This can be set inside the objects, where the light will be allowed to shine in all directions. Hence there is no ‘target ‘ boxes in point light properties. You may also need to turn down material transparency a bit.

If one just wants the light with no object showing, alter the material transparency to suit ones needs. In reality one doesn’t get a light without some sort of source. However, if one looks into a bright light, for example a lit 150 watt bulb. One normally cannot see the source filament. So a transparent source would be acceptable.

Color. **See Page 29.** **Default White**

Sets the primary colour of the light. Use the colour picker to set the colour, and the slider to adjust the tone.

Power. **See Page 29.** **Default 0**

Sets the power of the light in watts. A setting of zero will prevent the light from showing..

Efficiency. **See Page 29.** **Default 0**

Sets the efficacy of the power level, in Lumen per watt. Setting it to zero may turn off the light.

Samples. **Default -1**

The help file states *"the number of shadow ray to trace to compute direct light for this light source. This property is an hint and the render engine can ignore this information. -1 means use the default global value. "*

In many cases, with a point light, this won't make any noticeable difference.

Importance. **Default 1**

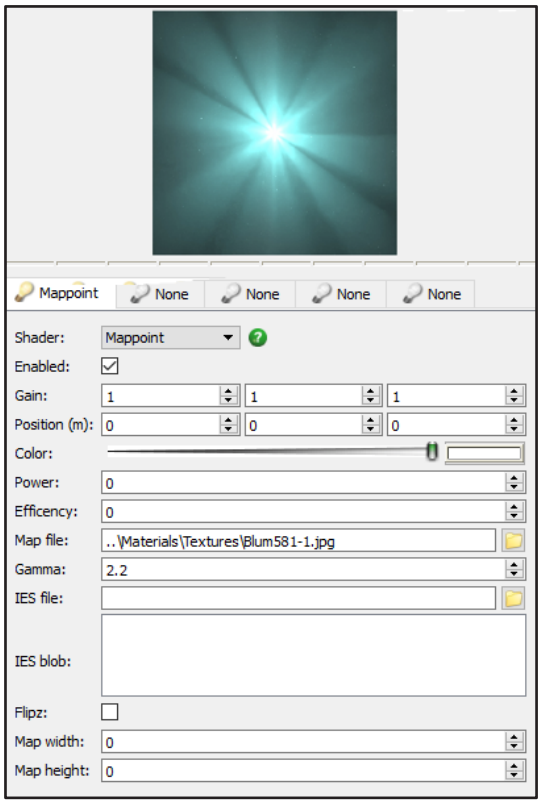
The help file states " Importance: How much processing power to spend on this light source compared to other light sources. Used to scale the light importance computed by the light strategy. For instance, if you set a uniform light strategy, a light with a user importance of 2.0 will be sampled 2 times more often than one with 1.0. If you use a power light strategy, the user importance will be multiplied by the light power. "

In many cases, if you have multiple of the same Lights, you will want to keep the importance the same. If you have multiple luminance, and using sun for example. One could set the sun as higher importance.

Radiance group. **Default 0**

How this works in TC. I have no idea.

Mappoint light.



The Lux Mappoint light emits light similar to point light. But with extra settings for adding a map, or IES dtata.

A map, is a texture whicih is normally a black and white image, where light is allowed to pass via the white areas, and reduced or excluded from the black areas.

IES (Illuminating Engineering Society) files, may be obtained from manufacturers or an IES library. And use real world data to simulate the light.

Enabled. **Default On**
Turns it on of off.

Gain, Position, Colour, Power and Efficiency, are the same as for Point light. Therefore only a brief description is stated here.

Gain. **See Page 28.** **Default 1, 1, 1**

Increases the power of the light, specific to the RGB channel chosen. Setting a gain channel to 0, turns off that colour.

Most often one would keep all the values equal to affect all the colours the same amount (grey scale).

Position. **See Page 28.** **Default 0, 0, 0**

Sets the position of the light relative to the source object in metres. See point light on previous page.

Color. **See Page 29.** **Default White**

Sets the primary colour of the light. Can be used with gain to isolate a particular colour.

Power. **See Page 29.** **Default 0**

Sets the power of the light in watts. A setting of zero will prevent the light from showing.

Efficiency. **See Page 29.** **Default 0**

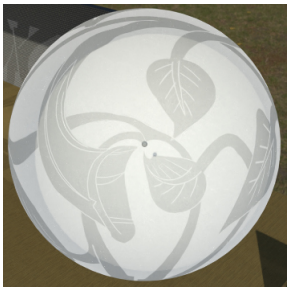
Sets the efficiency of the power level.

Map file. **Default “”**

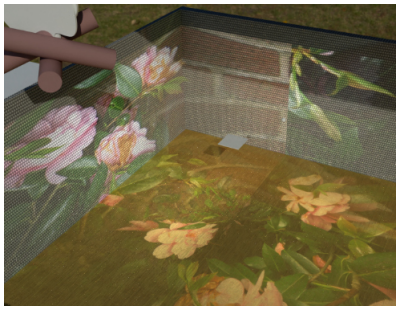
This is a 2d texture, which acts like projection, except it is projected as a sphere from the light source, as opposed to just an x, y image.

It can be bmp, jpg, tif or png formal. Not yet figured out how useful this could be. The image can be black / white, or full colour. Although it is projected as a sphere, some of the projection may be obscured by any shadow caused by the source object. See point light position on page 39

The image right is a black and white default image. It appears to allowed the light to shine more through the white parts, and restricted by the black parts.



As shown right, a colour image can be used. But it is still projection, so the surface on any objects (wall, floor etc.) will still show.



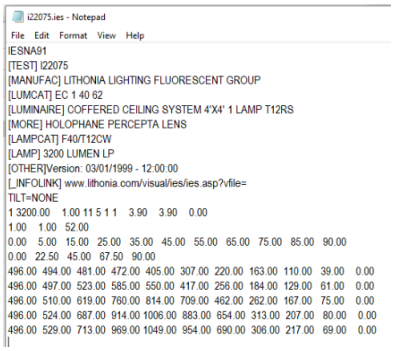
The pattern shows 360 degrees, but like any projection, it can stretch and contract depending on how the light hits an object. Shallower angles means more stretching.

Gamma. **Default 2.2**

Adjust the gamma value of the image map defined with .mapfile property, For example, if the projection is too light, adjust the gamma from 2.2 to something like 3.4 will darken and affect the tone of the image.

IES file. **Default “”**

An IES profile of emitted light. The render engine can ignore this information. -1 means use the default global value. Manufacturers measure the behaviour of their bulbs, and store their directional contribution into IES files (a text file). Those files are sometimes available for most of the bulb models directly from the manufacturers' web or via specialised sites.



i22075.ies - Notepad

File Edit Format View Help

```
IESNA91
[TEST] i22075
[MANUFAC] LITHONIA LIGHTING FLUORESCENT GROUP
[LUMCAT] EC 1 40 62
[LUMINAIRE] COFFERED CEILING SYSTEM 4'X4' 1 LAMP T12RS
[MORE] HOLOPHANE PERCEPTA LENS
[LAMPCAT] F40/T12CW
[LAMP] 3200 LUMEN LP
[OTHER]Version: 03/01/1999 - 12:00:00
[_INFOLINK] www.lithonia.com/visual/ies/ies.asp?vfile=
TILT=NONE
1 3200.00   1.00 11 5 1 1   3.90   3.90   0.00
1.00   1.00   52.00
0.00   5.00   15.00   25.00   35.00   45.00   55.00   65.00   75.00   85.00   90.00
0.00   22.50   45.00   67.50   90.00
496.00  494.00  481.00  472.00  405.00  307.00  220.00  163.00  110.00   39.00    0.00
496.00  497.00  523.00  585.00  550.00  417.00  256.00  184.00  129.00   61.00    0.00
496.00  510.00  619.00  760.00  814.00  709.00  462.00  262.00  167.00   75.00    0.00
496.00  524.00  687.00  914.00 1006.00  883.00  654.00  313.00  207.00   80.00    0.00
496.00  529.00  713.00  969.00 1049.00  954.00  690.00  306.00  217.00   69.00    0.00
```

IES blobs. **Default “”**

An IES profile of emitted light embedded in a blob. I’m guessing that this allows IES data to be manually added here, and embedded into the luminance, rather than using a separate file.

Flipz. **Default Off**

Flip the IES profile along the Z axis

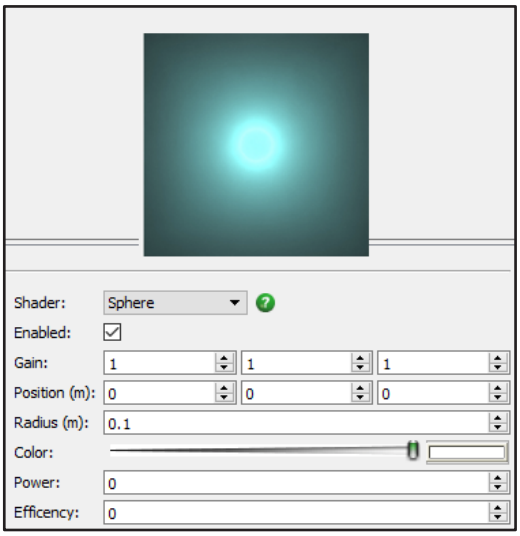
Map width. **Default 0**

Scale the image or IES map to the specified width. 0 means use the original size

Map height. **Default 0**

Scale the image or IES map to the specified width. 0 means use the original size

Sphere light.



The Lux sphere light emits light similar to point light. But whereas the point is an infinitely small light, the Sphere lights, allows for the adjustment of radius.

This makes it possible to achieve a more diffused effect, without the bright areas associated with the point light.

Enabled. **Default On**
Turns it on of off.

Gain, Position, Colour, Power and Efficiency, are the same as for other lights. Therefore only a brief description is stated here.

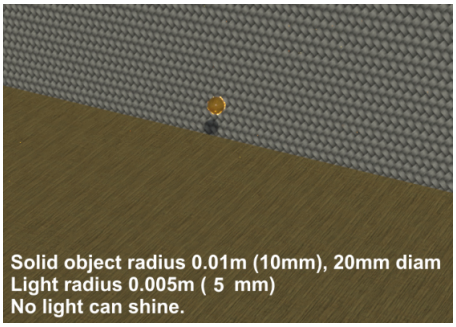
Gain. **See Page 28.** **Default 1, 1, 1**

Increases the power of the light, specific to the RGB channel chosen. Settig all channels to zero may prevent the light from showing. Setting just one channel, restricts the light to that one channel, For example, 1, 0, 0 restricts it to the red channel.

Most often one would keep all the values equal to affect all the colours the same amount (grey scale).

Position. **See Page 28.** **Default 0, 0, 0**

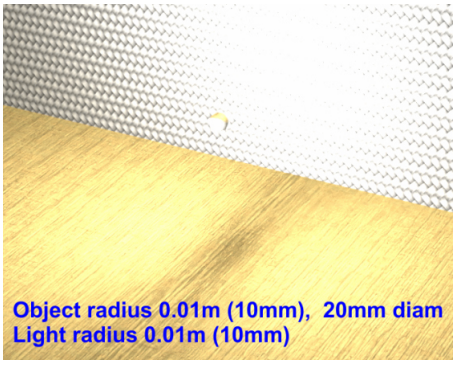
Sets the position of the light relative to the source object. Normally lights need to be set to outside a solid object and can be set inside on transparent objects. However because Sphere light has a radius, it can have the radius high enough so it is outside of solid objects.


Solid object radius 0.01m (10mm), 20mm diam
Light radius 0.005m (5 mm)
No light can shine.

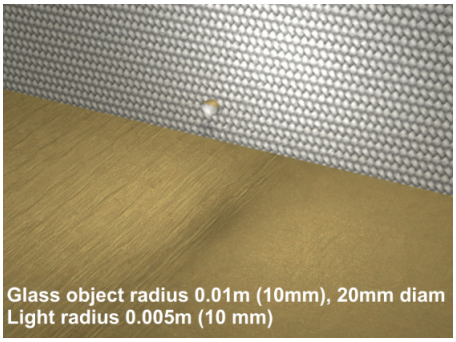
Radius. **Default 1 metre**

Radius is in Metres. This Sets the radius of the light sphere. It does not affect any associated object size. If the position is set within the object (i.e. 0, 0, 0). The radius should be at the minimum, sized large enough to be on or above the surface of the object.

For example. If the source object solid and is 20mm diam, and the light position is set at 0,0,0. Then the radius should be a minimum of 0.01 metres. To bring the light onto the objects surface.


Object radius 0.01m (10mm), 20mm diam
Light radius 0.01m (10mm)

If the source object is transparent (example - glass), then the light radius can be inside the object, but it will likely be affected by refraction and absorption. And thus reduce the power outside he object.


Glass object radius 0.01m (10mm), 20mm diam
Light radius 0.005m (10 mm)

Color. **See Page 29.** **Default White**

Sets the primary colour of the light. Can be used with gain to isolate a particular colour

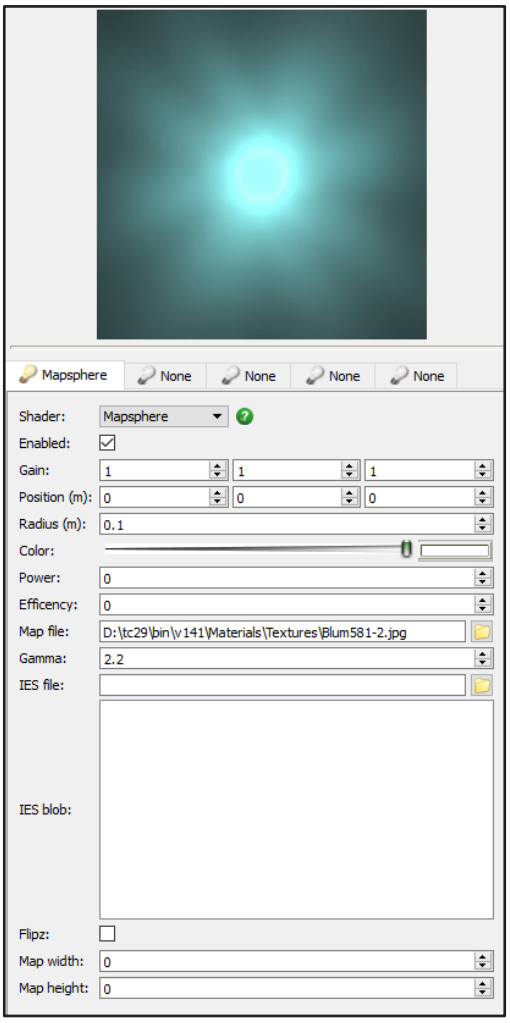
Power. **See Page 29.** **Default 0**

Sets the power of the light in watts. A setting of zero will prevent the light from showing

Efficency. **See Page 29.** **Default 0**

Sets the efficiency of the power level.

Mapsphere light.



The Lux Mapsphere light emits light similar to sphere light. But allowed for extra control via map and IES files.

Enabled. **Default On**
Turns it on of off.

Gain, Position, Colour, Power and Efficiency, are the same as for Point light. Therefore only a brief description is stated here.

Gain. **See Page 28.** **Default 1, 1, 1**

Increases the power of the light, specific to the RGB channel chosen. Settig all channels to zero may prevent the light from showing. Setting just one channel, restricts the light to that one channel, For example, 1, 0, 0 restricts it to the red channel.

Most often one would keep all the values equal to affect all the colours the same amount (grey scale).

Position. **See Page 28.** **Default 0, 0, 0**
Sets the position of the light relative to the source object. Normally lights need to be set to outside a solid object and can be set inside on transparent objects.

However because Sphere light has a radius, it just need the radius high enough so it is outside of solid objects.

Radius. **Default 1 metre**

See Sphere light on previous page.

Color. **See Page 29.** **Default White**

Sets the primary colour of the light. Can be used with gain to isolate a particular colour.

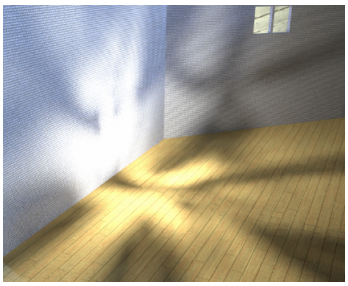
Power. **See Page 29.** **Default 0**

Sets the power of the light in watts. A setting of zero will prevent the light from showing.

Efficency. **See Page 29.** **Default 0**

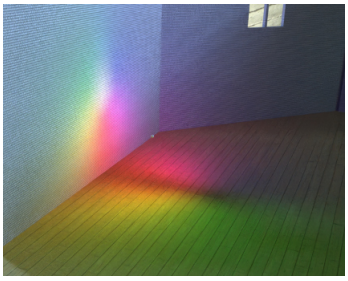
Sets the efficiency of the power level.

Map file. **Default “”**



This is a 2d texture, In bmp, jpg, tif or png formal. Not yet figured out how useful this could be.

Colours images can be used as the map. Though the effect is more of a specialised one-off effect.



Gamma. **Default 2.2**

Adjust the gamma value of the image map defined with .mapfile property.

IES file. **Default “”**

An IES profile of emitted light and the render engine can ignore this information. -1 means use the default global value.

IES blobs. **Default “”**

An IES profile of emitted light embedded in a blob.

Flipz. **Default Off**

Flip the IES profile along the Z axis.

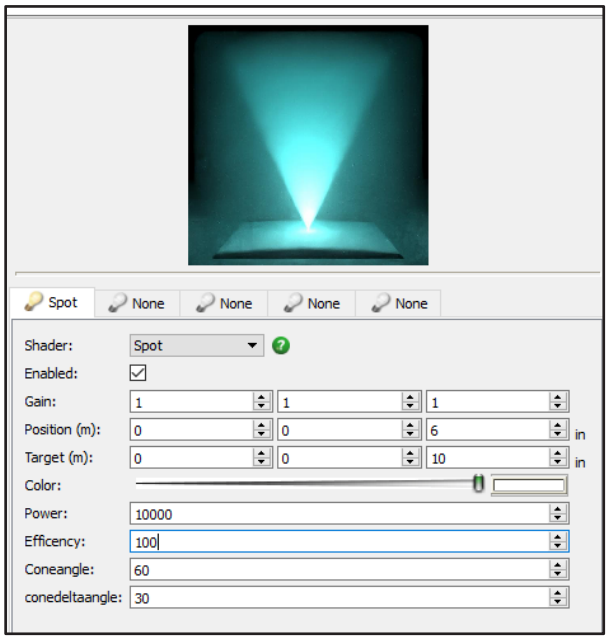
Map width. **Default 0**

Scale the image or IES map to the specified width. 0 means use the original size.

Map height. **Default 0**

Scale the image or IES map to the specified width. 0 means use the original size.

Spot Light (Luminance).



Spot luminance emits a cone shapes light, set via an inner and outer cone angle. there does not appear to be any setting for fall-off, i.e. the distance away from the light source, that the light begins to fade. It is assumed to be quadratic.

Enabled. **Default On**

Turns it on of off.

Gain. **See Page 28** **Default 1, 1, 1**

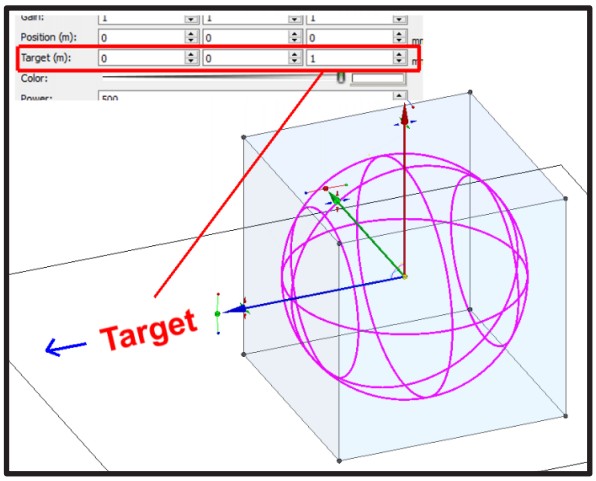
Increases the power of the light, specific to the RGB channel chosen.

Most often one would keep all the values equal to affect all the colours the same amount (grey scale).

Position. **See Page 28** **Default 0, 0, 0**

Sets the position of the light relative to the source object. Must be outside a solid object, but may be inside a translucent or transparent object (glass)

Target. **Default 0, 0, 1**



This is the primary direction one wishes the light to shine, and is in x, y and z coordinates of the source object. Example. In our solid sphere above, the Z axes is pointing where we want the light to shine. And the position is set to 0,0,0.01m. Therefore a target position of 0,0,1 will ensure success.

The target can be positive or negative numbers. To reflect a positive or negative object entity axes

A couple of things to keep in mind. If the target is set the same as the position, usually no light will shine. Therefore always set the target different to the position.

A second thing is. If you have a position of say 0,0,0.1m and wishes the light to shine in the positive z direction. One could set 0,0,1. However If one sets the target to 0,0,0.01m . i.e. the target is closer to the source than the position. The light will shine backwards towards the source. Therefore in most cases, you should set the target greater than the position.

Always remembering, the boxes are for object entity coordinates. Personally I just use either 0 or 1. There is often no need to enter decimal numbers for target.

Color. **See Page 29** **Default White**

Sets the primary colour of the light.

Power. **See Page 29** **Default 0**

Sets the power of the light in watts.

Efficency. **See Page 29** **Default 0**

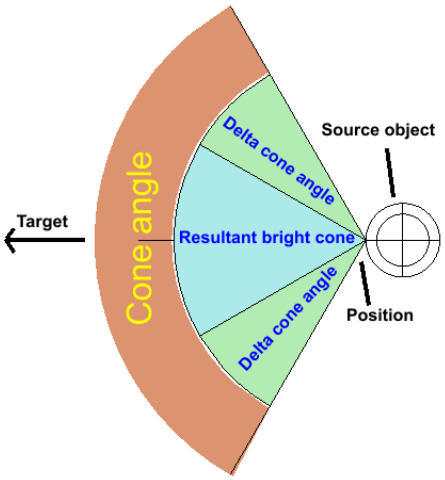
Sets the efficacy of the power level. In Limens per watt.

Coneangle. **Default 30 angle in degrees of the spotlight cone**
Conedeltaangle. **Default 5 angle at which the spotlight intensity starts to fade from the edge**

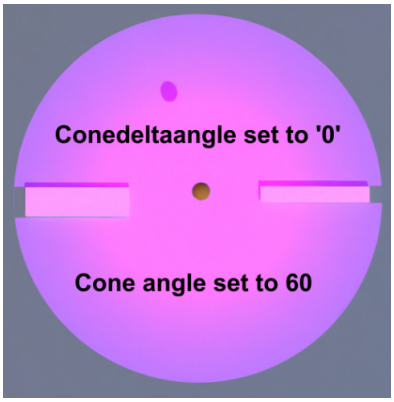
These have been put together as they are intertwined, altering one directly affects the other.

In the cone are two angles, the bright part, and then we have the outer fall off. Some rendering engines has a cone angle, and an additional cone angle, over and above the cone angle setting

In Lux, The cone angle is essentially the maximum angle of the light. i.e. both bright part and fall off, or fading angle. The actual angle is twice the angle set. Making the angle a radius of the cone, not the diameter.



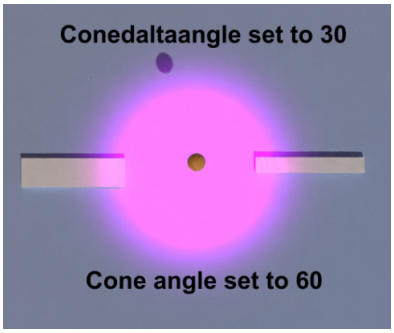
The brighter part of the cone angle is altered by the codedeltaangle. In that increasing the conedeltaangle, decreases the size of the bright cone, therefore more of the light is faded towards the edges.

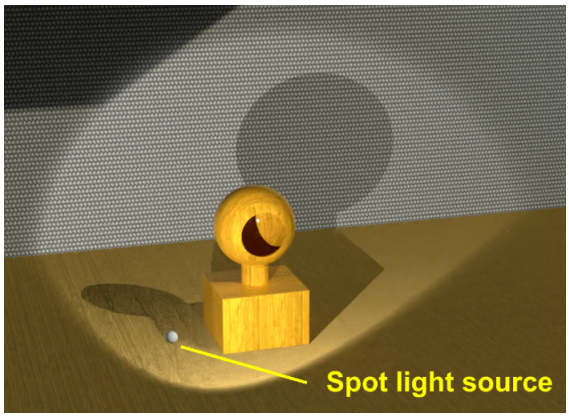


A 60 degree Coneangle angle, emits bright light as a 120 degree cone (2 x 60 degrees), if the delta angle is set to zero.

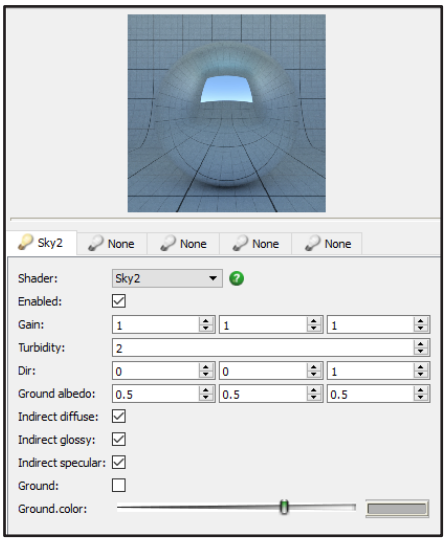
A conedeltaangle of 30 degree, will reduce the centre cone by 30 degrees all round.

Therefore the brighter inner cone will be 60 degrees (2 x 30 degrees). The area between the inner 60 degree and the outer max 120 degrees is fall off or fade.





Sky2 Light



The Luxcore help pages state

" The sun and sky light creates a lighting setup that simulates the light of the sun and atmosphere, based on the direction of a sun light in the scene and a parameter named turbidity which defines the clearness of the sky. Both the sun angle and the sky clearness influence the color of the light.

Note that the sun lamp is very bright (as it is in real life) so you may have to turn the Gain way down if you have other lights in the scene. The other lights may look like they are not even emitting, but they are actually overpowered by the sun. It is possible to use this light with sun only, sky only, or the default sun+sky. If you want softer shadows try adjusting the relative sun size parameter.
Note that the sun lamp's location in the scene does not matter, only its rotation. "

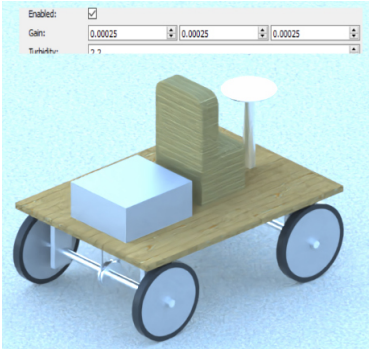
Enabled. **Default On**
Turns it on of off.

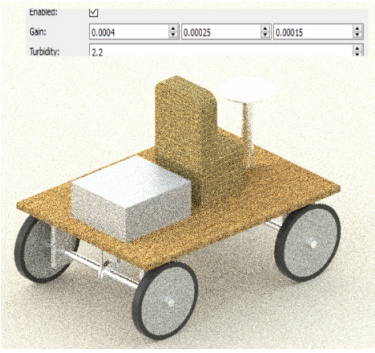
Gain. **See Page 28.** **Default 1, 1, 1**

Sets the Red, Blue, Green colour spectrum channel. All figures being equal, results in grey scale adjustment. The sky light has no power setting. Therefore the gain alone is used to increase the light power. Often a very small number is required, the images (right), have a gain of 0.002, 0.002, 0.002

At the default settings, and when all RGB entries are the same. Sky2 can give a blue haze. As shown in the first image right.

Changing the gain so that values are not equal. Can be used to compensate for the blue haze. As in the far image right.



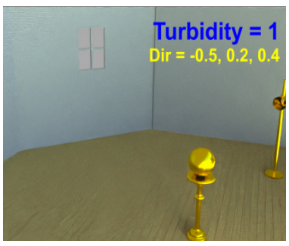


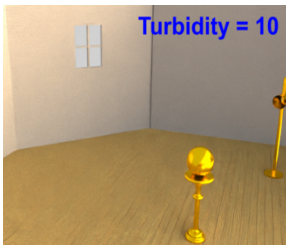
Turbidity. **Default 2.2**

Turbidity normally sets the amount of particles present in the atmosphere. I couldn't find much information about Lux turbidity. A turbidity of 1 normally means a perfect clear sky, while a turbidity of 5 means an average haziness sky.

The normal range of turbidity values is 1 to 30+. Although normally a higher turbidity, effectively darkens the scene. In a similar way that a cloudy sky means it is darker than a clear sky. In TurboLux, things are not that clear cut.

As can be seen, a higher turbidity can completely brighten the scene.





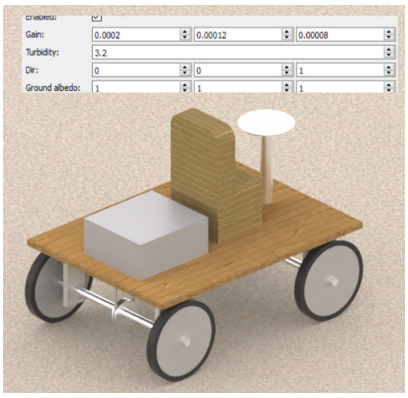
Dir. Default 0, 0, 0

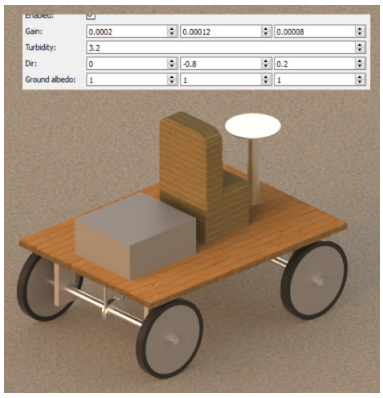
Direction vector of the sky's primary light source. The range is between -1 to 1. Though minus figures probably won't make much difference, for this type or light. This can affect how bright the sky is, for example 0, 1, 0 basically puts the sky light on the horizon. Setting the dir above '0' for the z direction, illuminates the sky. (Example 1, 0, 0.1).

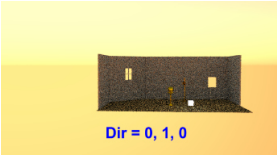
In the images right. We alter the Dir rather than alter the object.

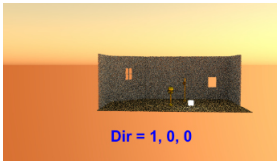
One thing to note, is that altering the direction can darken or lighten the effect, so it may need compensation, by increasing the gain.

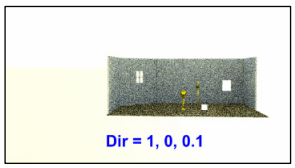
The actual colour of the background, can vary due to a mixture of Gain, Turbidity and Dir.









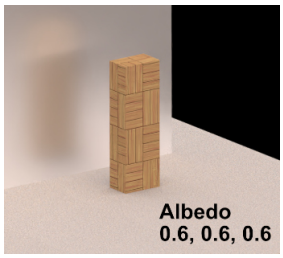


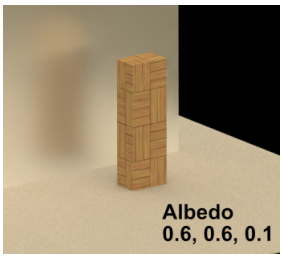
Ground Albedo. Default 0.5, 0.5, 0.5

Sets the ground albedo. This defining how much light is reflected back to the sky by the ground.

The three RGB channels can be used to change the light reflecting. In the far right, the blue channel was turned down, the RGB red and green = yellow.

Mostly one would leave all three channels equal, to avoid any colour changes





Indirect diffuse. Default On

"if the light source is visible by indirect rays on diffuse surfaces. This is an hint and can be optional supported by the render engine. It can be used to remove fireflies".

Indirect glossy. Default On

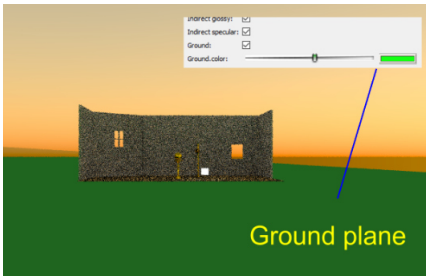
"if the light source is visible by indirect rays on glossy surfaces. This is an hint and can be optional supported by the render engine. It can be used to remove fireflies."

Indirect specular. Default On

"if the light source is visible by indirect rays on specular surfaces. This is an hint and can be optional supported by the render engine. It can be used to remove fireflies."

Ground. Default On

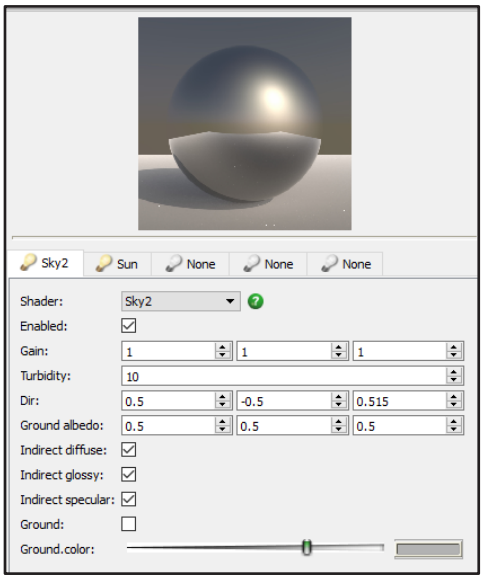
Turns it on of off. Sets a coloured plane on the ground, the orientation of which, is based on the object which the sky is attached to. Though the plane is at the drawings lowest point.



Ground color. Default Mid Grey (RGB 128, 128, 128)

Sets the colour of the ground plane. This is in RGB formal, which one can pick using the colour selector.

Sky Morning Light



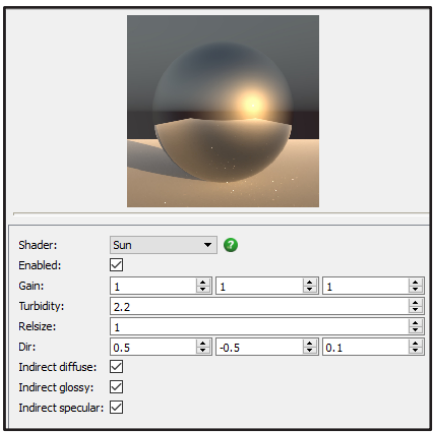
This is essentially the same as Sky light. But with the turbidity turned up. And a much lower direction.

Therefore all the same options apply, and won't be duplicated here.

The higher turbidity creates a hazier shy and the lower direction creates longer shadows.

See Sky luminance on page 48 for settings.

Sun.

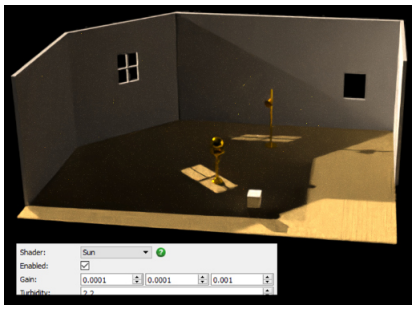


Sun and sky could be used together, or separately. Sun on its own differs by not having any ground settings.

Unlike a spot light which emits a cone. The sun emits parallel rays so all parts of the scene will be illuminated equally as per the direction set.

One extra settings is provided, and that being Relsize.

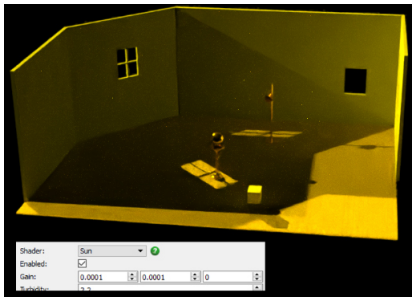
Enabled. **Default On**
Turns it on of off.



Gain. **See Page 28.** **Default 1, 1, 1**

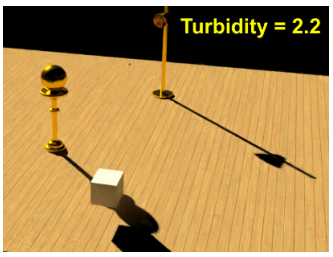
The sun can be very powerful, low settings may be required, for example
0.001, 0.001, 0.001. Keeping the gain to all the same figures alters the power of the sun. Changing the grey scale.

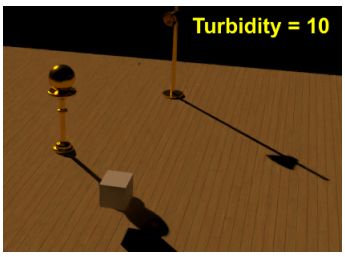
The colour can be changed if required. Example 0.001, 0.001, 0. Will change the sun colour to yellow (which is red and green but no blue). However such drastic changes in colour may look odd. (image on the right)



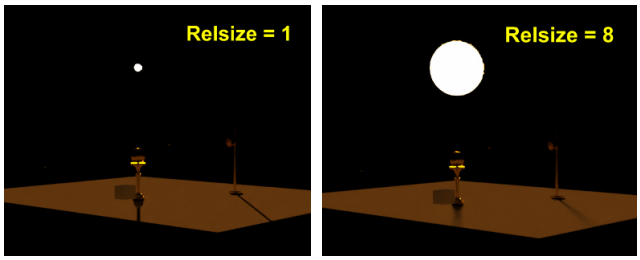
Turbidity. **Default 2.2**

Turbidity works well with sun. It creates haziness in the path of the sunlight, and thus can darken the scene. Whilst retaining some of the specular effect.





Relsize. Default 1

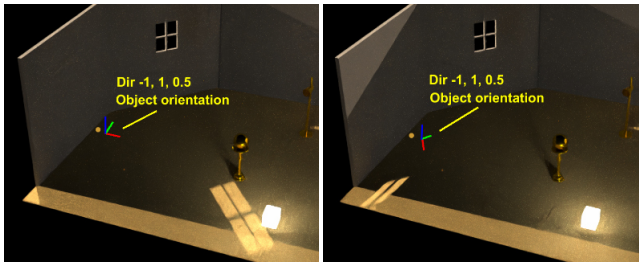


This determines the size is the sun sphere. It doe not affect the power (gain) of the sun brightness. However larger sun size and produce softer shadows.

If one needs to show the sun, for example, through sky, clouds etc. the the size can be adjusted to fit in. However if the sun can not be see, it can be left at relsize = 1, or changed to something like 4, to affect the shadows.

One way the relsize can make a negative difference, is if it is made too big. This can put the sun too big for the environment dome, and reduce the sun intensity.

DIR. Default 0, 0, 0



Direction operates as a vector. In the range of -1 to 1. This is the direction where the sun is located, not the way it will shine. Therefore, if one wishes the sun to shine in the positive z direction, us 0, 0, -1 (negative z).

If the sun luminance is attached to an object, the zero vector is taken from the objects 0, 0, 0 (centre extents).

This means that, if the object is rotated, the sun will also rotate,

Often the luminance would be attached to an object that cannot be rotated. Therefore changing the Dir vector is the best solution to change the sun direction.

Direction vector affects the x, y and z direction of the sun.

Indirect diffuse. Default On

Help file states *"if the light source is visible by indirect rays on diffuse surfaces. This is an hint and can be optional supported by the render engine. It can be used to remove fireflies"*.

Indirect glossy. Default On

Help file states. "if the light source is visible by indirect rays on glossy surfaces. This is an hint and can be optional supported by the render engine. It can be used to remove fireflies."

Indirect specular. Default On

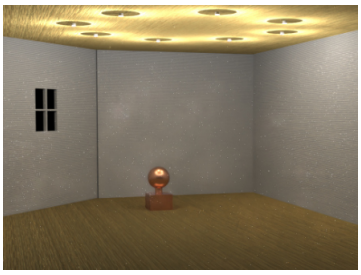
Help file states "if the light source is visible by indirect rays on specular surfaces. This is an hint and can be optional supported by the render engine. It can be used to remove fireflies."

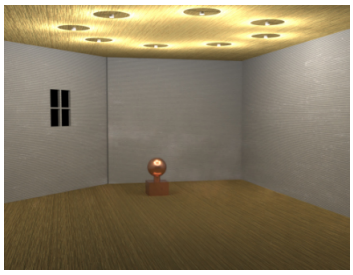
Combining luminance.

On many occasions one would may need to combine multiple luminance.

The obvious ones being combining sky and sun., The main thing to watch for is the gain, as combining luminance can make scenes too bright. And not forgetting to set the correct sun vector for ones drawing (see appendix 1, vectors).

There are many other drawings where multiple luminance is required, such as ceiling lights. One problem may be noise. Without some overall lighting, like area / HDRI etc. noise may be increased, and one may need to fiddle with the settings, to limit the noise in the scene.





The images right, have no scene lighting set. The only lights are 8 point lights attached to the ceiling.

The first image right, shows more noise from a standard path render.

The second right image, has reduced noise. The denoise setting was adjusted, and a longer render time allocated.
There is still some noise, so further adjustment is required.

Materials

Tlux (and Luxcore), provide a number of basic materials, which can be customised by the user. It does not have a ‘Default’ material. Therefore any created material will be a variation of one of the included materials.

One selects the type of material. Whether Matte, Glossy, transparent (Glass) etc. Which will load the default settings for that type of material. After which, the user is free to use just a colour, apply a texture (2d image file), add a bump map or normal map. Changed transparency etc.

The options available vary depending one the material type chosen. And will be covered late, in the material description

The material types are.

Matte.

Matte Translucent.

Rough Matte.

Glossy2.

Glossy Translucent.

Archglass.

Glass.

Rough Glass.

Mirror.

Metal2.

Carpaint.

Cloth.

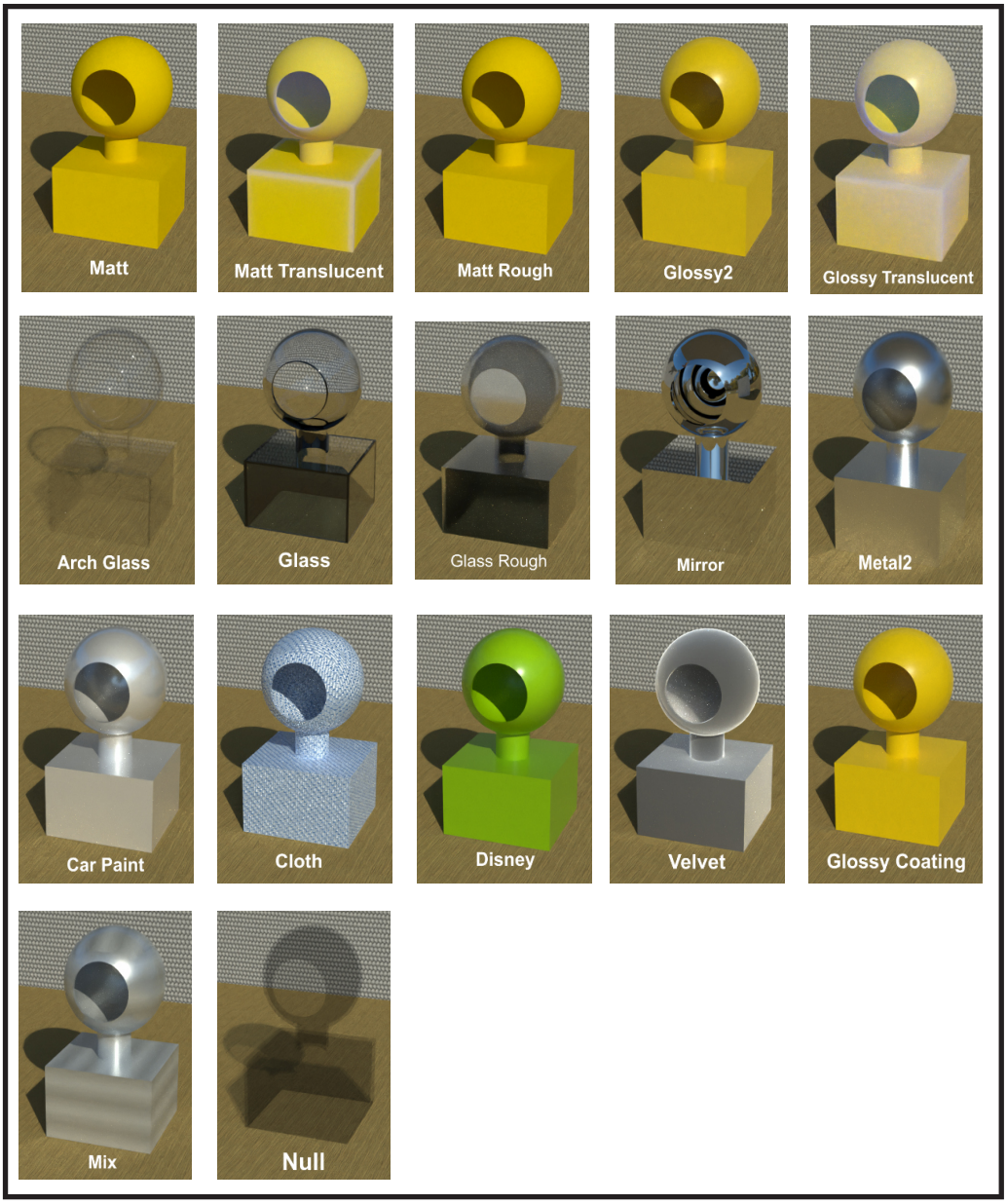
Disney.

Velvet.

Glossy coating.

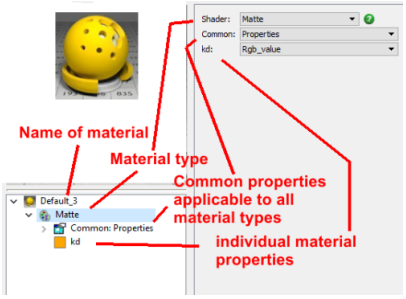
Mix

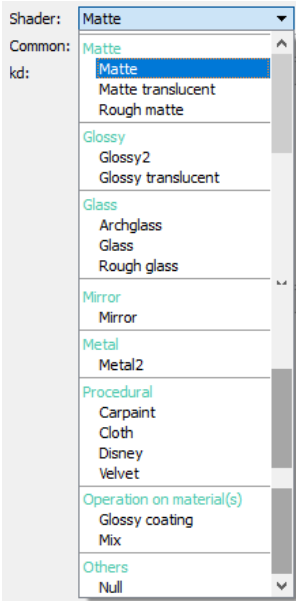
Null.



The first thing is the hierarchy.

The **material name** can be one of TC's built in material, like plastic, glass, leather etc. Or a default name changed to one own liking. This has no effect on the material itself.



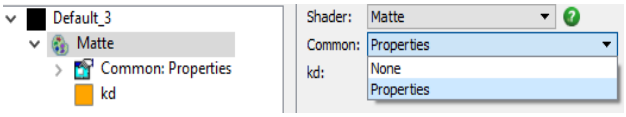


Material type or Shader.

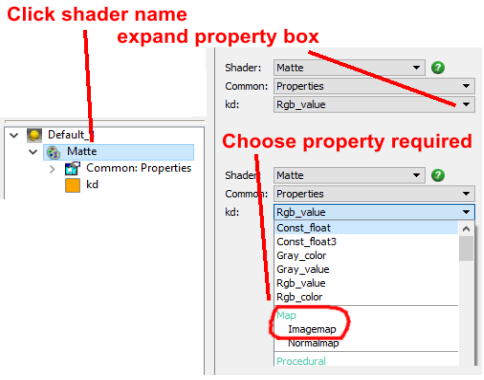
This is a standard shader, access from the first shader drop down box. This loads one of the primary materials, which will then open various properties depending on the material chosen.

Once a material shader has been chosen. The **drop down arrow head** in the left pane is used to expand more options. The first option is Common properties, with only two options in the drop down box. None and Properties.

Choosing common properties allows one to select, volume, emission, etc. Below common properties in the left pane, are all the properties associated with the chosen material shader. Matte only has one extra property (kd).



Most options in TurboLux are done through drop down boxes. For example. If one wanted to use an image to wrap the object, instead of a colour. Expand the property drop down box of kd, and choose image map from the drop down list.

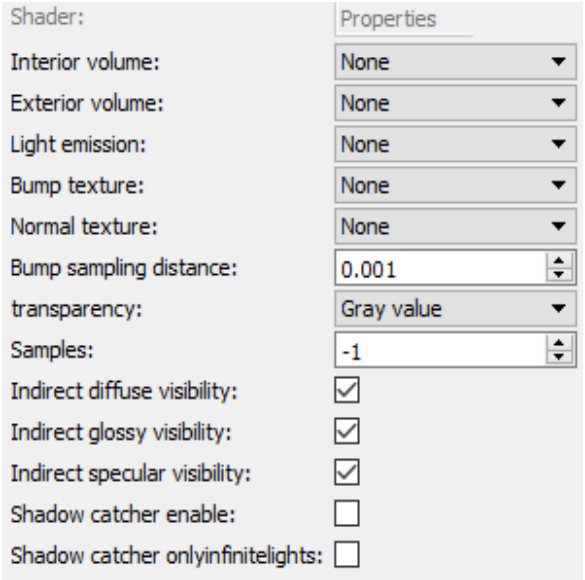


To change a property. Select it in the left pane, and it will open adjustable options in the right pane.

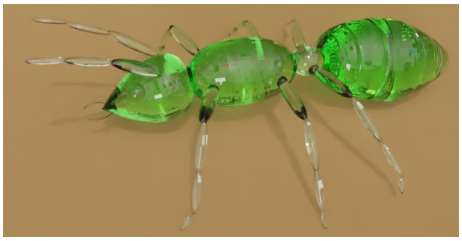
Common properties…

Common properties, simply means something that is inclusive in all shaders, There are some not applicable. For example Null shader cannot use a bump map. As there is nothing there to bump. Included in the common properties are -.

Property	Page number
Internal Volume	55
External Volume	61
Light Emission	62
Bump texture	64
Normal Texture	67
Bump sampling distance	
Transparency	68
Samples	70
Indirect diffuse visibility	70
Indirect glossy visibility	70
Indirect specular visibility	70
Shadow catcher	71
Shadow Catcher onlyinfinitelights	71



The following pages cover the common properties, going down the listing.



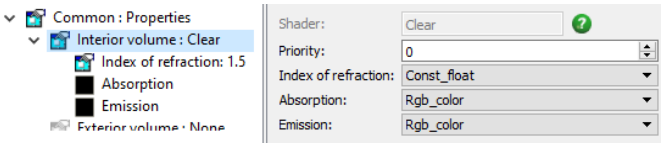
Interior Volume.

Its main function is to Control how light behaves inside an object, and can be used to put colour or coloured particles inside a translucent object. Interior volume would normally be left at 'None' for all opaque materials, like, metal, and plain plastic, etc. And also for transparent objects like plain glass, and architectural glass.

It can be used on glass or translucent objects, where thinner parts of the object with show clear or little colour, whilst thicker parts of the object show a deeper colour, or particles. It can also be used for liquids, smoke, dust etc..

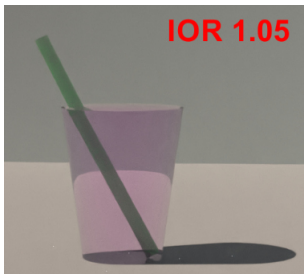
There are three volumes associated with this, Clear, homogenous, and hetorogenous.

Clear Volume.



From the Luxcore help file.
Clear is a simple volume that features refraction and absorption. It is primarily meant to be used with clear materials, such as colored glass. It can also be used to add some extra realism to translucent materials with little performance cost. It has two properties, an index of refraction(IOR) and an absorption color.

IOR. Default Const_float 1.5

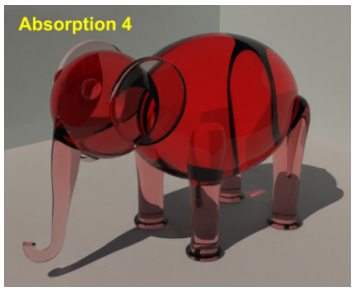


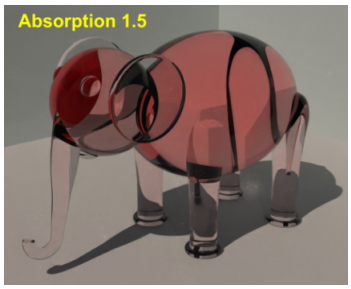
The index of refraction setting determines how the light refracts within the object. The IOR setting in Glass material is disabled when this setting is used. Likely to prevent any confusion which could occur with two identical settings in two different places.

IOR varies by material. For anyone not familiar with IOR, See page … for a brief overview. See page 155 for some example IOR values.

Absorption color. Change drop down box to Colordepth. Default RGB colour Black

First. It was advised on the TC forum to change the absorption drop down box, from rgb, to colourdepth (page 139). This allows for the color chosen to be the one displayed. With rgb the colour chosen would be inverted.





This is the colour absorbed by the object as the light passes through the material.

Depending on the absorption depth chosen, Where the material is thin the light can pass through easily and very little light is absorbed. Where the object is thicker, more of the light can be absorbed by the object, which can change the colour inside the object.

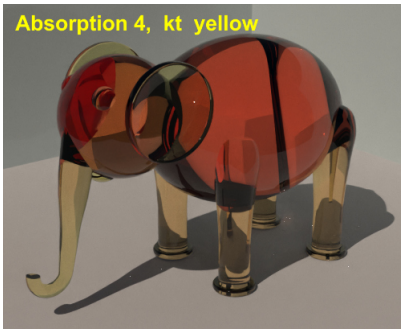
If using colour depth, this is the kt colour. If using rgb, calculate the inverted colour.

Absorption affect. Default 1

Absorption depth is often trial and error, as the depth varies with the size and shape of the object. Often the lower the depth, the more light has a chance to be absorbed and the darker the effect,. Higher values means the light may pass through the object before it can be absorbed.

This lends itself very well to objects with variable thickness,

The effect can depend on the transmission colour of the object. An ideal situation would be on clear glass, where the transmission colour of the glass is white (clear). In this situation, the absorption colour determines the colour of the light as it passes through the object. Thus making the process more predictable.

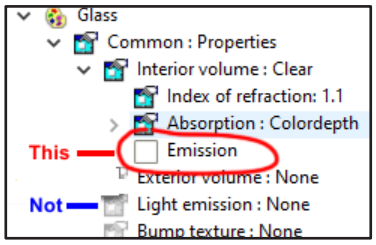


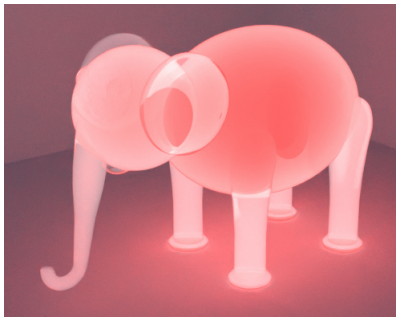
Using a **kt** (transmission) colour in the objects material settings, (not the colour depth settings) can change the colour of the absorbed light. As can be seen in fig right It is possible to get two colours in an object, yellow kt and red absorption.

For something unusual , One can add a transmission map as **kt**. The effects can be trial and error. But if one just wants something different as an ornament in a scene. it could be worth playing around with different maps..

Emission. Default RGB colour Black

Emission works with volume effects.



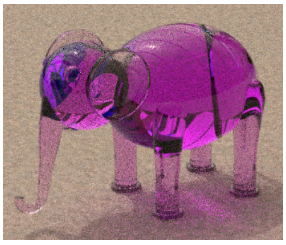


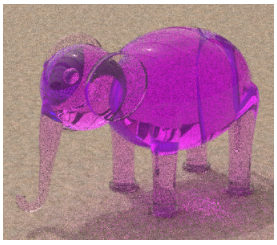
Because the emission is effectively inside the object, the effect may be more subdued than using the main light emission.

To get the requires luminance the emission may need to be greater then normal. The image right uses an emission of 100,000. NOTE this is volume emission. Not material emission.

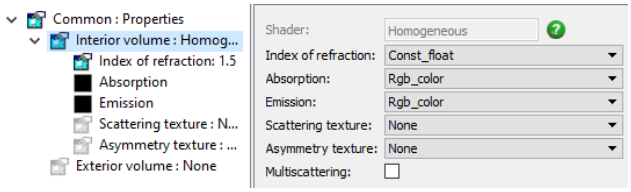
One problem with TurboLux volume, is black or dark areas If you look at the absorption elephant above, you will notice dark patches. These are not always easy to eradicate.

Reducing the IOR can often help. Another option would be to add an emission to the volume effect. Far right has volume emission set to white. And reduces the black effect. But because it is volume emission, instead of object emission, it does not overly brighten the scene.





The homogeneous volume



From the luxcore help file. *Represents a volume with an even distribution of microscopic particles. It can be used for subsurface scattering (SSS) such as milk or orange juice. It can also be used for atmospheric scattering such as a dusty room with visible sun rays.*

Please note that atmospheric scattering is very light. Set Scattering Color to full white and try setting Scattering Scale to 0.001. To simulate it efficiently, you should keep multiscattering disabled in your exporter. This will greatly reduce the noise in the scene. Multiscattering should be used with heavy scattering volumes, such as milk or orange juice. Homogeneous uses the same index of refraction and absorption parameters as the clear volume. In addition to those, it has two extra parameters, scattering color and scattering asymmetry.

IOR. **Default Const_float**

The index of refraction setting determines how the light refracts within the object. The IOR setting in Glass material is disabled when this setting is used.

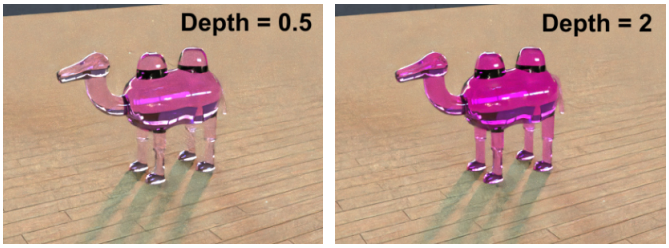
IOR varies by material, See page … for some example IOR values.

Absorption. Change drop down box to Colour Depth **Default RGB colour**

This is the same as for Clear volume. Better to change the setting from rgb to colour depth, using absorption drop down box.the drop

Depth (colour depth). **Default RGB colour**

Absorption depth is often trial and error, as the depth varies with the size and shape of the object. Often the lower the depth, the more light has a chance to be absorbed and the darker the effect,. Higher values means the light may pass through the object before it can be absorbed.



Absorption color. **Default RGB colour**

Although this is the same as for Clear volume. With the introduction of scattering colour below. The effect can change depending on the scattering and absorption colours. It can be sometimes be preferable to let the absorption take care of the colour, and leave scattering at grey scale. This will alter the scattering to work without hanging the actual colour (it can make the colour darker, or lighter).

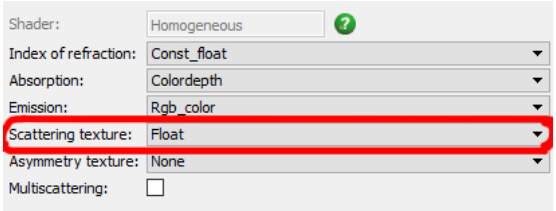
Using a transmission colour in the objects material settings, can change the colour of the absorbed light. And produce some odd effects,

Emission. **Default RGB colour**
Emission works with volume effects,

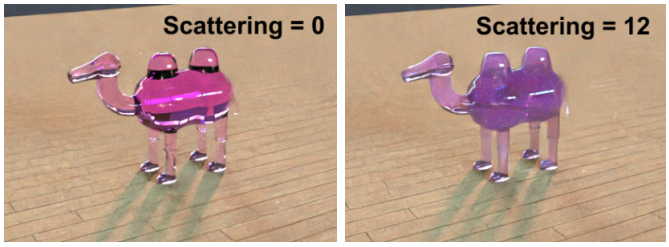
Scattering Texture. **Default None**

From the luxcore help file. *The scattering color determines the color and density of the particles. Higher values are denser. This control can also affect the color of your volume, but that will be determined primarily by the absorption color.*

Fig 182



While the scattering color has red, green, and blue values, it is not limited to this range. You can specify values higher than 1 (pure white). In fact, for heavy interior volumes you will often need to use values in the 30-1000 range. To help deal with this wide range of values, your exporter will present you with a set of RGB values and a "scale factor" that the RGB values will be multiplied by.



Scattering asymmetry. **Default none**

From the luxcore help file. *The asymmetry value determines the scattering direction. A positive value is primarily forward scattering (light is scattered in the same direction as the ray was traveling) and a negative value is backwards (back towards where the ray came from). Clear particles have a more forward scattering and opaque particles a more backwards scattering. The default value 0 scatters light evenly in all directions.*

In LuxCoreRender, SSS is done in the following way:

You take a material that allows light to shine through it. This can be Null (completely transparent), Glass, Glossytranslucent or Mattetranslucent.
If you use Glossytranslucent or Mattetranslucent, set a dark diffuse color and a bright transmission color. That is because reflected light cannot be transmitted (due to energy conservation), so if you have a bright reflection (diffuse) color, less light will be transmitted.
Use a volume with scattering support as interior volume, most of the times you want the Homogeneous volume.
To control the scattered color, I would recommend to use the absorption color of the volume. So if you want that white light gets a red tint when scattered in the material, set the absorption color to red.
Remember to also set the absorption depth of the volume to the depth where the light should be completely red, e.g. 1 cm.
Leave the scattering color of the volume white and use the scattering scale to control the scattering strength.

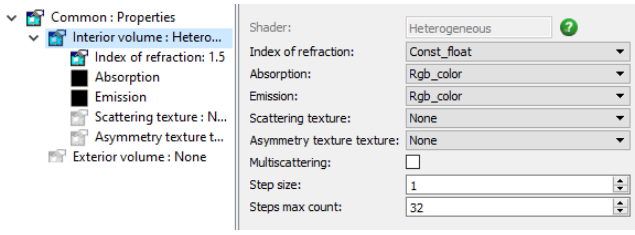
Homogeneous can be used for volumetric effects, like light rays in a dusty room.

Fig… uses a 3d box inside the room , the box is set to homogeneous volume, with Absorption = black, scattering 0.1, and IOR 1.1 scattering was required only in the direction of the rays, so asymmetry was not set.

Multi scattering (tick box). **Default Off**

Turns of / off multi scattering. For most renders one would probably leave multiscattering turned off. I have read that turning it on can increase rendering times. And for the most part, it is unnecessary.

Heterogeneous Volume



From the luxcore help file.
The heterogeneous volume is the most powerful (and also the slowest) of the three volume types. It has the same functionality as the homogeneous volume, but it also includes ray-marching support. This means it can deal with volumes with varying internal properties, such as clouds, smoke, and ground-hugging fog. It has the same properties as the homogeneous volume, with one extra setting, step size.

The heterogeneous volume can be used as the exterior volume as well, although this comes at a substantial performance cost (even when not using multiscatter) and is not recommended in most cases.

The clear and homogeneous volumes evaluate volume properties only once as it enters the volume. They assume these properties hold constant all the way to other side of the volume. For volumes that are continuous, this is a helpful optimization. However, if the volume is not continuous (such as a cloud), this causes blurred details and the edges of the volume container to become visible. In order to handle changing details, heterogeneous subdivides the volume transit path into several sub-paths (a process known as ray-marching) and evaluates the properties again at each sub-point.

NOTE. The following example use a Null material, so there is no outer shell. It is just the volume itself which is displayed.

Null material. Only the volume is displayed

IOR.

The index of refraction setting determines how the light refracts within the object. The IOR setting in Glass material is disabled when this setting is used. Using IOR on a Null material, likely won't make any difference.
.

Absorption (Changed in the drop down box to Colordepth).

This is the same as for Clear volume. Better to change the setting from rgb to colour depth, using absorption drop down box. Uses kt to adjust the colour of the absorbed light.

Depth (colour depth).

Absorption depth is often trial and error, as the depth varies with the size and shape of the object. Often the lower the depth, the more light has a chance to be absorbed and the darker the effect,. Higher values means the light may pass through the object before it can be absorbed.

Absorption color.

Although this is the same as for Clear volume. With the introduction of scattering colour below. The effect can change depending on the scattering and absorption colours. It can be sometimes be preferable to let the absorption take care of the colour, and leave scattering at grey scale. This will alter the scattering to work without hanging the actual colour (it can make the colour darker, or lighter).

Using a transmission colour in the objects material settings, can change the colour of the absorbed light. And produce some odd effects,

Emission

Emission works with volume effects,

Scattering texture.

According to a tutorial I saw, it said ‘Scattering works slightly different with heterogeneous’.

However, except for its interaction with step size, I didn’t find much difference in TurboLux. This may simply be because I have not found the correct settings, or the different way TurboLux is programmed compared with Luxcore.

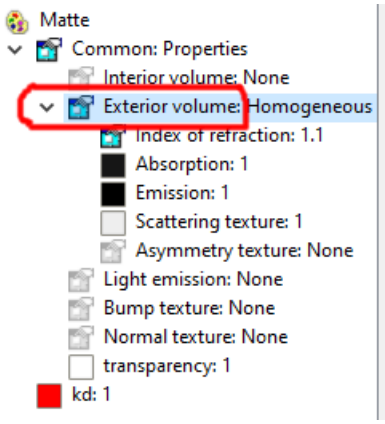
Step Size

Step size defines the spacing between the ray-marching sub paths, in meters. Smaller steps will show greater volume detail, but are slower to render. Setting too large of steps may result in a blocky or noisy appearance of the volume.

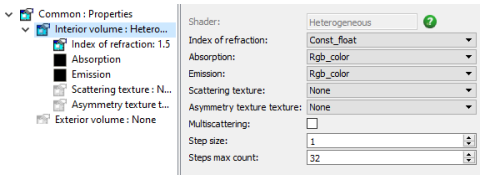
At the time of writing, I found very little difference,

External Volume.

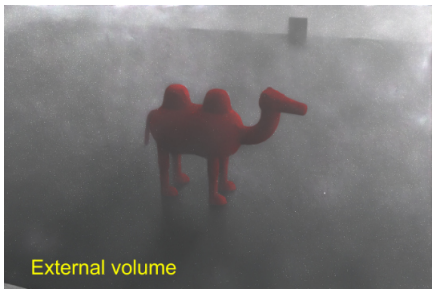
Use with caution. It is use is to put particles in the scene, outside an object, I can produce good effects, like fog, or light rays for instance, but overuse can severely impact on rendering times.

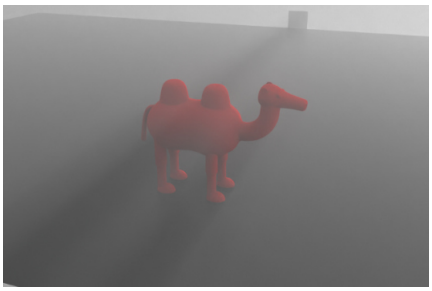


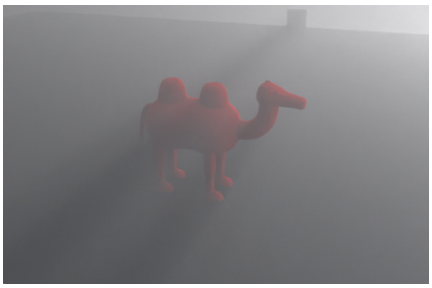
Volume parameters are the same as for interior volume.



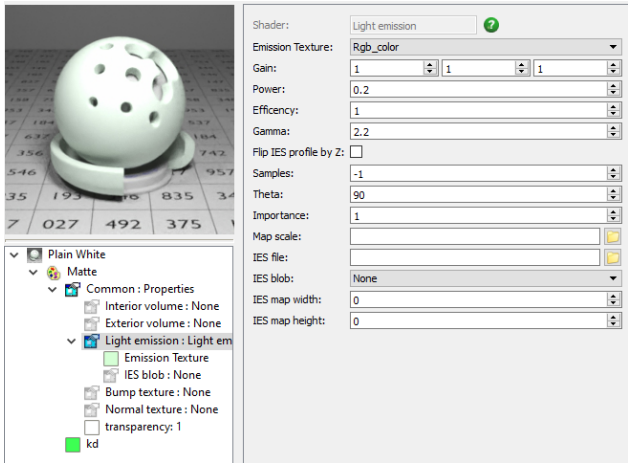
Used to add a fog or a smoky room.







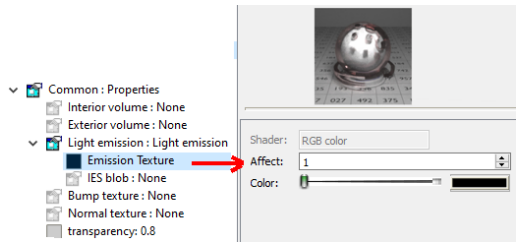
Light emission. Default None



Note. The volume options have a separate emission, that works with volume absorption. This is not the same as this main light emission setting.

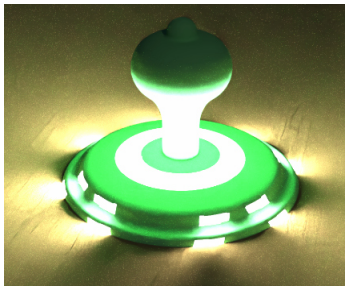
Light emission allows an object to emit light like a light bulb, without the need for attaching a luminance. It also allows one to use IES light date for more accurate lighting.

One difference to using a Luminance, is that the object itself emits the light, and becomes bright. Whereas with luminance, one can set the light away from the object.



Emission texture. Default RGB colour

Although called a texture. Mostly it would be set at grey colour or rgb colour. This opens another setting where the colour can be set. This is the colour of the light, but it will be affected by other settings like the power.



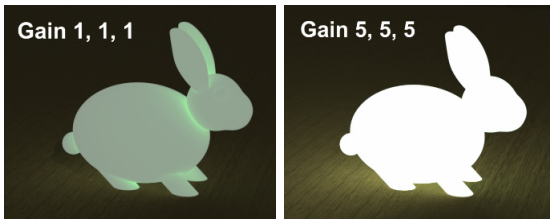
Emission texture can also be an image. Where white areas allow for emission, and black area hide emission.

Unfortunately. It highlights a problem with TC wrapping. TC only has planar box wrapping (unless you use TC's UV wrapping tools). This means on boxes, curved objects or even on spherical objects, TC will wrap 6 separate faces, Therefore emission will occur from the sides as well as the top. Which may or may not be acceptable.

There are workarounds if the above is not acceptable. Like creating the finished object out of multiple objects. And using two materials.

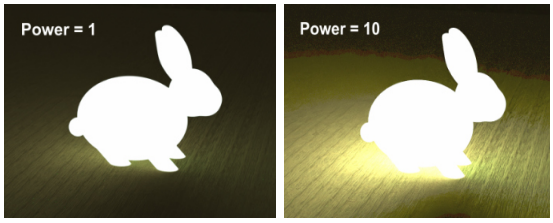
Gain. Default 1, 1, 1

Gain works as a multiplier for the power setting. But acts on individual rgb colour channels. Whereas power acts on all three channels at once. For example if the colour is set to purple (1,0,1), and the gain is 1,0,0, then only the red channel will show light.



Power. Default 0

Lamp output power in watts. If turned to zero, along with efficency then no light will shine. Power Increases or decreases the brightness and distance the light will shine. Works in conjunction with both efficacy and gain.



Efficacy. Default 0

Luminous efficacy in lumen/watt. If turned to zero, along with power then no light will shine.

Gamma. Default 2.2

Gamma value of the image map defined with .mapfile property. This has no effect if a map file is not used. If a map file is used then it allows one to alter the gamma if the image, Saves having to go back and forth to a paint program to alter the image.

Flip IES profile by 2 Default false.

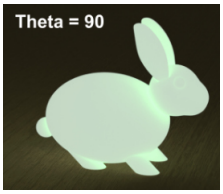
Flip the IES profile along the Z axis

Samples. Default -1

Luxcore help. The number of shadow ray to trace to compute direct light from this material. This property is an hint and the render engine can ignore this information. -1 means use the default global value.

Theta. Default 90

How directional the light is emitted, set as the half-angle of the light source. Default is 90°. Smaller values mean that more light is emitted in the direction of the face normals and less to the sides



Remember. Theta is light emitting from an object, not from a separate light. Therefore theta will look different to normal, as, rather than illuminate form the whole object, it can emit from different parts of the object.

Importance. Default 1

A hint how often to sample this light source compared to other lights

Map scale. Default “”

The name of an optional image map that can be used to scale the emitted light.
This should allow one to use a 2d map. Unfortunately in TC it only allows an IES file.

IES file. Default “”

An IES profile of emitted light

IES blob. Default None

An IES profile of emitted light embedded in a blob

IES map width. Default 0

Scale the image or IES map to the specified width. 0 means use the original size

IES map height. Default 0

Scale the image or IES map to the specified height. 0 means use the original size

Bump map. Default None

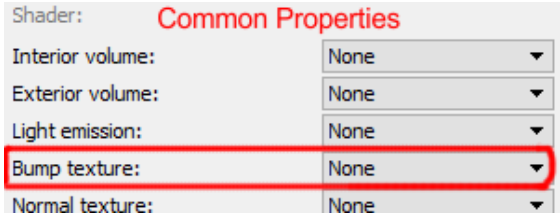
A bump map is a purely visual appearance, of a raised or lowered surface. Unlike displacement maps it does not alter the geometry. Instead it calculates shadows based on a bump map image. Essentially it fakes a 3d effect. When rendered, the program alters the areas, where it finds the difference between light and dark. Which is much quicker than actually altering the geometry (as displacement maps do).

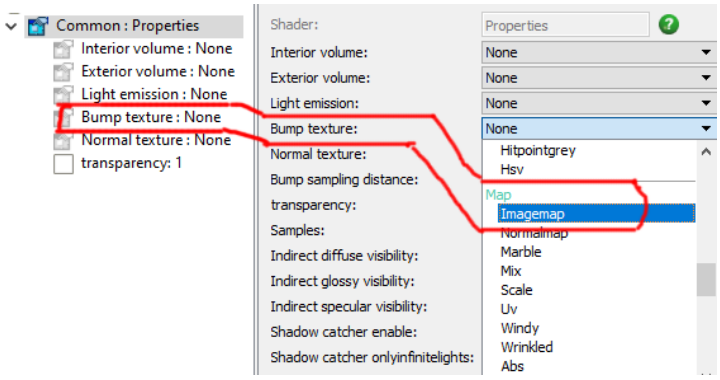
Bump maps are 2D pictures, which are calculated based on the RGB colour value. It is calculated to show black (RGB 0,0,0) as high areas and white areas (RGD 255,255,255) as low areas. With varying degrees in between these values.

One drawback with bump maps, they can be sensitive to physical light and colour changes.

Bump maps are set via common properties.

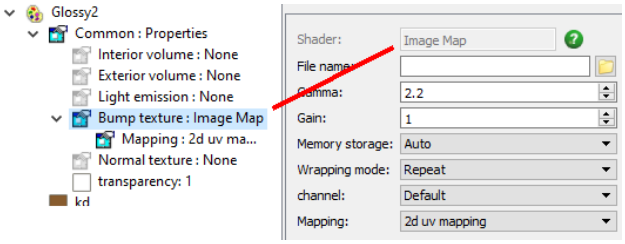
Use the drop down box to change the bump map setting to the required function. Most likely this will be 'Image map'

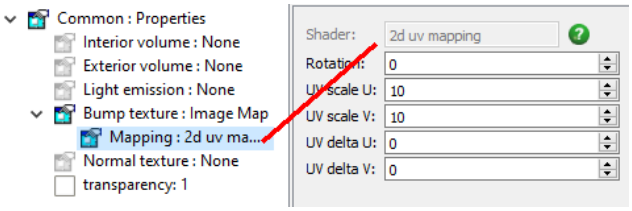




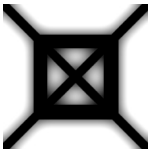
From there, clicking on the bump texture, shown the imagemap selections. Including setting the imagemap file name.

Clicking on mapping 2d, allows one to set the image map scale, position and rotation.



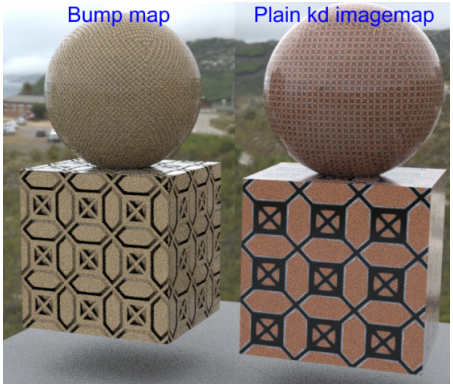


The image used for the bump map is shown in fig right It is mostly a black and white image, with grey scale shading,



The difference between applying an image as just a standard kd map, is shown in Fig right

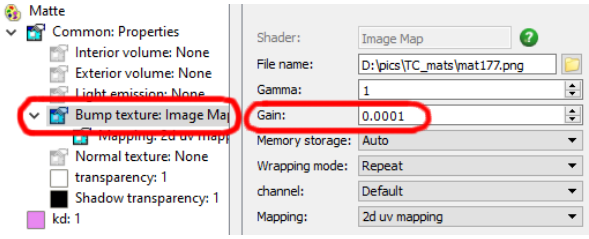
To get the same effect of a bump map, using just a kd image, would require quite a bit of work in an image editor. And you would need multiple images, loading a new one each time the camera position was changed.



Bump map height.

At the time of writing, There is no bump map height entry. This is not included as standard within Luxcore, and in Blander appears as a separate node entry box. Therefore will require programming into TurboLux separately. Which has not been done at present.

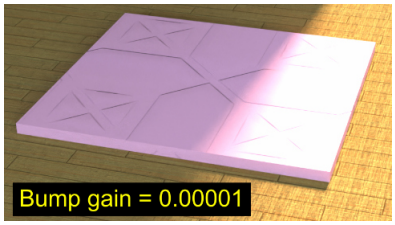
Instead, we can use the gain control, which alters the image used in the bump map - imagemap settings.

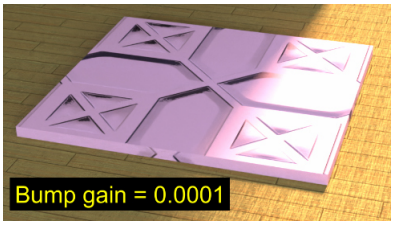


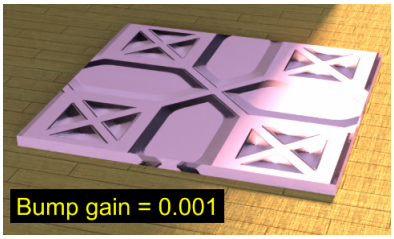
The scale of the gain needs to be pretty small. A shallow bump could be down to 0.00001. With the large bump, being about 0.01.

The three images on the right show, differences in gain values.

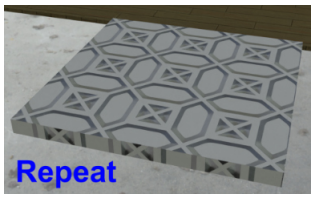
As can be seen the lower the value the shallower the bump map effect.

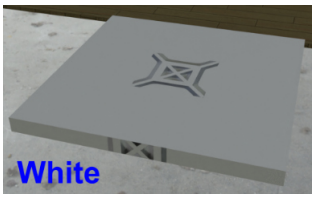


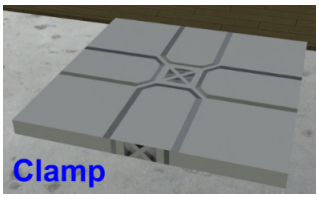




Wrapping mode appears pretty useless in Cad work. As can be seen, the wrap is odd.



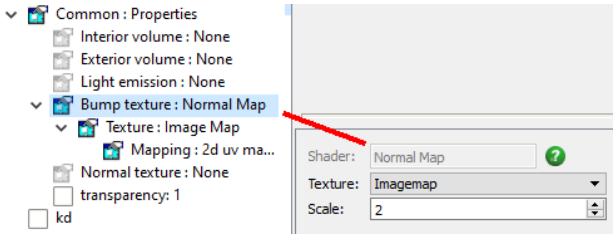




Bump Map - Normal Map setting

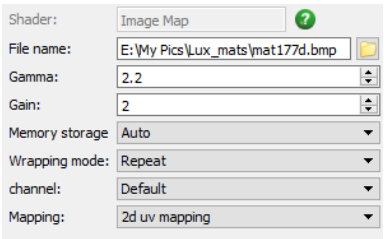
It is possible to use a normal map plugged into the bump map setting.

First open common properties, and in the bump map drop down box, select Normal map. (instead of image map).



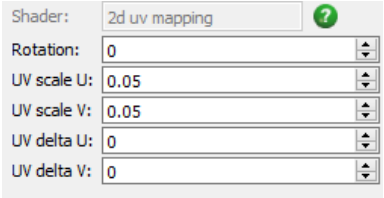
Next. Open the drop down, in the normal map texture. And select image map.

This brings up the standard image map entry screen. Select the normal map file name, adjusting the gamma and gain if really necessary.



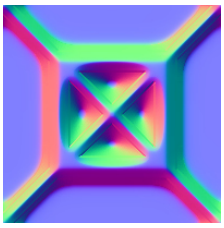
As normal maps are processed differently, adjustment to gamma and gain may not be required. The default values will often be satisfactory.

Setting the UV scale and offset is the same as normal images.



There are two main differences to using a bump - normal map as opposed to a standard bump - image map, these are.

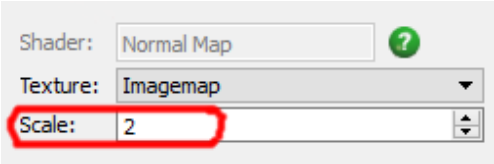
Normal maps use a separate algorithm, for processing special RGB images used as the normal map. These maps won't work properly as standard image bump maps.



The second is, the inclusion of a scale entry, on the normal map screen.

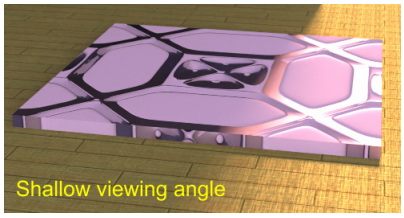
This allows for adjustment without having to mess with the gain control.

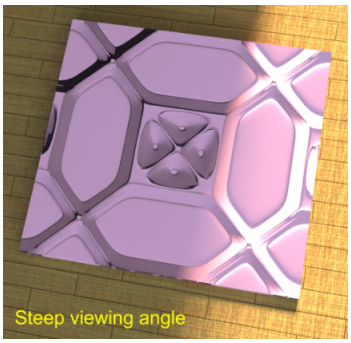
At low scale values (on the image map page), normal maps may not look as good as grey scale bump maps.



Just like bump maps, they are more realistic when viewed at steeper angles, than shallow angles..

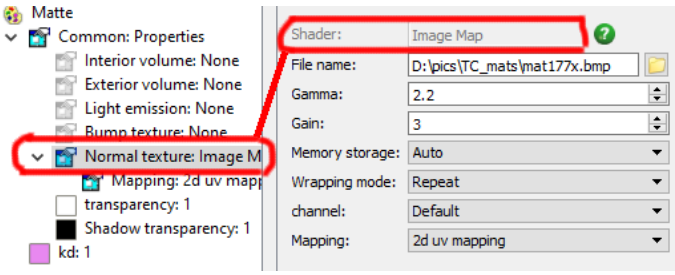
However normal maps can look a better at shallower angles, than using a plain image map as a bump map,





Normal Map Default None

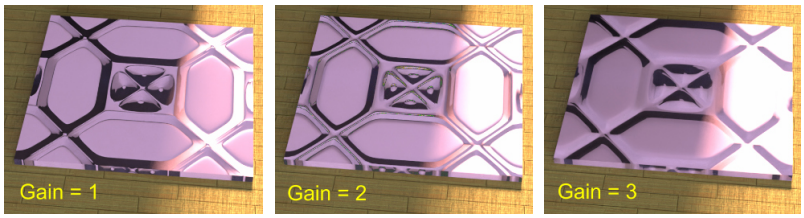
Normal map with image map.



The first option is putting an standard image map into the normal map drop down box.

This isn't the best way to do it, as, just like grey scale bump map, there is no scale factor available. Because the map is RGB colour, the gain is not the same as bump image map.

Increasing the gain from 1 to a greater value, changes how the map is placed on the object.



Normal map with Normal map.

A second option would normally be to plug 'Normal map' into the Normal map drop down box. Unfortunately. At the time of writing, this does not work correctly on my computer. So cannot be commented on in this document.

From Luxcore
Normal mapping is the process of using a texture to vary the surface normal of an object. It can be used to add details to a lower-poly mesh from a generated texture, or by using data baked from sculpting or displacement.

Usage
Normal maps are loaded via the image map node and by enabling the Normal map option. The node can then be connected to any material bump slot. The normal map bump height can be scaled with the Height setting. A height of 1.0 would be the default height of the texture.

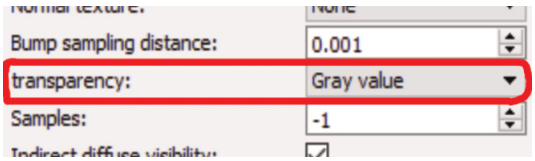
Unlike regular bump maps the scale of a normal map is relative. For instance, 0.5 is half the bump of the texture. Thus you'll mostly want to to leave the height setting at 1.0 for normal maps. As with any image map, the object must have UV coordinates for using a normal map.

Mixing With Other Textures
The normal map texture acts a special float(grey-scale) texture, and can be mixed with other float textures using the math node to produce complex bump maps.

Personally. I think it better to use Common properties - Bump map with the normal map image selected, instead of image map.

Transparency.

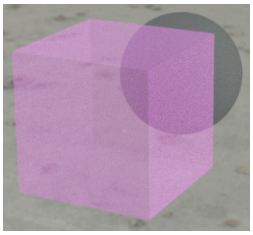
Default Grey colour (1)



When using a transmission colour, Black is transparent and white is opaque. This is important when using a transmission map plugged into the property.

Transmission works with numbers, colours or textures. It can be useful not just for glass type materials, but also for things like, plants / trees, mesh, Trellis, figures and Volume type materials. Sometimes it is preferable to avoid using high reflectance value along with transparency.

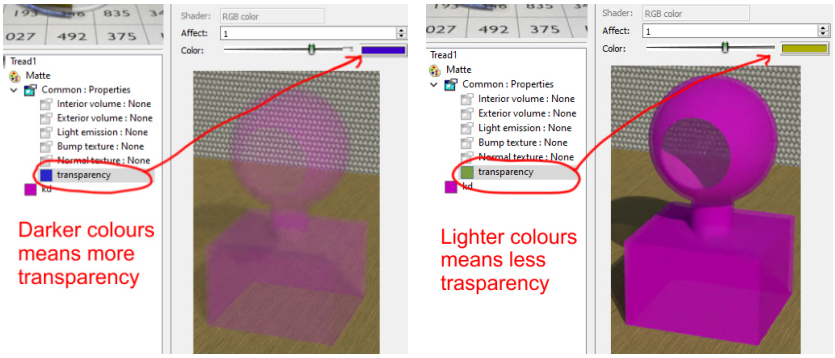
Transparency numbers



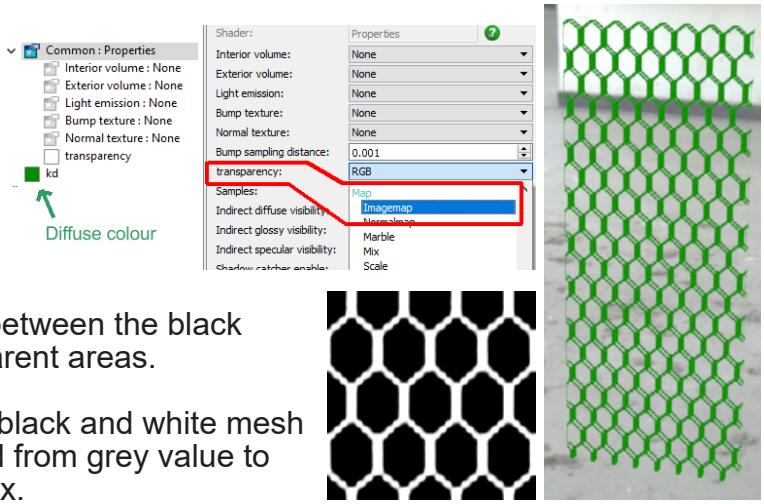
When using a number, 1 (white) = opaque and 0 (black) = transparent. With varying degrees of transparency in between. This is the simplest use of transparency.

Transmission colour.

If the input is changed to RGB colour for example. Transparency will work, But it appears to work on the tone of the colour, not the actual colour itself. For example dark blue is close to full transparent, bright yellow would only be partly transparent. The transparency colour has no affect on the diffuse (kd) colour of the material.



Transparency map.



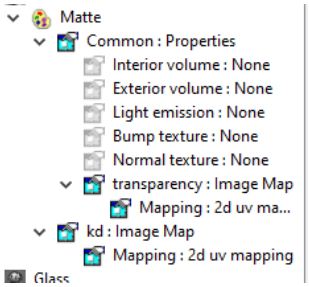
Transparency maps are a powerful function of transparency.

The simplest map is a black and white image, which produces good contract between the black transparent, and the white non- transparent areas.

The images on the right were a simple black and white mesh image. The transparency was changed from grey value to an Image map, using the drop down box.

The matte material uses a green diffuse colour. The mesh scale was altered in the image map settings as normal.

This second example is the classic plant with transparency. It uses two images, and therefore, both the Transparency, and kd (diffuse) settings, were changed to Image map.



The transparency uses the black and white image, and kd uses the coloured tree.

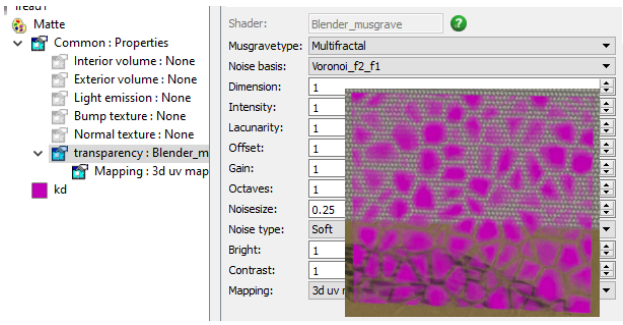
It is important to remember that the image mapping settings / scale, must be set the same for transparency and kd.

The first fig right Shows one problem that can occur. The image was mapped onto an object a few millimetres thick. As can be seen, the image is also mapped onto the edge of the object, giving a poor finish.

The picture in second fig right was mapped onto an extruded line, giving a sheet object. Thus with no edges, the tree looks far better.

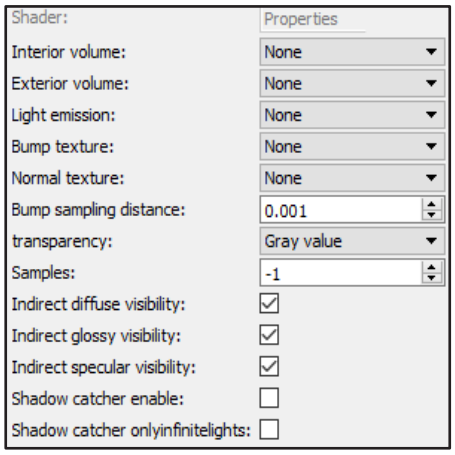
Transparency also works with other functions, besides colour and image. For example the blender maps. Though sometimes the effects can be unpredictable.

The fig right, uses a blender map ‘Musgrave’ as the transparency map.



One can also use a bump map with transparency. Though the effect may not look as good as bump maps on normal surfaces. Trial and error.

Other common properties.



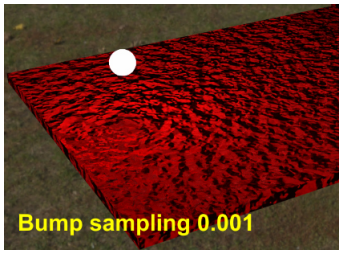
Bump sampling distance. Default 0.001

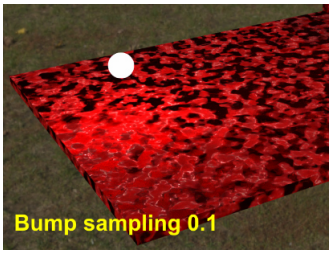
set the offset used to sample bump/normal mapping.

Sets the sampling distance between points of a procedural texture.

These include procedural texture like, brick, dots checker, and the blender noise textures (blend, clouds Fbm etc.).

In most instances, to avoid confusion, it may be better to adjust the bump map properties, and leave bump sampling distance at 0.001





At the time of writing, with the following items. It is unknown how they work within TurboLux.

Samples. Default -1

number of rays used to sample this material. -1 means use the default global value. This property is an hint and the render engine can ignore this information.

Indirect diffuse visibility. Default ON

if this material is visible in indirect rays. This property is an hint and the render engine can ignore this information.

Indirect glossy visibility. Default On

if this material is visible in indirect rays. This property is an hint and the render engine can ignore this information.

Indirect specular visibility. Default On

if this material is visible in indirect rays. This property is an hint and the render engine can ignore this information.

Shadow catcher

From the Luxcore help files. ……. Overview
Since version 1.6 LuxCore materials have the shadowcatcher option. It is intended to be used for compositing, e.g. to integrate a rendered 3D object into a photograph.

A shadowcatcher material will be transparent where hit by direct light, while shadowed areas will stay opaque.
The opacity of the shadow can be controlled by the opacity/transparency of the material.
The color of the shadow can be influenced by the material color.

Any material can be a shadowcatcher, but it does not make sense to use a purely specular material without any diffuse reflection, like glass or mirror. Usually matte is the material best suited for a shadowcatcher.

Supported Engines … The shadowcatcher option is supported by the Path and Tiled Path engines (both CPU and OpenCL), but not by the Bidir and BidirVCM engines.

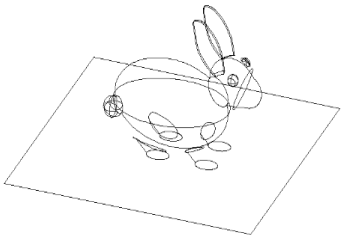
It is possible, however, to use the alpha channel generated by a rendering using the path engine to compose the images.

Tips
When using a sky light as environment, enable the "groundcolor" option and set the groundcolor to black
When using an imagemap as environment (e.g. a HDRI), enable the "only sample upper hemisphere" option (available with infinite luminance).
Doing this prevents the environment to cast lights onto the shadowcatcher from below, leading to better alpha transparency.

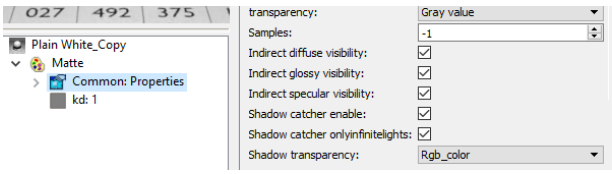
The fig right right shows an object against a HDR environment. There are no other objects in the scene, before the shadow catcher plane was added.

Looking at the rabbit, the human brain, knows that in real life, a rabbit wouldn't be floating in mid air. So assumes it is sat one the floor.

To add to the effect, we will add a touch of shadow to show where the ground is.



The simplest shadow catcher is either a 2d rectangle converted to surface (works with ACIS in platinum only). Or an extruded line (Acis and TC surface).

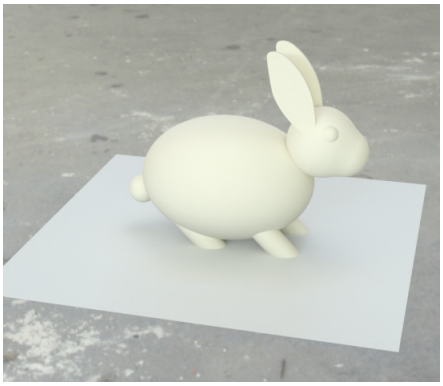


The **shadow catcher tickbox**, is included with the materials common properties. Shadows on a matt floor are not shiny. Therefore we choose a matt material. And assign the material to the plane object.

Ticking the shadow catcher box, tells the program to just show areas of the object not lit by direct light.

The shadow catcher normally works with all lights, However - .

If the '**Shadow catcher onlyinfinitelights**' is ticked. Then only infinite lights will be caught. Like HDR, Sun, Infinite and distant. It doesn't shadow catch spot or point lights. If there is a sun and a spot light, the shadow from the sun will be caught.



Looking at the figure right, we can see that the shadow created by the chosen HDRI is very subtle. But is good enough for this example.

The plane is left visible, as it shows where the shadow will be created.

The finished render (fig right), has shadow catcher ticked, which hides most of the plane except for the shadow.

And even with such a subtle shadow, it looks better than the original image on the previous page.

Different lighting will produce different shadow effects, If darker shadows are required, use a sun or even a spot light with 'infinite light only' turned off.

The drawback, is one sometimes needs stronger lighting to get darker shadows, depending on the scene, and existing lighting.

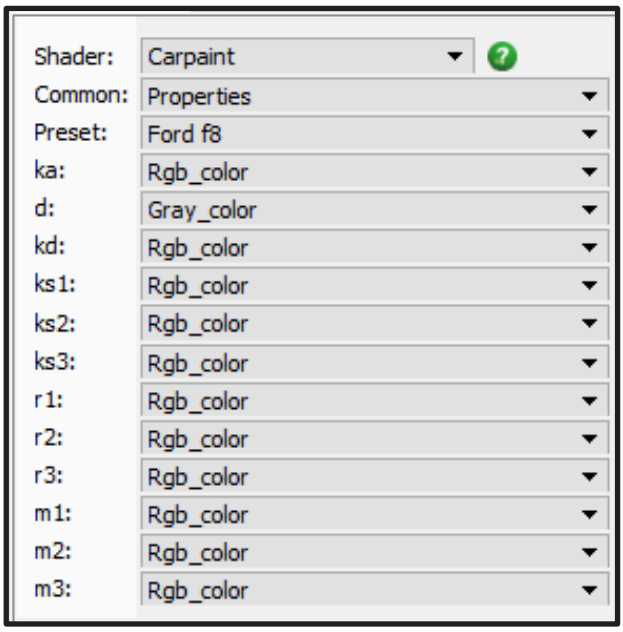
Transparency,

Altering the transparency may occasionally be required to remove any edges, still present when shadow catcher is used.

Shadow Transparency.

Use along with transparency, to alter the object which the shadow catcher is applied to. This option may only show up in certain circumstances.

Car Paint (Text from help file)



Car Paint is an advanced glossy material, simulating a diffuse surface with multiple reflective coatings. It works well for car paint, obviously, but is also well suited to any sort of "high-tech"-looking painted surface, such as painted aircraft skin, starships, futuristic weapons, etc.

Presets. **Default Fordf8**

Presets are available for some quick paint jobs. These are derived from the paper that the car paint material itself was based on. The actual values used by the presets can be found here.

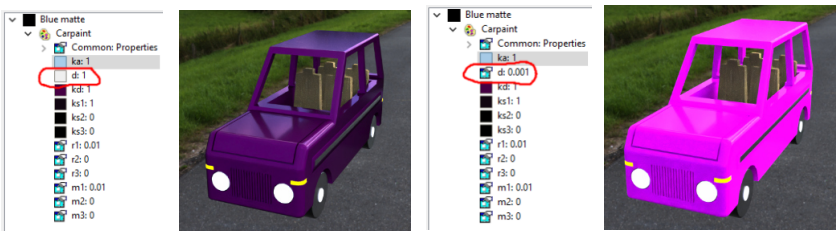
Absorption Color and Depth

Color (ka). **Default RGB_color**

Absorption means that the light which reaches the car paint is not reflected but "swallowed" by it. If your light source is white it consists of even amounts of red, green and blue. If you set the absorption color to red, the car paint will absorb the red and reflect the green and blue (which is cyan). Absorption works on the diffuse color of the car paint, and not on the specular layers.

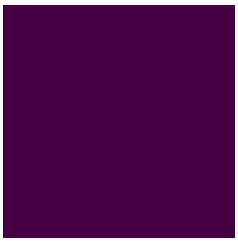
Depth (d). **Default Gray_color**

Depth works as a scale factor for the absorption color. Increasing it makes the car paint absorb more. Setting this to 0.0 or the absorption color to black will disable absorption.



Diffuse Color (kd). **Default Default RGB_color**

The diffuse color is the base color of the material. Try to not go above 0.8 as no color in the real world is that saturated/bright.



Parameters

The parameters may seem complicated at first, but when the basics are understood it's not that difficult. You work with three gloss "layers" in car paint and each layer has three parameters that can be controlled. In a nutshell, they are specular color (color/ brightness), R (fresnel power) and M (roughness).

For example the first layer's parameters are "Specular color 1", "R1", "M1". (note: "layer" is a bit of a misnomer, as they do not actually stack, they are merely chosen at random. But it will be used here as a term of convenience)

When setting up the gloss layers, you may find it easiest to "solo" each layer by turning the specular color on the other two layers to full black, allowing you to configure your M and R values for each layer one at a time. You will find that using higher values for M and R (such as R=.9, M=.2) produces a more "metallic" look for that layer, while lower values (such as R=.01, M=.02) produces a more "glassy" or "waxy" appearance.

Specular Color (ks 1,2,3). **Default RGB_color**

This controls the color and brightness of the reflections for each layer. Increasing the value of this color will result in that layer reflecting more light, producing a stronger reflection. Setting this color to green, the reflection of a gray object will appear green on the surface of the car paint. The reflection of a red object will appear yellow (the combination of red and green).

Example. The three primary colours, Red, Green, Blue, are unique, I.e. One cannot be made using the other colours.

R1-3 (r1,2,3). **Default RGB_color**

The R value controls the strength of the fresnel effect. With low R values, the reflection will only be near full strength when viewed from a very steep angle. At high values, it will be almost evenly distributed across the material. Generally, you want to use a lower R value for your brightest, most glossy layer, and increase it for the duller layers. How much will depend on the paint you are simulating.
R=1.0
R=0.1
R=0.01

M1-3 (m1,2,3). **Default RGB_color**

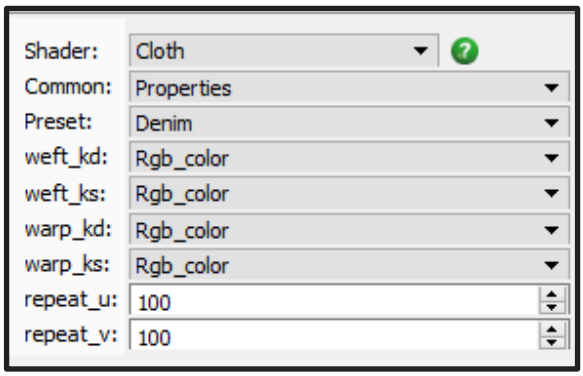
This one controls how strongly the reflections are scattered. It can be set between 0.0001 and 1.0. Higher values will make the reflections scatter more.
M=0.001
M=0.01
M=0.1

Changing the parameters and adding 2 and 3 settings can change the effect quite a bit.

But it is a lot of trial and error, to get the desired finish. And for most normal car drawings, not worth the effort.

Cloth

The Cloth material simulates a woven fabric. The model's parameters are too many to expose individually, but presets are available for typical fabrics such as denim, polyester cloth, wool and silk. The material is based on the PhD thesis "The Appearance of Woven Cloth" by Piti Irawan. LuxCoreRender's implementation is an adaptation of the code in Wenzel Jakob's Mitsuba renderer, which was again based on code by Irawan.

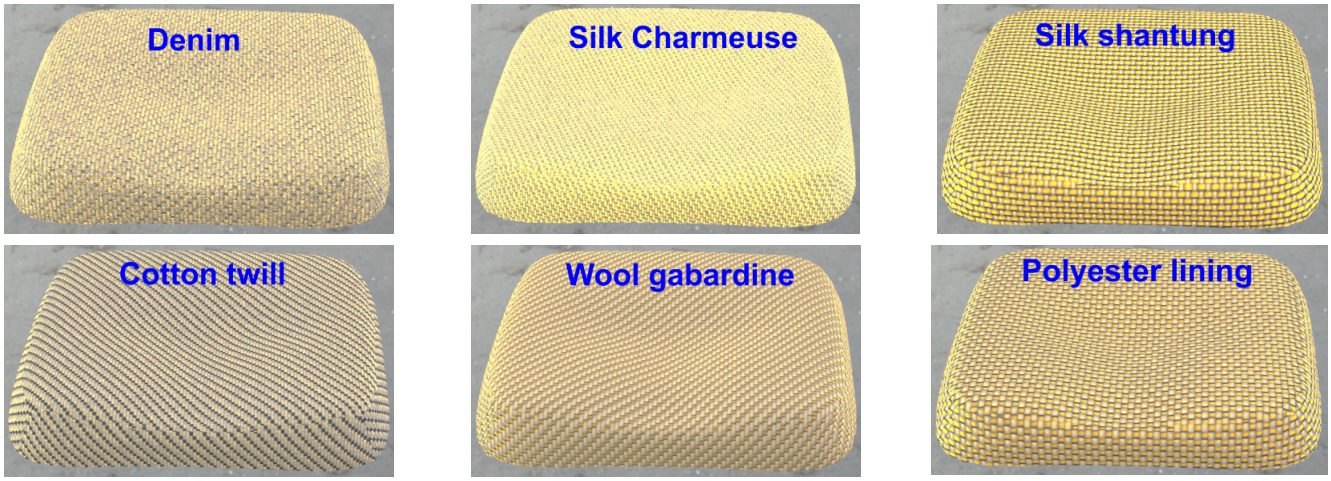


Important note: the object you are assigning this material to will have to be UV unwrapped for this material to work properly.

Preset Name. **Default Denim**

Six presets with different stitch patterns are available. More advanced cloth materials can be created by using textures instead of solid colors on the diffuse and specular channels, and by mixing with other materials such as velvet or glossy translucent.

This alters the weave, not the actual colours. The six different weaves are shown below.

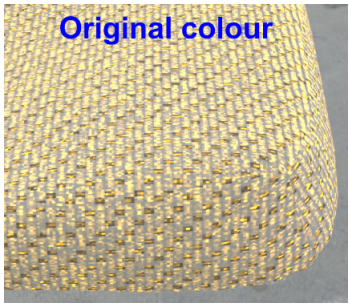


Warp and Weft Diffuse Color - kd. **Default RGB_color**
Diffuse base color of the warp and weft fibers.

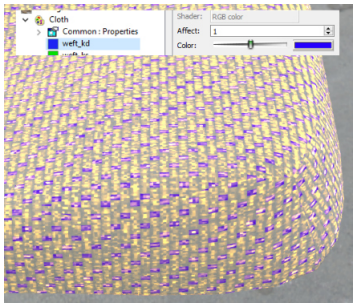
Warp and Weft Specular Color - ks. **Default RGB_color**
Specular color of the warp and weft fibers.

Warp (the lengthwise stretched fibres) and Weft (woven in / out fibres) settings, alter the colour of the fibres making up the cloth. The specular colour would normally be a grey scale or a deeper shade of the warp or weft. The colours below is exaggerated, simply to show the effect of altering the setting.

Default settings



Weft kd change to blue.



Weft ks changed to green

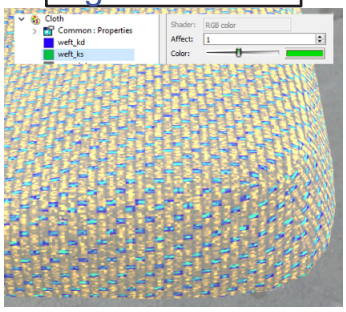
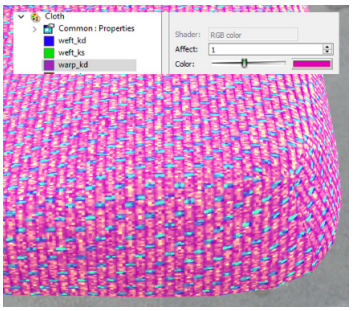
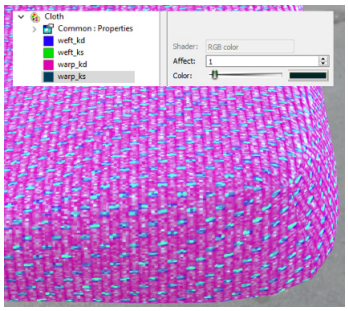


Fig right shows altering the Warp settings.

Warp kd change to dark pink.



Warp ks changed to dark blue

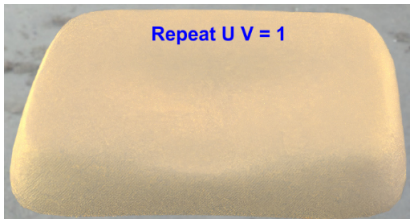


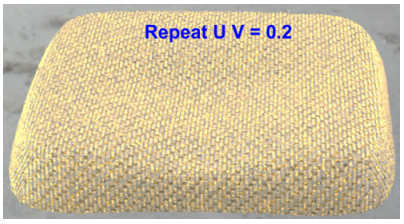
Repeat U and V. Default 100

Alters the scale of the weave in U and V direction according to the UV unwrapping.

By altering the scale, one simply enlarges or shrinks the weave. Like most scale setting in T-Lux, smaller numbers creates a larger weave, and larger numbers = smaller weave.

A default of 100, would likely produce such a small weave that it cannot be seen. Small numbers like 0.2 or smaller may be required.



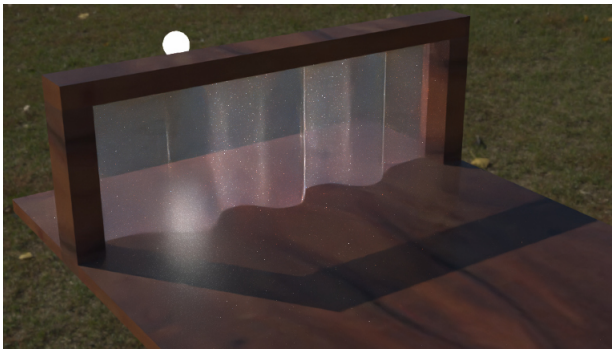


Cloth other settings.

Whilst the common properties will work with cloth. One would not normally set most of them. The cloth is built 'on the fly' so to speak like lightworks, and thus cannot correctly accept an image bump map. Similarly one would not have cloth emitting light.

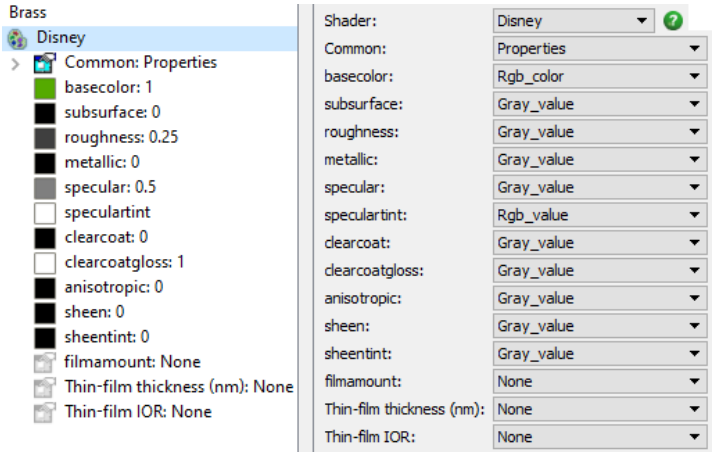
Transparency can be used with cloth. For example, used as curtains.

Apply the material, and in common properties turn down the Transparency. To produce a translucent curtain effect.

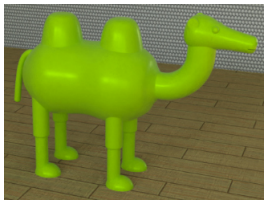


Disney.

Disney probably allows the most user control under one heading. Whilst other materials have been pre-programmed to cater for specific purposes. Disney allows more freedom, at the expense of more settings to fiddle with and get confused by..



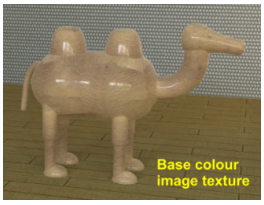
Base color. Default RGB_color

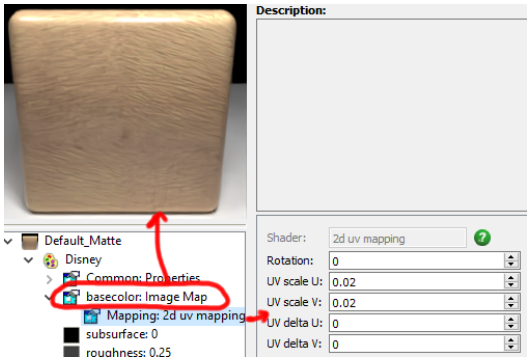


Base colour can be a plain colour, or a texture.

For metals, leave at a grey colour for metals like silver, steel, etc. change the colour as appropriate for gold, brass, copper and bronze. For other objects, for example, plastic, paint, etc. the colour could be anything, but may benefit from some sub surface scattering.

One can also use an image map as a base texture. This is simply done by setting the base colour to Image map.

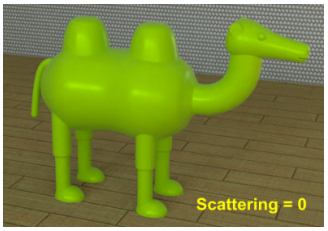


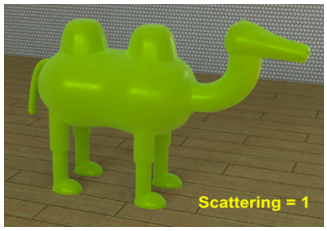


Navigate to the desired image on ones computer. And set the necessary UV scale.

Sub surface. Default Gray_value

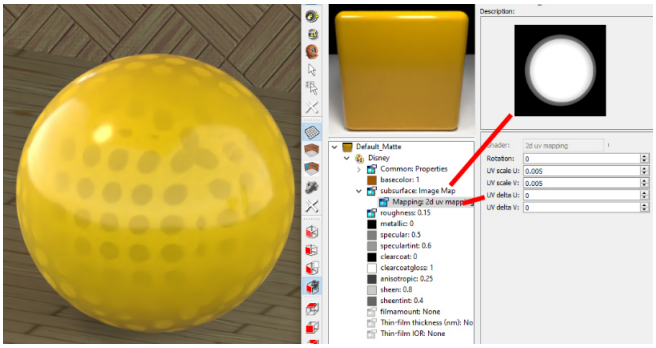
Often used for organic materials, and some types of plastic. It scatters the light as it penetrates the surface.





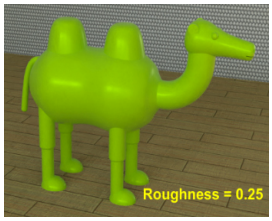
One can use an image map as the scattering. Though the results can be unimpressive.

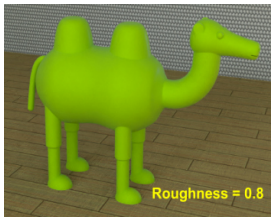
Change the drop down from grey_value, to image map.



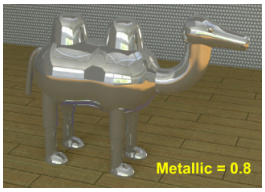
Roughness. Default Gray_value

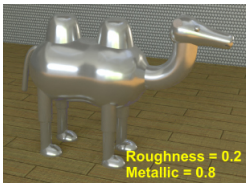
Determines if the surface is rough (more matte) or specular. Most materials have a degree of roughness. Even if only a very small amount. Use values between 0 and 1





Metallic. Default Gray_value

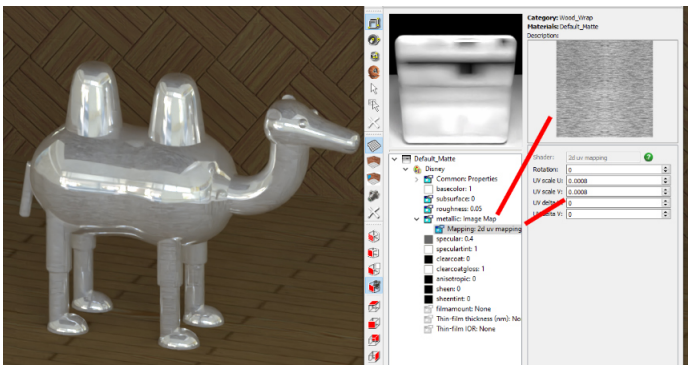




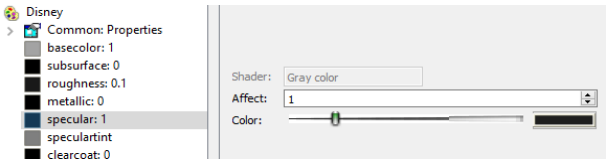
Gives the selected base colour or texture, a metallic shine. A good alternative to using the Metal2 materials.
Use values between 0 and 1

The metal part of Disney, can accept an image map. Which will change the surface texture of the metal.

This can be subtle, as in the image right. Or quite dramatic, If a black and white image is used.



Specular. Default Gray_value
Specular tint. Default Gray_value

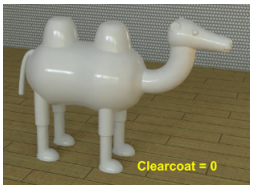


Sets the amount of specular effect (shiny bits) to apply to the object.

Setting the tint towards black will reduce the amount of specular to apply. (I changed the drop down to grey colour).

Clearcoat. Default Gray_value
Clearcoat gloss. Default Gray_value

Adds a surface coating to the material. Like applying a coat of varnish or epoxy.



Anisotropic. Default Gray_value

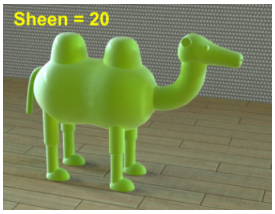
Changes now the specular effect is depicted on the object. Value normally ranges from 0 to 1.

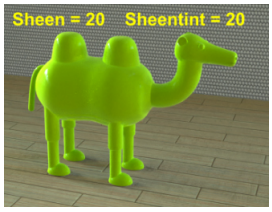
Unfortunately it only has one setting, therefore cannot easily be switched from horizontal specular to vertical.

Sheen. Default Gray_value
Sheen Tint. Default Gray_value

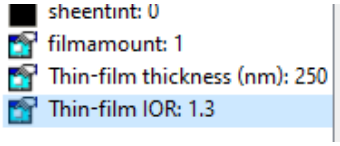
Adds an external sheen to the object in a similar way that velvet does.

Tint allows for a different colour to be used for the sheen. But some coloured effects will look odd, and unrealistic.





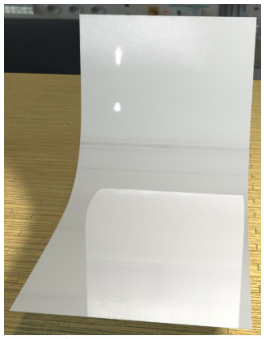
Filmamount. Default Gray_value

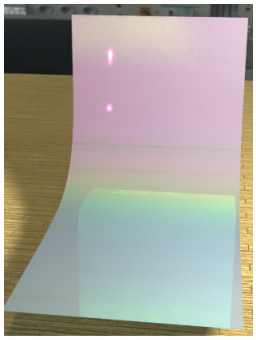


Linear interpolation between no film (0) and full film effect (1) for artistic control.
This property is optional, if it is not defined, the coating is disabled.

Film thickness (nm). Default Gray_value

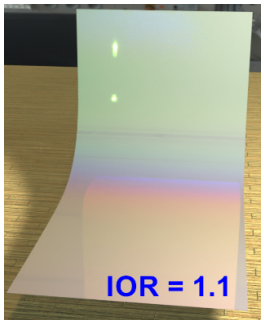
thickness (in nanometers) of a thin-film coating on the surface.
This property is optional, if it is not defined, the coating is disabled.

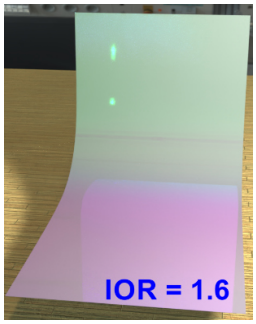

Original colour


Thin filled applied

Yep this is nanometres, it does say its a thin film. Don't go too thick, It is possible to apply too thick a coating, that the effect diminishes. Between 200 and 400 should be about right.

Because light is being refracted inside the file, the base object colour, roughness etc. all play a part in the final render.

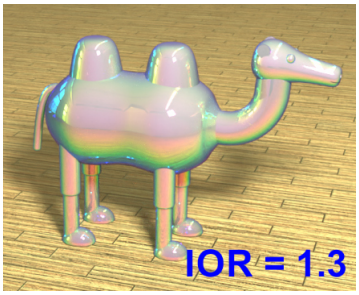




Film IOR. Default Gray_value

Refractive index of the thin-film coating.
This property is optional, if it is not defined, the coating is disabled.

IOR plays an important part of the effect, Setting too high an IOR will reduce the effect. Because the film is primarily water (even a bubble), with imperfections in it like oil. The IOR should be fine at 1.33.

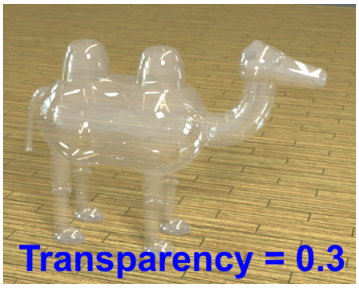


Disney Other.

Transparency (Opacity).

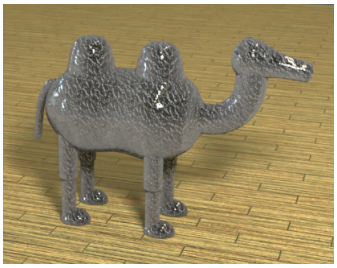
Like all materials, common properties has a transparency settings.

Unlike some materials, the settings within Disney can make the transparency look a lot more realistic. Especially for things like plastic, where a high specular reflection, and low roughness is required..



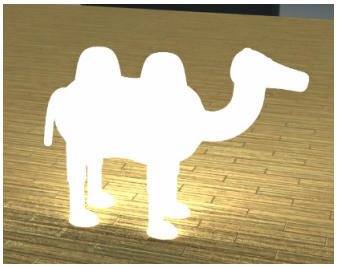
Bump

A bump map can be applied to the material, but if a top coating is used, this can affect how the bump map looks.



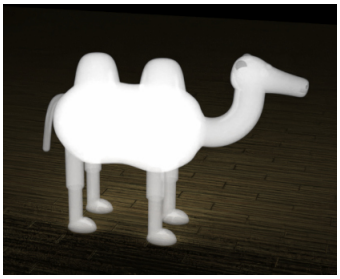
Emission

Adding common properties emission, simply emits light from the object itself, Therefore one can lose object definition, when trying to get a decent about of light from the object.



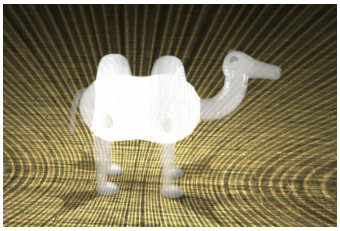
Volume emission.

Makes an object glow a bit, but doesn't supply a lot of light, unless one cranks up the emission quite a bit.

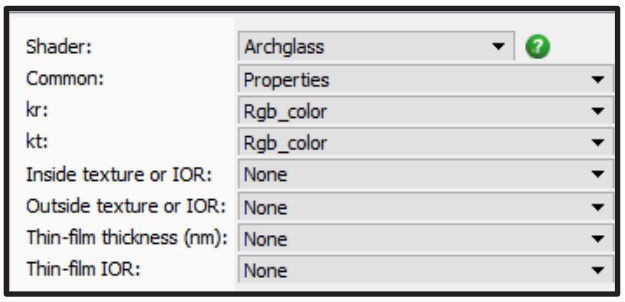


Combining various elements, bump map, emission, luminance, can produce unusual effects. However they really have no purpose in general CAD work.

Note. There is a single emission light supplying background illumination. All the actual lighting effect is coming from the object.



Architectural (glass)



For thin sheets of glass, the lack of refraction in architectural glass does not matter

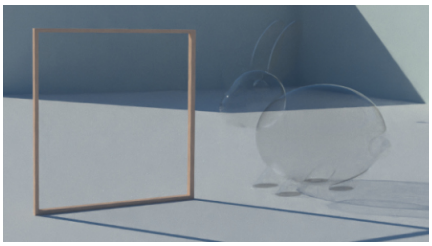
Luxcore help - This option disables refraction during transmission. The effective result is that transmission will appear as though the glass had an IOR of 1.0, while reflections will use the specified IOR. The resulting glass will show only the transmission/reflection colors and volumes effects.

You could enable this option for a thin sheet of glass such as a window. The render output difference will be minimal, and it will render much faster, as there is less to compute. However, disable it on any curved glass as it will make the object look fake.

There are several other important features to architectural mode, besides reducing the necessary calculations of the material. Because there is no refraction during transmission, architectural glass is transparent to shadow rays and alpha, neither of which is the case in non-architectural mode. This is especially important for window glass. Because it is transparent to shadow rays, it is possible to perform direct light sampling through the window to outside light sources, greatly improving the render efficiency. And because it reflects but still propagates alpha, it allows you to have realistic interior reflections and absorption on the windows, while still being able to composite in a backdrop for the view outside of the window.

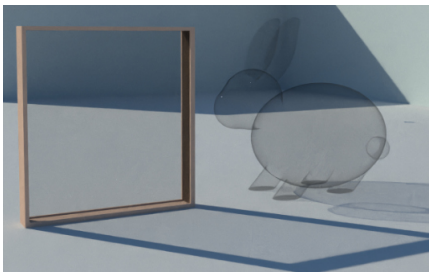
As can be seen in fig right the default setting in TC shows very little reflectance and no refraction. The only definition for the glass is on the edges.

This figure uses a kr of white.



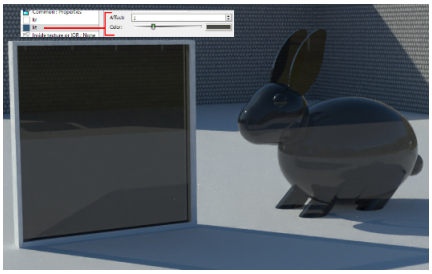
kr (reflection). **Default Gray_value**

Reducing the kr reflection can darken the glass slightly. It can add definition to the edges of the glass. Whilst the glass itself remains clear.

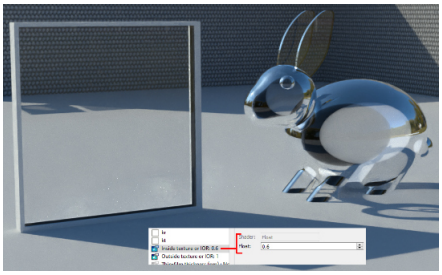


kt transmission). **Default Gray_value**

Reducing the transmission, is like tinting the glass.



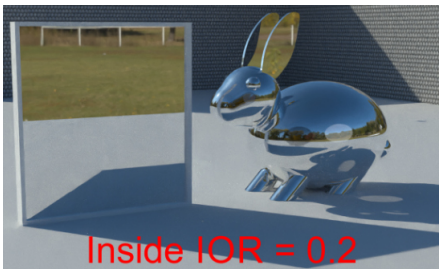
Inside texture or IOR Default None



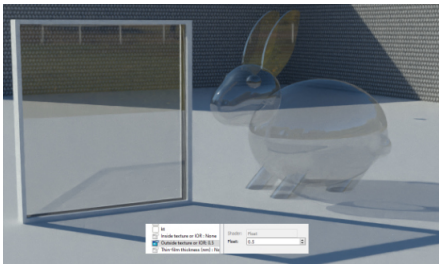
Note. For this, I changed the drop down from rgb colour, to Float.

Adding an inside IOR, seems odd, because unlike normal glass, it uses a special algorithm which isn't affected the same way using IOR. Therefore numbers above 1, generally don't make much difference.

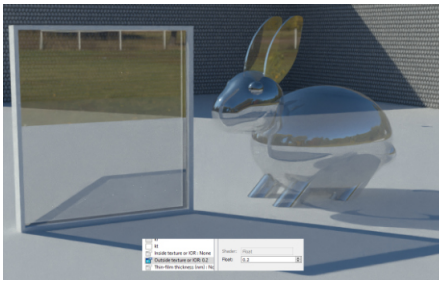
However turning down the IOR makes the glass more reflective.



Outside texture or IOR Default None

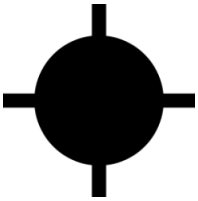


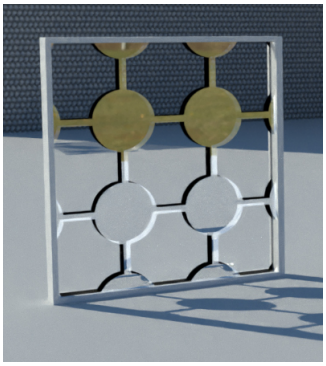
Outside texture.IOR changes the surface. In which lower values increase reflection at the surface. And unlike Inside texture, even low values appear to retain some transmission.



Using an image with inside texture, or a combination of inside/outside. (change the drop down box to image map). Can produce unusual results .

The black parts become reflective, and the white parts remain transparent.

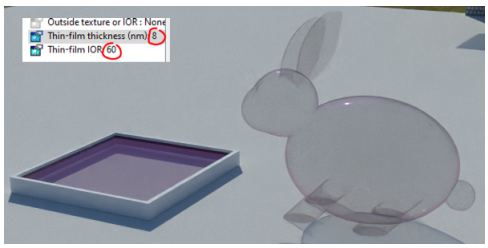




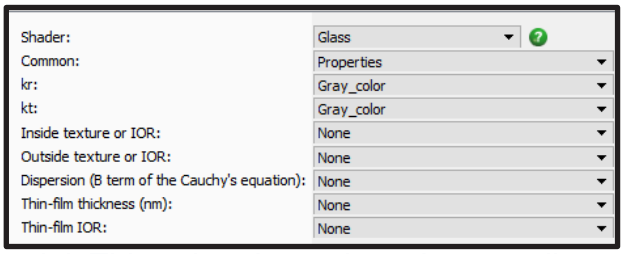
Thin-film thickness (nm) Default none thickness (in nanometers) of a thin-film coating on the surface. This property is optional, if it is not defined, the coating is disabled.

Thin-film IOR Default none Refractive index of the thin-film coating. This property is optional, if it is not defined, the coating is disabled.

Adds a thin nanometre film on the surface of the object, which will often put colour on the surface, or part of the surface. This is sometimes purple or green in colour. Depending on the direction of light.



Glass



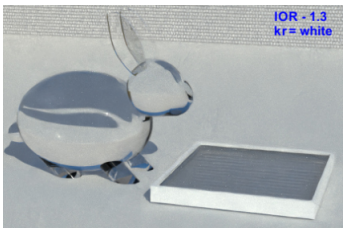
Luxcore help - The glass material represents a smooth dielectric substance. It works well for materials such glass, crystals, clear liquid, transparent plastic, and gemstones.

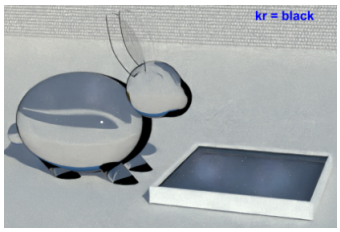
Luxcore help - The index of refraction for the material. This value determines the overall appearance of the material, and should be set according to the material you are trying to simulate. In combination with dispersion (see below), it should be set with the Cauchy-A coefficient.

In TurboLux, the glass material by default has a clear volume pre set. This was disable for many of these tests, so the settings of just the glass will be evident.

kr Reflection Color Default 0.1 texture or constant reflected color of the material

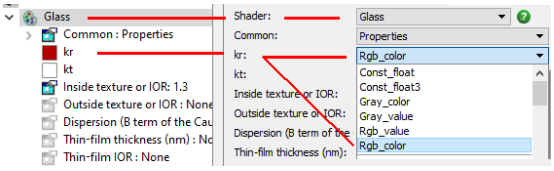
Luxcore help - This is the color of light reflected from the surface.





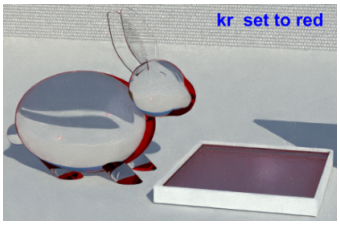
Alters the surface reflection of the material.

For clear glass this would be left at white. Changing kr to black whilst kt is still white, appears to mainly affects the reflection at shallow angles.



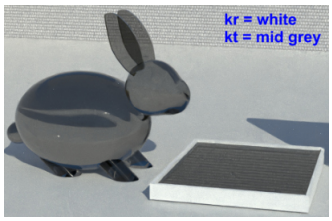
kr material come preset to grey colour, but this can be changed to colour using the drop down box against kr, in glass properties.

As can be see, the colour changes in the same way in which happened simply by turning the colour to black.



Kt Transmission Color Default 0.1 texture or constant transmission color of the material

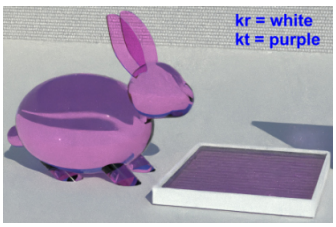
Luxcore help - This is the color that the light will be filtered to after being transmitted. The color given here will be applied to the material uniformly, you may wish to derive the color from the objects volume instead.

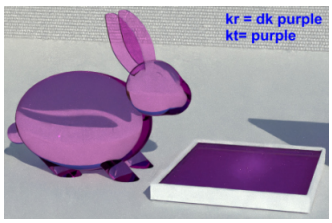


The kt setting may be set at grey colour, but as with kr, it can be change to colour, by altering the drop down box against kt, in glass properties.

In grey scale, it is like tinting the glass, the darker the grey colour the darker the transmitted colour.

Changing the kt colour to purple, simply changing the colour of the glass. If kr is also changed to dark purple, it becomes darker where light hits at shallower angles. (Inside IOR setting is 1.3)

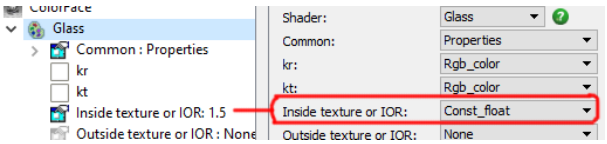




Inside texture or IOR

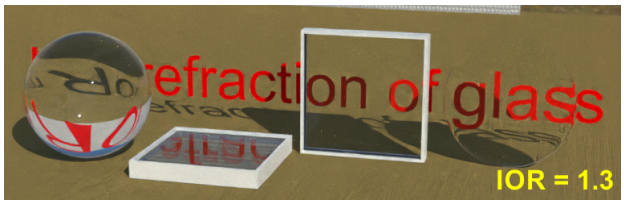
Default 1.5 inside texture or constant index of refraction (only for volume-less rendering)

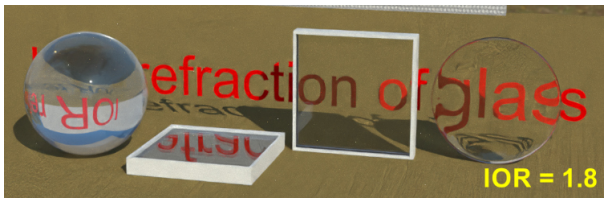
NOTE. For these tests, the Inside IOR drop down box in glass properties, was changed from grey scale, to constant float.



This alters the IOR setting which affects how the light refracts inside the glass material. The normal IOR for glass is around 1.5.
But depending on the glass, IOR can vary from 1.3 to 1.8.

This simple example uses a sphere, two flat sheets, and an ellipse which acts like a magnifying glass.

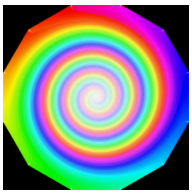


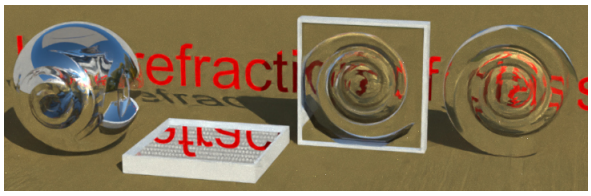


As can be seen, the sphere and ellipse produce a magnifying effect, The vertical sheet produces almost no distortion. The sheet lying down, is fussy at just 1.3, because with a thick sheet of glass and low IOR, one is getting reflection from the top surface, and also reflection from the bottom surface. Which, at such a shallow viewing angle produces two images of the text.

It is possible to use an image map in the IOR setting, however different objects or orientation can react in unexpected ways. As can be seen below. The sphere being dense, refracts a lot of the outside scene (a hdr map in this case), the vertical sheet and ellipse act as expected. But the shallow viewing angle of the horizontal sheet shows no effect, and becomes reflective.

This simply shows that refraction is highly dependent on the viewing angle.

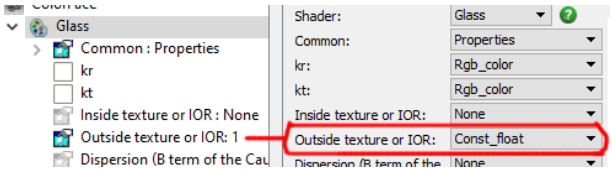




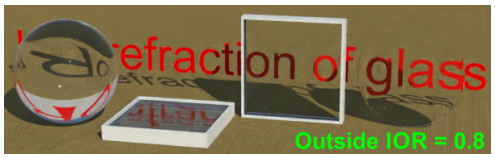
Outside texture or IOR

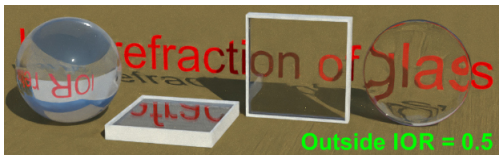
Default 1.0 outside texture or constant index of refraction (only for volume-less rendering)

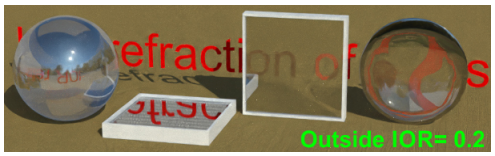
NOTE. For these tests, the outside IOR drop down box in glass properties, was changed from grey scale, to constant float.



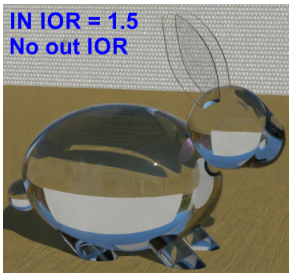
Outside IOR appears to be like a film on the surface of the object, similar to a lens having a polarising or UV coating. Note strong IOR effects can be obtained from reducing the IOR not increasing it like Internal IOR. (One can still use values larger then 1).

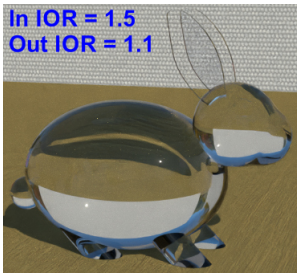






Using both internal and external.





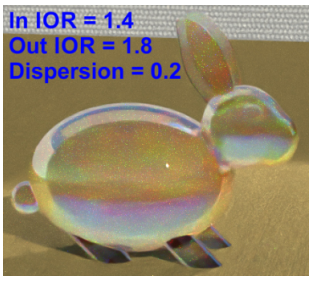
When using both together. To avoid one overpowering the other, it may be better to keep Outside IOR at just above 1.

This can sometimes help if one gets too much black areas of the glass.

Dispersion

Luxcore help - Dispersion means that the IOR is calculated as a function of the wavelength, i.e. the color, of light. Setting a value greater than 0 puts the glass material in a spectral mode. This will make rendering slower, but increases the realism by adding color effects.

LuxCore uses a two-term Cauchy-equation to model dispersion. Dispersion is measured by the Cauchy-B coefficient. Realistic values range from roughly 0.003 to 0.0015. For further information, see Glass Material IOR and Dispersion.



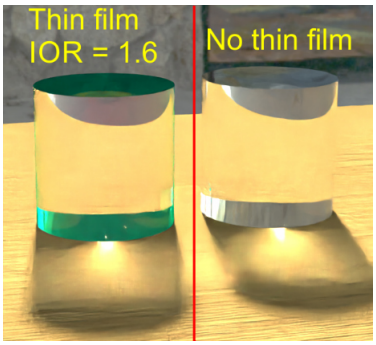
Dispersion creates a specular coloured effect with the glass. It is totally trial and error, to get the effect needed. Especially if inside and outside IOR is also used.

Thin-film thickness (nm) Default none thickness (in nanometers) of a thin-film coating on the surface. This property is optional, if it is not defined, the coating is disabled.

Luxcore help - This allows you to define a film coating on top of the glass. This usually results in a rainbow colored reflection. It has two controls, a thickness and IOR. The IOR should match that of your film material (Cleaning solution, which can leave such a film behind, has an IOR of ~1.3-1.35, due to being almost entirely water). You can vary the thickness to achieve the desired effect. A thickness of several hundred nm will result in the most pronounced effect.

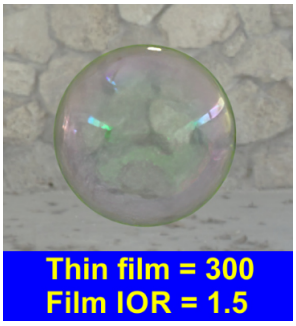
Thin-film IOR Default none Refractive index of the thin-film coating. This property is optional, if it is not defined, the coating is disabled.

Adds a thin nanometer film on the surface or the glass. This will change the colour on certain surfaces.



The two images right, have the same glass properties. Except. The first on has a thin film applied.

The last image uses thing film on a hollow sphere (well, its 600mm diam x 0.5mm thick). And is more of a bubble effect, rather than simply a coloured glass effect.



For more 'odd' effect, one could also add dispersion. (not pictured).

Glass with common properties

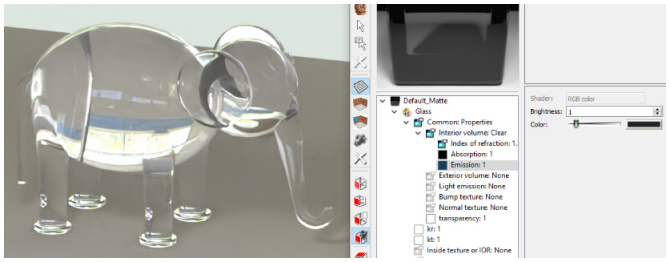
Glass comes preset with clear volume. Though this can be changed by the user.

Volume emission.

If we look at the original image, we can see one problem with Glass material. And that is 'black' areas In the glass. Which detract from the effect.

One way to reduce these black areas. without unduly affecting the render, is.

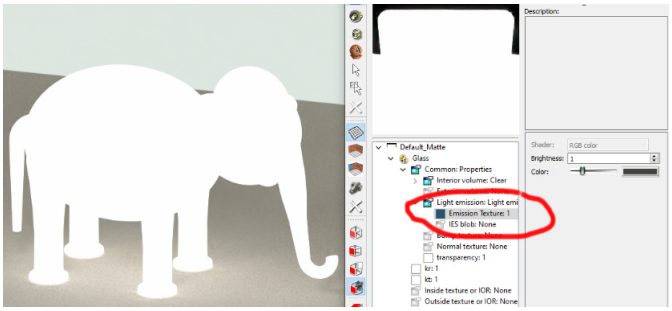
Adding a small amount of volume emission. Which can reduce the black areas. Whilst this will make the glass less clear in places. This will be acceptable in many circumstances.



Light emission.

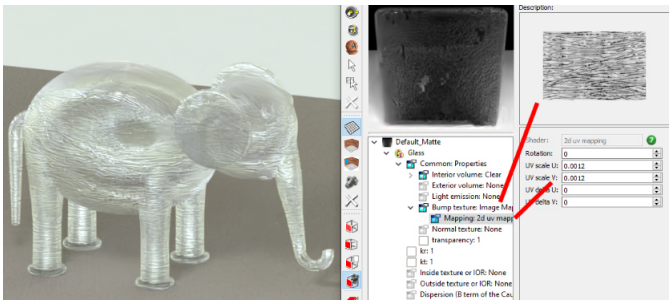
Like most materials. Light emission can be applied to glass objects. And as normal, this will essentially white-out the object.

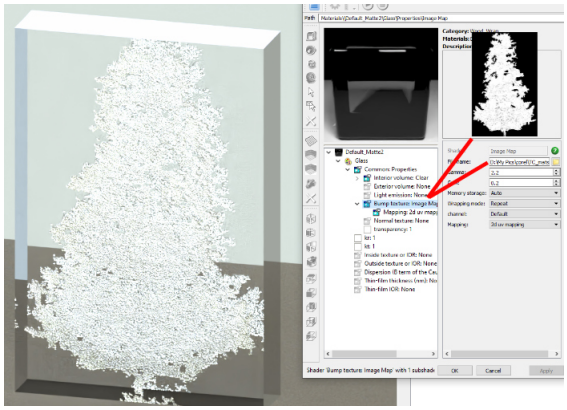
The light is emitted from the surface normal's, in all directions.



Bump map.

Bump maps work with a glass material. There are so many variations. It is impossible to show all variations.



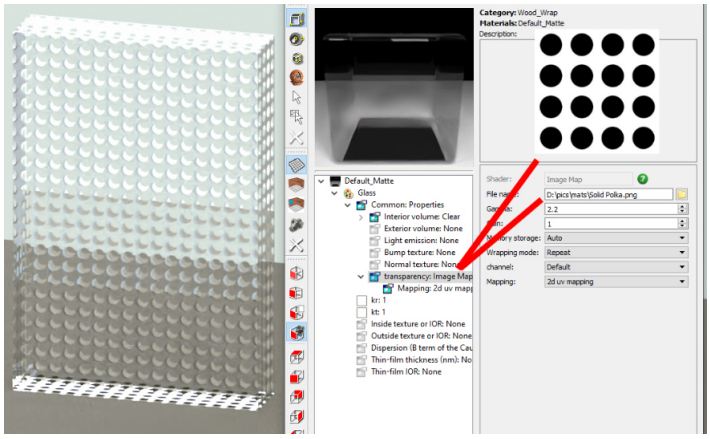


Bump can also be used to 'etch' a pattern into the glass. Though obviously it is cosmetic, in that it doesn't alter the geometry like displacement maps do. The map was applied only to front and back faces.

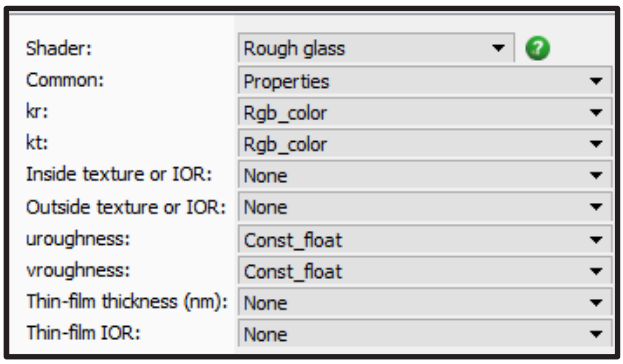
Transparency.

As glass is generally transparent, This extra option my seem odd. But because glass has an outer shell, Transparency will still work.

It works well with image maps.



Rough Glass



The Rough Glass material represents a rough dielectric surface. It works well for frosted glass and some kinds of ice.

For the most part. Rough glass works the same as regular glass, except the dispersion setting of regular glass is replaced with u and v roughness settings. **Rather than repeat everything, See Glass for further information.**

kr (reflection). Default RGB_color texture or constant reflected color of the material
This is the color of light reflected from the surface.

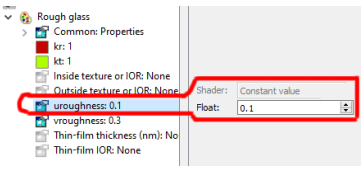
kt transmission). Default RGB_color texture or constant transmission color of the material
This is the color that light will be filtered to after being transmitted. The color given here will be applied to the material uniformly, regardless of the object's thickness. You may wish to assign a volume to the material instead to get more accurate absorption in the glass. If you use a volume, set this color to full white.

Inside texture or IOR Default None texture or constant index of refraction (only for volume-less rendering)
The index of refraction for the material. This value determines the overall appearance of the material, and should be set according to the material you are trying to simulate.

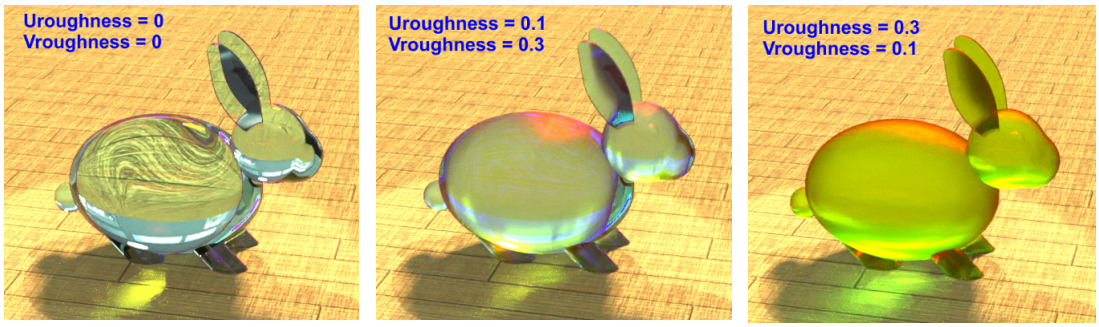
Outside texture or IOR Default None texture or constant index of refraction (only for volume-less rendering)

uroughness Default Const_float texture or constant roughness value along u coordinate of the material
vroughness Default Const_float texture or constant roughness value along v coordinate of the material

Luxcore help - This determines how shiny the material is. Lower values are shinier, with 0 being a perfect reflector and 0.8 being matte. Values between 0.8 and 1 are an unrealistic "super-matte" and should be avoided.



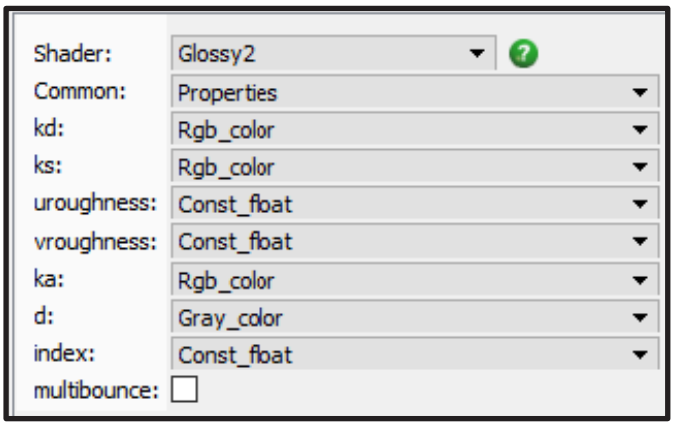
Changing the roughness Affects how the roughness looks. A setting of zero, is almost clear, setting U and V roughness changes the effect from vertical to horizontal roughness. Or vise versa.



Thin-film thickness (nm) Default none thickness (in nanometers) of a thin-film coating on the surface. This property is optional, if it is not defined, the coating is disabled.

Thin-film IOR Default none Refractive index of the thin-film coating. This property is optional, if it is not defined, the coating is disabled.

Glossy2.



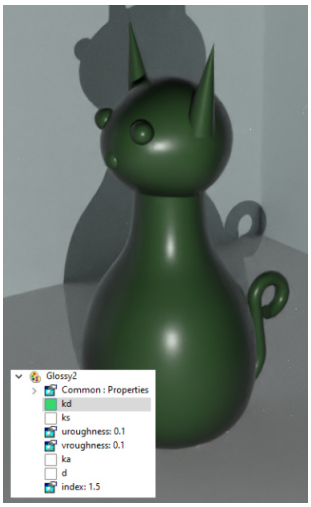
Glossy2, is like adding a gloss varnish on top of a base colour, or a glaze on pottery.

The settings include diffuse, specular, roughness, absorption and IOR. It shares the same common properties as other shaders

This makes it a versatile material for shiny surfaces.

kd Diffuse Color Default RGB_color texture or constant diffuse color of the material

Luxcore help - This is the color of the matte base layer, and will be the main color of the material.



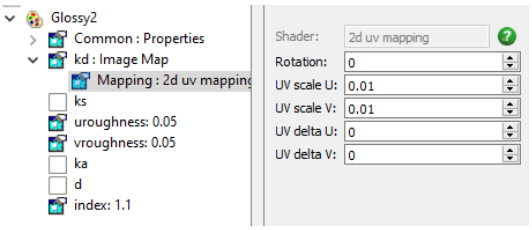
At its basic, kd is just colour,
The diffuse colour can also be supplied as an image. This will override the single base colour, and use the colours of the 2d image. In the same way one would use wrapped image, and then apply a reflectance.

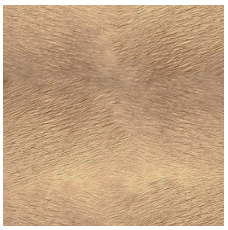
NOTE. The object will look a deeper colour if ka (absorption) is set to a white or light colour. For a more accurate kd, reduce the absorption to a darker grey.

Using a 2d map,

Like most shader, glossy2 can accept a 2d image. This is done by changing the kd property via the drop down box, to image map.

The material will remain glossy, (see IOR and roughness for changing the specular effect)



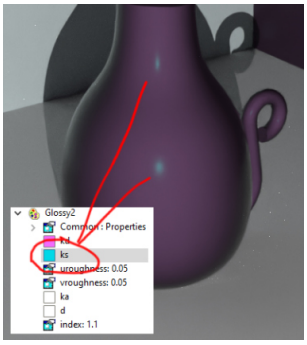


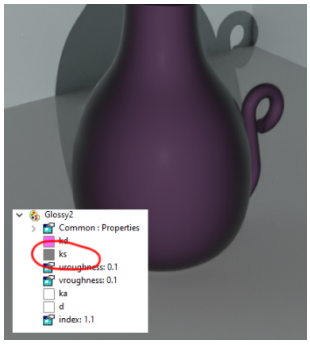
ks Specular Color Default RGB_color texture or constant specular color of the material

Luxcore help - This is the color of the varnish coat. In many real world materials, all color is in the base rather than the gloss, so this color should often be left as gray. (For example, the paint on porcelain is applied before the glaze, so your paint texture should only be on the diffuse channel). Darker colors will make the material less shiny, especially at shallow angles. Optionally, you can specify an index of refraction for the coating material, however this will prevent you from coloring the coating, it will always be gray. This can be useful if you want to specify a specific material making up the coating. For example, a wet floor would have IOR = 1.333, the IOR of water.

The specular color should not be set higher than about .25, and many everyday materials will have much lower values, such as .03-.05.

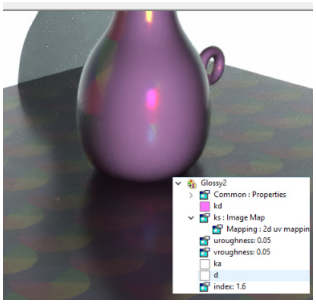
Whilst colouring the specular factor, is a simple matter of changing the specular colour, in many cases this will not look natural.





Reducing the specular colour towards black, will reduce the specular effect. Things like leather, may have a specular effect, but it is usually not shiny like plastic. Therefore reducing the specular effect may be necessary.

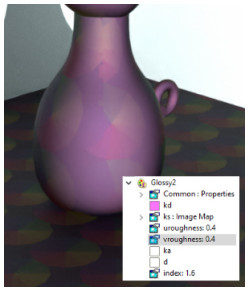
Plugging a 2d image into the specular slot, can change the effect, as lighter areas will remain specular, and darker area will be flatter. However it doesn't just change the specular effect, it imprints the image design into the varnish of the objects.



uroughness value 0 to 1 Default Const_float texture or constant roughness value along u coordinate of the material

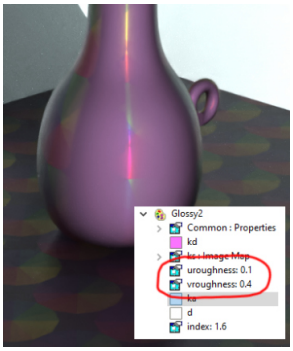
vroughness value 0 to 1 Default Const_float texture or constant roughness value along v coordinate of the material

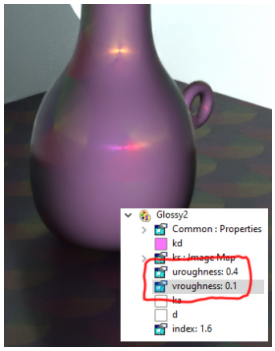
Luxcore help - This determines how shiny the material is. Lower values are shinier, with 0 being a perfect reflector and .8 being matte. Values between .8 and 1 are an unrealistic "super-matte" and should be avoided.



Small amounts can use the same specular effect roughened up a bit, simply making the shiny bits a little less shiny, Large amounts makes it more flat with no shiny bits. NOTE. The odd colours are because I forgot to change the image map back to colour.

If one used a specular image map.
Using 0.3 to 0.4 for both u and v, can gives a nice sheen to an object without the bright spot normally associated with specular.





The u and v roughness also controls the anisotropic effect.

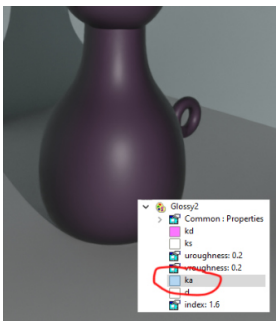
If both u and v are the same, the specular effect is isotropic. i.e. the effect is calculated the same in any direction.

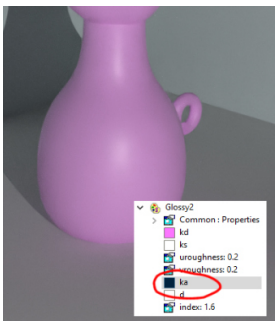
By changing just one of the roughness settings, the effect becomes anisotropic. In that the effect is calculated differently depending on the direction of the each roughness setting.

ka Absorption Color

Default RGB_color texture or constant value of coefficient of absorption of the coating layer

Luxcore help - These allow you to specify the color and depth of light absorption by the varnish. Note that since this is an absorption color, it will seem to work "backwards". Setting it to blue will cause blue light to be absorbed, leaving you with a yellow-orange appearance. You can invert the color before plugging it into this slot to make it easier to achieve the desired look. To disable this option, set the color to full black or the absorption depth to 0 nm. Conversely, the absorption depth needs to be greater than 0 to see an effect.





Normally one would leave this at a grey scale colour. This can then be used vary the amount of light that hits the base colour kd. Setting it to white, means a lot of light / colour is absorbed and make the object a deeper colour. This is similar to applying a finish coating on many hardwoods.

Setting it to black means no absorption, and the full base colour (kd) shows through.

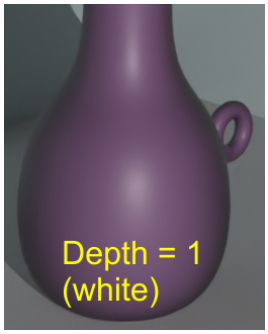
Setting ka to an rgb colour, will normally invert the specified colour. For example, Using full red (rgb 255, 0, 0), will absorb red and change the displayed colour to cyan. One can change the drop down box, of Glossy2, from ka = rgb color, to ka = color depth. This appears to internally invert the colour chosen before being calculated, thus, what one sets, is what one gets. However, personally I would leave absorption at a grey scale, so the original kd object base colour is the controlling factor.

d Coating Depth

Default Gray_color texture or constant value of the depth (thickness) of the coating layer for absorption effects. (0 = disables)

Alters the depth of the varnish in nanometres (nm). Although preset to d = grey colour, one could easily change this to d = constant float, and use values between 0 (no absorption depth), to 1 max absorption depth.

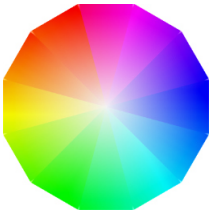
Unusual effects can be created, using a 2d image file to supply the absorption colour. The pattern applied to the varnish will be an invert of the original image.

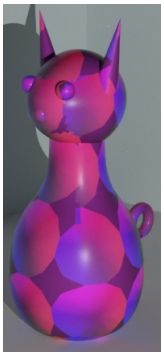


Using a 2s image.

A 2d image can be added as an absorption map, (change drop down box in glossy2 - ka to image map).

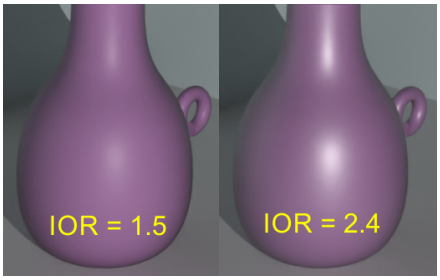
The map will be applied only to the absorption layer, so it will likely completely change the object look. Depending on other settings, this can mix with the kd colour.





The drawback is that he effects can be unpredictable. Therefore it may still be preferable to apply the image to kd, and use absorption as grey colour, to adjust the layer (for example. varnish)

Index (IOR). Default Const_float IOR of the coating. IOR overrides color Ks if both are specified



The IOR factor will change how the specular affect works. A higher number will increase the specular effect. Low numbers decrease the effect. If using an image as the IOR. Darker colours will increase the specular effect, and lighter colours reduce the effect. and it will imprint the 2d design into the varnish (normally as a grey scale).

As with all settings, they all combine to produce the desired effect. Therefore the finished IOR effect, will also be determined by the roughness settings.

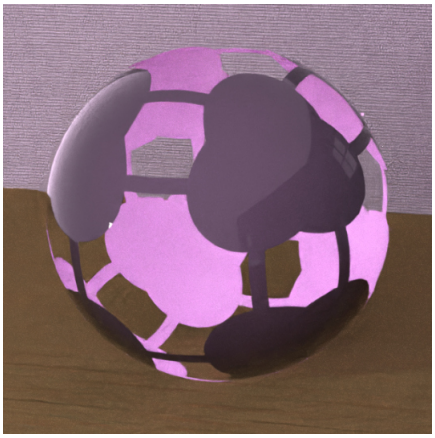
Glossy 2, other options.

Like all materials, it is possible to add extra effects. For example, applying a bump map. (Common properties)

Bump maps affect how the specular effect is produced on the surface. On a glossy surface, normally the specular effect is smooth. Applying a bump map (normally a grey scale image), will change the specular effect of the surface. Like applying a thin finish coat on a rough piece of wood, instead of sanding the wood smooth first.

Note. A bump map may darken the finish, so one may need to compensate by adjusting other factors.

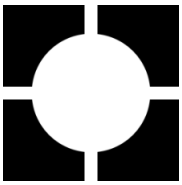
Transparency (common properties).



For simple transparency, one can adjust the colour. A better use for transparency is to apply an imagemap.

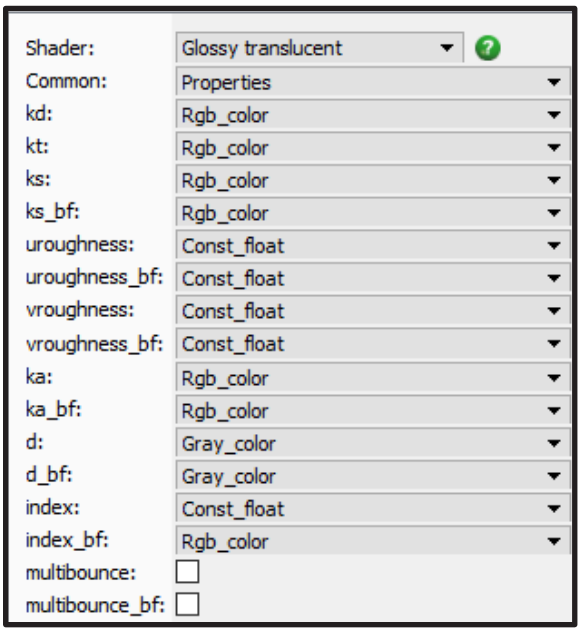
Black areas, become transparent, whilst white areas remain glossy.

As can be seen, the glossy affects are produced on the surface, so the inside will be simply kd.



The Glossy coating material allows for inside and outside adjustment.

Glossy Translucent



The two difference between this and Glossy, is the inclusions of Transmission colour and Double sided (bf)

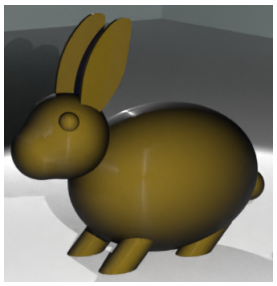
Luxcore help - This material is well suited to most organic materials, including skin, internal organs, leaves, eyes, milk, ketchup, alien slime, etc. It can also work well for soap, and certain plastics.

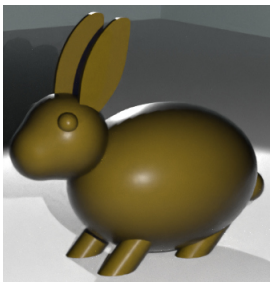
Note. This shader may seem complicated due to the many settings. This is simply because most functions have a normal face, and back face (bf) option.

For many solid objects, one can simply ignore the bf settings. As they will play no part in the render.

The bf setting is like ticking double sided in RedSDK.

kd (Diffuse Color) Default RGB_color texture or constant diffuse color of the material
Luxcore help - This is the color of the matte base layer, and will be the main color of the material, assuming it is set above 0.4 - 0.5 or so. The diffuse color is the color reflected off the surface and any light not reflected is transmitted. So in order to get a highly translucent material you need to use a darker diffuse color.

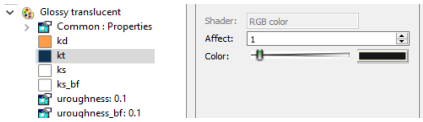


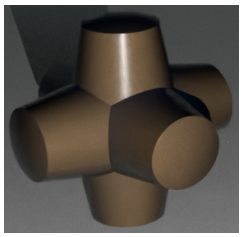


By default the results may not be very impressive. Though the objects shown right are not the best for this shader. By default the v roughness is set at 1, this is too high for a glossy item. Reset roughness settings to 0.1, or smaller (including bf settings). Should give a better glossy surface to start off with.

Because of other settings, the actual colour may be darker than the diffuse colour which has been set..

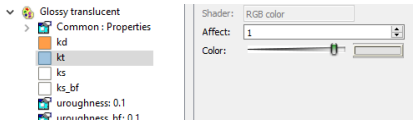
kt (Transmission Color) Default RGB_color texture or constant transmission color of the material
Luxcore help - This determines the color of light that transmits through the material. For best results, you should leave this as 1.0 (full white) and assign a volume to the material and use that to give the material its internal color. This allows for volumetric color absorption and subsurface scattering. Color from the transmission setting is applied uniformly to transmitted light, and can make your object look unnaturally uniform in color, so using a volume is recommended.

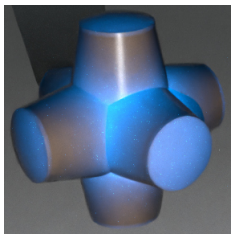




This works with transmission through an object, therefore thick solid objects, are likely to not show any effect by just altering the transmission.

Fig's right. Are a hollow object with a light source inside. Changing the transmission allow more, or less light to transmit through the material.





At the time of writing, the colour shown is the invert of the kd, kt appears to only use the grey scale tone irrespective of what kt colour is chosen. Unknown if this is by design or fault.

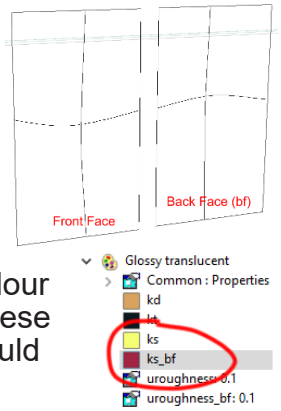
ks (Specular Color) Default RGB_color texture or constant specular color of the material
ks bf (Specular Color) Default RGB_color texture or constant specular color of the material

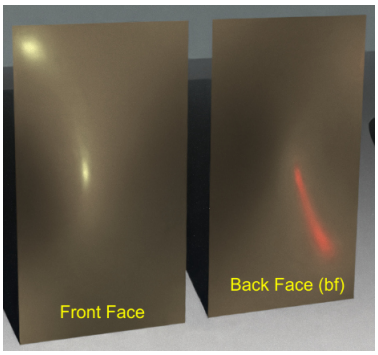
Luxcore help - This is the color of the varnish coat. In many real world materials, all color is in the base rather than the gloss, so this color should often be left as gray. (For example, the paint on porcelain is applied before the glaze, so your paint texture should only be on the diffuse channel). Darker colors will make the material less shiny, especially at shallow angles. Optionally, you can specify an index of refraction for the coating material, however this will prevent you from coloring the coating, it will always be gray. This can be useful if you want to specify a specific material making up the coating. For example, a wet floor would have IOR = 1.333, the IOR of water.

The specular color should not be set higher than about .25, and many everyday materials will have much lower values, such as .03-.05.

This is the first of the front face / back face settings. The images right are curved faces, to shows the specular effect. Note. These are a sheet object (zero thickness), to ensure they are not affected by ka.

Specular colour is just that, it doesn't alter the specular effect. For specular effect use the roughness settings. Simply set the desired colour required. Most of the time, one would leave these settings are at a grey scale colour. Rarely would one need an actual RGB colour.

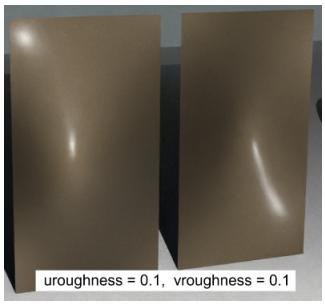


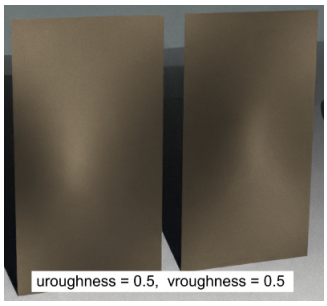


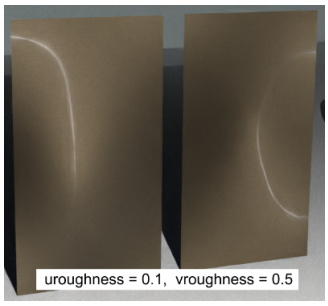
uroughness Default 0.1 texture or constant roughness value along u coordinate of the material
uroughness bf Default 0.1 texture or constant roughness value along v coordinate of the material
vroughness Default 0.1 texture or constant roughness value along u coordinate of the material
vroughness bf Default 0.1 texture or constant roughness value along v coordinate of the material
Luxcore help - This determines how shiny the material is. Lower values are shinier, with 0 being a perfect reflector and 0.8 being matte. Values between 0.8 and 1 are an unrealistic "super-matte" and should be avoided.

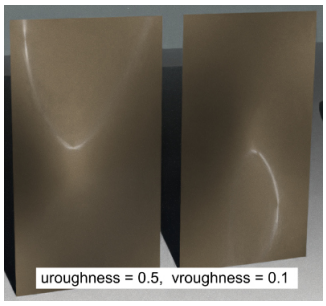
Roughness has two functions. Lower values are high specular, higher values tone down the specular value. If both U and V are the same figure, it is usually classed as isotrophy

The second is an anisotrophy effect. In that using a different value for u and v, changes the way light is reflected off the object. This is a little bit more complicated, due to having a front and back face to alter.



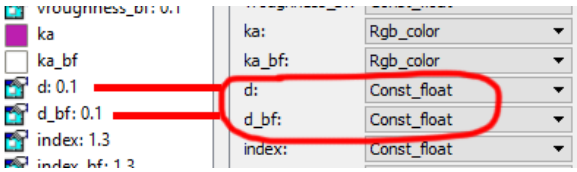




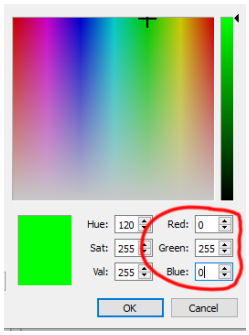


ka (Absorption Color) Default RGB_color texture or constant value of coefficient of absorption
ka_bf (Absorption Color) of the coating layer
d (Absorption Depth) Default 0.0 texture or constant value of the depth (thickness) of the coating
d_bf (Absorption Depth) layer for absorption effects. (0 = disables)
Luxcore help - These allow you to specify the color and depth of light absorption by the varnish. Note that since this is an absorption color, it will seem to work "backwards". Setting it to blue will cause blue light to be absorbed, leaving you with a yellow-orange appearance. To defeat this option, set the color to full black (0.0).

The first think I'd do, would be to change the depth drop down box from grey colour to constant float. Because this is the varnish depth, so doesn't need a colour setting.



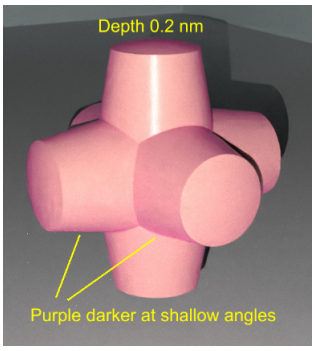
Colour, is the colour that is absorbed by the coating leaving the remaining colours to be reflected from the surface. So the actual colour is the 'invert' of the colour one selects. For example, one sets a green colour (rgb 0, 255, 0). Therefore the invert is purple (rgb 255, 0, 255).



Remember. This is the coating, and has a depth setting (d and d bf).

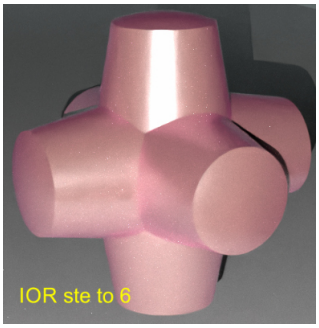
Applying a thick coating, will obscure the diffuse kd colour.

Applying a very thin coating can show more of the original kd colour when light hits at steep angles, whereas light hitting at shallow angles will show more of the absorption colour.



Index (IOR) Default Const_float IOR of the coating. IOR overrides color Ks if both are specified
Index_bf (IOR) Default RGB_color

See separate page 155 for IOR. Normally one would keep to specific IOR values for realism. However, Sometimes it pays to think beyond the norm. For example fig right has the IOR increased to 6, to give a more metallic copper look.

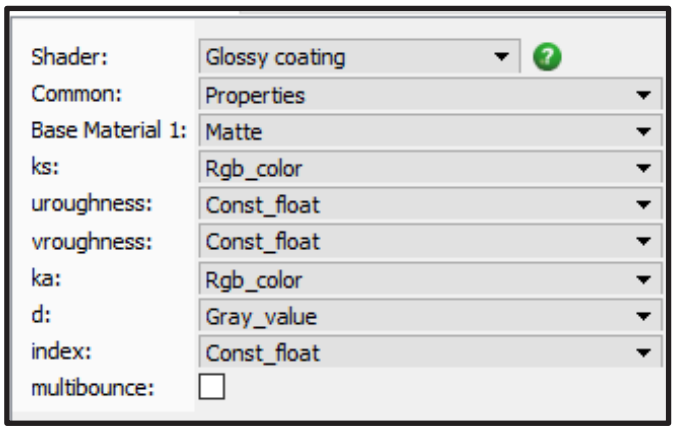


Multibounce
Multibounce_bf Default Off simulation of asperity (velvet-like reflection) on the glossy surface
Luxcore help - The option will simulate light being scattered at the surface by things such as fuzz or fine hair. It is useful when using glossy for a skin or cloth material, and will give a soft, fuzzy appearance.

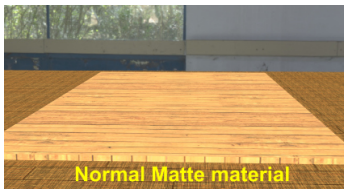
I have no idea if this works in TurboLux or not.

Glossy coating

Glossy coating is an "empty gloss coating" material. It is used to add gloss layers on top of other materials. It is something of a special glossy material that has no base reflector of its own, but rather takes another material node as its base.

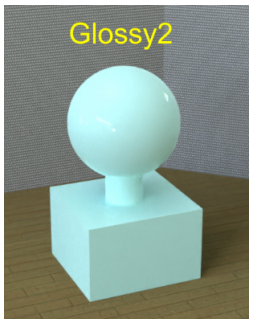


The glossy coating can be used to add a surface finish like gloss or satin varnish to other materials. Obviously some materials, like glass, already have a gloss surface, but material like Matte, or glossy2 can be used as a base material.



Base Material Default Matte. name of base material, the coating layer is added on top of this material

The base material is the reflector beneath the glossy coating. If you use the matte material, the result will be the same as the regular glossy material. The material specified here can be any material type, including null, mix, layered, or another glossy coating material. The only exception is that the initial node in the chain is not glossy coating, as all glossy coating materials need to have a base material specified.



ks Specular Color Default RGB_color texture or constant specular color of the material

This is the color of the varnish coat. In many real world materials, all color is in the base rather than the gloss, so this color should often be left as gray. (For example, the paint on porcelain is applied before the glaze, so your paint texture should only be on the diffuse channel). Darker colors will make the material less shiny, especially at shallow angles. Optionally, you can specify an index of refraction for the coating material, however this will prevent you from coloring the coating, it will always be gray. This can be useful if you want to specify a specific material making up the coating. For example, a wet floor would have IOR = 1.333, the IOR of water.

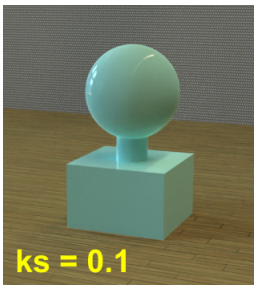
The specular color should not be set higher than about .25, and many everyday materials will have much lower values, such as .03-.05.

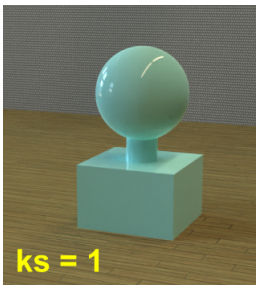
If using a base material like glossy2, the specular in glossy2 may interfere with the coating ks.

Materials like Matte. Lend themselves very well to an additional specular coating.

Although one can use a colour with specular, most of the time one would leave it as grey scale.

In the images on the right. I change the drop down box to 'float'. This prevents the colour from changing.



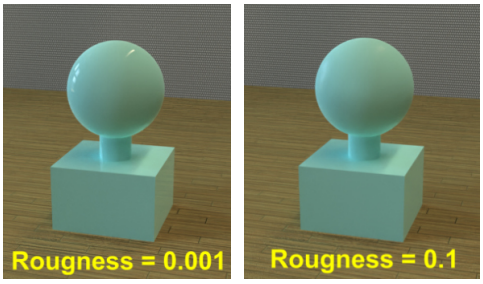


uroughness Default Const_float texture or constant roughness value along u coordinate of the material

vroughness Default Const_float texture or constant roughness value along v coordinate of the material

This determines how shiny the material is. Lower values are shinier, with 0 being a perfect reflector and .8 being matte. Values between .8 and 1 are an unrealistic "super-matte" and should be avoided. If your exporter uses the exponent setting to control roughness, higher values are shinier, with 0 being matte. To convert exponent to roughness use the following formula: ROUGH = SQRT(2 / (EXPONENT + 2))

Turning up the roughness will reduce or eliminate a specular setting. A setting of 0.001 is gloss with minimum roughness. A setting of 0.1 will show a tiny bit of sheen. A setting of 1 is matte rough.



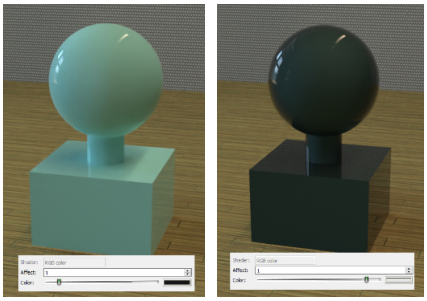
ka absorption Color Default RGB_color. texture or constant value of coefficient of absorption of the coating layer

d absorption Depth Default Gray_value texture or constant value of the depth (thickness) of the coating layer for absorption effects. (0 = disables)

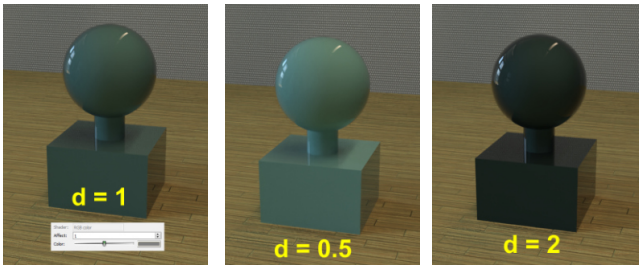
These allow you to specify the color and depth of light absorption by the varnish. Note that since this is an absorption color, it will seem to work "backwards". Setting it to blue will cause blue light to be absorbed, leaving you with a yellow-orange appearance. To defeat this option, set the color to full black (0.0).

If the absorption (ka) is set towards black, the dark colours are absorbed leaving the lighter colours to pass through the coating and show the base colour.

If (ka) is set towards white, the white light is absorbed by the coating. Allowing dark colour to pass through. This makes the coating darker, or even black.



Altering the depth (d) will absorb more or less light. Thicker absorption depth will restrict a lot of light, leaving a dark object.



Index (IOR) Default Const_float IOR of the coating. IOR overrides color Ks if both are specified

May or may not make any difference.

Matte Material

Matte material is a flat non-specular texture applied to an object. It can be used on its own, with an image map, bump map, transparency, or with volume.
Although the matte material is flat, it will still cast and receive shadows.

Apart from common properties, there is only one option available,

Kd (Diffuse). Default RGB_color. texture or constant color of the matte material

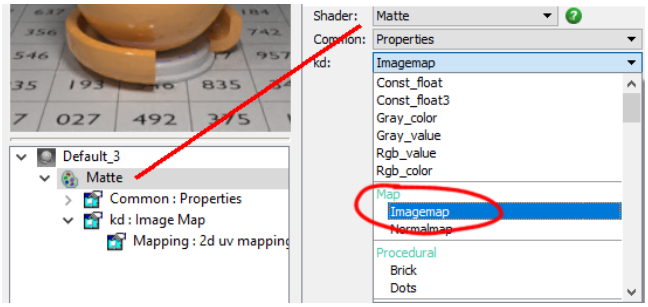
Luxcore help - "This is the main color of the material. Please avoid using a color value above 0.8 as no material in the real world is that bright/saturated".

One simply sets the desired colour for the object. Using the rgb colour picker, and affects slider. Ensure you are altering the 'Active Drawing' material, not the TC database material.

The Sigma setting which is commonly in Matte material, has been implemented as a separate material. See Matte Rough material.

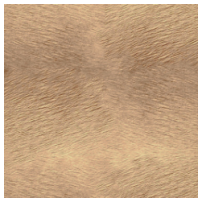
2D image

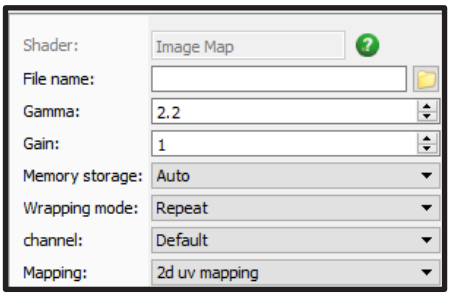
Matte material lends itself well to using a 2d image as the kd diffuse colour. Images can be black and white or full coloured.



To use. Click the material name to bring up the shader. Expand the 'kd' drop down box. Locate Imagemap and select.

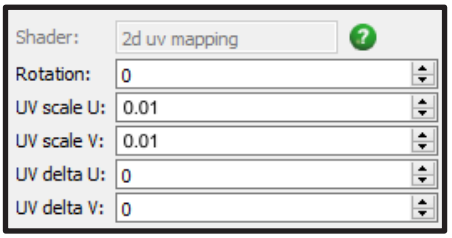
This opens up two more options.
(See page 110 For image map description).





Click kd imagemap in the left pane of the render manager (Active drawing)
In the dialog, select the appropriate file, by clicking the folder to bring up the file open dialog.

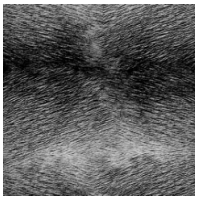
On the 2d uv mapping page section, enter the desired uv scale, and a delta distance if required.



Other options, like using blender maps. (clouds, voronoi etc), also work well with Matte, especially when combined with transparency.

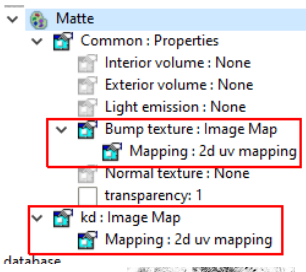
Matte… Other Options.

Bump maps (common properties).



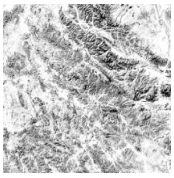
Being a flat wrapping, Matte lends itself well to bump maps. Though it can darken the original colour. Therefore you may need to compensate for the if it is not desirable Coloured images can be used as bump. But a black and white image is likely to give the better results.

Bump maps can be applied to a plain kd colour, or by changing kd to use an image map.



Applying to a base colour , the bump alters the light as it hits various parts of the objects.

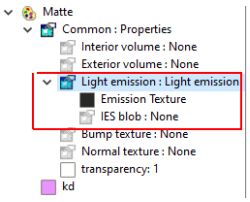
And thus changes the effect from a plain flat colour, to multiple shades. From black to white.

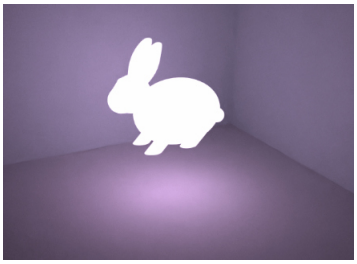


Used in conjunction with an image. i.e. setting kd to use an image map. Will completely change the look of the object, though because it is still a matte colour, one doesn't get any specular effect.

The most obvious use of this is when using rocks, gravel, hair / fur, brick and plain (unvarnished) wood.

Light emission.





When one looks at a 'lit' light bulb, one sees the glow, one doesn't see any specular effect of the glass itself. Therefore the simple matte shader can be used for creating a light emitting object.

The image right, is a completely black scene, no lights, no scene, and no luminance. It is simply lit by the object itself.which effectively operates like a point light, in this instance.

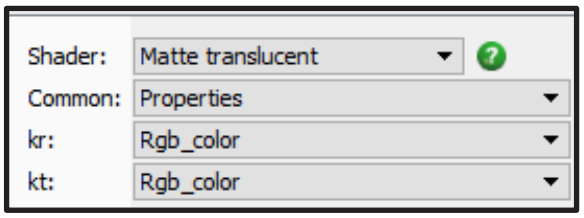
Volume.

Matte, can also be use with a volume effect. Though as it does not have a transparent shell, it may be necessary to reduce the transparency. This will allow the volume to show through.

Transparency (Common properties)... *Luxcore help* - "Values 0 to 1. Sets whether the objects is opaque = (1), transparent = (0) or anywhere in between"

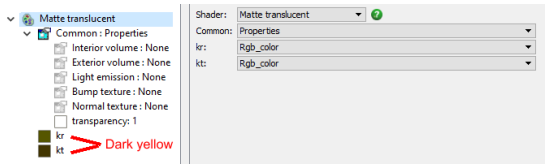
When using colours, this can be opposite to some render engines. Black becomes transparent and white is opaque. Lowering the value means a more transparent object. Being a matte material, with transparency it can look a bit bland.

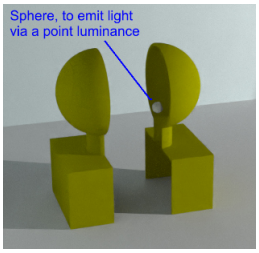
Matt Translucent Material

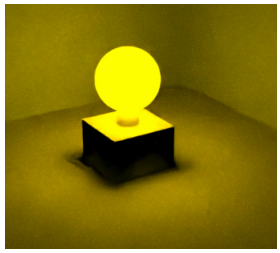


Luxcore help - Matte Translucent allows for transmission of light through the material. It is well suited to plastic lamp covers, where glossy translucent would simply add calculation time, as well as dull polymers, some dull organic materials (such as cheese or banana peels) and some kinds of textiles.

Although this texture has a kr reflective colour, it is not specular like a normal glossy surface. It is still a matt material. Being translucent, it will let light through, but one cannot see though it. Fig … shows a simple setup, with a hollow object (2mm thick), and a single sphere inside. This sphere will be a point luminance, to provide the light.







All other lights were turned off, no sky, sun etc. The only light in the scene being the point light inside the object. When rendered one can clearly see the light emitting through the object,

kr (Reflection color). Default RGB_color. texture or constant reflected color of the material

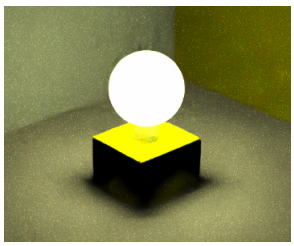
Luxcore help - This main color specifies the color of light that is scattered off the surface of the material. Any leftover light will pass through the material. To get a highly translucent material, this value needs to be a dark color to allow light to pass through it. To get an overall dark material, select a dark reflection color and a dark transmission/volume color and that will make the material absorb light.

In the above scene. If kr is made a light colour, the scene renders almost black, because the light is reflected back onto itself and not through the material.

kt (Transmission color). Default RGB_color. texture or constant transmitted color of the material

Luxcore help - This determines the color of light that transmits through the material. This color is applied uniformly to transmitted light, regardless of the thickness of the object and can make your object look unnaturally uniform in color. For best results, you should leave this as 1.0 (full white) and assign a volume to the material and use that to give the material its internal color. This allows for volumetric color absorption and subsurface scattering.

In the above scene. If transmission is made a light yellow colour, the white light of the sphere passes straight through and is not change to yellow by the transmission colour.

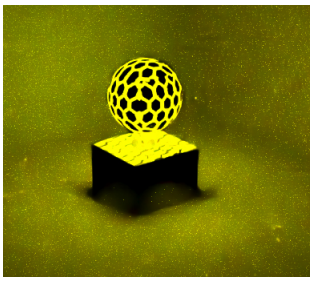


Matte Translucent … Other Options.

The same options as for Matte material, applies to Matte translucent. Obviously being translucent, the effects will not look exactly the same.

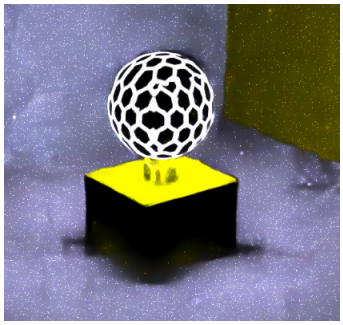
2d image.

Applying a 2d image to reflectance kr can mean white areas reflect the light back to the source. Whereas black areas allow light to flow through.

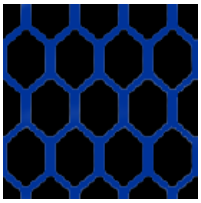


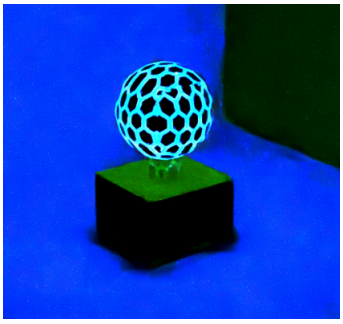
Adding a 2d image to kt will also change how the light passes through the object.

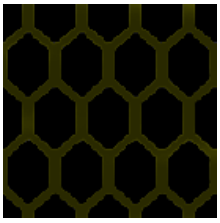
However because white coloured images allow light to pass easily, the white light is transmitted straight through.

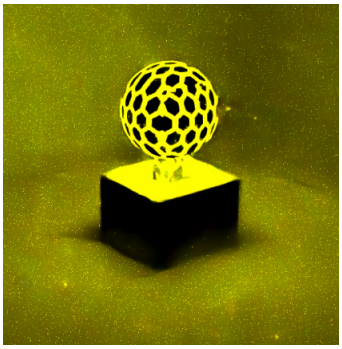


Changing the white areas to a darker colour will subdue to white light and allow a coloured light to be passed through the object.









Bump map.

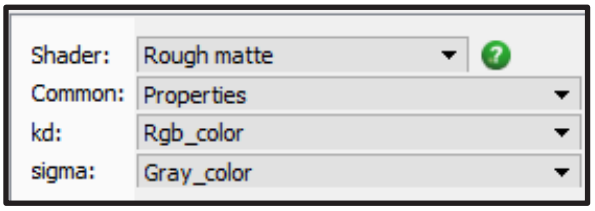
Although one can add a bump map. Because Matt Translucent does not have a kd, setting, the map will likely not create a proper bump. This is because the map would have to be applied as a reflectance of transmission.

Transparency.

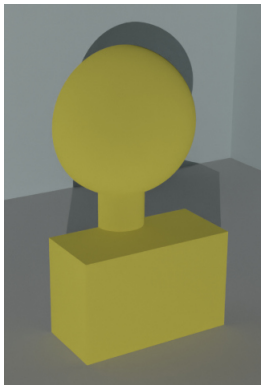
Although transparency including transparency map, can be used. Too much transparency will defeat the purpose of using a translucent material.

Rough Matte

Rough matte differs from normal matte, by having a sigma (roughness) setting. Kd (diffuse) works the same as a normal matte material.



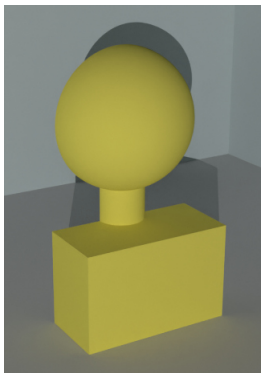
Kd. (Diffuse colour). Default RGB_color. texture or constant color of the matte material



Sets the base colour of the material. (sigma set t0 white)

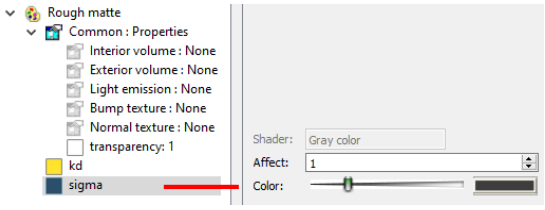
Sigma Value 0 to 45 Default Gray_color.

Luxcore help - "The sigma parameter determines how rough the surface appears, with zero being perfectly smooth. A very rough material will have a sigma of no more than 30 degrees in the real world. The rougher the surface the more light is reflected back at grazing angles, giving it a flatter look".



The sigma setting, affects how the light bounces off a slightly roughened surface. The actual material remains matte, but it can gain definition because the light reflects at differently angles.

Setting to white, and the material looks flat.

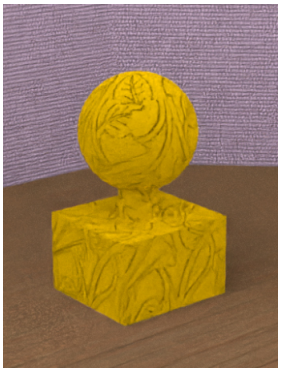


Setting towards black gives increased roughness, and shading as the light reflects.

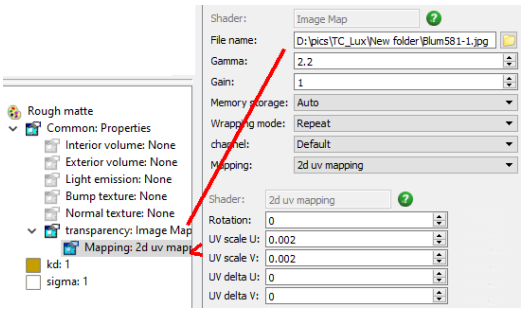
Note. The actual effect will differ depending on the lighting, and scene setup. A plain colour scene background, like white, will likely produce poor results.

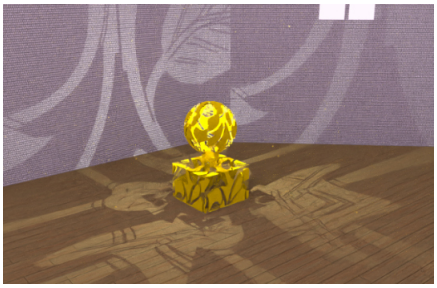
Bump map

Bump maps work fine with Matte rough.

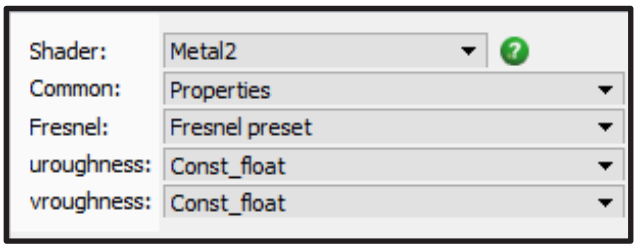


Transparency map can produce good effects. Applying a black and white image as transparency (in common properties), means the black parts become transparent. Using a point light inside the object, it creates a pattern showing through the transparent parts of the object.





Metal2



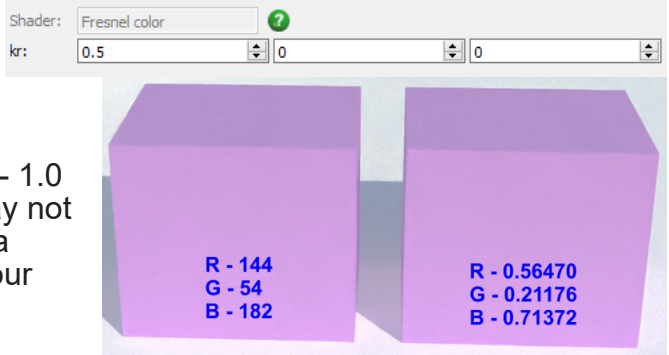
Basic Metal is one of the simplest materials, until one starts trying to figure out complex Fresnel effects, using custom NK data. However Normally one would never need to delve into NK (refraction wavelengths).

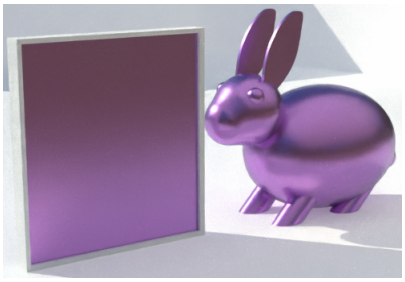
Although one can get highly polished metals, many (if not most) have a roughness, This may be minor like chrome, to rough like iron. However by altering just two settings, one can change it, to create a metal to suit.

The shader does not have a simple 'kd - diffuse selection, one has to use the Fresnel selection drop down box.

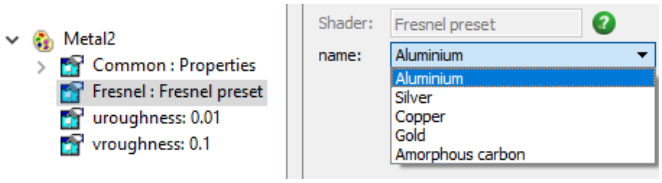
Fresnel Color (kr) Default Preset.
Fresnel colour is preset to use RGB values, and this does not have a colour picker.

Colour uses sRGB input, meaning it uses 0.0 - 1.0 values, instead of 0 - 255 values. Whilst it may not be totally accurate due to gamma correction, a rough guide is simply to divide each RGB colour channel by 255

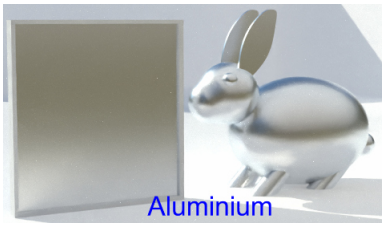


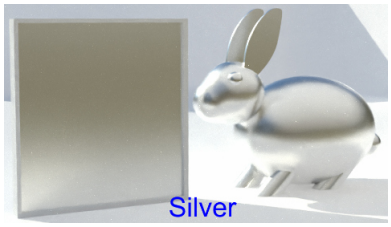


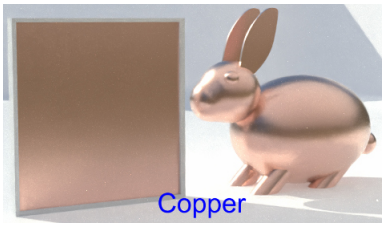
Fresnel Preset
This defines the base color of the metal. Metal presets can be loaded by using the drop down box next to Fresnel (as above). .

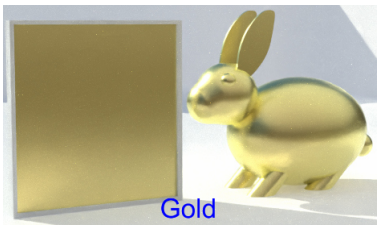


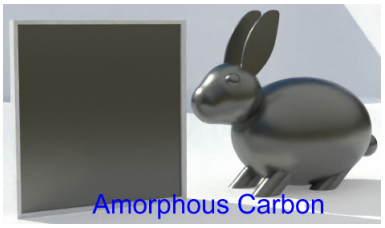
There are no user available settings when using one of the presets. And there are only 5 presets available.











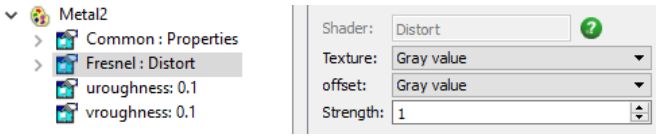
Fresnel Sopra
Fresnel Luxpop

Fresnel NK files with measured data can be used as well in either the Sopra or Luxpop formats. NK (N - refractive index, K - Extinction coefficient wavelength.

Sopra and Luxpop are text files, downloaded from the internet. However, from what I read they either don't work, or don't work as expected. Feel free to search the internet.

Fresnel Distort.

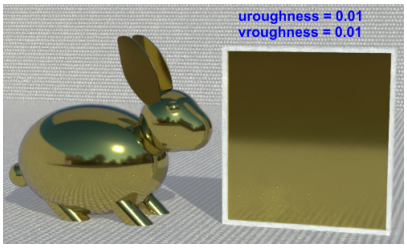
At the time of writing, this doesn't work on my computer using 2022.

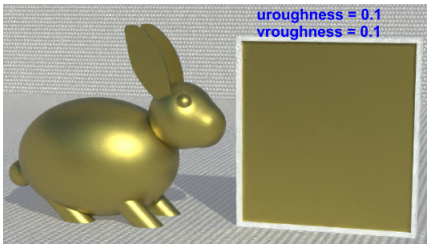


uroughness. Default Const_float.
vroughness. Default Const_float.

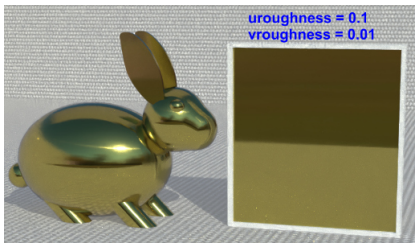
This determines how shiny the material is by varying the roughness of microfacets. Lower values are shinier, with 0 being a perfect reflector and 0.8 being matte. Values between .8 and 1 are an unrealistic "super-matte" and should be avoided.

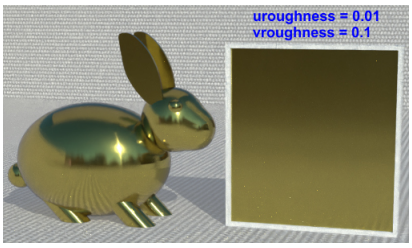
If both u and v v roughness are the same value, the light / surface is isotropic. Often producing a round or ellipse specular effect. Increasing or decreasing the values the same, alters the object it from mirror to rough matte.





If the u and v factors are different, it produces an anisotropic effect, and the specular affect is stretched in the u or v direction. This may be more natural in some circumstances.





Metal other options.

Internal Volume. External volume, Transparency, etc. have no real place with Metal2.

Bump Map.

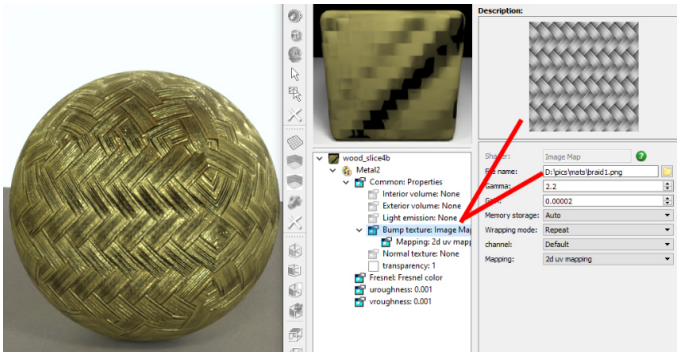
Bump can be used, Though the metal aspect of Disney allows more control.

With metal2 there are three options. Apply a bump map only. Applying a bump map + U and V roughness maps. And lastly, U V maps only.

Bump map only.

Like many bump maps, it can slightly darken the material applied. On many occasions this can be a good thing, but there are occasions where the darker spots, don't really add to the effect.

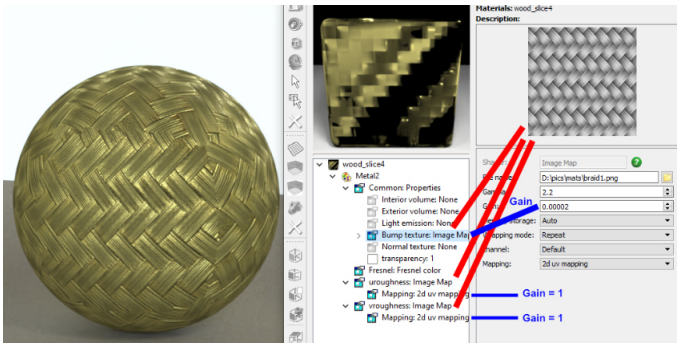
Very low values of gain may be required, eg. 0.00002



Bump map and U / V maps.

Using bump and U / V maps, may be overkill, but it can help if the bump map alone, produces unwanted dark spots.

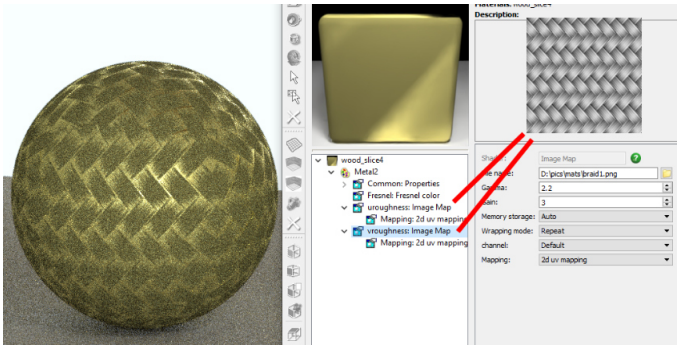
Whilst the bump map is set to 0.00002. The U / V gain is left at 1



U / V maps only.

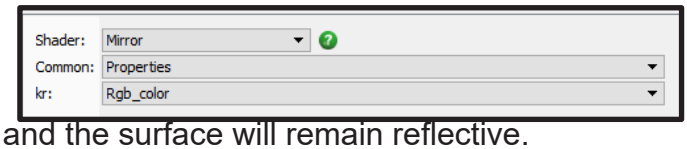
Using just the U and V maps, can lack any depth.

In some instances this is perfectly acceptable, especially if a smooth patterned finish is all that is needed.



Normal maps, may give unsatisfactory results, without a lot of trial and error.

MIrror

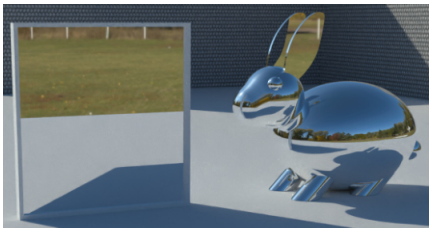


Mirror creates a highly reflective surface.
Using kr, the colour and tone can be altered, and the surface will remain reflective.

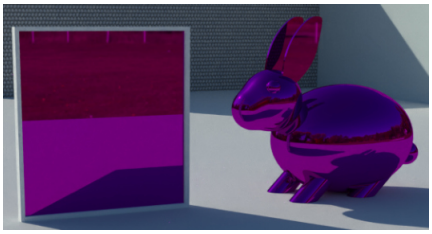
Kr (Reflection Color) Default RGB_color texture or constant reflected color of the material

This is the color of the mirror. Adjusting it will control the apparent "brightness" of the "mirror world." A value of around .55-.65 is a good starting point.

In TC there is only the 'kr' setting (in addition to common properties). Which adjusts the reflectance. The kr setting itself has two adjustments. The colour, which sets the base colour itself. And the amount, which affects the tone. The latter should normally be kept between 0 and 1. The box will accept larger numbers than '1' but this can override the colour, so defeats the object of using a colour.

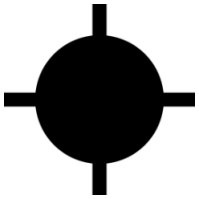


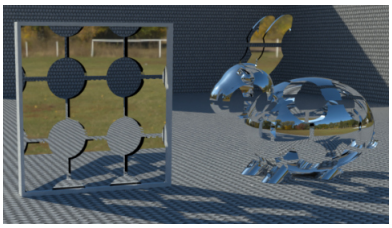
Although one can use an image map or a bump map. The results can sometimes be - not good. Better to use glossy or Disney if you want a shiny surface on an image map.

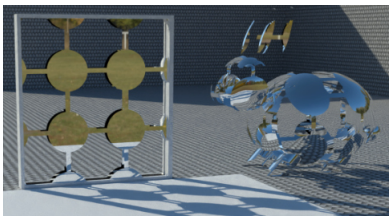


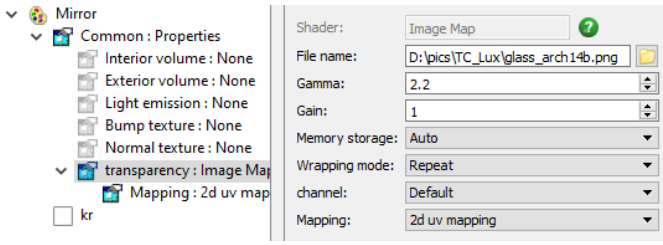
Transparency (common properties). Some unusual effects can be achieved. With white remaining a mirror reflection, and darker colours becoming more transparent

The figures right, are simply black and white images. They are applied by changing the drop down box against transparency, from rgb to image map. The images are then applied to the transparency as shown below. Don't forget to alter the image scale.in the 2d uv mapping page.









The Luxcore help files mentions Thin film. But as yet this has not been inplemented in the mirror material.

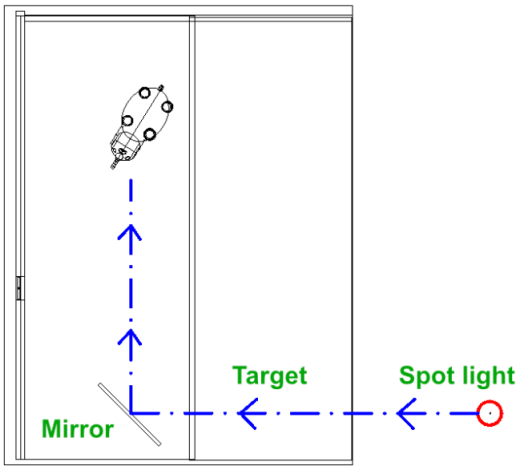
This allows you to define a film coating on top. This usually results in a rainbow colored reflection. It has two controls, a depth and IOR. The IOR should match that of your film material (Cleaning solution, which can leave such a film behind, has an IOR of ~1.3-1.35, due to it being almost entirely water). You can vary the thickness to achieve the desired effect. A thickness of several hundred nm will result in the most pronounced effect.

Mirror reflection.

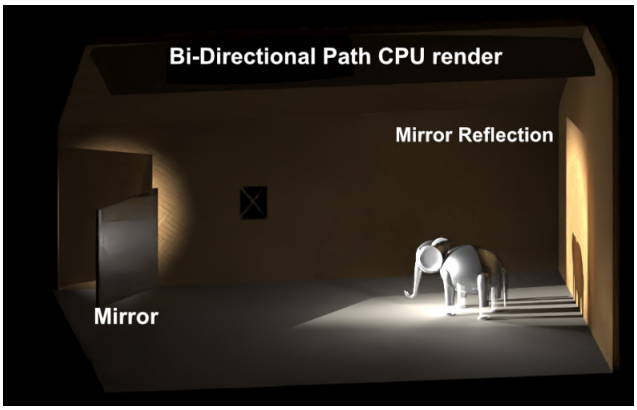
If a mirror is used in the drawing And Bi-Directional path CPU rendering is used. Light will be reflected off the mirror. This does no happen with RT path, nor Path (GPU or CPU).

This should be taken into account, if light reflection is not required.

It can be used to create certain effects of ones own making. This simple example uses a spot light as the source. Aimed at a mirror.

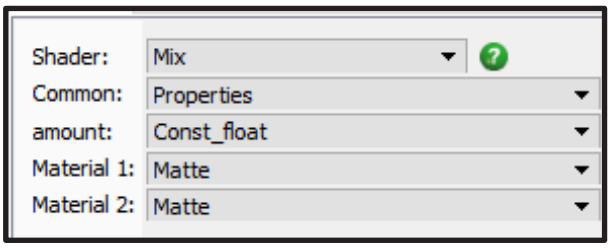


As can be seen in the finished render. The object if illuminated only from the reflection via the mirror.



See also ‘Volumetric’ in Appendix 7, for another mirror example.

The Mix material



LuxCore help file.. This is a special material that is used to combine two other material settings.

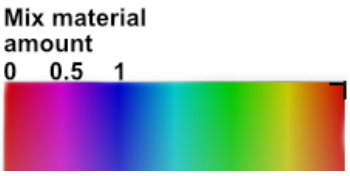
It is possible to use one mix material as a component of another mix material. This allows for complex shader combinations.

The difference between this and the Mix property available in the drop down menu's. (see page 117 Is that this works on whole material, whereas the mix property works on colours textures, and properties.

Some materials, there is no point in mixing, for example, mixing glass with Architectural or rough glass, is pointless.

Amount Default Const_float texture name or constant

This specifies how much each component material contributes to the final result. A mix amount of 0 results in only the first component, 1 results in only the second component, everything else is somewhere in between. The mix amount can also be controlled with a float(grey-scale) texture, to allow the blending to be localized. Note that mixing volumes is not possible. Any volume used by Material 1 or Material 2 will be ignored. The Mix material itself can however have a volume.

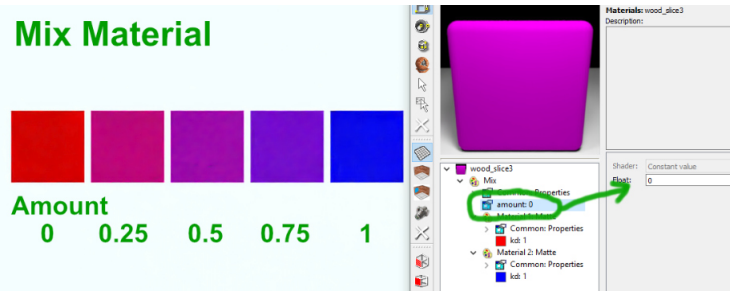


The image right, mixes two matte materials, one set to a red colour (amount 0), and one with a blue colour (amount 1).

The five squares, are simply rendered with different 'Amount' settings.

This is just a demonstration of the mix material. Obviously in practice one would not do this. But simply use the correct colour in one Matte material.

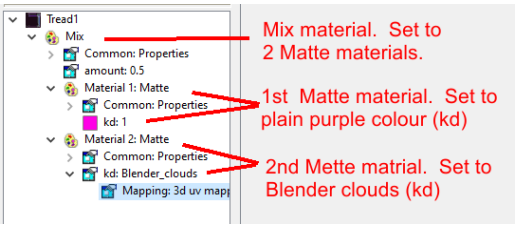
The power of Mix material comes when using images or blender textures.



Material 1 Default Matte texture name or constant

Slot to attach the first component material.

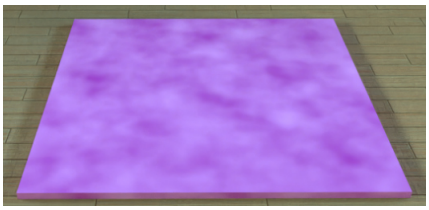
To use properties like, Colour, Blender procedural textures (clouds etc.), imagemap, Marble etc. select a desired main material like matte, then add the the property as kd to the material.



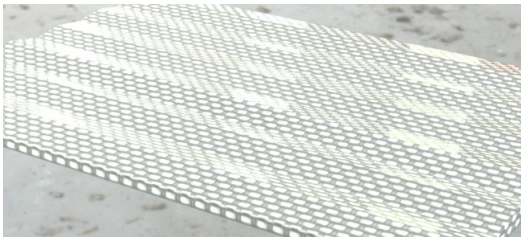
Material 2 Default Matte texture name or constant

Slot to attach the second component material.

As with the first material, Select a main material, then select the desired property using kd



The figure right, simply mixes cloth with metal,



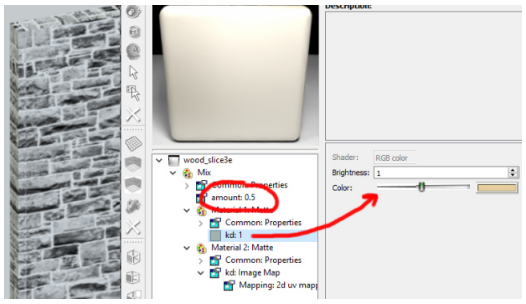
Mixing materials can save a lot of tweaking of a material in a paint program. Especially when mixing a colour with a texture or blender map.

A simple example is a wall. Say one has an image of bricks. We could adjust in a paint program, then if it doesn't look good, re-adjust in a paint program etc.

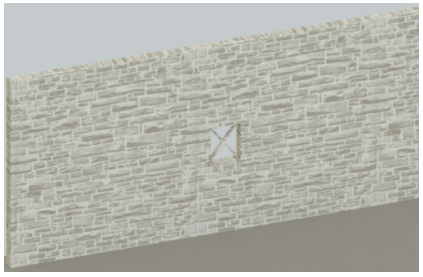
As an experiment we could use the Mix material.

Here we assign the brick texture to a matte material.
As Mix 2. We assign a matte colour to Mix 1.

And set the amount to 0.5



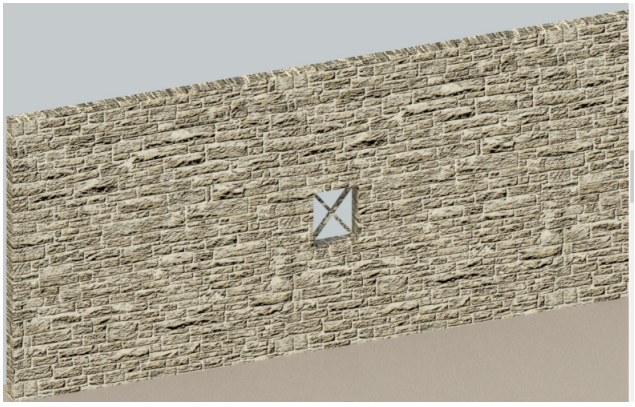
A test render shows the wall is too flat and lacks definition.



To overcome the above problem. We can tweak the settings.

First we add a bump map to the mix common properties.
Then we adjust the colour of mix 1.
Lastly tweak the gain of mix 2 texture.

The final result is shown right.

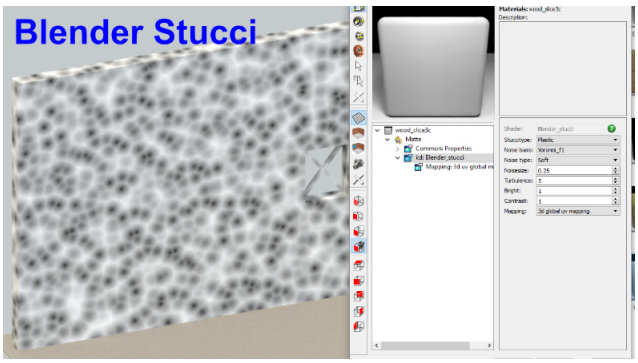


Where Mix does shine, is mixing a colour with Blender procedural textures. With the Blender stuff, one cannot adjust in a paint program, so have to mix 'on the fly' so to speak.

Often blender maps are used for bump maps. But there is nothing wrong with using them as normal kd maps.

If we just apply a Blender texture, it is normally grey scale colour.

So we need some way to add colour to it.

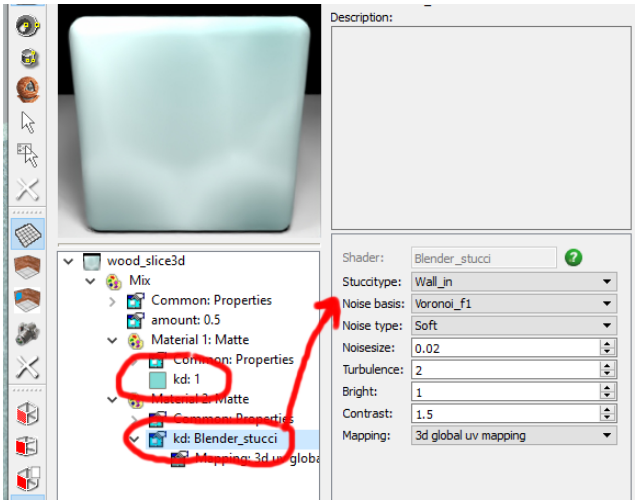


Here we can use the Mix material.

We apply the colour to Mix 1. In this case blue.

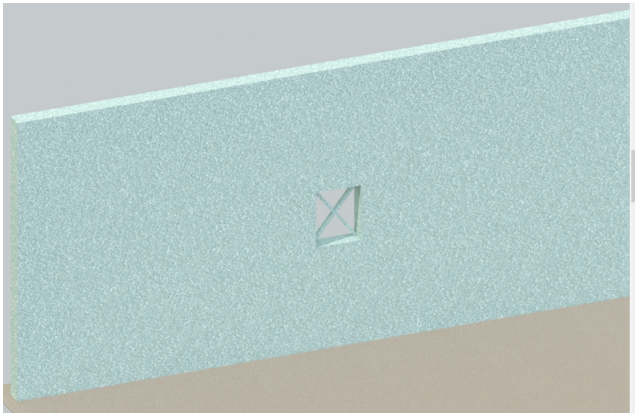
For Mix 2. We change the drop down to Blender Stucci. And set the desired properties.

Amount can be left at 0.5 for now.

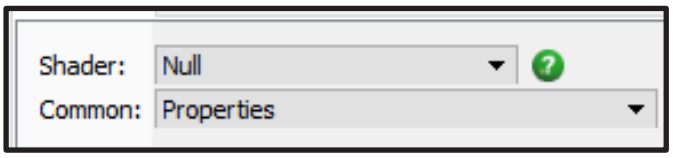


The final render. Note bump map was not used, as this is for a fine textured wall.

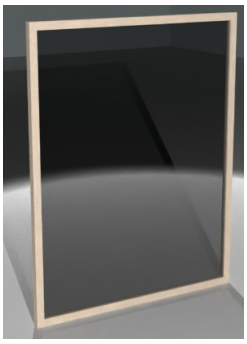
Also a bump map may slightly darken the final result.



Null Transparency

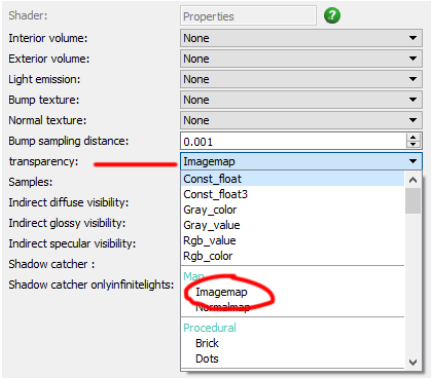


Null is a special material that does not have a visible surface. Null can be used in combination with a volume, making the object act as a container for the volume.



Null has no separate properties of its own. All settings are done through the common properties settings. One thing to note about common properties transparency. Normally with other shader, black is transparent. However because null is by default transparent, black is now opaque. This is especially important if using a transparency map.

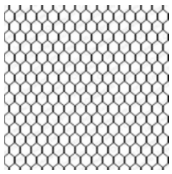
If one renders null at the default settings, the object will appear invisible. Therefore one would add one of the common properties to the material. The fig right Has common properties, Transparency set to 0.5 . (black = 0, white = 1). For a tinted plain look

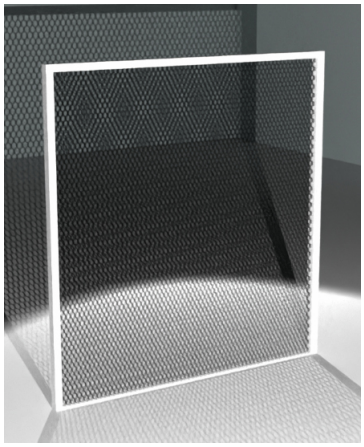


A transparency map can quickly be used with Null, or to mask out certain areas.

Open common properties, in the drop down box for transparency, choose Image map.

Remember the transparency works opposite to normal. Therefore, choose a map where white will be transparent (normally its black for transparent).





The fig right shows a simple mesh used for a screen. The white areas become transparent, and the black area become opaque.

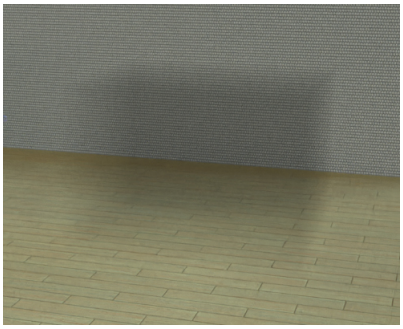
Although coloured images can be used, because this has a null depth, the results can look very flat. And not realistic.

Bump map.
Because Null does not have a surface, there is nothing to bump. So If you wish to use a bump map, try the Matte material or possibly Disney.

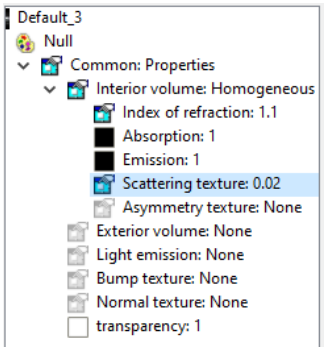
Null and the Mix Material
Instead of mixing a material with null to make it semi-transparent, the transparency parameter of the material should be used, because it is much faster.

Null and Volumes
The null material can also be used with volumes to create a "volume-only" material. This is because LuxCoreRender uses the regular material settings to define the surface of the material, and the medium settings to define how light is filtered as it moves through the material. Since null defines a transparent surface, the only thing that appears is the color

In a simple form, the volume is just particles inside a null shape.



As can be seen in the fig right. The volume has no sharp edges, normally associated with a box.

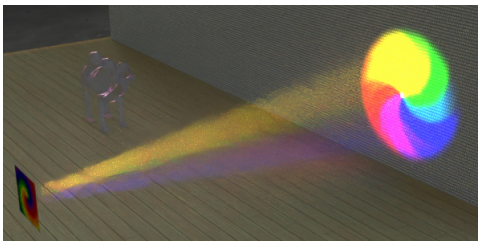


Null works very well for light rays. In this instance homogeneous volume is use, so one has access to the scattering settings, which clear volume does not have.

A null box, as shown in the image above, is put into the scene, for the light to pass through.

Absorption should be turned down, either to black, or towards black.

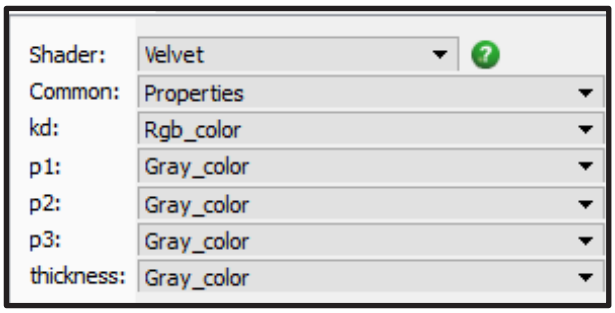
Scattering is turned down so the volume is not solid where the light passes. In this case, the scattering drop down box of homogeneous volume, was set to float. Then the float was set to 0.02.



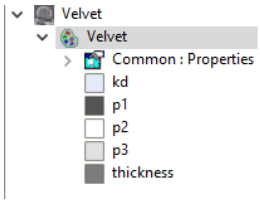
The colours of the projected image, changes the colour of the particles inside the null volume.

Velvet.

The Velvet material represents a fuzzy, velvety surface. It works well for velvet and other types of cloth. Its appearance can vary wildly with lighting conditions, so you may find it easier to use in a mix with another material such as glossy or cloth.



kd (diffuse) Default RGB_color color of the material's fuzz



Kd alters the base colour of the velvet material. The default being a blueish grey effect. However actual velvet comes in a wide variety of colours.

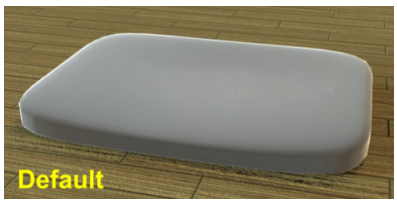
One would probably have to play with the p1 to p3 settings

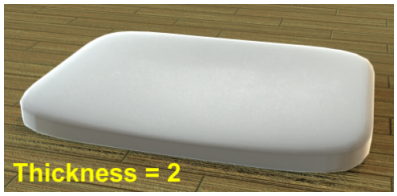
Velvet is one of those materials where it is just trial and error.

P1. Default Gray_color polynomial that influences the fuzz
P2. Default Gray_color polynomial that influences the fuzz
P3. Default Gray_color polynomial that influences the fuzz
These are three polynomials that define the appearance of the fuzz. They are rather abstract, so its best to just play around until you get a good result.

Thickness Default Gray_color height of the fuzz
This defines the thickness of the "fuzz" along the surface. Higher values will give a more pronounced fuzzy-look.

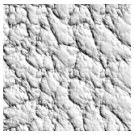
Increasing the thickness, can lighten the effect, and increases the edge brightness.

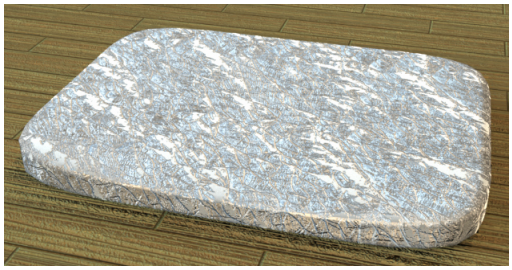




Bump map.

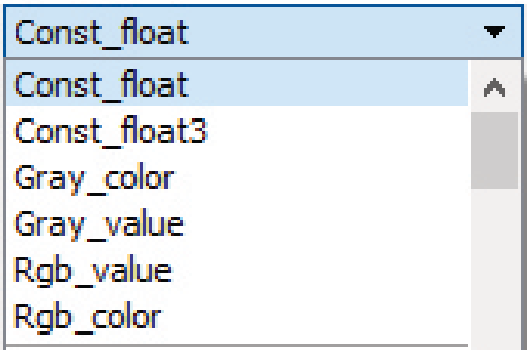
Whilst it is possible to apply a bump map via common properties. It can produce some unusual and unexpected results.





General material property input.

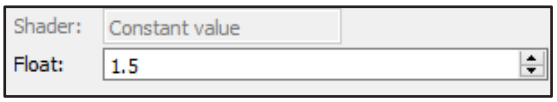
Values



Mostly the correct field will be set as the default. But many allow for a user to change it to ones liking.

A simple example is transparency. The default is often 'Grey value'. But Transparency accepts an image as a map or mask. Therefore using the drop down box, the user can change 'grey value' to 'image'.

These six inputs are use my many materials, where a field require a simple input.

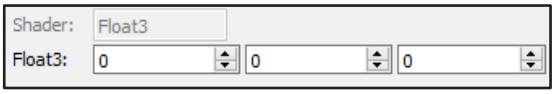


Constant value (float) Default 1

is a simple floating point number. And will normally allow for positive and negative numbers. Although it allows for large numbers to be entered, a particular property may ignore entries above a preset value.

Example, Transparency is between 0 and 1.

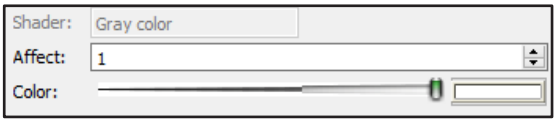
This can be used for many number inputs, Including IOR. Transmission, dispersion, etc.



Float 3. Default 0, 0. 0

Is a labelled by Luxcore as a constant RGB value. Each box represents a colour channel, Generally the colour value would be between 0 and 1.

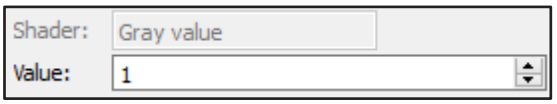
See colour chart, appendix 6 for examples.



Grey colour. Default 1.0

A single entry box with slider. Although there is a colour picker, the output will be restricted to grey scale, irrespective of what colour is chosen.

The **affect.**
Different functions depending on use. Can act like a tone control. i.e. After choosing a colour, reduce the affect towards zero to reduce the tone, (makes the colour darker).



Grey value Default 1

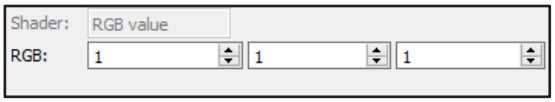
Similar to constant value. Some shader properties only recognise values of 0 and 1, even though greater values can be entered.

For example, if use for a grey diffuse colour (kd). Numbers 0 to 1 would be valid, numbers above 1 would be treated as white.

RGB Color. Default 1

Similar to Grey colour but acts on all the RGB colours. Therefore selecting the colour sets it to the correct colour.
As above, decreasing the affect darkens the colour.

RGB value Default 1, 1, 1



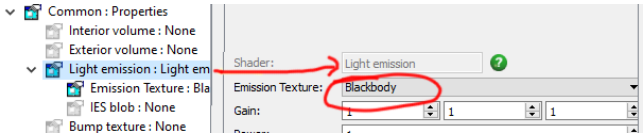
Usually input between 0 and 1. Higher numbers are often treated as 1.

As rgb values are normally 1 to 255. A rough equivalent would be ‘ value / 255 ‘ . If the red channel was 182, it would be 182 / 255 = 0.713725.

This is an approximation. See internet for correct calculation between RGB and sRGB used in luxcore.

See colour chart in appendix 6 for example colours

Emission Spectrum.

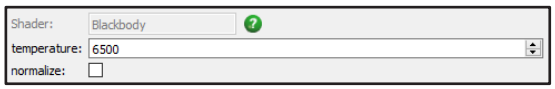


Blackbody.

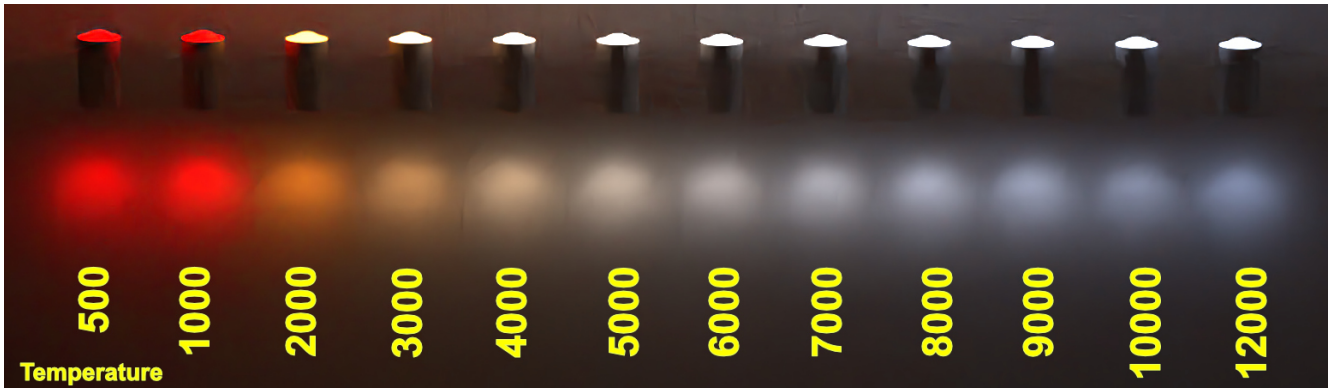
By indicating a color temperature, you can easier get the correct light color in your scene.
The target mood of a scene can be achieved by picking the right color temperature
Some general color temperatures for common light sources are: Candles (1850K). Incandescent (2700). Halogen (3000K). Fluorescent (4200). The sun (5000-6500K).

The blackbody property can be applied to the light emission in common properties.

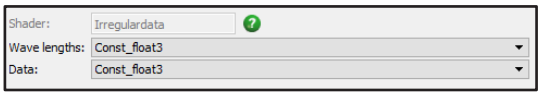
Temperature. Default - 6500
emission temperature



The images below show the temperature colour, Note. These only alter the colour, therefore power gain etc. still needs altering in the light emission settings.



Normalize (tickbox). Default false
Normalize the output. Haven’t found it any difference in TC yet.

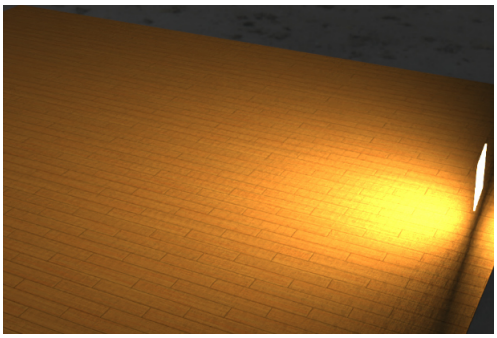


Irregulardata.

Wave lengths. Default - const_float3 580, 620, 660

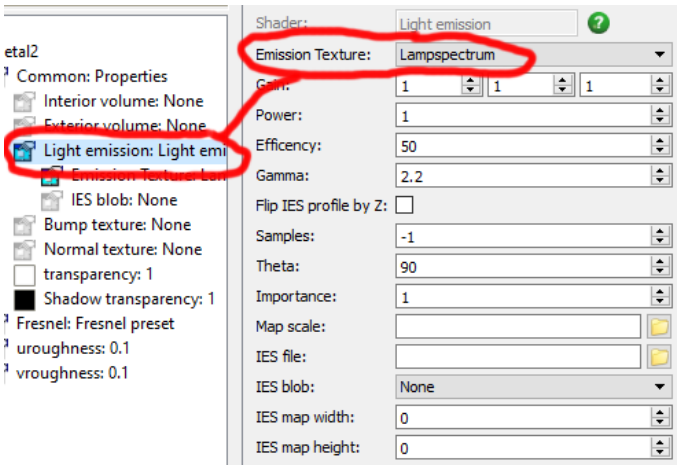
Data. Default - const_float3 0, 0.000015, 0

data (length should match wavelength's length)



Whilst it is possible to create lighting effects. At the time of writing, I cannot find much information as to how the wavelength and data interact.

Lampspectrum. Default - BlackBody 2900K

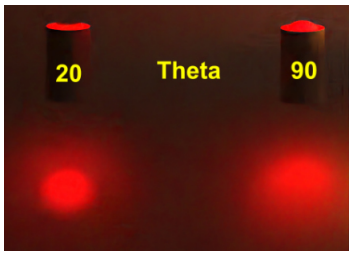


Lamp spectrum is a set of preset textures, that can influence the emission of an object.

Use the drop down box of the light emission to change the setting to Lampspectrum.

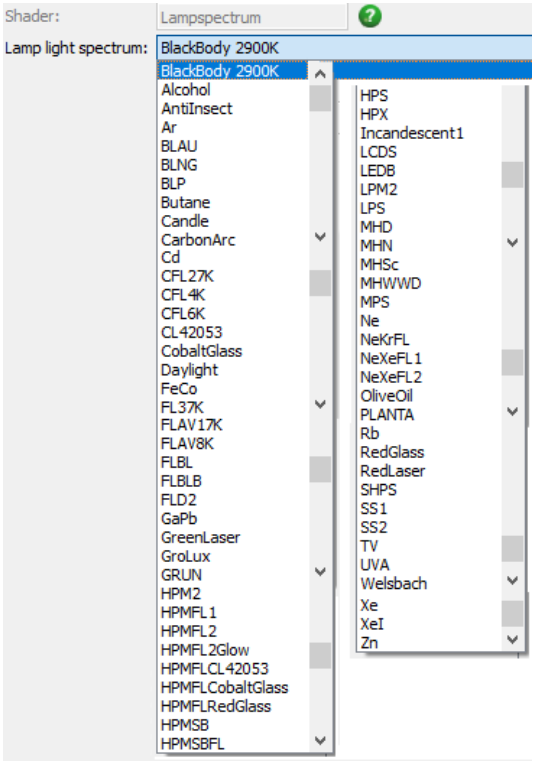
Lamp spectrum has quite a few options. Most are similar to Map point luminance. With the ability to use IES files. See Map point luminance

There are two main differences. A theta value, which affects the angle of the light emission.

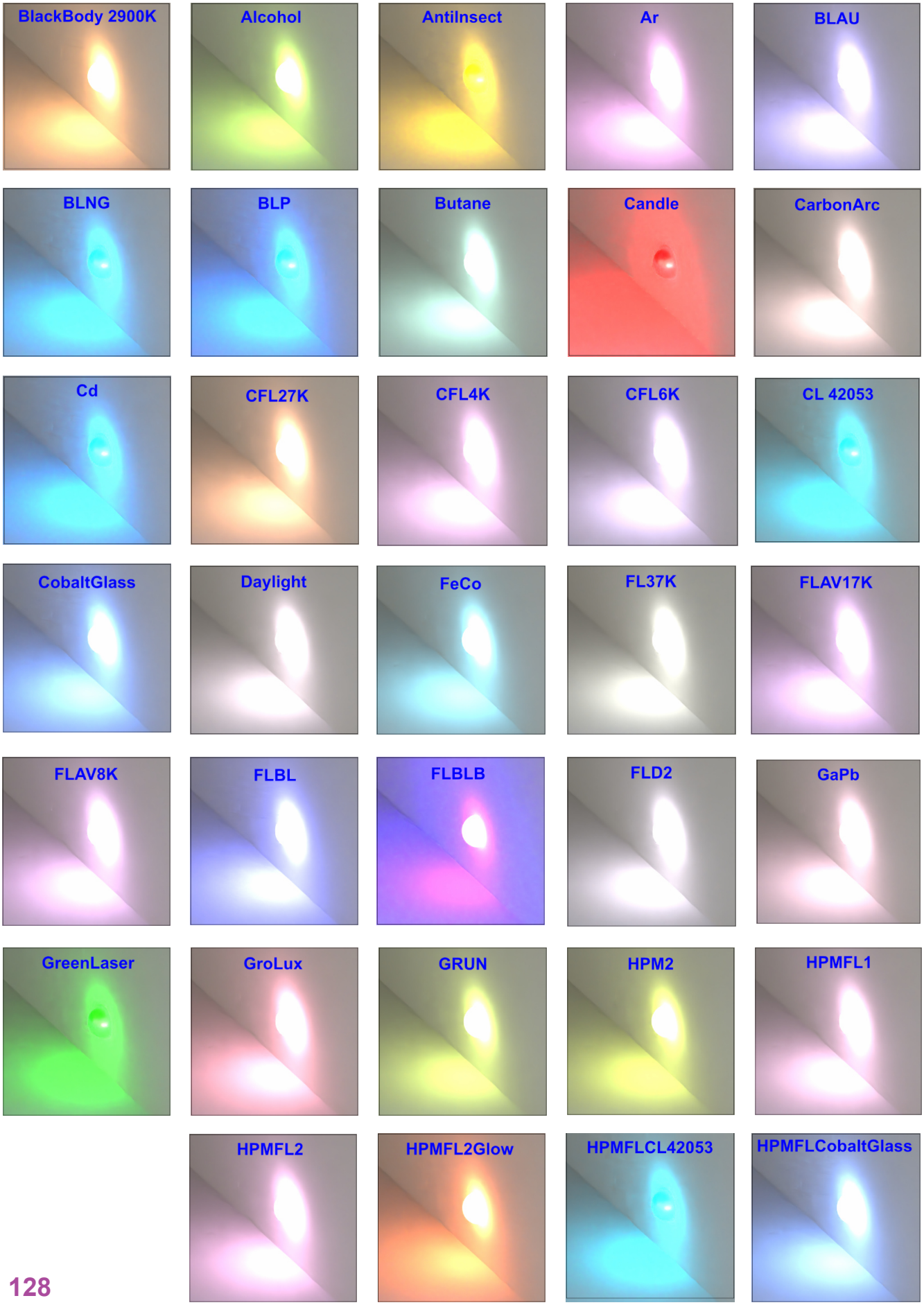


The second is Emission texture. These are a list of preset elements which change the colour temperature, and how the light shines.

Most are abbreviated, for example Xe = xenon. Zn = Zinc, Most take some google searching. Although I could have included a description against each. I cannot be certain, if the preset data of the light, follows what was found on google. So I don't want to include what may be false information.



All the following lights were rendered in the same file. So just the emission is affected when changing to a different preset.



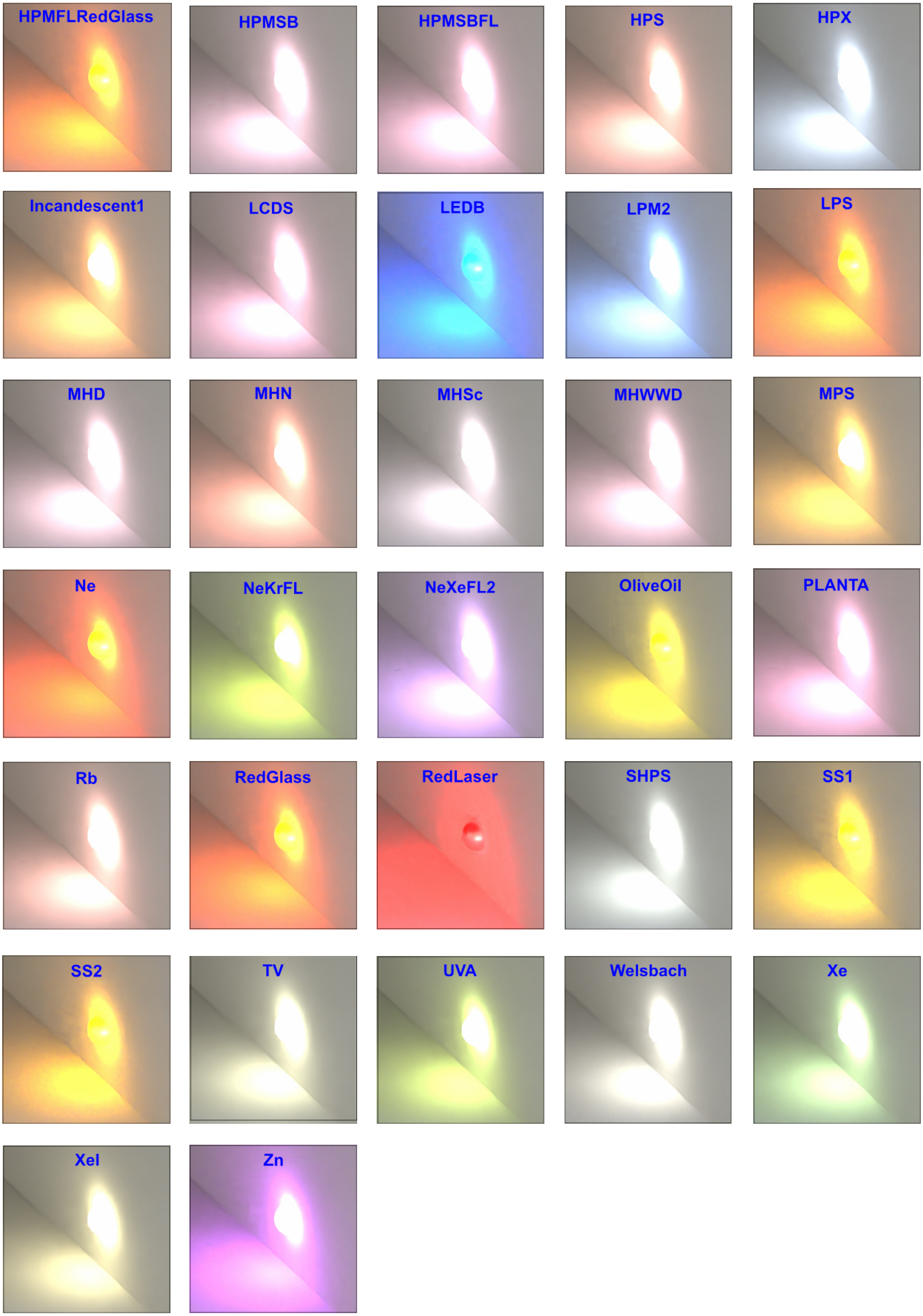
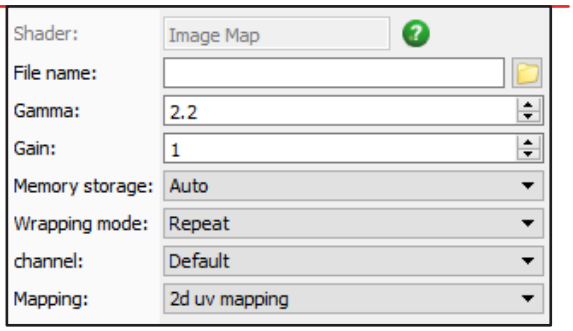

HPMFLRedGlass
HPMSB
HPMSBFL
HPS
HPX
Incandescent1
LCDS
LEDB
LPM2
LPS
MHD
MHN
MHSc
MHWWD
MPS
Ne
NeKrFL
NeXeFL2
OliveOil
PLANTA
Rb
RedGlass
RedLaser
SHPS
SS1
SS2
TV
UVA
Welsbach
Xe
XeI
Zn

Image mapping.

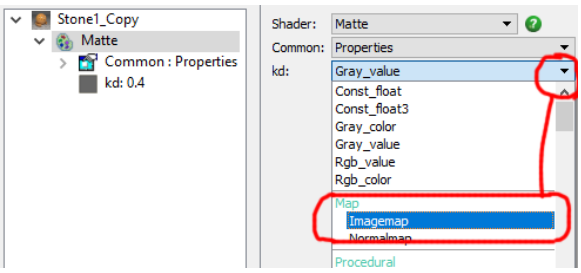


Commonly known as a texture. This applies a image wrap on a 3d object. The wrap can then be further enhanced by including a bump map, normal map, or transparency map (in common properties)..

Image wrap is in most render engines, In Lightworks it is wrapped image, In RedSDK its texture.

One can apply an imagemap to various items by changing the drop down box in a material property.

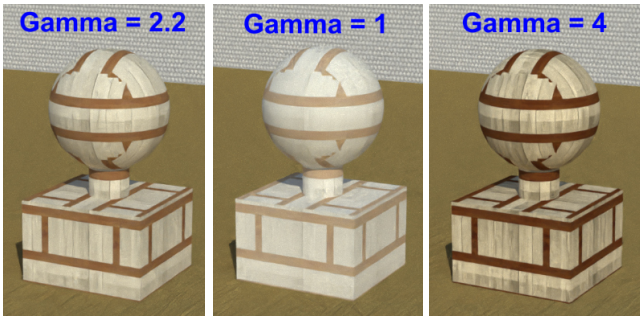
This example changes the matte kd drop down from rgb to image map.



File name. Default blank name of the image file
Image files can be bmp, jpg, tiff or png. Simply click the folder icon and navigate to the required file.

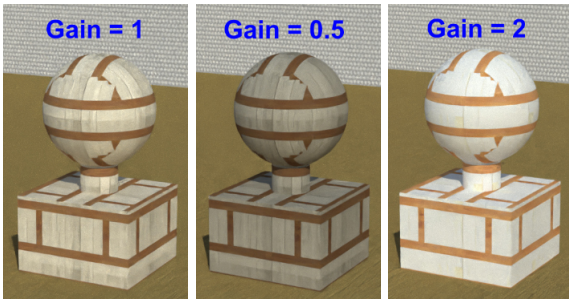
The required file can vary depending on circumstance, for example, bump maps are generally grey scale images, with high black to white contrast. Whereas kd texture wrapping would be full RGB colour images.

Gamma. Default 2.2 gamma correction value used in the image



A gamma of 2.2 is normal for bitmap images. HDRI's are often 1. Increasing the gamma, is similar to decreasing the brightness and contrast in the mid tones of an image. Decreasing the gamma, increases the brightness and contrast.

Gain. Default 1 image values are multiplied by gain value



Increases or decreases the brightness of each pixel in the image. Increasing gain brightens the image, decreasing the gain darkens the image.

Images can easliy be affected by the lighting in the scene, therefore it is preferable to set the scene correctly, before altering the gain. The gain can be used, if the lighting makes the wrapped object too bright or too dark.

Memory storage. Default auto select the type of memory storage
Auto (lux decides on best setting). Byte (8 bit). Half (16 bit). Float (32 bit).

Leave at auto, unless you have a specific need to try different bit rates. I have no idea how this affects TurboCAD.

Wrapping mode. Default Repeat select the type of image map wrapping mode

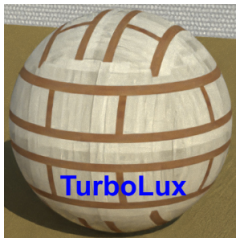
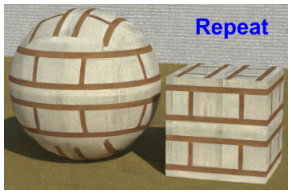
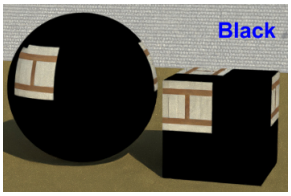


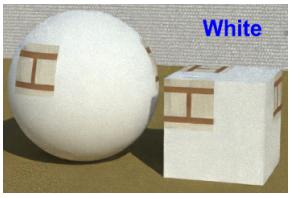
Image wrapping in TurboLux is, not the best in my opinion. It wraps in 2d plane or box mode, with no option to change this. So may not look as good on curves surfaces. It is labelled as 2d uv wrapping, but it probably uses uv face wrapping, not uv object wrapping. This may improve in time.

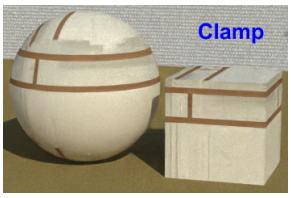
An alternative is to use TC's UV mapping tools, which produces a much better wrapping, at the expense of more fiddle to set up on some shapes.

There are four wrapping modes. These are Repeat, Black, White, Clamp. Though I cannot think of any reason, why one would want to use anything other than repeat.



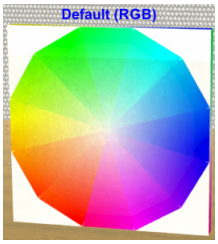


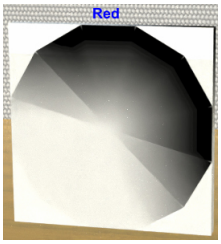


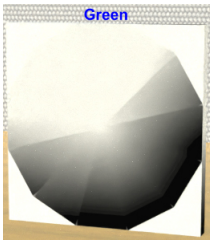


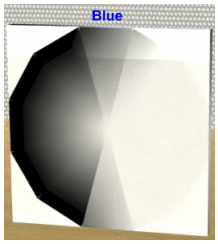
Channel. Default 'Default' extract from the image only the selected channels:
Default (RGB). Note - except for rgb, the following are turned into grey scale when extracting the channel. The seven channel options are
Red
Green
Blue
Alpha
Mean
Coloured mean
Rgb

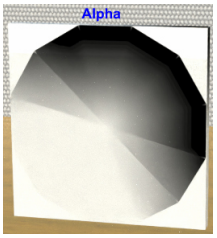
No idea why one would wish to extract a greyscale colour channel. But I assume its some specialised function.

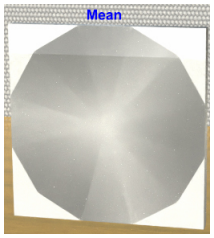


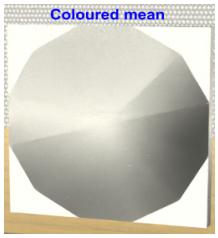












Mapping. Default 2d uv mapping

Allows access to 2d map settings. This allows one to set u anv v scale, rotation and offset.

Wrapping images onto an object. (the basics).

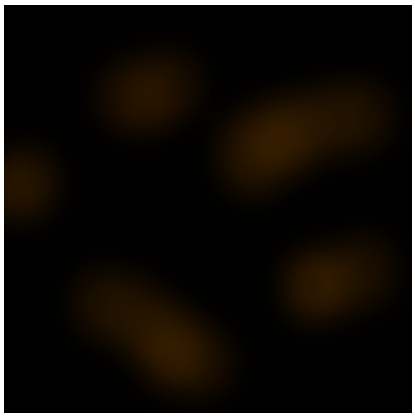
Obviously the first thing we need is an object. For this example, we will use adjustable smooth jaw spanner.

Rendering normally in Turbolux, will generally leave a plain finish. Even if we apply roughness, absorption etc.

So what if we want to make if a bit different.

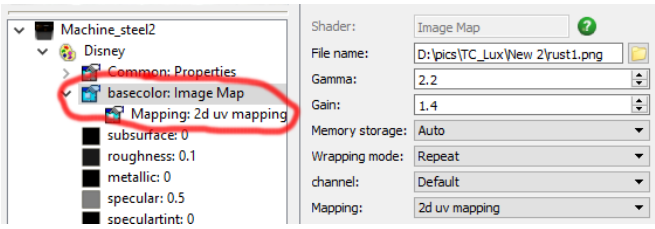
A simple way is to apply an image instead of a colour. The image to wrap, was created in a paint program, and is very simple for this test.

A page was filled with black, and then faint patches of brown added.

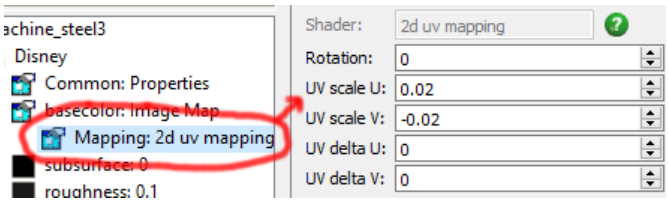


Here. The Disney material was assigned to the spanner. The drop down box next to kd, was changed from rgb, to Image map.

The file name, button was clicked, and navigated to the image required.

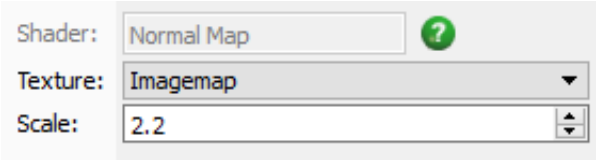


Lastly, in the 2d uv mapping page. We set the desired U and V scale.



Rendering the new wrapping. We can see the added brown patches are quite subtle. But gives the spanner a touch more of a. 'used' look. Rather than just being plain black

.Normal map



A normal map is a special type of bump map. Usually RGB in colour, whereas normal bump maps are often grey scale in colour. It uses a different algorithm than normal bump map.

Normal maps are often created in a dedicated program. But can be created using some paint programs.

This works well if the normal map is plugged into the bump map entry. Although there is a separate normal map entry, in common properties. At the time of writing, his doesn't work as good as using bump map - normal map.well,

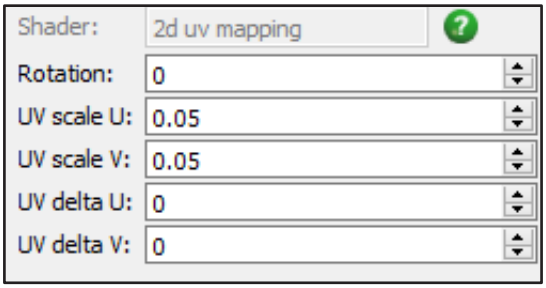
See the page on bump maps for further information.

A big difference to bump map is the inclusion of a scale box.

Scale. Default 2.2

Scale is basically the height of the map applied to the object.

2d UV mapping



2D mapping allows access to the 2d mapping scale and delta settings. These are necessary to tell the program how big you want to map the wrapping onto an object. And any offset needed for a correctly positioned wrap.

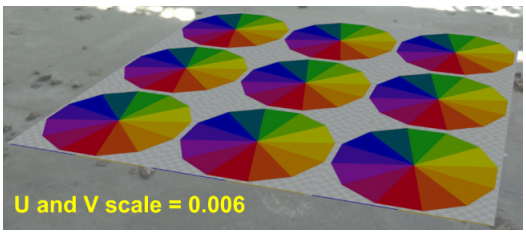
If using TurboCAD's UV wrapping tools instead of using standard wrapping, the scales may be 1 (or -1)

Rotation. Default 0

Many times one will want to orientate the image in a particular direction, Whilst one can simply alter the image in a paint program, this option allows T-Lux to change the orientation within TC. This allows a single image in multiple materials, and orientated a different way.

Type in the number of degrees in an anti-clockwise direction.

UV Scale U Default 10
UV scale V Default 10

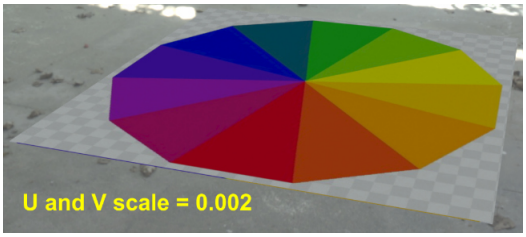


The number of times one wishes to wrap the image. This is an inverse scale. In that the smaller the number, the larger the image when wrapped.

This is wrapped as 1 unit divided by the object size in units.

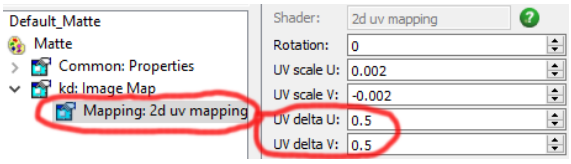
Example, we want the texture to wrap once on a 500mm square object.

Therefore, 1 wrap / 500 units = 0.002. So we type 0.002 into the scale u box and -0.002 in the scale v box.



See page 149 for more information on wrapping scales. Including wrapping to real life scales.

UV delta U Default 0
UV delta V Default 0

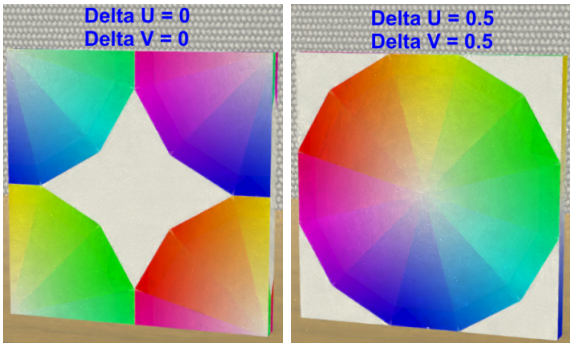


These are offset values,

Often an image may not wrap in the desired position on the object. Therefore we can use an offset (delta) value to 'shove' the texture into place.

These values are by texture, i.e. 1 = 1 texture, 0.5 = half a texture. Normally one would be using values between 0 and 1.

In the image right, the texture is half incorrect, in both U and V. Therefore we simply enter 0.5 in both U and V delta boxes to correct the wrap.

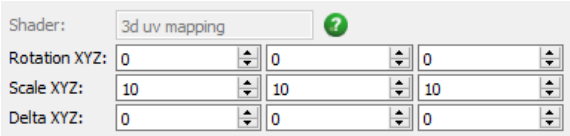


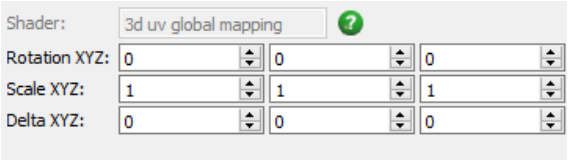
3d UV mapping.
3d global UV mapping.

3d mapping is only available with specific shaders. It is normally available in the shaders associated with Blender. (Brick, Dots etc).

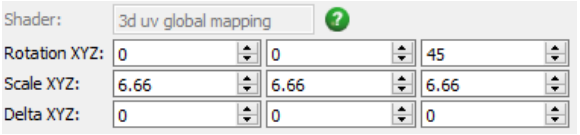
There are two options, 3d uv mapping. And 3d uv global mapping.

The difference is to fold. First how they calculate the u and v scale. And secondly the start location of the wrapping process. (see next page).





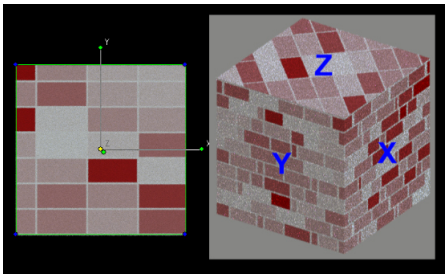
Rotation. Default 0, 0, 0



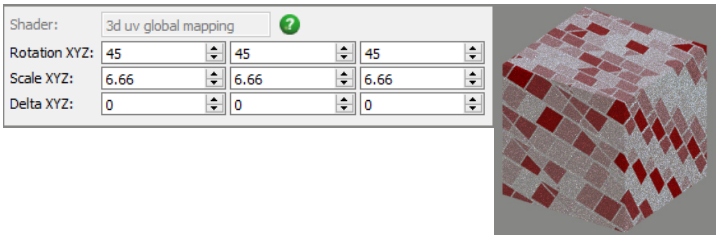
Rotates the texture around the object axes. . As this is 3d wrapping, there are three boxes, x y and z Axes

In the example far right. We rotate Z. And type 45 (degrees) in the z box.

This rotates the wrapping around the z axes of the object.



This can get unpredictable if we rotate around multiple axes.



Scale x,y,z Default 10, 10, 10

Like 2d mapping, the normal UV scale for a single wrap is taken as 1 divided by object size, however this differs when one chooses global wrapping. Which uses a different calculation.

3d uv mapping (same as 2d mapping). Calculates based on 1 wrap per drawing unit. So a 500 unit (mm) square would be 1 wrap divided by 500 units) = 0.002 scale.

3d uv global mapping. This appears to be related to the standard unit of measurement, i.e. the SI unit for metric is Metres. Therefore calculated as 1m
Example. If you have an object 500mm square. That is half a metre (0.5m). Therefore one would enter 1 m / 0.5 m = 2. So enter 2 and -2 in the u and v scale boxes.

If we have a 150 mm square. The two ways would be -

Uv map	1 unit / 150 mm	= 0.00666 and -0.00666
Uv global map.	1 m / 0.15 m	= 6.666 and -6.666

It is personal preference as to which one prefers to use.

The above assumes a single wrap of the texture on the object. Many textures

Delta x,y,z Default 0, 0, 0

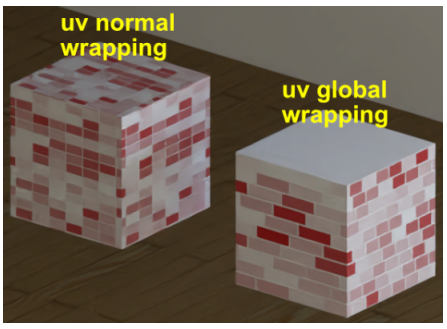
Delta values work the same as for 2d mapping, discussed earlier. However because 3d wrapping is often associated with procedural textures, rather than image textures. There are few occasions when one would need to move the texture, using the delta fields.

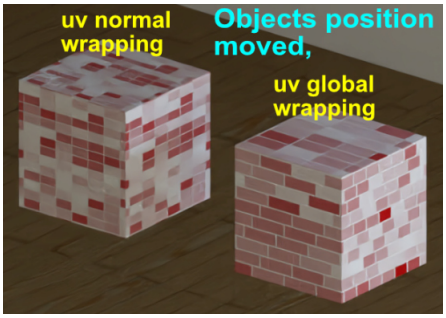
Wrapping origin. (second difference).

Uv wrapping uses object origin. This means if the object is moved, the wrap will always look the same.

Uv global wrapping. Appears to use global work plane origin. This means that if the object is moved, then the pattern will change.

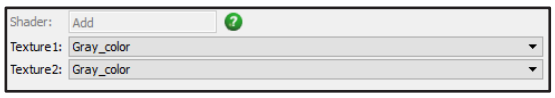
This is normally not a problem, unless one has deliberately oriented an object to show a specific pattern. IN which case best to use normal 3d UV wrapping mode, rather than global.



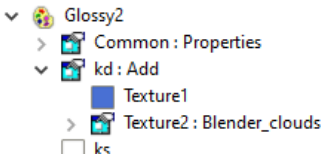


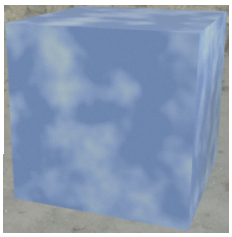
Operations and maths

Add. Default Texture1 - grey_color - white
Default Texture2 - grey_color - white



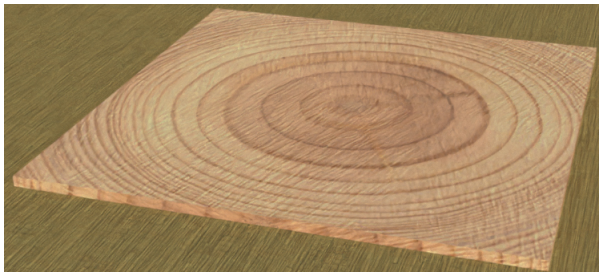
This can be used to add two values together. These values can be numbers, colours, or textures, or a combination. Adding two colours is a bit pointless. But a colour and a texture, or two textures can create some unusual effect.

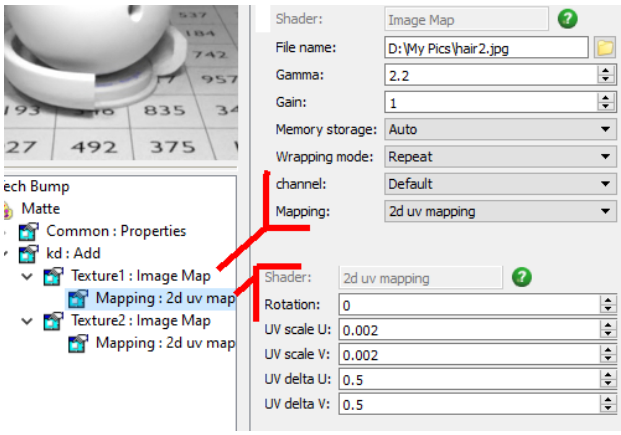


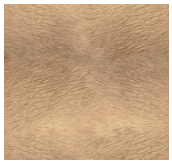


The image right, adds a blue texture to blender clouds. Unlike the mix property, the add property does not have a balance adjustment. Though one can still adjust the individual texture properties.

This last example adds two images. Say you have a wood end image that is too smooth. You wish it to show more of a cut surface. Adding a second image to represent the cut marks. Should show the desired surface on the object.



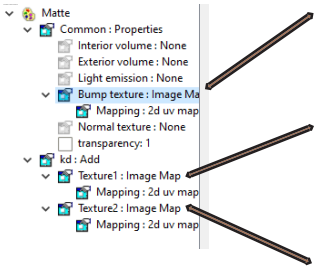


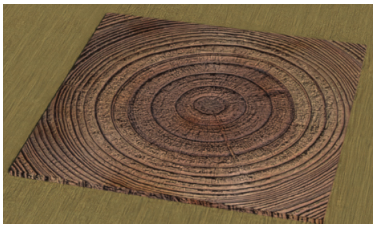


Unusual effects can be obtained by combining properties,

The image below uses a flat ring pattern for the main image, a patterned image for the subtraction. And a wood ring grey scale image for the bump map.

The patterned addition, is just to the change the colour slightly in certain places, Because wood is rarely uniform in colour.



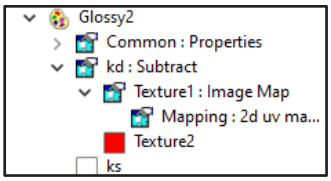


Subtract.

Default Texture1 - grey_color - white
Default Texture2 - grey_color - white

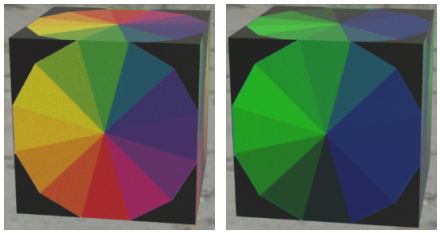
This subtracts the second value / texture from the first value / texture. Subtracting one colour from another, is not really necessary, but subtracting a colour from a texture, or two textures works well.

The setup right, subtracts the red channel from a 2d texture.



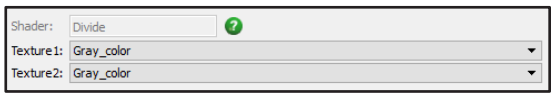
The first image shows the original wrapping without subtraction.

The second image is the finished subtraction. As can be seen, after subtracting the red channel, the result comprises of just the green and blue channels.



Divide.

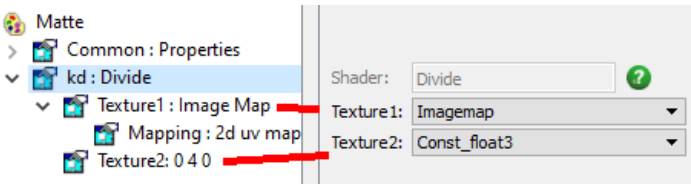
Default Texture1 - grey_color - white
Default Texture2 - grey_color - white

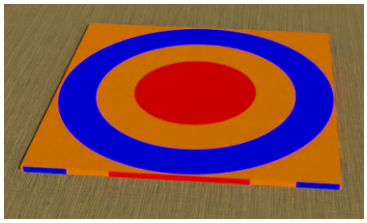


If the second entry/ texture is zero then zero is returned (cannot have a divide by zero).

Using float3 we can target a specific colour. Here we have Yellow (RGB 255,255,0). Blue (RGB 0,0,255). And Red (RGB 255,0,0). If we affect just the green channel, It should only change the yellow, as this is the only colour with green channel included.

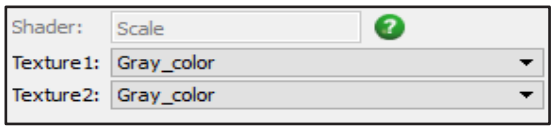
By using a setting of 0, 4, 0 in float3. The red and blues are unaffected, and the green 255 is divided. So the yellow parts become roughly (255,64,0)





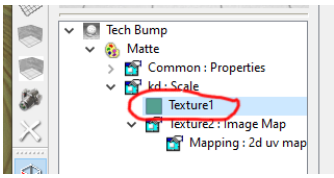
Scale. (multiply)

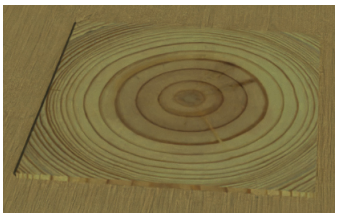
Default Texture1 - grey_color - white
Default Texture2 - grey_color - white



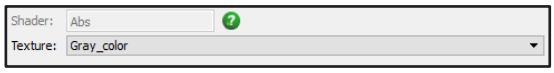
Output value: float or color.
The scale texture mixes textures by multiplying their color values, where black is zero and white is one.

Multiply can be used the change the colour of a texture.
For example, using a green colour to change the image, whilst retaining the wood end grain effect.



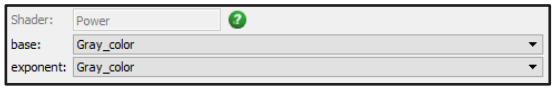


Abs. Default Texture1 - grey_color - white



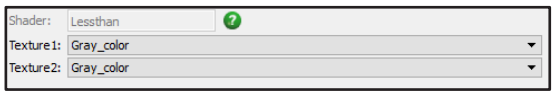
Ensures a value passed to a material is a positive number.

Power. Default Texture1 - grey_color - white
Default Texture2 - grey_color - white



pow(base, exponent)

Lessthan. Default Texture1 - grey_color - white
Default Texture2 - grey_color - white



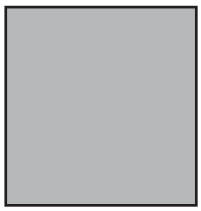
Returns 1 if texture1 < texture2, returns 0 if otherwise. Creates a black and white image, based on the pixel value difference, between the two images.

More of a specialised texture alteration. Take the grey scale equivalent of a texture (Texture 1). Set the comparison colour (texture 2). If the colour in texture 1 is less than the comparison colour, it gets turned black. If its greater it gets turned white. Example.

Image

grey scale

Comparison

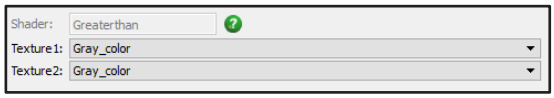


final

The yellow grey scale, is less then the comparison, therefore it is changed to black, The red and blue are both greater than the comparison, so are turned to white.

Greaterthan. Default Texture1 - grey_color - white
Default Texture2 - grey_color - white

Returns 1 if texture1 > texture2, 0 if otherwise. Creates a black and white image, based on the pixel value difference, between the two images.

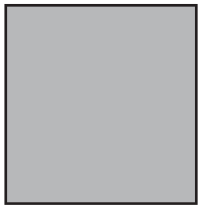


Opposite to Lessthan. Example

Image

grey scale

Comparison

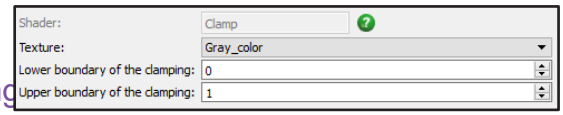


final

Blue and red are greater than comparison, therefore turned to black. Yellow is less and so turned to white

Clamp.

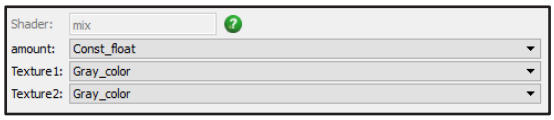
Default Texture - grey_color - white
Default Lower boundary of the clamping 0
Default Upper boundary of the clamping



Clamps the texture between two values. Probably a specialised function, which can eliminate high (1 - white), or low (0 - black) points/

Mix.

Default Amount - Const_float - 0.5
Default Texture1 - grey_color - white
Default Texture2 - grey_color - white



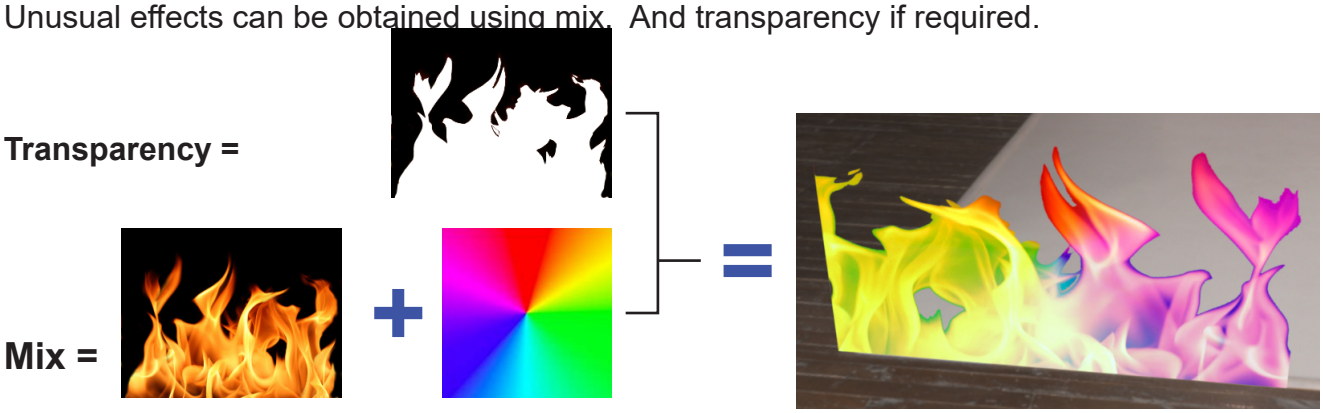
Mixes two texture and / or colours together similar to add, but allows for varying the amount of mix.

Amount Allows one to show more of the first texture, or more of the second texture. Lower settings show more of the first texture, higher numbers are biased towards second texture

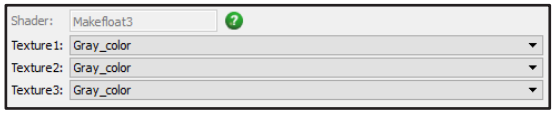
Texture 1, Texture 2

The drop down boxes can be changed to imagemap, allowing two images to be mixed together for special effects. Mix allows for one image to be used in multiple materials. And saves creating multiple images in a paint program.

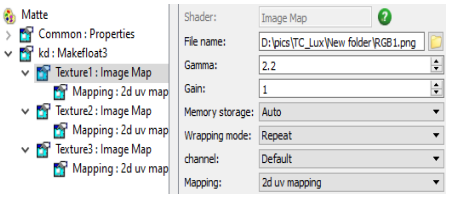
Unusual effects can be obtained using mix. And transparency if required.

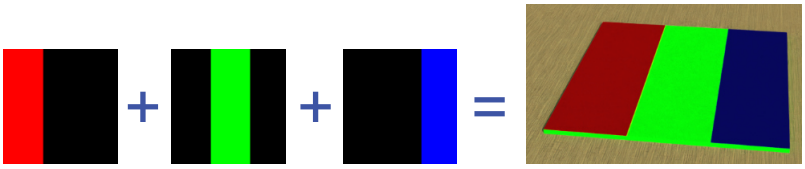


MakeFloat3. Default Texture1 - grey_color - white
Default Texture2 - grey_color - white
Default Texture2 - grey_color - white

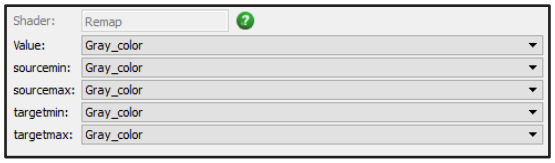


Combines three colours or textures into one float3 texture.
An example of texture combining is as follows. Black areas of a texture is treated as transparent for the purposes of adding textures.





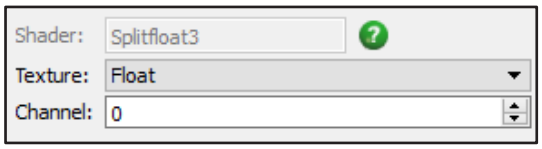
Remap. Default Value - grey_color - white
Default sourcemin - grey_color - white
Default sourcemax - grey_color - white
Default targetmin - grey_color - white
Default targetmax - grey_color - white



This texture brings the value from the source range into the target range.

Note: The value is clamped into source range before performing the remap operation, and the resulting number is then clamped into the target range.

Splitfoat3. Default Texture1 - float - 0
Default channel - 0



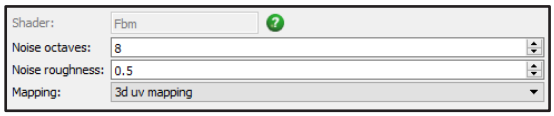
Extract a channel from a float3 texture. Can for example be used to get the R, G or B channel of a color texture, or to get the X, Y or Z components of a vector texture. The result is a black and white texture, With white being extracted and black the remaining colours. Valid channel is 0, 1 or 2)

In fig … below, we have an image texture with Red, Blue, and Yellow (consisting of red and green).
Channel = 0, this extracts the red, and as a consequence because yellow includes red and green, it is also extracted. Leaving just the blue circle as black (fig…)

Channel = 1, extracts the green channel. Therefore Yellow is the only colour containing green, so the red and blue show as black.

Channel = 2. Extracts blue. Resulting in red and yellow turning black. And the blue circle is white.

Noise

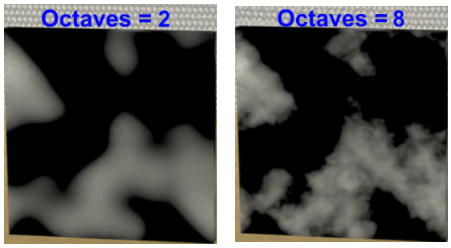


Fbm. (fractal Brownian motion).

Fbm texture generates fractal noise, which is very suitable for use as a bump map.

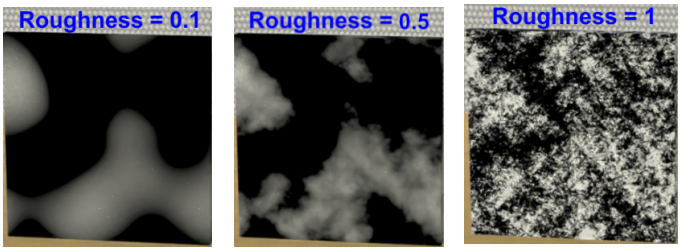
As with all wrappings, set the scale, in this case 3d mapping instead of 2d mapping

Noise octaves. Default 8
The range of noise frequencies - higher values create more detailed patterns.



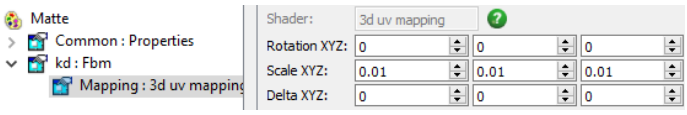
Noise roughness. Default 0.5
The noise roughness - low values generate smooth shapes, higher values add detail.

The images right, have the default octave of 8.



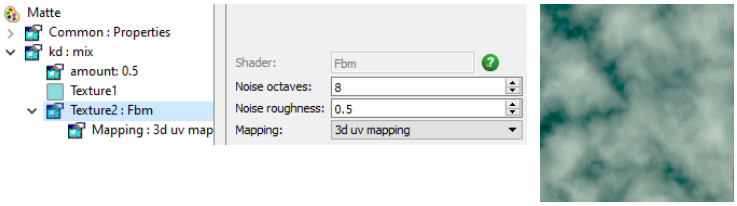
Mapping. Default 3d uv mapping

Allows selecting 3d mapping, or 3d global mapping. Which affects how the scale factor is applied. See page 134



Example.

The image on the right used the mix property to add the fbm map, and a blue kd diffuse colour.



Fbm Other.

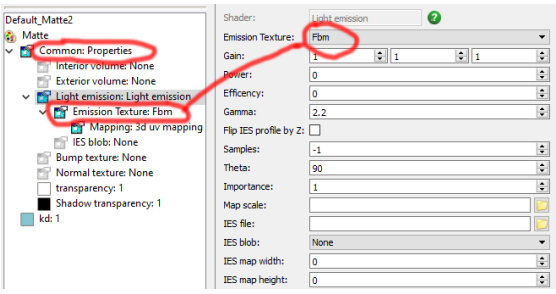
Fbm Noise can be used to create a variety of effects, using the material common properties.

Emission.

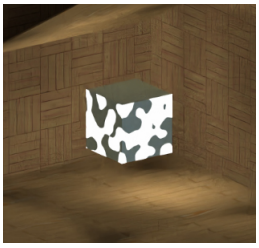
Emission can be applied to an object with fbm.

In the common properties - light emission properties, the texture is changed to fbm.

Set the power and efficency. Also set the uv scale as appropriate. As can be seen the light is produced by the fbm pattern.



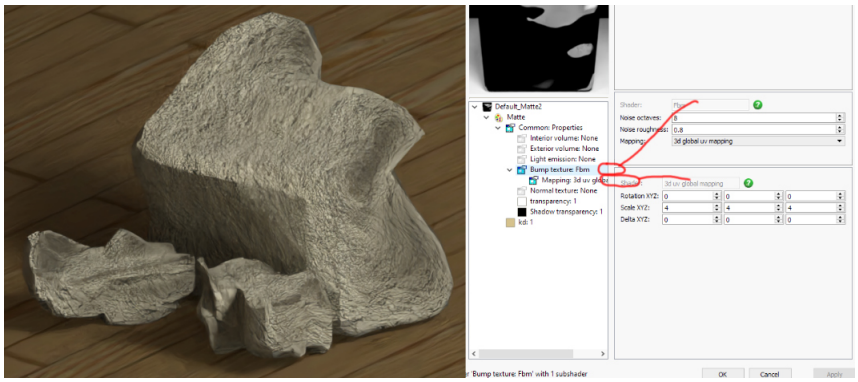
I have no idea why one would want to do this. But this shows that it is possible.



Bump map.

Fbm lends itself well to the use as a bump map. Not like a geometric pattern, but as a rough surface, like stone, rocks, and even clouds.

This simple example, uses an fbm bump map to roughen up an smesh (smesh is a surface object in TC Platinum). Adding a noise texture like fbm, is a simple way to add roughness, without having to hunt down an appropriate image on the internet.

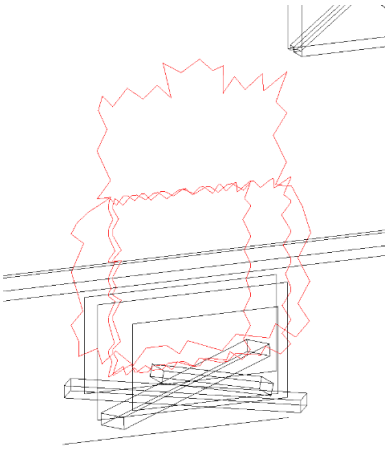


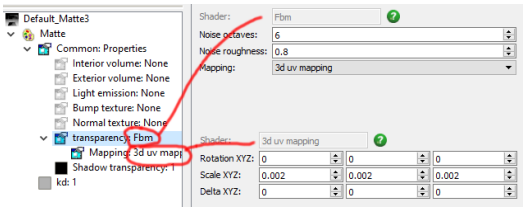
Transparency.

Fbm can be plugged into common properties transparency. Here the black areas will become transparent, and the white areas will take on the materials kd colour or texture.

The red objects in the far image right, are a flat 3d plane, onto which the matte material is applied. Three objects were created, all using the same material. This can give a better effect than using just one plane.

Common properties Transparency, is set to fbm. And the properties / scale adjusted as appropriate.



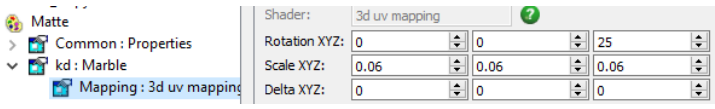


The fire is just a kd image with added black and white transparency map.

Marble.

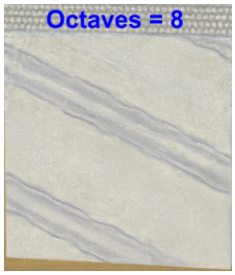
The marble texture generates a marble pattern by distorting a layer pattern.

As with all wrappings, set the scale, in this case 3d mapping instead of 2d mapping.



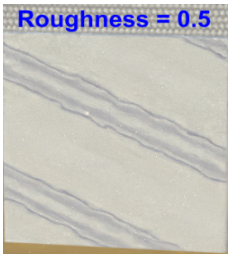
Noise Octaves. Default 8

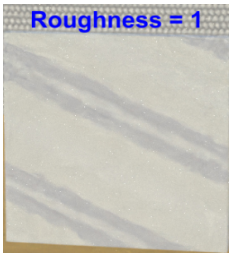
The range of noise frequencies.



Noise roughness. Default 0.5

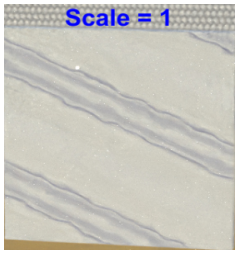
The amount of noise. Higher values, reduces the sharp edges of the effect.

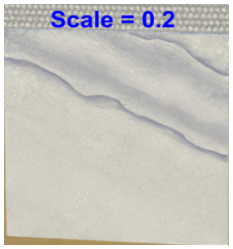


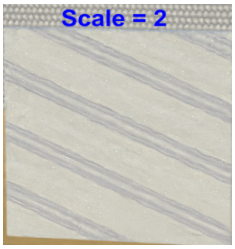


Scale. Default 1.0

The scale of the noise
High values is like zooming out to show more pattern.

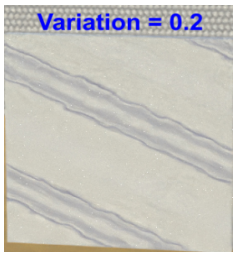


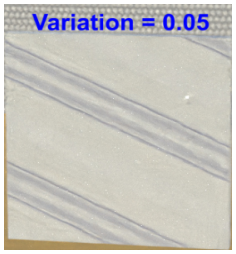


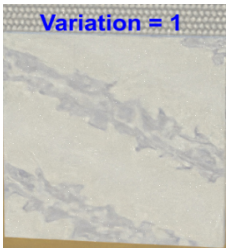


Variation. Default 0.2

Additional variation in the pattern by offsetting through the marble layers.







Mapping.

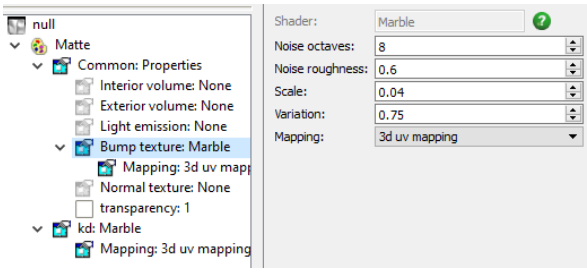
Both normal 3d mapping and 3D global mapping is available. 3D global is the simplest.

See page … for mapping details.

Bump map.

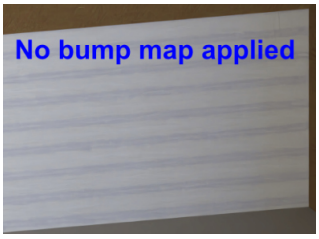
Marble can be used as a bump map. Though the results can be unpredictable. Unlike an image map which is predefined, with marble,it is a matter of rendering and seeing how it looks, then adjust properties accordingly.

Applying a bump map, is the same as other bump maps. Though there are all the marble settings to consider. All of which affect how the bump map os produced.



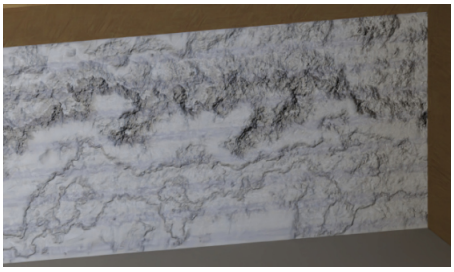
It is all these varied settings which makes the finished render unpredictable.

The render on the right, is the plain material without any bump map applied.



When the bump map is applied using the settings shown in the dialog box above. The image completely changes.

The marble effect can still be seem as the background pattern.

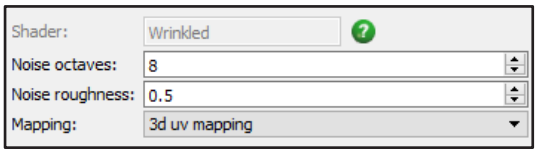


Transparency.

Marble does not really work well with transparency. Probably because it does not contain any black or dark areas, for the transparency to use.

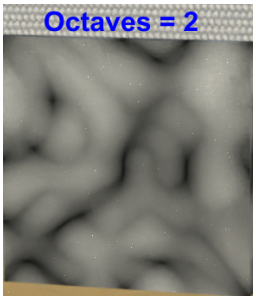
Wrinkled.

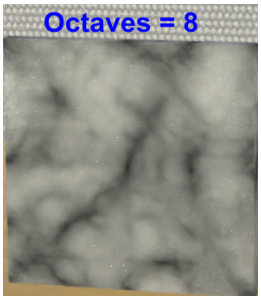
Output value: float only The wrinkled texture generates turbulence noise, which is very suitable for use as a bump map.



Noise Octaves. Default 8

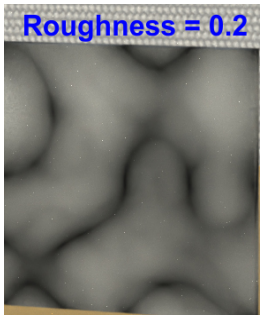
The range of noise frequencies. Higher octaves produces a sharper result.

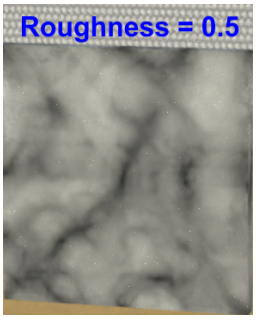




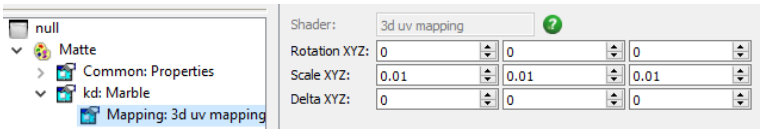
Noise roughness. Default 0.5

How smooth or sharp the noise is.





Mapping.



Both normal 3d mapping and 3D global mapping is available. 3D global is the simplest. See page .. For details of wrapping.

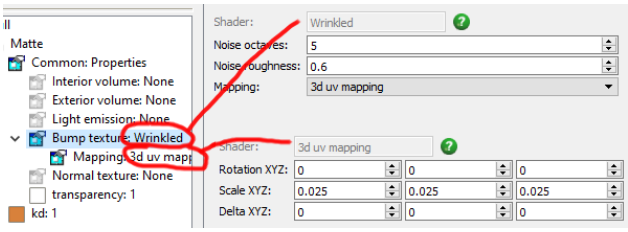
Wrinkled other.

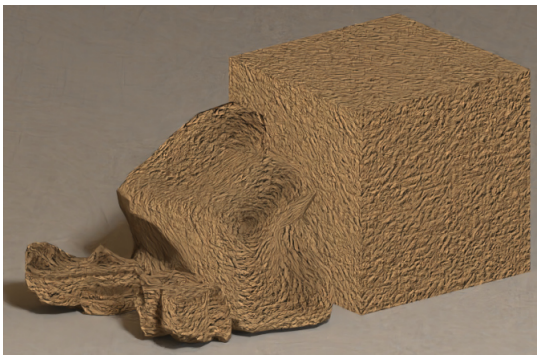
Bump map.

Wrinkled makes quite a good bump map.

Simply open common properties. Against bump map, drop down the box to wing wrinkled.

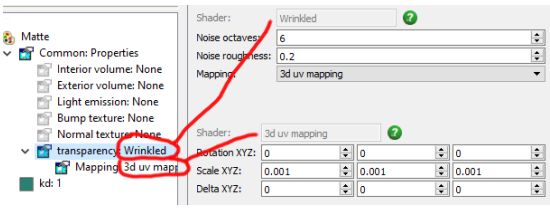
Set the properties and scale as appropriate.

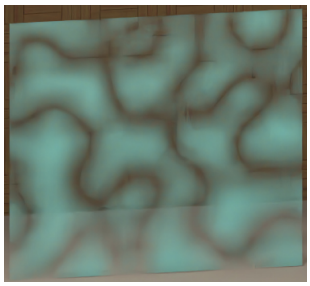




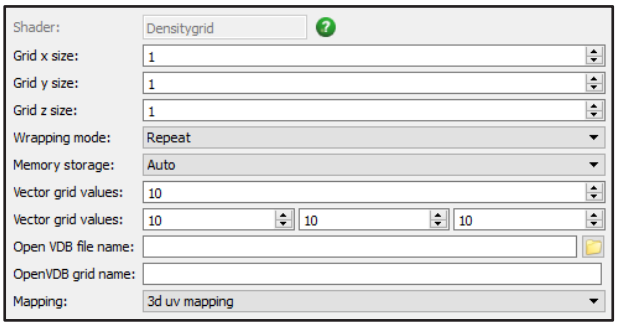
Wrinkled as transparency.

Wrinkled can be used with some transparency. But because wrinkled generally has thin black or dark areas, the results can be a bit disappointing.





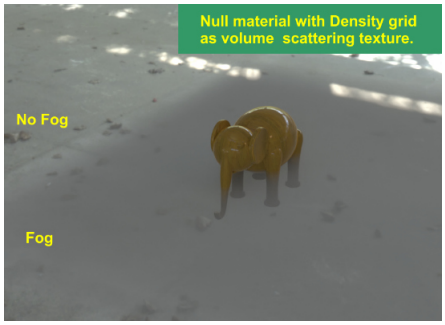
Density grid.



You can use ".data", ".data3" or ".openvdb.file" to define the 3D grid values.

OpenVDB is an open source software library for working with sparse volumetric data. It is designed for use with CGI. So probably has little use in CAD.

Can be used for ground fog. But, I have no idea about cgi. So cannot give any suggestions as to the use within TC.



grid x size. Default 1
grid y size. Default 1
grid z size. Default 1

Wrapping mode. Default Repeat

Sets the type of grid wrapping mode: "repeat", "black", "white", "clamp".

Memory storage. Default Auto

Sets the type of memory storage: "auto", "byte", "half", "float".

Vector grid values. Default 10

vector grid values (total count: nx * ny * nz).

Vector grid values. Default 10, 10, 10

vector grid values (total count: nx * ny * nz * 3).

Open VDB file name. Default “”

OpenVDB file name.

Open VDB grid name. Default “”

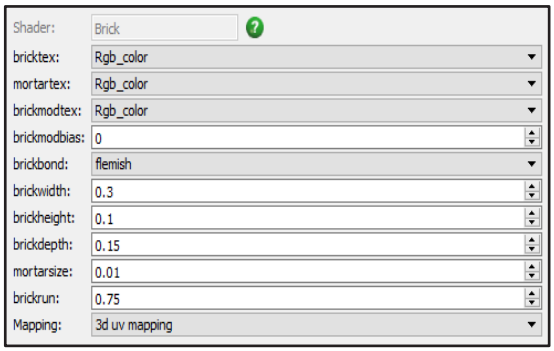
OpenVDB grid name to read from the file.

Mapping.

Both normal 3d mapping and 3D global mapping is available. 3D global is the simplest.

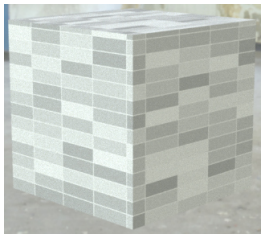
Procedural texture

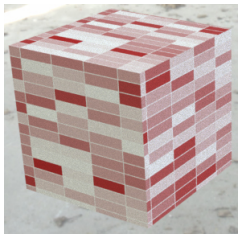
Brick.



The brick texture is similar to how LW works,it calculated the brick dimensions etc. instead of using an image.

Although it only has two colour selections, It can automatically add intermediate colours between the two values. This depends on the brickmodbias setting.





Brick uses a plain colour on each brick, and doesn't have any texture. Therefore although the brick colour is more random than using a brick image texture, it can look a bit plain.

bricktex Default RGB_color 128, 0, 0
The first brick colour, in RGB format

mortartex Default RGB_color White
Mortar colour in RGB format. Generally a grey colour, though some houses use red mortar.

brickmodtex Default RGB_color White
The secondary brick colour in RGB format.

brickmodbias Default 0

Sets the relationship between the first and second brick colour. This is a -1 to 1 number. A value of '0' is mid range. A value towards -1 is baised to the bricktex colour (first colour). A value towards 1 is biased at the brickmodtex (second colour).

brickbond Default "running"
The type of brick bond required. (drop down box). Flemish, English, herringbone, basket, chain, and link. NOTE AT the time of writing, these are not realistic brick runs.

At the time of writing. The ones in TC does not look anything like the ones in Luxcore - Blender.

brickwidth Default 0.3
Width of brick in metres. example UK brick = 0.215

brickheight Default 0.1
Height of brick in metres. example UK brick = 0.1025

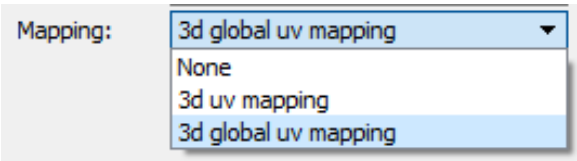
brickdepth Default 0.15
Depth of brick in metres. example UK brick = 0.065

mortarsize Default 0.01
size of mortar in metres. example UK brick = 0.01

brickrun Default 0.75

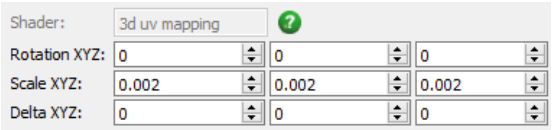
Depending on the value chosen, this can put a row of bricks, with all the same colour. Example, 3 rows staggered colours. 1 row all the same colour. 3 rows staggered.

mapping Default 3d UV mapping settings



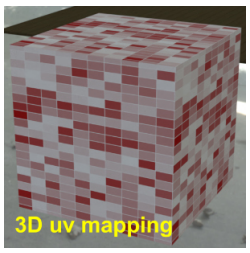
Brick uses 3d uv wrapping instead of the normal 2d. There are two 3d wrapping modes, 3D uv mapping, and 3d global uv mapping.

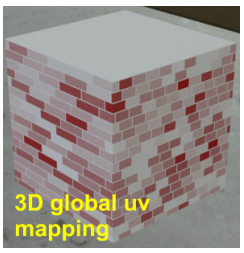
The plain 3d mapping doesn't look the best, and needs a scale factor. Some of the brick bonds don't work properly.



3D global uv, uses global scaling. So the scale factors can normally be left at 1. Also The brick bonds work correctly.

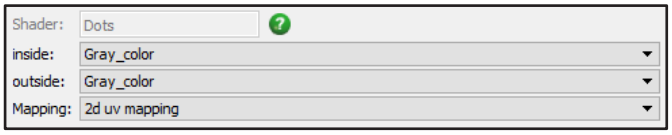
The above makes 3d global uv, a better choice.



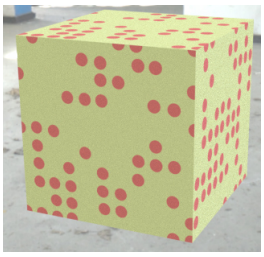


Other thoughts…. Bump maps. Unlike predictable image wrapping, This may not lend itself well to the normal bump maps. And doesn't have a roughness setting. But one can use a procedural function to add some roughness to the bricks. Something like Blender - noise, may give reasonable results.

Dots (procedural)



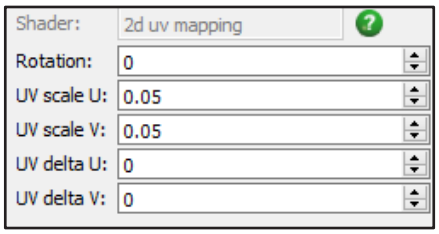
Simply puts a series of dots on an object. The colours of which, are controlled by the inner and outer colours.



Inside Default 'grey colour'
Sets the dot colour of the mapping. Although the setting is by default grey colour. The drop down box can be changed to RGB.

Outside Default 'grey colour'
Sets the colour of the background outside the dots.

Mapping Default '2d uv mapping'



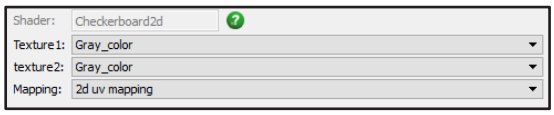
Sets the 2d UV mapping scale. i.e. how large the dots will be. This works the same as for image wrapping. Larger scale numbers = smaller dots. At a scale of 1, the dots can be so tiny, they cannot be seen.

One drawback is the way TurboCad wraps spherical objects. When Compared to some other rendering programs, it is certainly not the best . As can be see from the image on the right, it is wrapped onto six faces, even if the faces are spherical.

Because this is a procedural texture, one cannot simply create a 'tiled' image to compensate for the box type wrapping.

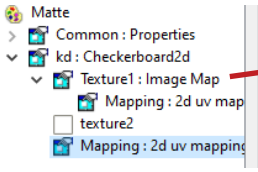
Checkerboard2d.

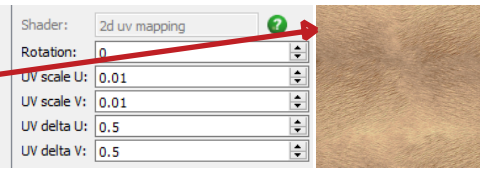
Default Texture1 - grey_color - 70, 70, 70
Default Texture2 - grey_color - 180, 180, 180
Default mapping - 2d uv mapping

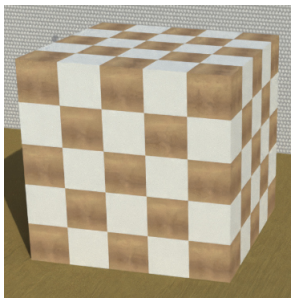


Creates a checker pattern. This also allows for colours and textures (images) to be used in the pattern.

One can use two colours, a colour and texture, or two textures. The example below uses a texture and a white colour for the squares.



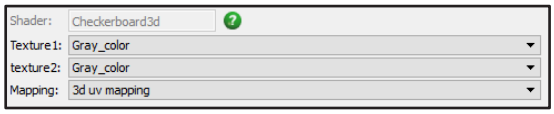


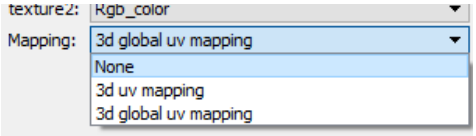


Checkerboard3d.

Default Texture1 - grey_color - 70, 70, 70
Default Texture2 - grey_color - 180, 180, 180
Default mapping - 2d uv mapping

Because of the way TC completes the wrappings, as planer wrap, not proper uv wraps, the z scale and delta may not work correctly on curved objects, or not at all.

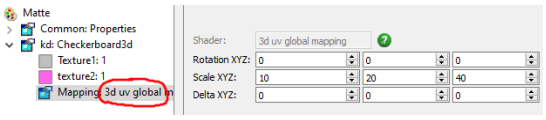




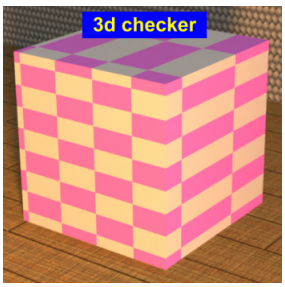
3d mapping has two options, 3d uv, and 3d global uv.

For ease, this example uses 3d global uv.

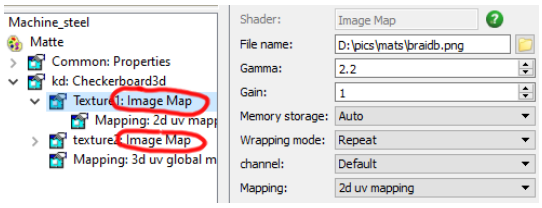
By using different figures in the x, y, and z, scale boxes. A different number of checker patterns can be obtained for each face.

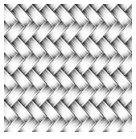


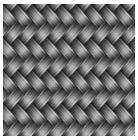
Note at the time of writing, this is planer wrapping. Therefore may not look correct on curved objects.

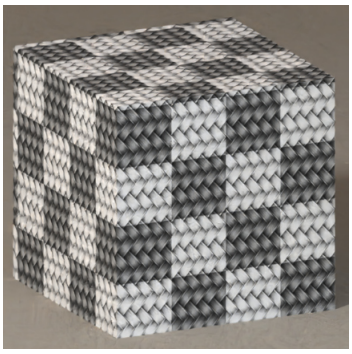


As stated two textures works well with this pattern.



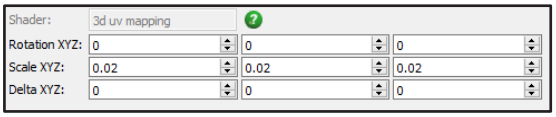




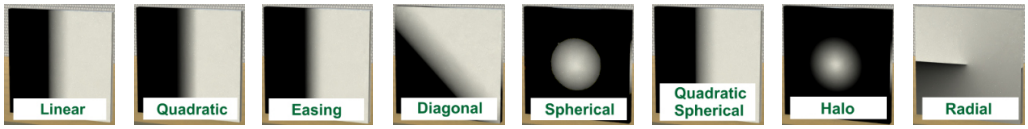


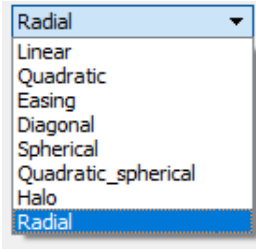
Blender maps Blender - Blend

From the blender manual. *The Blend texture generates a smoothly interpolated progression. This is one of the most frequently used procedural textures. You can use blend textures to blend other textures together (with Stencil), or to create nice effects (especially with the Mapping: Normal trick). Note. Remember that if you use a ramp to create a custom blending, you may have to use No RGB, if the Mapping value needs an intensity input.*



The blend function has a number of preset blend types.

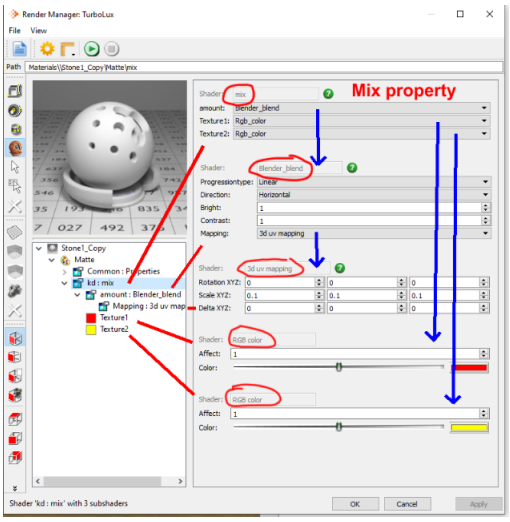




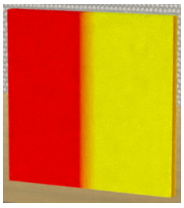
There are way too many variations on the use of the blend types. So here are just a couple.

This setup just blends red and yellow colours on a single object, Note, this blends two colours where they join, and is not the same as adding together etc.

The Mix property is used here, with the blend attached to the amount drop down, and the RGB colours attached to texture1 and texture 2.



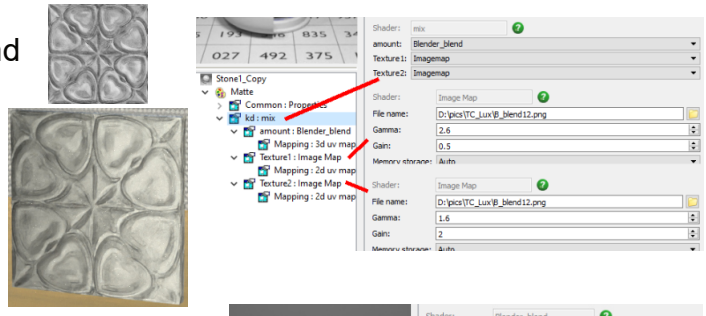
When rendered the result is shown here.



OK not very exciting, but it just shows what happens when two colours are linear blended.

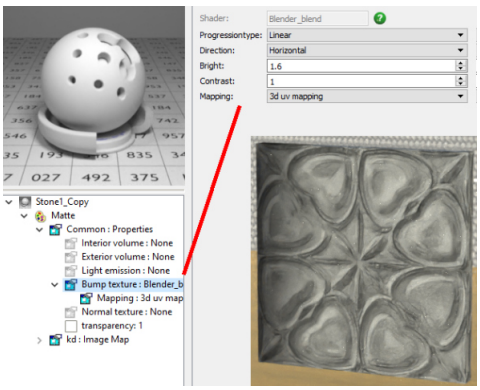
It is possible to blend textures, or even blend the same texture, using different settings.

This is not a lighting effect, the same texture was blended, with the right side slightly lighter than the left.



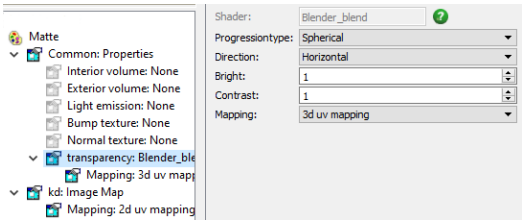
Using blend as a bump map can produce nice results. In this little example, the linear blend was applied as bump, and normal image used as kd.

The shadow effect is down to the bump. No lighting was altered. The drawback is, it may not alter if one changes the lighting power or location. Like a normal bump map may do.

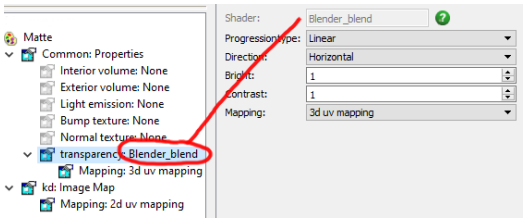


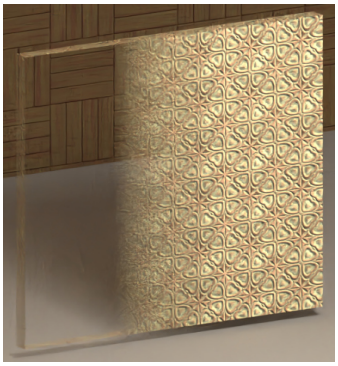
Common properties - transparency, can produce unusual effects.

This first one uses spherical blend. And, OK. It looks weird, but it is just done to demonstrate the effect.

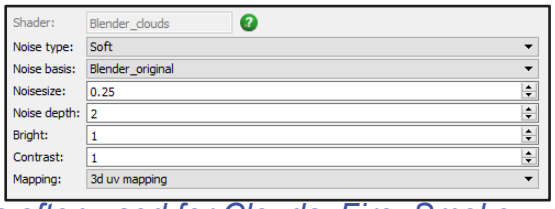


The second example uses linear blend with transparency, to hide part of the image.





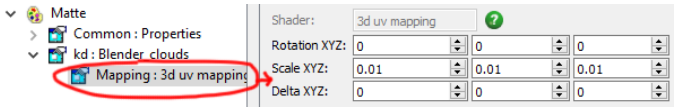
Clouds



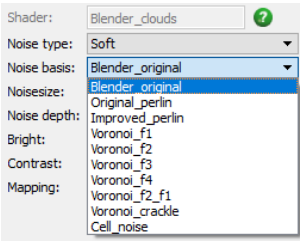
From the blender manual. *Clouds represent Perlin noise. In addition, each noise-based Blender texture (with the exception of Voronoi and simple noise) has a Noise Basis setting that allows the user to select which algorithm is used to generate the texture. This is often used for Clouds, Fire, Smoke. Well-suited to be used as a Bump map, giving an overall irregularity to the material.*

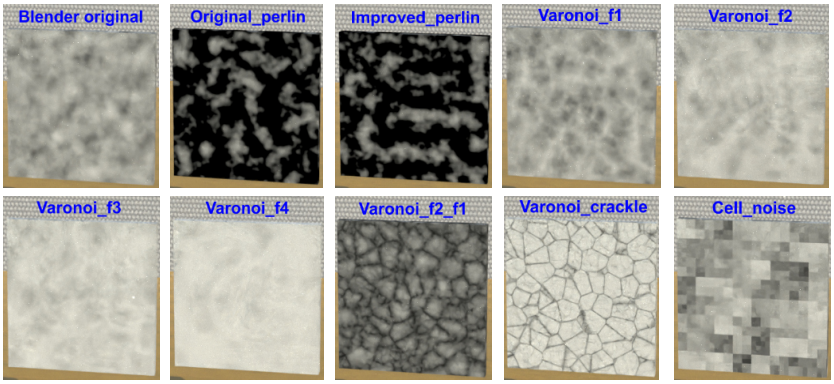
Although the clouds base textures are grey scale, they can be turned into sky and colours in a few ways. Turning into smoke, could be done using them as a transparency map.

As with all wrappings, set the scale, in this case 3d mapping instead of 2d mapping.



There are ten maps, with each being varied using size and depth. The ten options are.





Noise size.

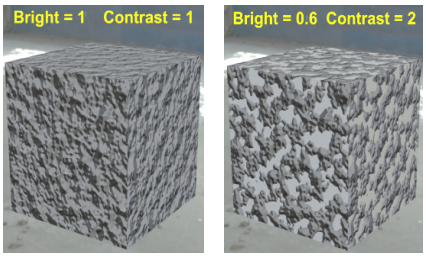
This works as standard scale.
larger numbers = larger noise.
smaller numbers = small noise.

Noise Depth.

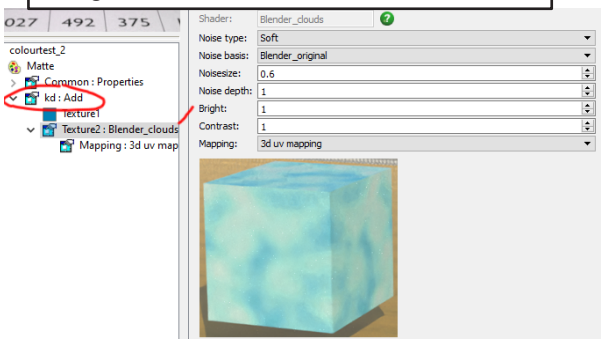
Integer numbers 1 to 25.
Higher number means a sharper noise..

Bright and Contract.

The same as an image program. These can make a big difference, when used as a bump map.

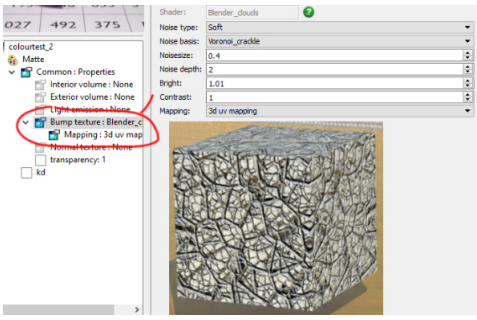


Using cloud with 'Add'. To colour the effect



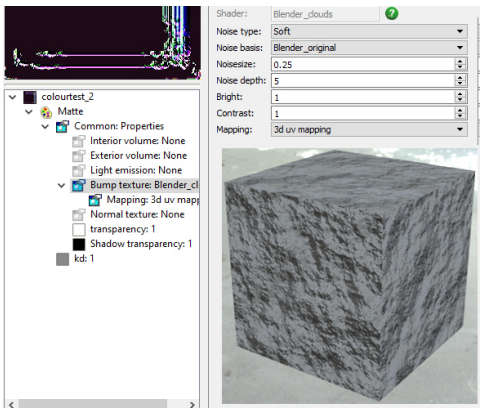
Various commands can be used with noise. On the left is a simple Add. Where the cloud is added to a blue colour.

Using cloud as a bump map



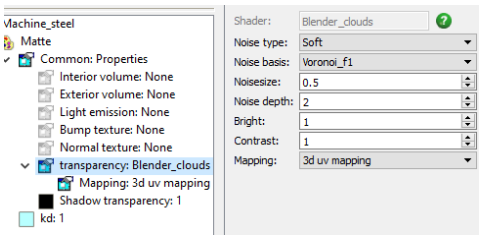
Bump map.

Although other noise textures may be the obvious choice for bump maps. Cloud can also be used to good effect.

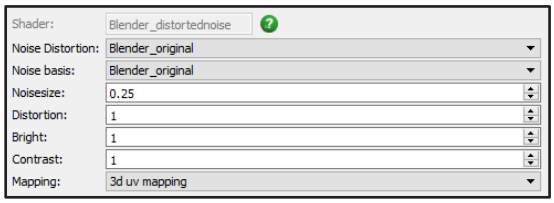


Transparency.

Cloud will work with transparency. As it can have good definition between light and dark.

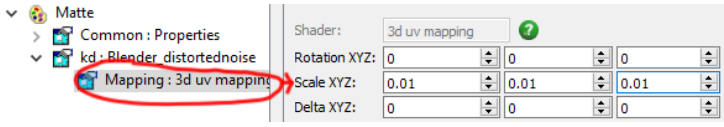


Distorted Noise

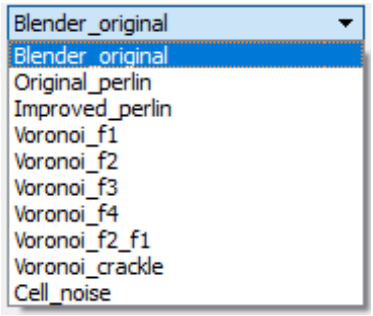


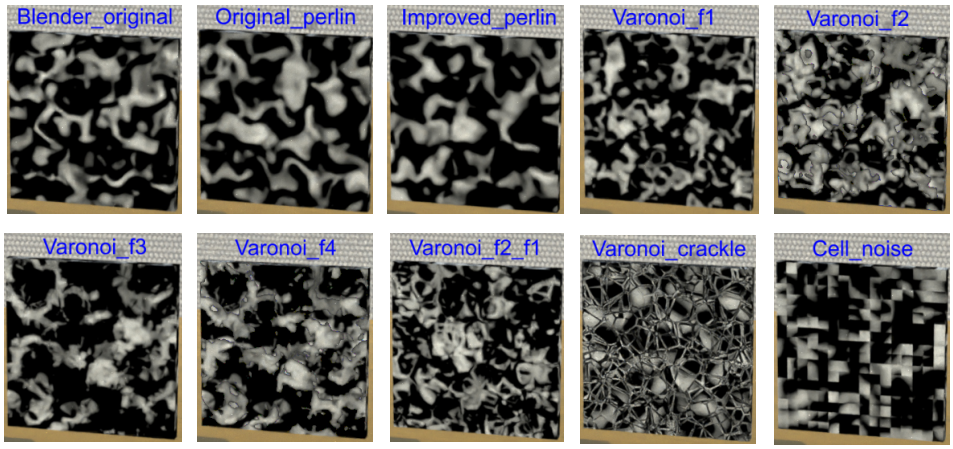
From the blender manual. *Distortion Noise takes the option that you pick from Noise Basis and filters it, to create hybrid pattern. It is often used for grunge but is also very complex and versatile.*

As with all wrappings, set the scale, in this case 3d mapping instead of 2d mapping.



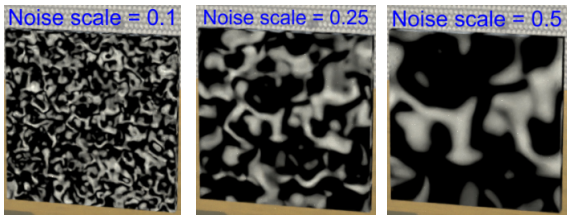
I have no idea why there is bland distortion and blend basis are doing the same job.
Like other noise properties there are ten options. Most, apart from crackle and cell, are just variations of the same theme.





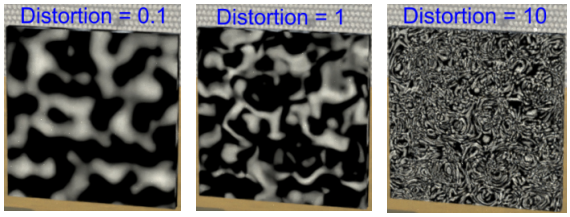
Noise size.

This works as standard scale.
larger numbers = larger noise.
smaller numbers = small noise.



Distortion.

Seems similar to Noise scale, but the opposite way.
larger numbers = small noise.
smaller numbers = larger noise.

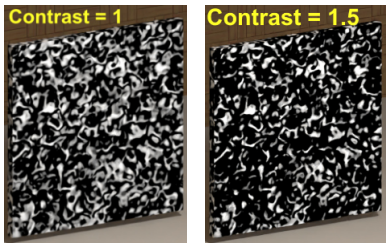


Brightness

The same as an image program. These can make a big difference, when used as a bump map.

Contract.

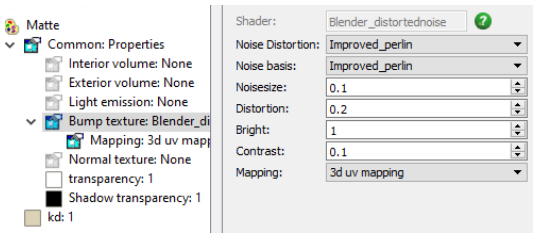
The same as an image program. These can make a big difference, when used as a bump map.

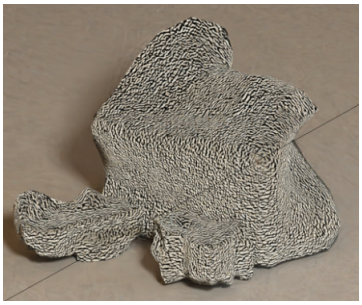


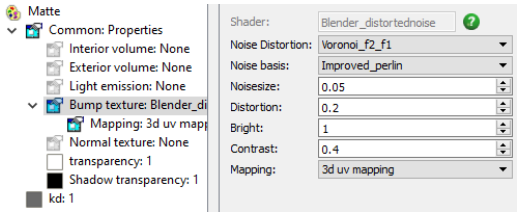
Using as a bump map.

Because of the variation between black and white. It may seem like the perfect map for using bump. However, in practice the results can be less than satisfactory, at least without lots of experimentation.

That said. It will still work with bump maps. But one may have to do a bit more fiddling than normal.

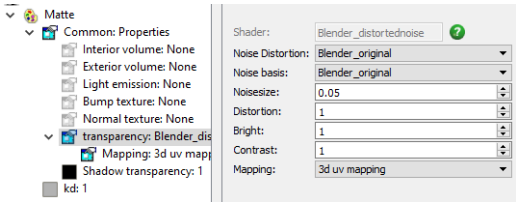


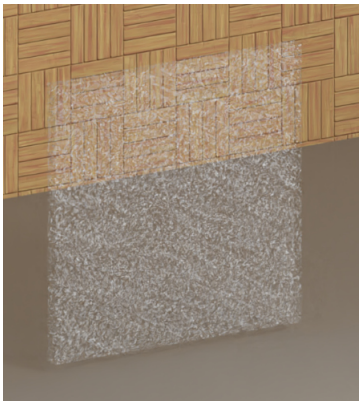




Transparency.

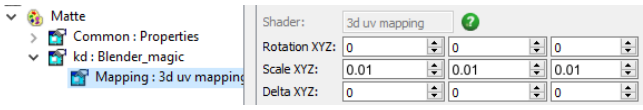
Distorted noise can be added as a transparency map. The effect can fiffer quite a lot, depending on the settings



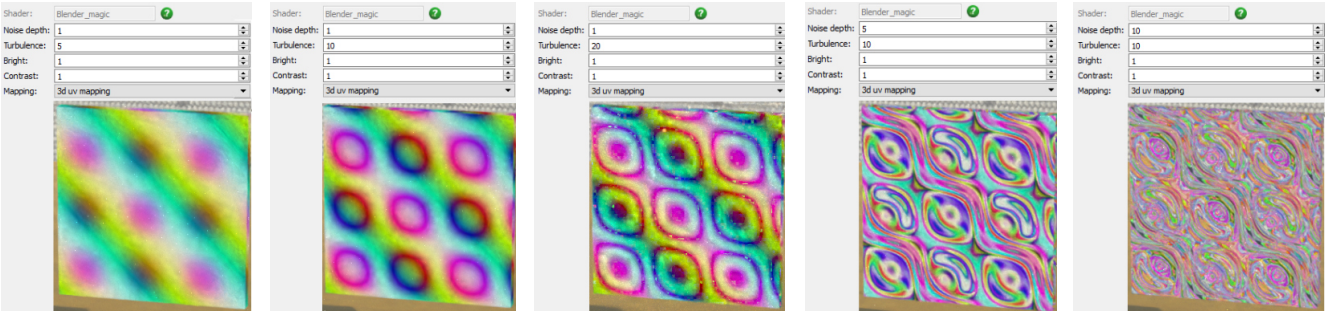


Magic

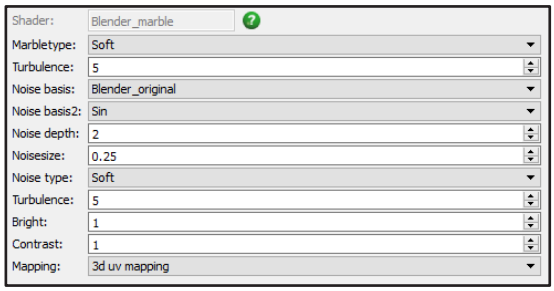
From the blender manual. *The magic texture is not frequently used. It can be used for "Thin Film Interference", if you set Mapping to Reflection and use a relatively high Turbulence. The RGB components are generated independently with a sine formula.*



This one is just weird.



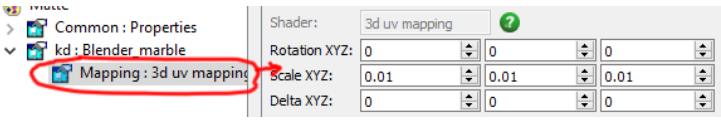
Marble



From the blender manual. *The marble texture is used to generate marble, fire, or noise with a structure. Bands are generated based on the sine, saw, or triangular formula and noise turbulence.*

Another odd one, billed as marble, It doesn't really look like marble But plugging it in as a bump map can really be effective.

As with all wrappings, set the scale, in this case 3d mapping instead of 2d mapping.

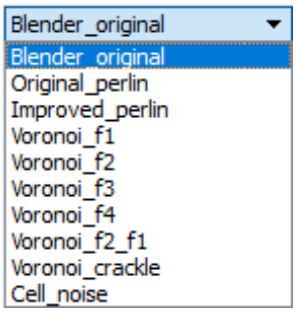


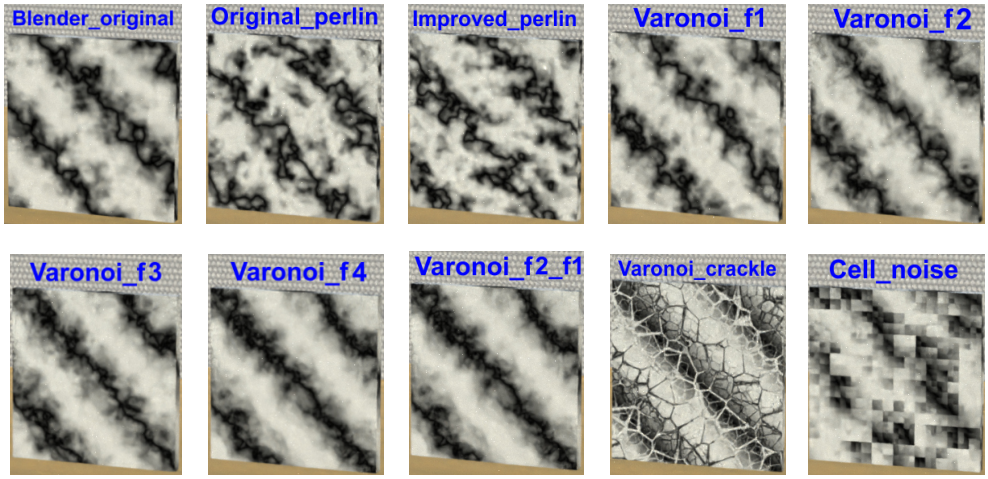
Turbulence.

Doesn't appear to do much on my computer.

Noise basis.

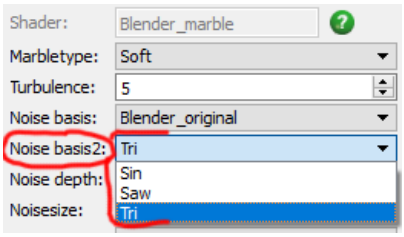
Noise basis, is the same as most blender maps. One can choose from 10 different effects / calculations.

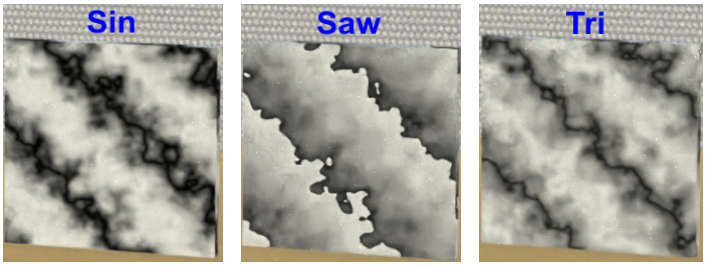




Marble - Noise basis2

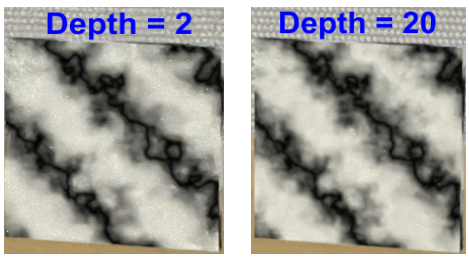
Can be switched between 3 different algorithms for calculating the effect. Sin. Saw and Tri. Sin and Tri, give the normal striped pattern. Saw can be much sharper, and similar to a wave.





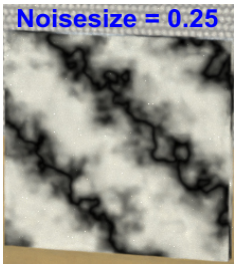
Noise depth.

Appears to just change the contrast slightly. This can have an effect on bump maps. Range 0 to 25.

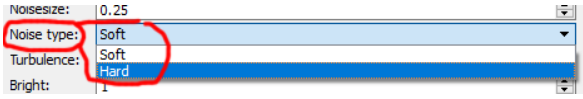


Noise Size.

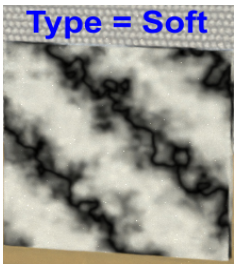
Blurs the relationship between light and dark. Small differences in size can make quite a difference. Especially when used as a bump map.



Noise Type.



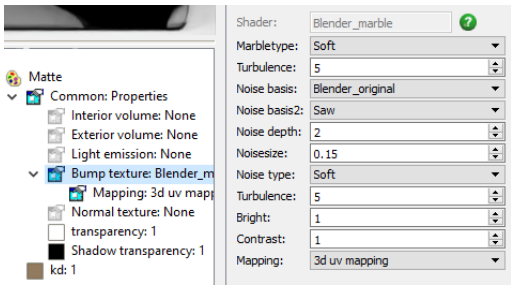
Similar to altering the sharpness of a bit map image.

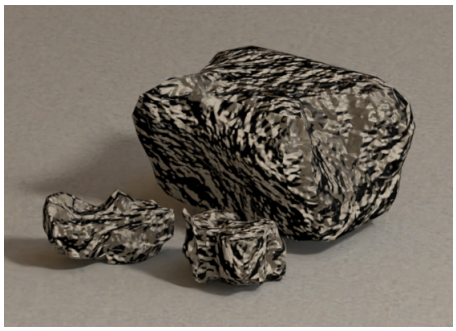


Bright and Contract.

The same as an image program. These can make a big difference, when used as a bump map.

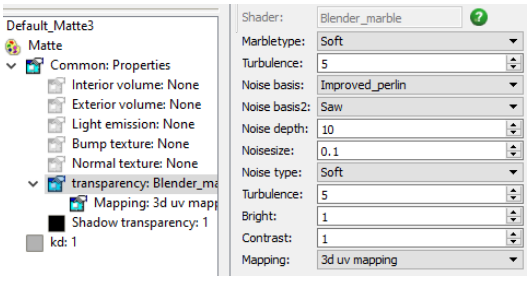
Bump map. It acts the same as other blander noise maps.

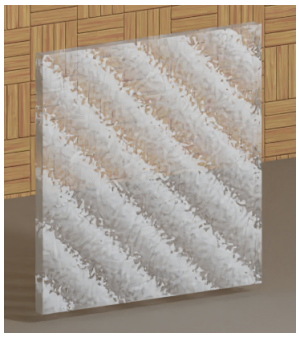




Transparency. (common properties).

Transparency can use the blender marble map. But still results in a striped effect.

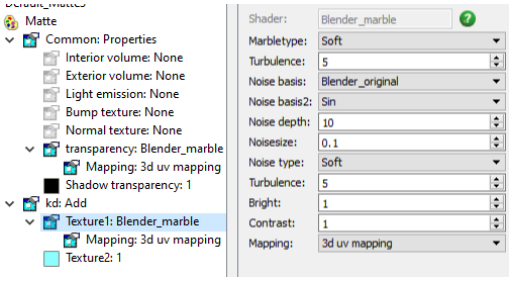


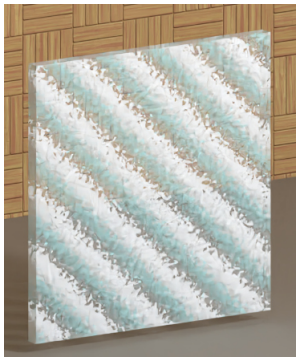


Other.

The effect can also be used with add / subtract etc. The images right, use kd (diffuse) as the source for the add commend.

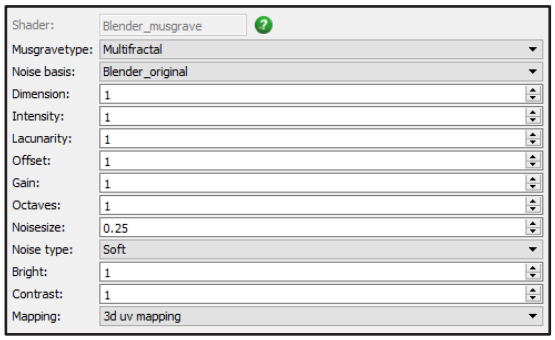
This changes a portion of the colour used in the map.



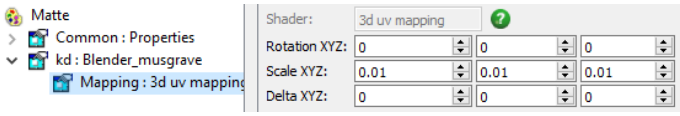


Musgrave

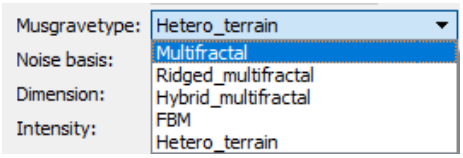
From the blender manual. *The musgrave texture is used to generate organic materials, but it is very flexible. You can do nearly everything with it.*



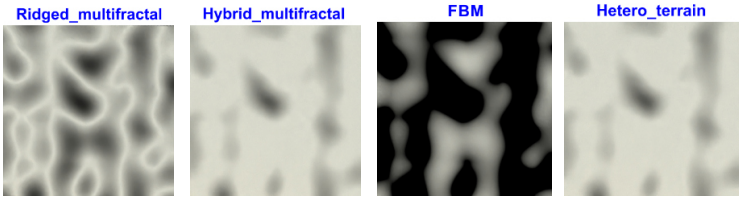
As with all wrappings, set the scale. In this case 3d mapping instead of 2d mapping.



Musgrave type.

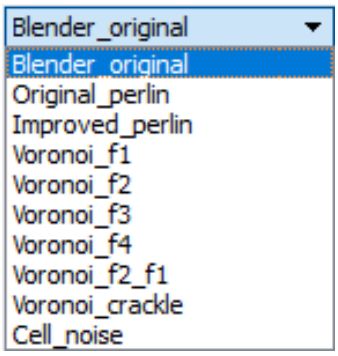


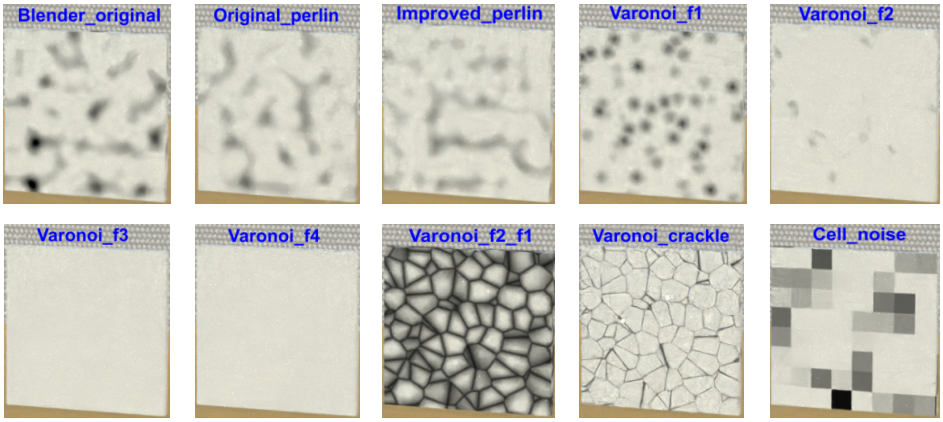
Musgrave has 5 different types, which can be combined with the 10 different noise basis, to give a huge number of variations. Way too many to discuss here.



Noise Basis.

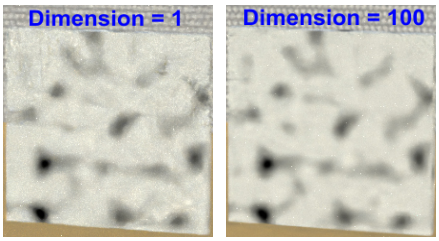
Like other noise properties there are ten options.





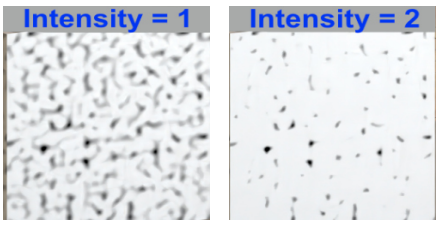
Dimension.

No idea what this setting is suppose to do.



Intensity.

The intensity of the white areas. High intensity can 'white-out; the map.



Lacunarity.

From Wikipedia. A specialised term in geometry referring to a measure of how patterns, especially fractals, fill space, where patterns having more or larger gaps generally have higher lacunarity.

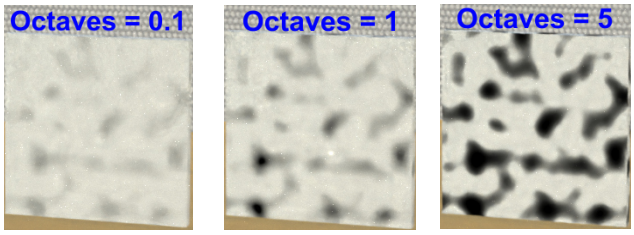
No known usage at time of writing.

Offset. No known usage at time of writing.

Gain. No known usage at time of writing.

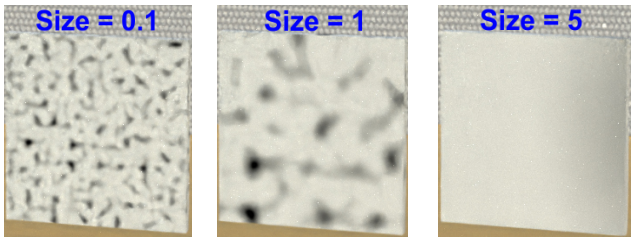
Octaves.
Increases the strength of the effect. At low octaves the effect is more grey noise.

At higher octaves, the effect is more black.



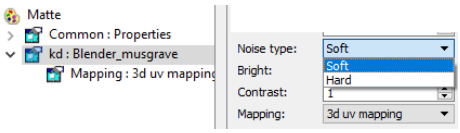
Noisesize.

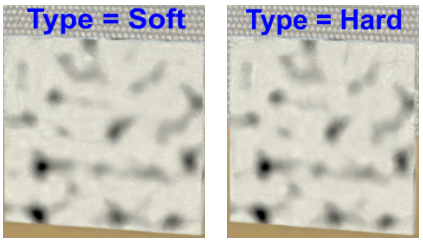
Size of the particles used to create the effect. Too large a size, will not show the effect.



Noise Type.

May require a specific circumstance to make any difference.No noticeable difference.



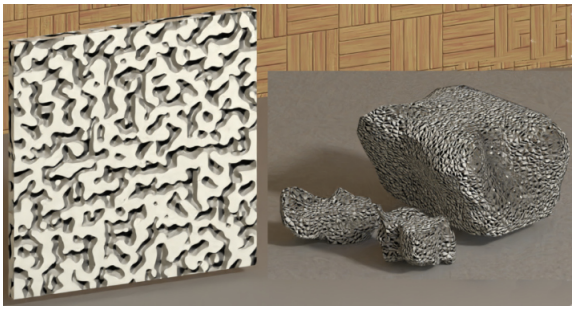


B**right and Contract.**

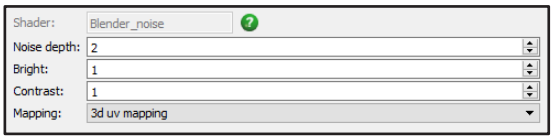
The same as an image program. These can make a big difference, when used as a bump map.

Bump.

Makes a good and varied bump map.

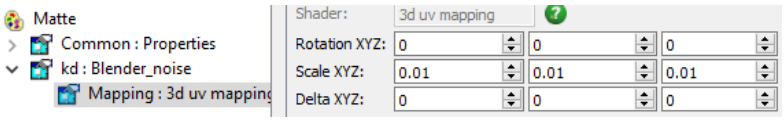


Noise.



From the blender manual. *Although this looks great, it is not Perlin Noise! This is a true, randomly generated Noise. This gives a different result every time, for every frame, for every pixel.*

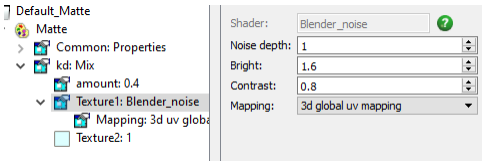
As with all wrappings, set the scale, in this case 3d mapping instead of 2d mapping

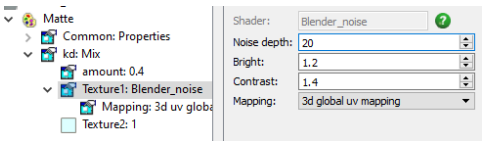


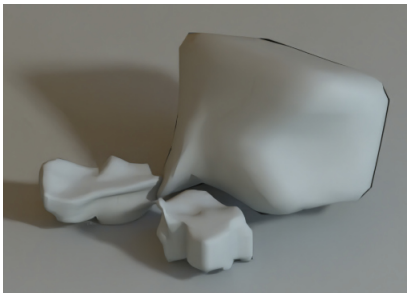
The noise property is an odd one. Most blender noise properties, Mustgrave voronoi for example, are used a bump map or transmission maps. However this noise is so fine, it doesn't really work as a bump map.

Using noise as part of, for example a mix property or Add property works. Though the result can be hardly noticeable.

The following uses the mix property in the KD section.





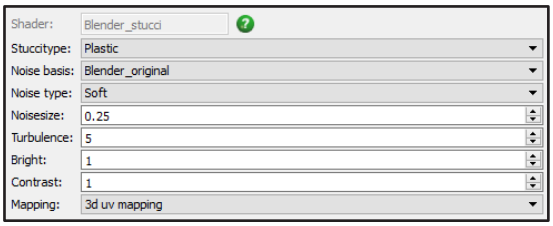


As can be seen, the noise doesn't really change, it is the brightness and contrast that changes the render.

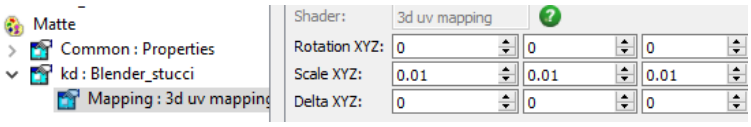
There are better noise options if one wishes to get a propper bump map.

Stucci.

Stucci is a good option for fine bump maps.

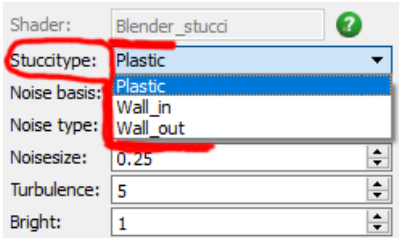


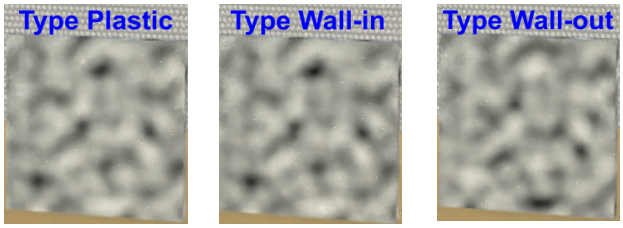
As with all wrappings, set the scale, in this case 3d mapping instead of 2d mapping



Succi type

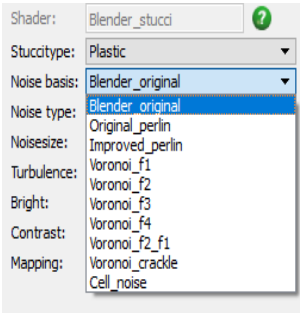
There are three types, which vary the effect, with wall-out being an invert of wall-in. This makes it easy to reverse any bump map effect, by simply changing the type.

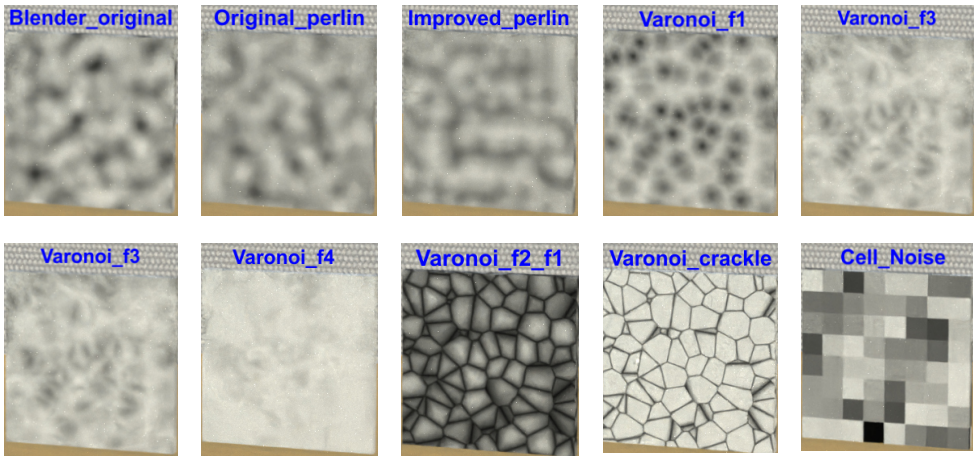




Noise Basis

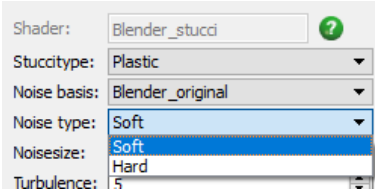
There are ten noise basis items, like most other noise ttextures.

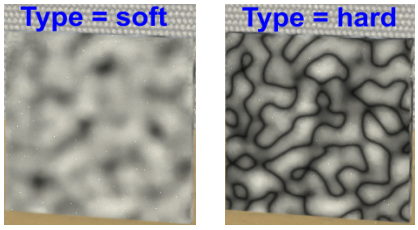




Stucci. - Noise type.

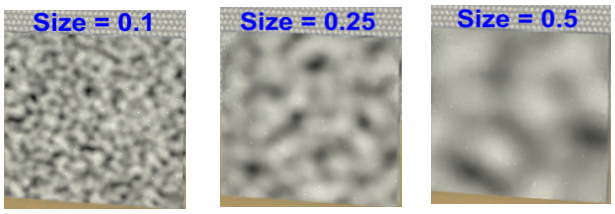
Choose either hard or soft.





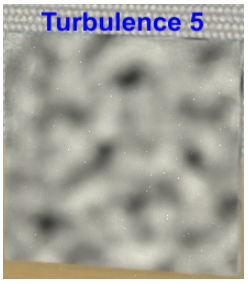
Noise size.

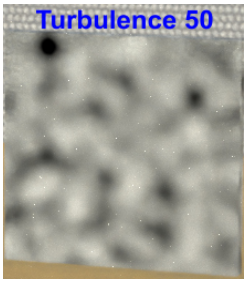
Increase or decrease the size of the noise



Turbulence.

Changes how the noise is displayed.





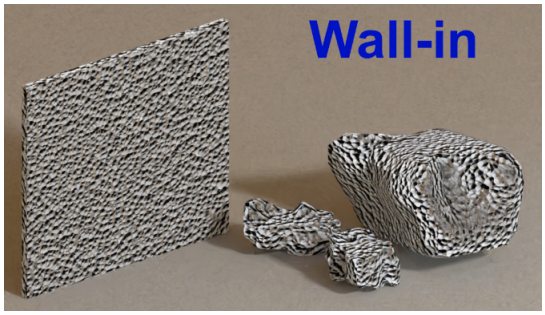
Bright and Contract.

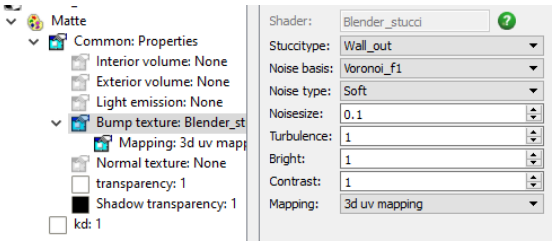
The same as an image program. These can make a big difference, when used as a bump map. Adjusting the brightness and contract will change how any bump map looks.

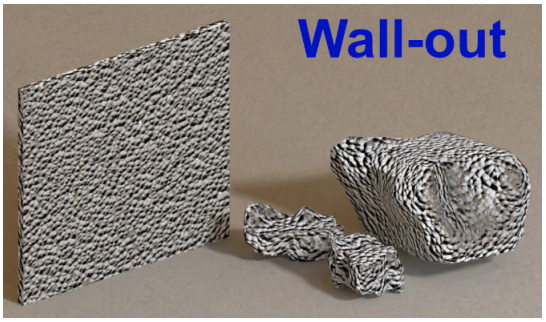
Bump map.

Like many of the Blender types. Stucci works well as a bump maps.

The two bump maps here use the same settings, except, the stucci type was changed between wall-in and wall-out.



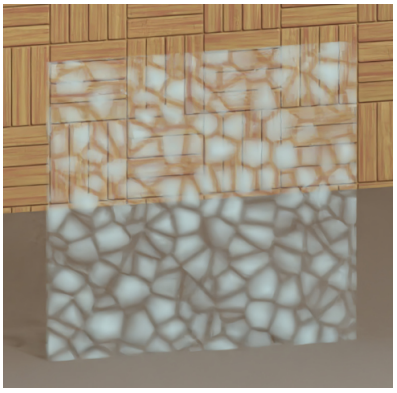


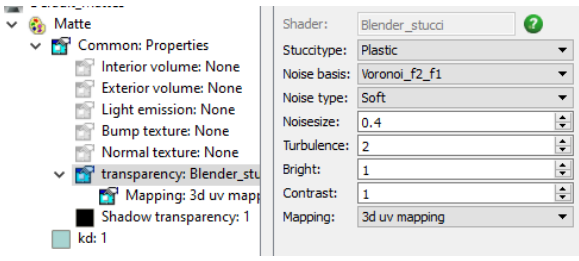


Trasparency.

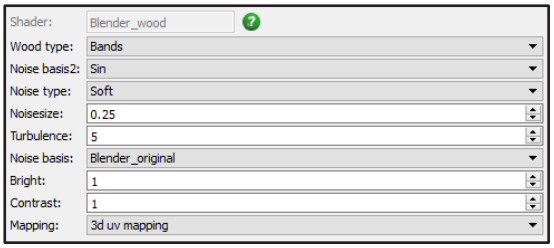
Stucci will work with common properties - transparency.

In this example the noise basis was changed to voronoi_f2_f1.



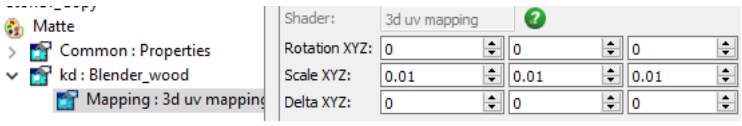


Wood.



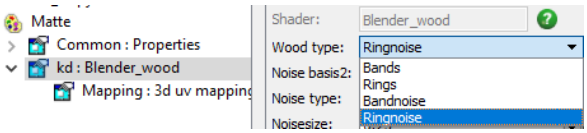
From the blender manual. *The wood texture is used to generate wood and ring-shaped patterns.*

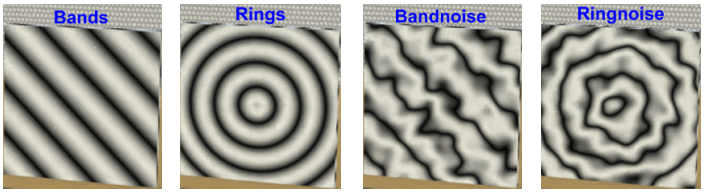
As with all wrappings, set the scale, in this case 3d mapping instead of 2d mapping, one can use standard scale mapping, Or 3d Global UV map.



Wood Type.

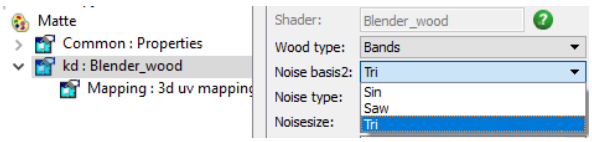
There are four types to choose from, though the last two are just noisy variations of the first two.

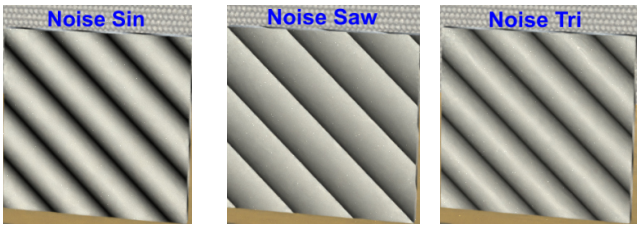




Noise Bands.

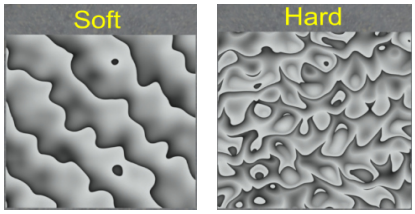
The three noise bands are Sin, Saw and Tri.





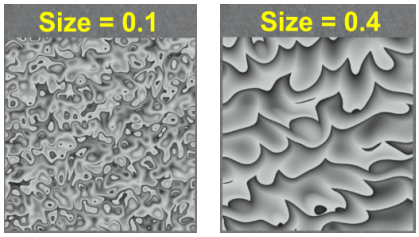
Noise Type. Hard or soft.

Mainly works with band noise, and ring noise wood types. And changes the pattern created for the wood.



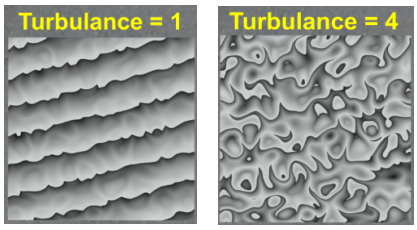
Noise Size.

The size of the generated noise. Although one can enter numbers above 1. The main change to the pattern is between 0 and 1.



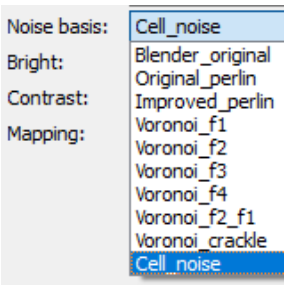
Turbulence.

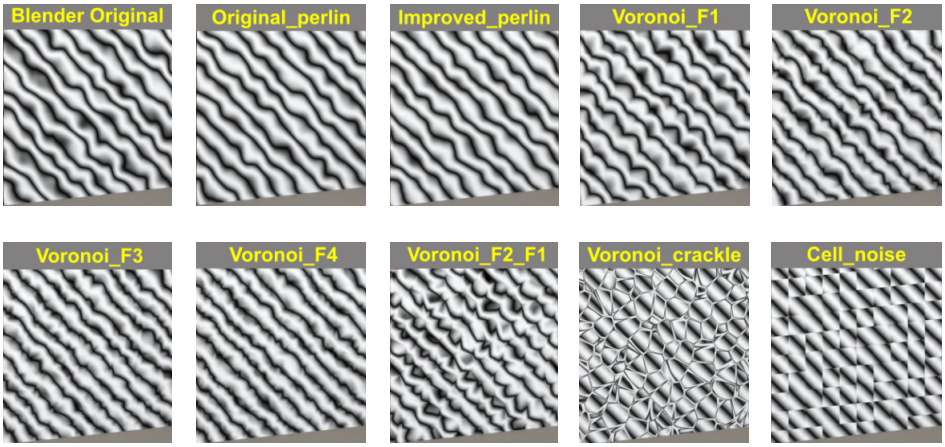
Higher numbers add more randomness to the effect. The effect differs depending on previous settings. Including wood type.



Noise Basis.

The first 8 are simply slight variations. Voronoi crackle adds a cracked effect, and Cell Noise is a pixel effect.



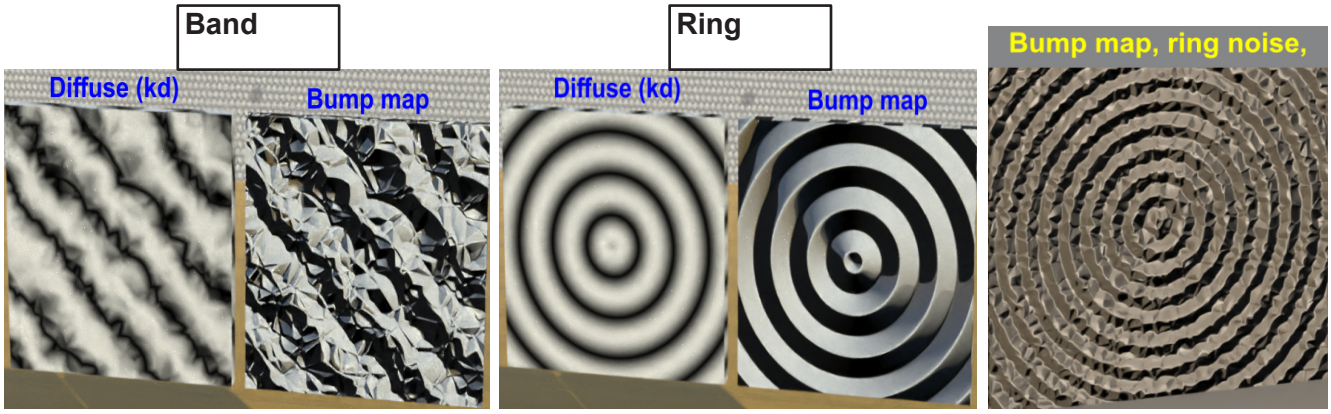


Bright and Contract.

The same as an image program. These can make a big difference, when used as a bump map. Adjusting the brightness and contrast will change how any bump map looks.

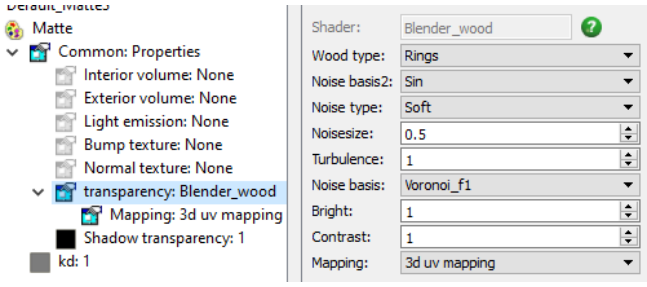
Using Blender_wood as a bump map.

Works to give a stripey effect to any bump map. Whether diagonal, or circular.



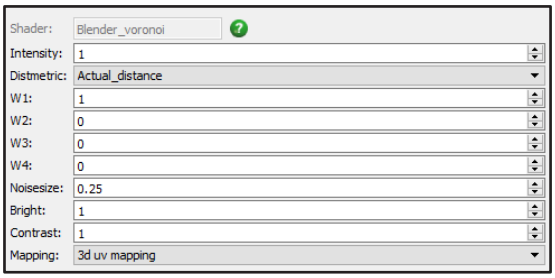
Transparency.

Wood noise can be used wit common properties - transparency. Not sure why one would want to use it like this. But this just shows that it is possible.



Voronoi.

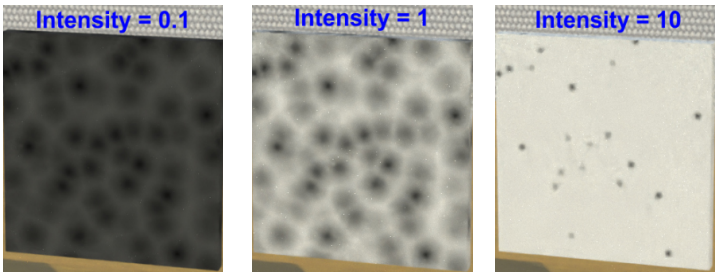
From the blender manual. *The Voronoi texture is used to generate very convincing Metal, especially the "Hammered" effect. Organic shaders (e.g. scales, veins in skin).*



Intensity.

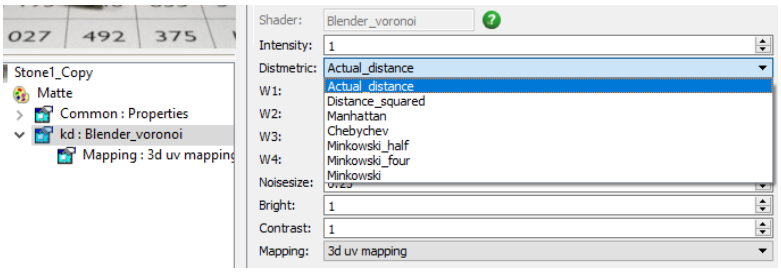
Lightens or darkens the background for the effect.

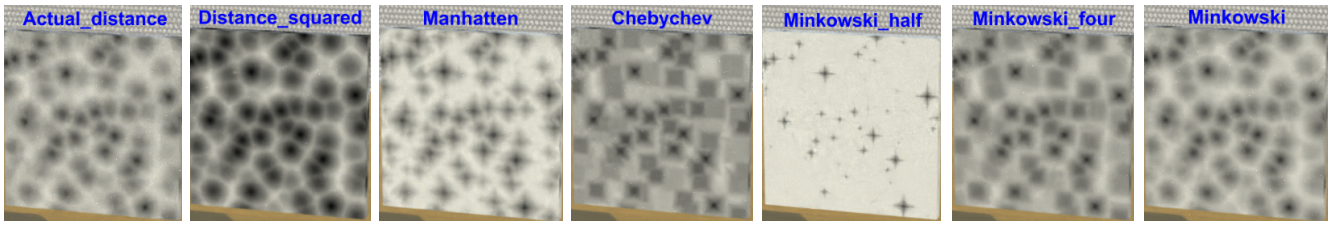
A high intensity means the background stands out a lot more than the spots.



Distmetric.

Various pre-programmed effects, some appear to be more star effect than the hammered effect the blender manual mentions.

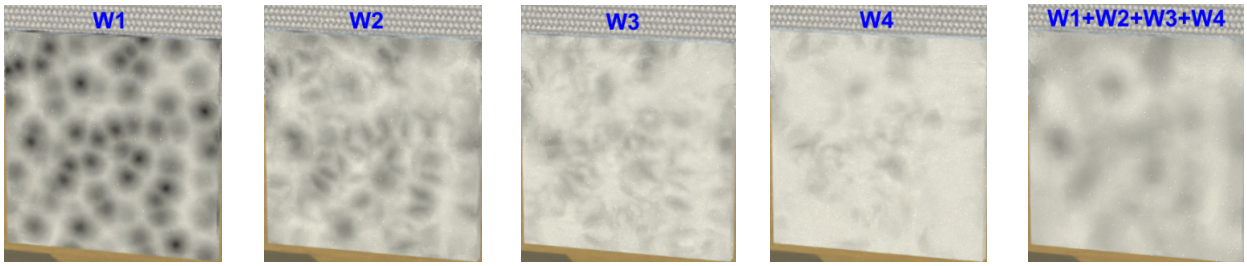




W1 to W4.

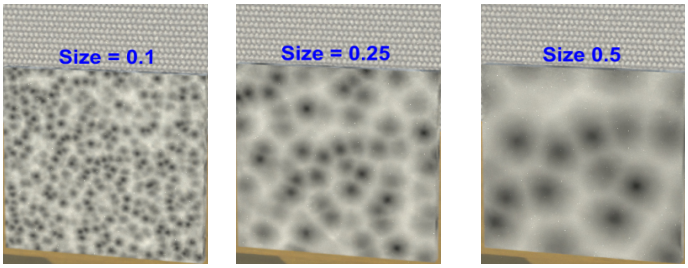
Five preset noise calculations.

W1. **W2.** **W3.** **W4.** **W1,W2,W3,W4**



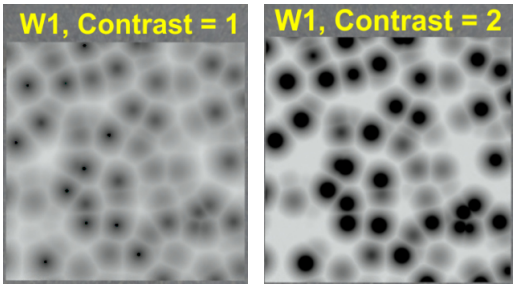
Noise size.

Increases or decreases the size of the particles, for a hammered finish, a larger size may be required.



Bright. Contrast.

The same as an image program. These can make a big difference, when used as a bump map.

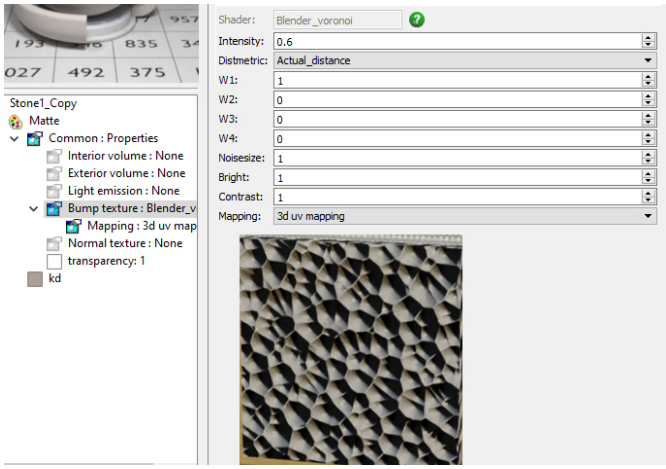


Bump map.

Like most of the procedural textures. As 2d they are of limited use.

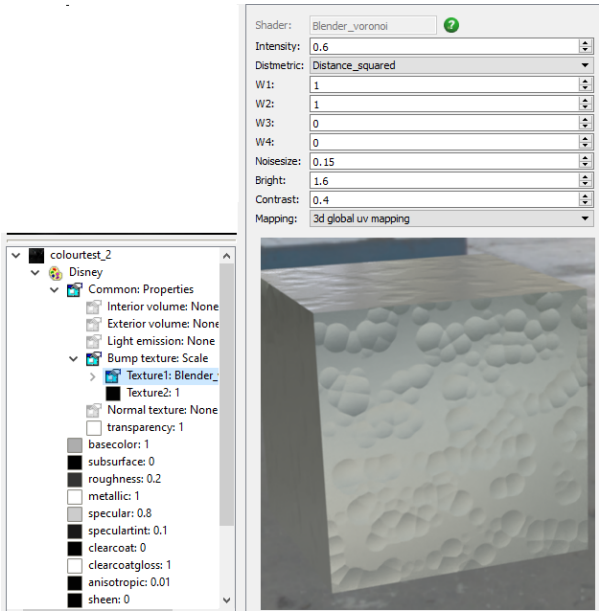
They are especially useful for use as 3d bump map.

There are way too many variations to display here.



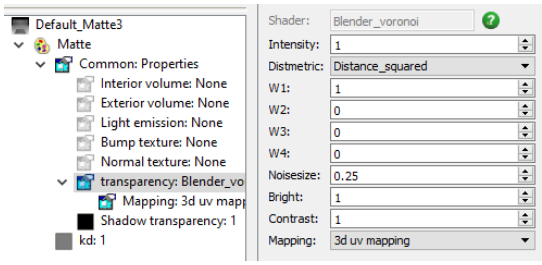
Using Voronoi to give dents in a material.

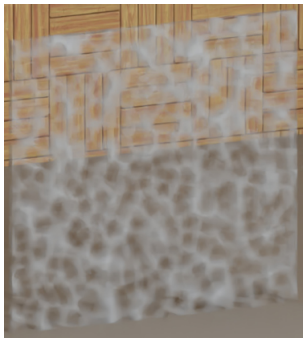
Note, a lot of it is just trial and error.

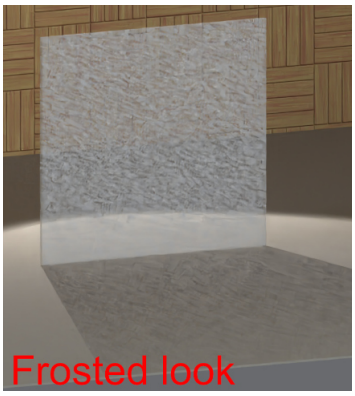


Transparency.

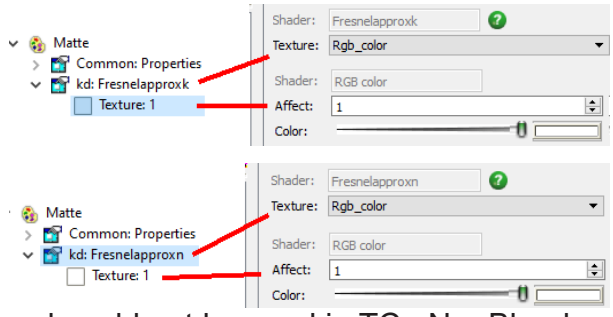
As with most maps, Voronoi can be used as a transparency map.







Textures for loading optical data



Fresnelapproxk. Default - rgb_color White

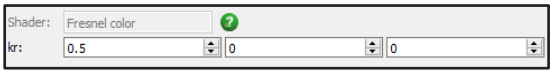
Fresnelapproxn. Default - rgb_color White

texture names or constants

Fresnel k and n are wavelength data, and may be available from the internet. Though the one I found could not be used in TC. Nor Blender for that matter. Too specialised for me

---

Fresnel color. (Reflection color). Default - 0.5, 0, 0

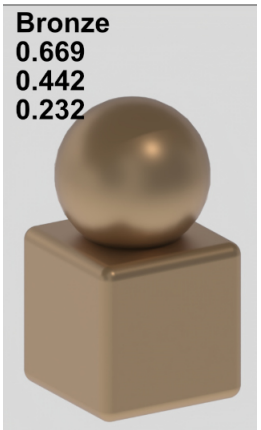


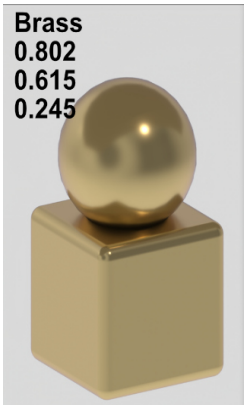
Used as an option in the metal shader.

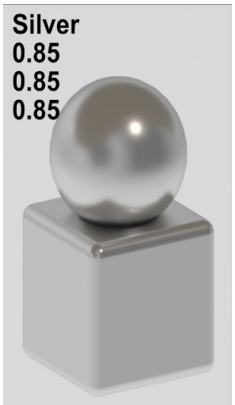
Reflection colour. Allows on to specify ones own colour, for example Bronze.

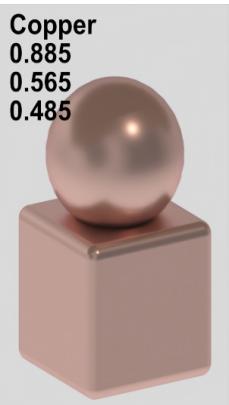
See page … for colour codes.

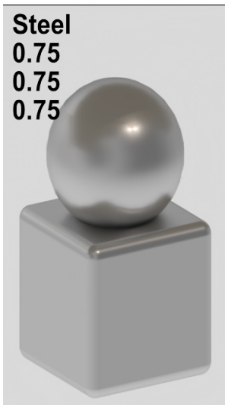
More example colours below.

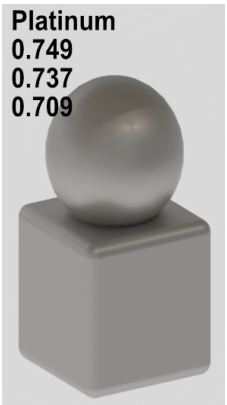


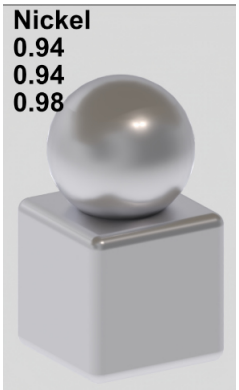


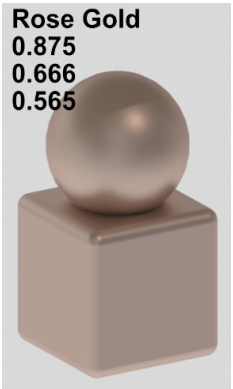


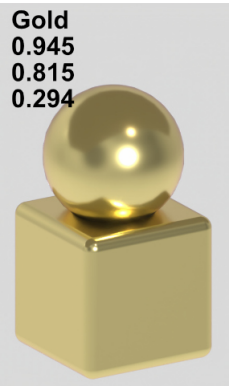


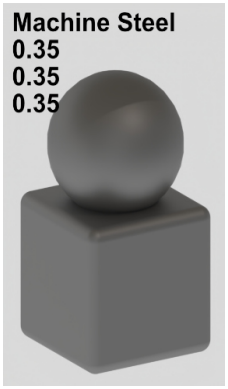


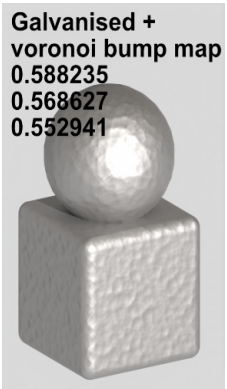








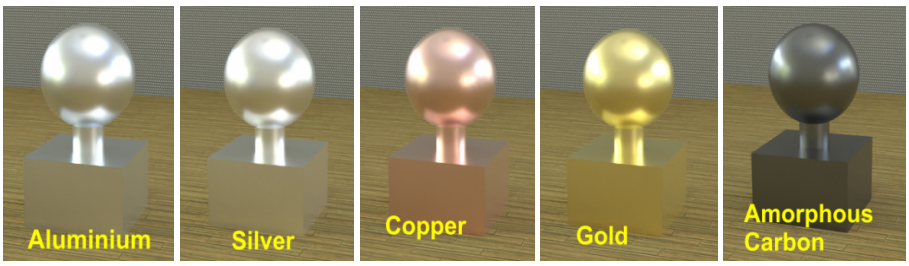




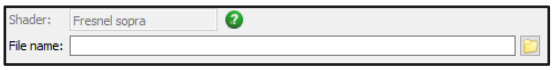
Fresnel preset. Default - Aluminium

fresnel preset ("aluminium", "silver", "copper", "gold" or "amorphous carbon")

These are pre programmed metals. Saves having to look up the metal colours.

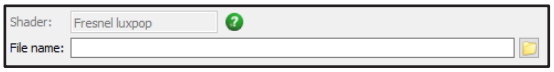


Fresnel sopra. Default - file name



nk file path (e.g scenes/luxball/Au.nk)

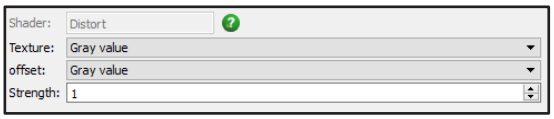
Fresnelluxpop. Default - file name



nk file path (e.g scenes/luxball/Cr.nk)

Text files from the internet. (From what I read).

Distort. Default - Texture - gray_value White
Default - Affect - gray_value White
Default - 1

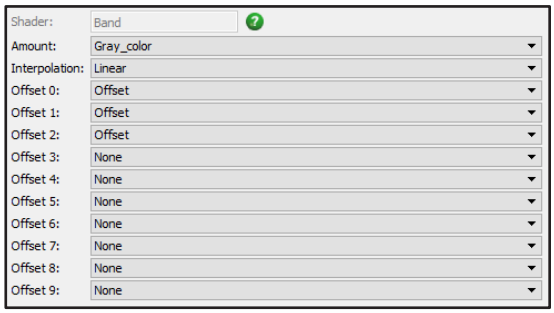


Texture
Affect
Strength

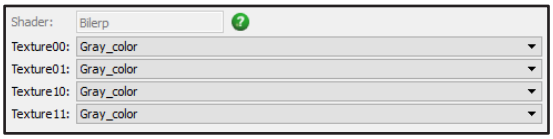
Doesn't appear to work on my installation, at the time of writing.

Band.

No idea how this works.



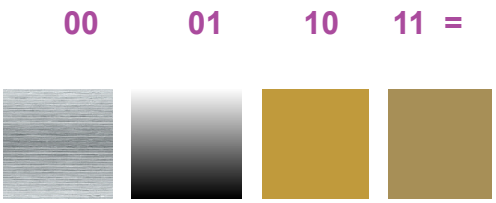
Bilerp.

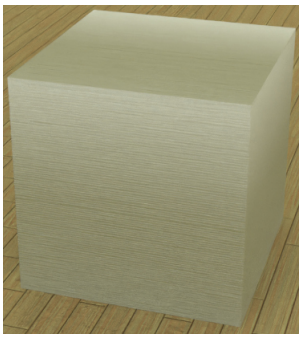


The bilerp texture takes four colours or textures as input and interpolates the colours in between.

Normally it can use uv, planar, cylindrical and spherical mapping. Except TurboLux cannot currently change the type of mapping.

I have no idea how one would normally use this property within T-Lux.





Colordepth.

Color depth, also known as color at depth, is a texture designed to make setting absorption simpler. Normally, you set absorption by defining the color that will be absorbed, resulting in a color control that is essentially "backwards". This texture allows you to instead set the desired result color and how far the light should penetrate before reaching that color.

Depth Default 1

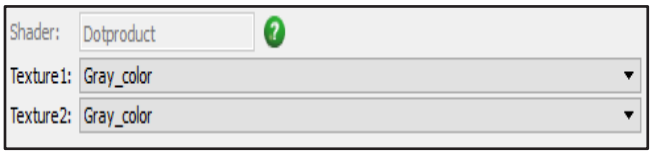
- The depth in meters at which the specified transmission color is reached

Transmission color Default - RGB_color White

The target color of the transmitted light as it reaches the defined depth

Use Colordepth for depth and colour of Absorption. This will mean that the colour selected will not be inverted as is the case with RGB. The depth good alteration for the amount of colour absorbed.

DotProduct.

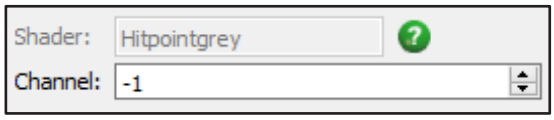


At the time of writing. I've no idea what this does.

Hitpointalpha. Default

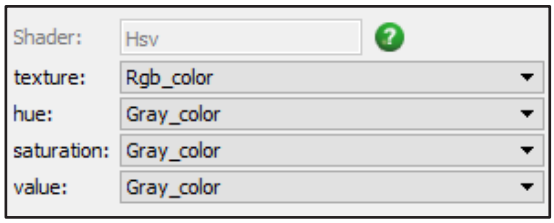
Hitpointcolor. Default

Hitpointgrey. Default



No idea what these are used for withing TC.

Hsv.
Changes the hsv of the selected texture.



Texture Default - rgb colour
By default, Texture is set to rgb color, but for just a colour, its simpler just the alter a materials diffuse colour, This function works well for changing a texture image. Use the texture drop down box to change selection to 'imagemap'. Then follow the normal procedure for inserting image, and setting image scale etc.

Important to remember that this changes a texture (image or colour), it doesn't simply colour an object, like one could normally do within TC colour palette.

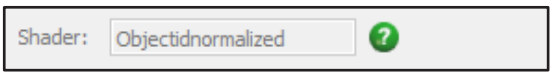
Hue Default - Gray_color
Alters the hue of the image

Saturation Default - Gray_color
Alters the saturation of the image

Value Default - Gray_color
Alters the value of the image

Objectidcolor. Default - blank

Objectidnormalized. Default - blank

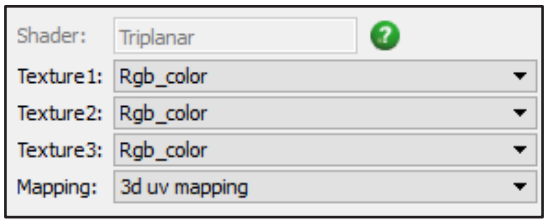


It allows you to access the object ID in a material.
By default, object IDs are assigned by hashing the object name (so you get a seemingly random, but stable value). But you can also assign custom object IDs in the latest v2.2beta1 (in the object properties in Blender).
The object ID is a 4 byte unsigned int.

There are three modes:
- Color: the lower 3 bytes are interpreted as R, G, B values -> you get "random" colors
- Normalized: the returned value is in the range 0..1 -> you get "random" grayscale values
- Raw: you get the raw object ID (range 0..0xffffffff), this is for power users who want to do their own calculations

I doubt these are of any use within TC.

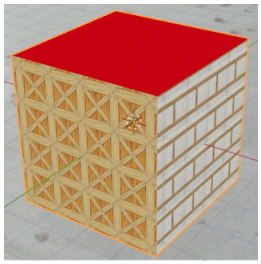
Triplaner



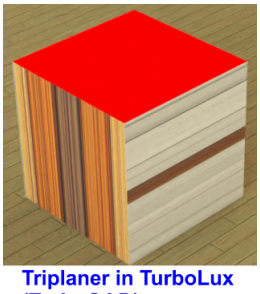
Triplaner puts a different colour or texture on the x, y, and z faces.

At the time of writing. This works in blender, but doesn't work properly in TurboLux.

Texture 1 goes on + and - X faces (as depicted by ref point). Texture 2 goes on + and - Y faces. Texture 3 goes on the + and - Z faces. Note changes in the ref point (through 3d add for example), may not take effect straight away. On may need to force the change, (sat healing, place workplane, close / reopen TC etc.



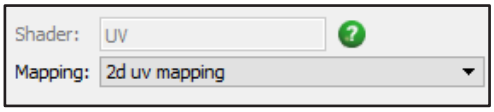
Trilpaner in LuxCore (Blender)



Triplaner in TurboLux (TurboCAD)

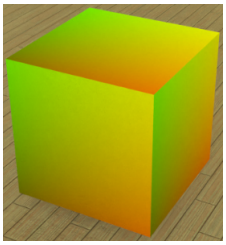
The textures can be colours, textures or images. Simply change the drop down boxes shown above, and select the relevant property.

Uv (test). Default - mapping - 2d uv mapping

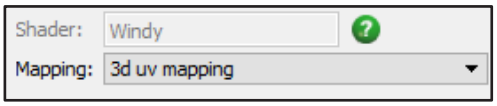


Created a uv pattern. No idea why.

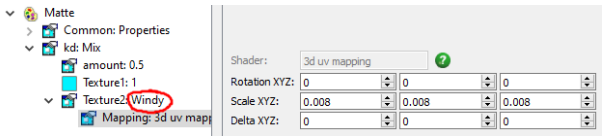
This procedural texture has no parameters aside from the support of 2D mapping.



Windy. Default - mapping - 3d uv mapping

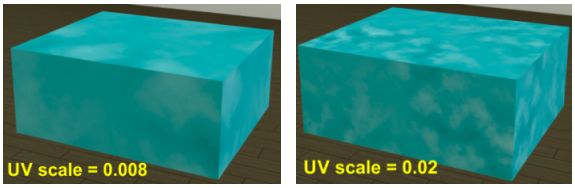


This procedural texture has no parameters aside from the support of 3D mapping.

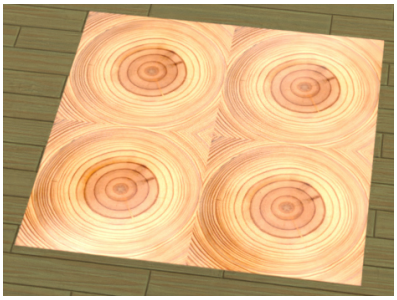


Windy can add colour variations to plain colours, when used with mix parameter for example.

Increasing the uv scale, increases the effect.



Windy can also be used with images. The wrapped wood end (right) would normally be identical on all four patterns. But using mix and windy. Some variation can be applied,

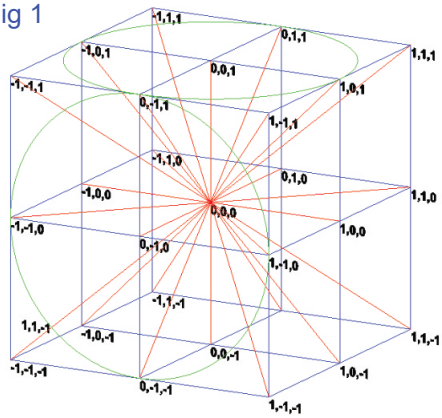


Appendix 1

Vectors

Whilst the normal way of thinking of angles in TC is degrees or Radians, With material and light, including sun, there are also vectors. Although not always strictly necessary, often they need to be entered in the range of -1 to +1.

Fig 1

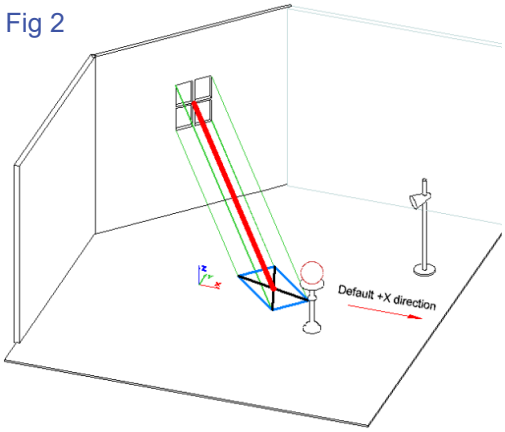


The origin, would be the centre of extents of the drawing (not an object). Classed as being at vector 0, 0, 0 (this is not world coordinates)

As can be seen in Fig 1, all directions can be contained within the -1, 0, +1 matrix. There are a few things in TC which can use this type of coordinates, Including RedSDK Sun and TurboLux sky and sun position. And for these examples, we will use sunlight in TurboLux.

Sun vectors are generally represented on a spherical surface, However they can be represented on two 2d planes, x,y. And x,z (or y,z).

Fig 2



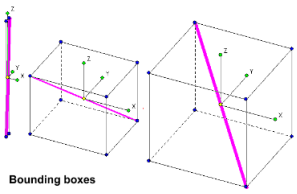
There are a few ways we can get the correct vectors. One can use vector calculations (a later example). Here are a few examples. They all start with the setup shown in Fig 2. In all cases we are trying to get the sun to shine at the blue rectangle drawn on the floor. We use the red 3d polyline as the direction needed.

NOTE. Some images may be shown 'over complicated' . This is for illustration purposes only. And does not need to be included to obtain the correct vectors. But in all these examples,. **The red direction line is needed.**

If the sun is pointing in an orthogonal direction. i.e. pointing straight down, or in the direction of the orthogonal horizon. Then it is simply a matter of deciding whether the directions are + or minus. Other than that, we need some way of getting the x, y, and z vectors.

The quick method. (Selector Entity coordinate system to Extents)

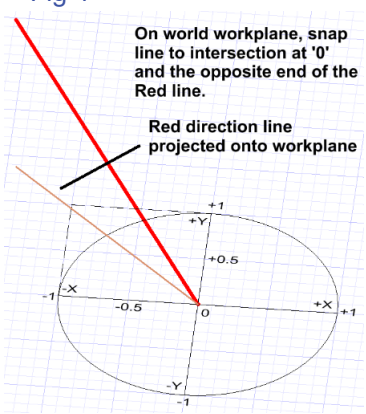
Fig 3



First, set selector coordinate Entity system to Extents.

2D and 3D object have a bounding box. Irrespective or what non-ortho direction is used. Fig 3.

Fig 4



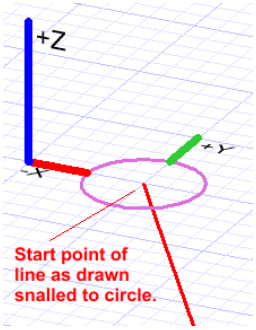
What we need first is to decide on whether the sun x, y and z vectors are pointing at the origin point (0), from the positive of negative direction.

In Fig 4, looking at the red direction line, (I projected the tan line onto the workplane for clarity). We can see the X direction is -X, Y direction is +Y, and Z is above the horizon, therefore +Z.

The middle image in Fig 3 above would be -X, -Y, +Z.

Draw a circle on the world workplane. This is strictly so that, if, when we re-size the red 3d polyline, it gets too small to see, or if we click away from the line. we can still easily find it again.

Fig 5



Make a copy of the red line, Relocate reference point onto the start vertex. If you cannot remember which way you drew it. The start vertex can be found by checking the start vertex coordinates, in the selection palette - metrics.

Snap the direction line to the centre of the circle. I drew the axis marker for visual reference only. Fig 5.

Fig 6

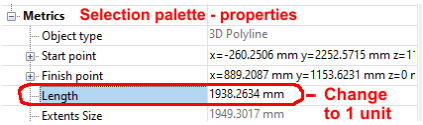
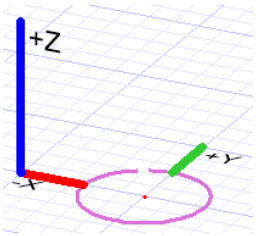


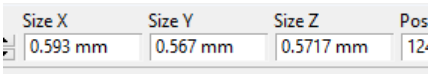
Fig 7



Access the selection palette - properties section, and expand metrics.

Change the length of the line to 1 unit. i.e. 1 mm, 1 inches etc. Fig 6. As can be see on the right. The line can become pretty small, hence why the circle makes it easy to find. Fig 7.

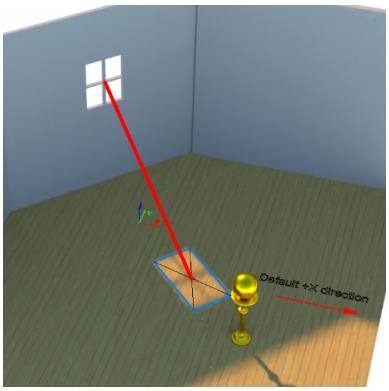
Fig 8



Select the line, and read the sizes in the inspection bar. Sizing the object down to 1 unit, means the sizes will fall into a 1 x 1 matrix. Fig 8.

Next. Assign a positive or negative direction. As we found on the last page, X direction is -X, Y direction is +Y, and Z is +Z.

Fig 9



Therefore the three vectors we need are X = -0.593, Y = 0.567, and Z = 0.5717, Type these into the sun direction (turbolux scene) or via a sun luminance. Rendering the scene, we can check if it is pointing as we would like. Fig 9.

IMPORTANT NOTE.

The above may only works if the 3d line bounding box In selector extents = Entity, and it aligns with the x and y world directions. For example. If the drawing (not the view) is rotated, the bounding box is also rotated Fig 10. This means the above procedure probably won't work. Therefore we need to get the objects bounding box, aligned. The simplest way is to select workplane by world, and snap a new 3d polyline into place.

Fig 10

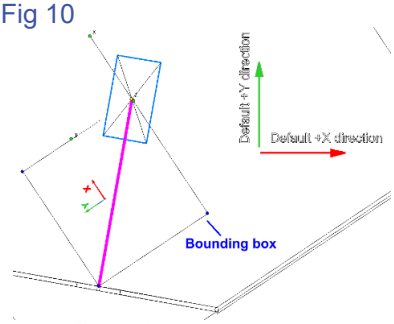
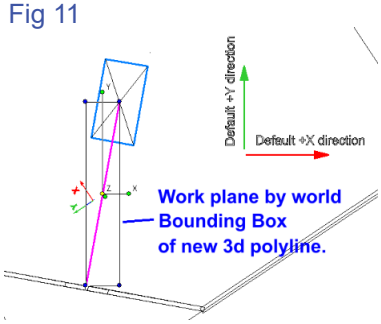


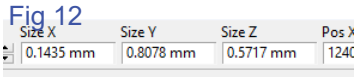
Fig 11



Now we have the bounding box correct, Fig 11. We can look at the drawing and determine the new x,y,z directions.

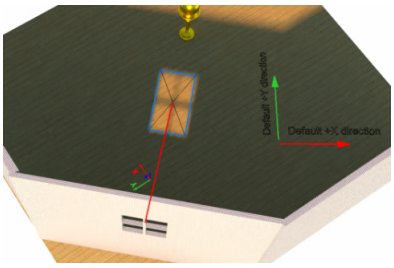
From the strike point (the blue rectangle. We can determine the line (sun) is shining from -x direction, -y direction, and above the horizon so + z direction.

Fig 12



Completing the steps as previously stated. On the inspection bar we have X = 0.1435, Y = 0.8078, Z = 0.5717
Applying the + or - sign to the inspection bar size fields. In this case both x and y are in the minus directions.

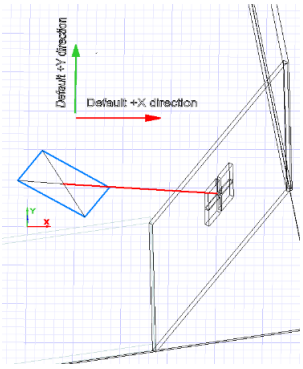
Fig 13



We get. X = -0.1435, Y = -0.8078, Z = 0.5717.

What if the drawing is rotated, and for whatever reason, you prefer not to change to extents. Drawing a new direction line may not help..

Fig 14



One could then use a **fully manual approach**. Where the coordinate system doesn't matter.

The same start drawing is used as previously, but rotated. Fig 14.

We will use the grid to get a 20 x 20 area on the world workplane.

Draw a circle to represent the spherical surface. Snapping the circle to a major grid line will make it easy to view as 20 x 20.
For reference, draw lines to represent +- X, and +- Y

Fig 15

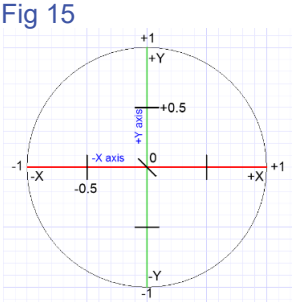
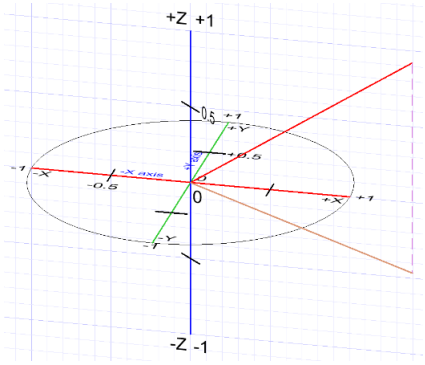


Fig 16



In front view, we add a a line to represent + and - Z direction (shown blue). Fig 15.

Make a copy of the red 3d direction line. The reference point is relocated to the target vertex. And snapped to the centre of the circle.

The diagonal 'tan' line in Fig 16. represents the red direction line as 2d on the world workplane. Simply switch to world workplane. C snap a 2d line to centre of the circle, and V snap to the end of the red direction line. Because it is a 2d line, it will project the end point onto the workplane.

Fig 17

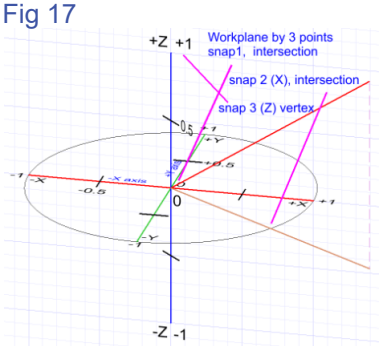
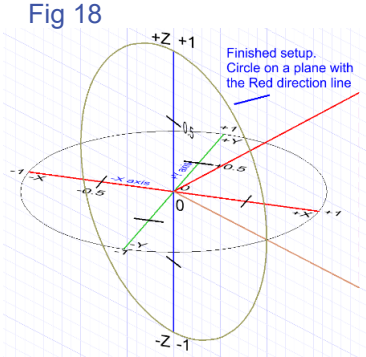


Fig 18



This is used for the z direction. To set the workplane.

I added the + and - markers for illustration purposes only.

Set the workplane, to that of the red sun direction line. Fig 17.

Workplane by 3 points, snap to the centre of the circle (c). Intersect snap (i) where the 'Tan coloured' line crosses the circle. And finally vertex snap (v) to the top of the +Z line.

Snap a new circle on this workplane. Fig 15. The same size as the original circle, which can be typed in, or 'v' snap to Z, or 'I' snap to 'tan' line and circle, or snapped to grid (g).

From here there are multiple options. But as this is estimation, we will stick to 2d only.

As we still have the workplane set to the direction line. Use View menu - camera - plan - by workplane. Or choose UCS plan (ctrl + shift + w) (Depending on ones TC interface)
.

Fig 19

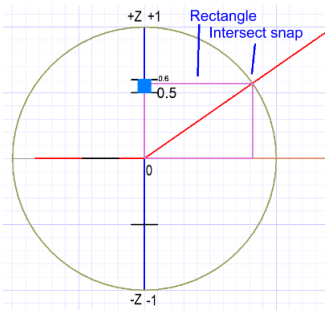
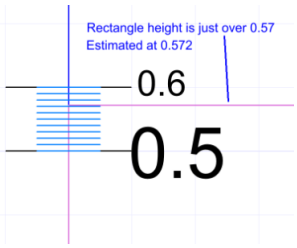


Fig 20



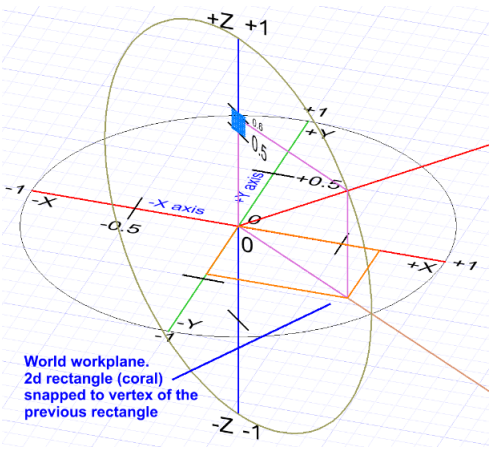
Draw a rectangle, snapped to centre of the circle. And intersect (i) snapped to the junction of the red direction line and circle edge. (shown orchid). Fig 19.

Using the grid we can estimate the junction on the Z axis. (this is the z vector). Fig 20.

I drew extra lines for clarity.

Switch to ISO view and World workplane. draw a rectangle, snapped to centre of circle, and the vertex of the previously drawn orchid (z direction) rectangle. Shown coral in Fig 21.

Fig 21



Switch to world workplane.

Estimate the coral rectangle vectors, as shown below. Fig's 22 to 24.

Fig 22

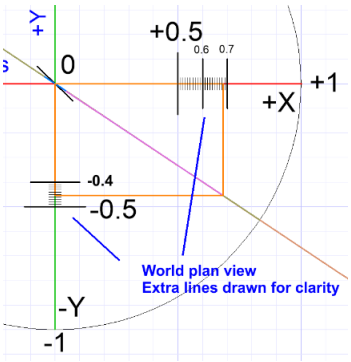


Fig 23

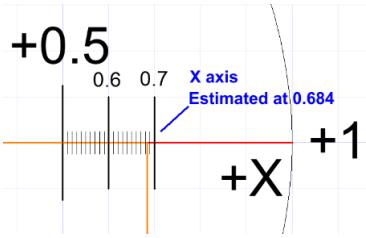
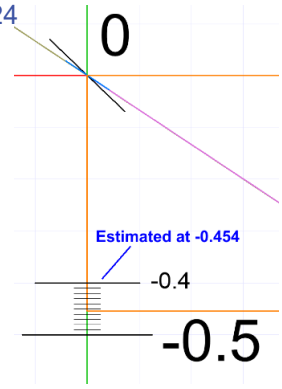


Fig 24



These give us our final vectors, X = 0.684, Y = -0.454, Z = 0.572. Entering these into the sun direction settings, Fig 25.

we can check our estimation. Fig 26.

Fig 25

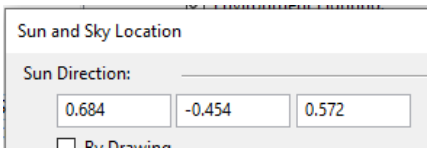
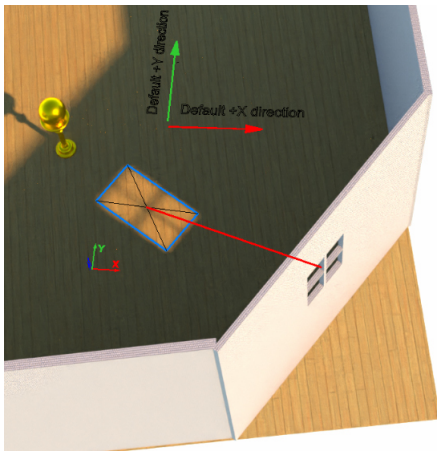


Fig 26



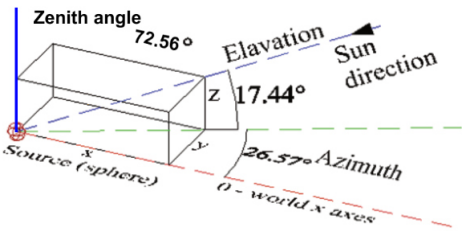
In case anyone doesn't wish to estimate. One can calculate using the grid and the rectangles. In our scenario, each grid space is 80mm, and each quadrant is 10 x 10 to grid squares. Therefore = 10 squares x 80 mm = 800mm total.

X vector (coral rectangle length) is 546.6122 mm / 800 mm = 0.68327
Y vector (coral rectangle width) is 363.3633 mm / 800 mm = 0.5442 (change sign to minus)
Z vector (orchid rectangle height) is 457.3644 mm / 800 mm = 0.57171

The last example, is more mathematics. And uses angles rather than drawing grid lines.

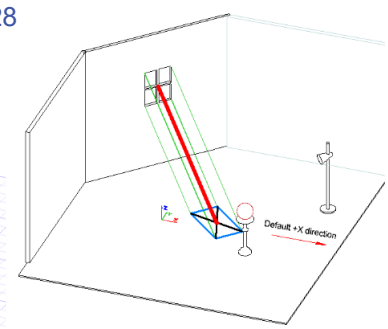
This relies on calculating the zenith angle, and the azimuth angle. Fig 27. And is the most accurate as it takes away much of the 'human error' factor, by reducing the amount of snapping required.

Fig 27



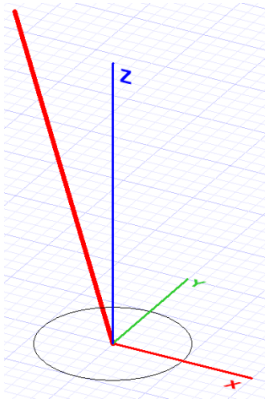
The angles required are from world +X in an anticlockwise direction to the sun direction line. 0 to 359.999 etc.

Fig 28



And the angle from +Z to the sun direction. 0 to 90. Assuming the sun is above the horizon.

Fig 29



We will use the original drawing for this example so we can compare figured to the 'quick method'. Fig 28.

The setup is very simple.
We have our X, Y and Z indicators.
A circle, which is only for reference to the world workplane.
And our sun direction line, snapped to the XYZ intersection.

Fig 30

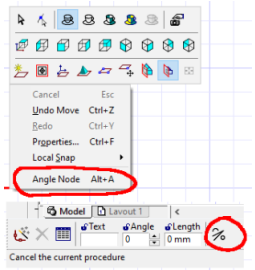
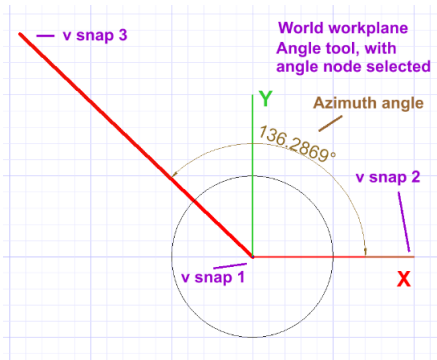


Fig 31



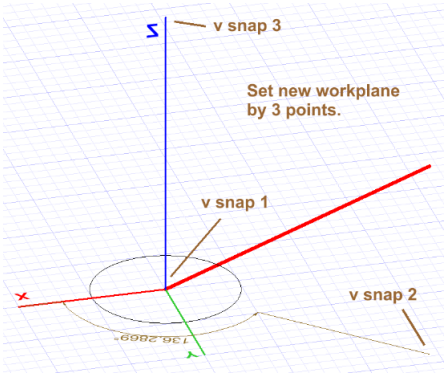
For the first angle, we need it from world +X to the direction line, anticlockwise. We can use the angle tool, with 'Angle node' selected (inspection bar, or right click menu) Fig 30.

We can use three vertex snaps to obtain the angle. As shown in Fig 31.

This we will call angle A and is 136.2869 degrees

NOTE. There are multiple ways to obtain the Zenith angle. This is an example of one way. It may not be the simplest, nor the quickest way. But it works.

Fig 32



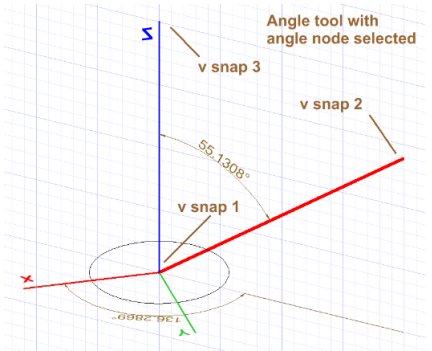
To get the correct zenith angle, we need a new workplane by 3 points, in line with the direction line.

For once, the angle lines come in handy as a snapping point. Snap the workplane as shown in Fig 32.

Last setup is to get the zenith angle. Again using the angle tool with angle node selected. As shown in fig …

And that is all the setup we need. We now have the z angle for the vector calculation. 55.1308 degrees. Fig 33.
We will call this angle B

Fig 33



So we have our two angles. A = 136.2869. And B = 55.1308. (but we need them in Radians). One can convert first, or just add Radians into the calculation.

For the vectors, we will use three calculations. One each for X, Y and Z. For 'c' in the following calculation we use 1 as the units size. Because we are only interested in direction, not magnitude. (I got the calculations via Internet).

The following calculations are for the x, y and z boxes in the sun direction entry dialog box.

X = c * sine (B) * cosine (A) - - - = 1 * Sin (Radians (55.1308)) * Cos (Radians (136.2869))

Y = c * sine (B) * sine (A) - - - = 1 * Sin (Radians (55.1308)) * Sin (Radians (136.2869))

Z = c * cosine (B) - - - = 1 * Cos (Radians (55.1308))

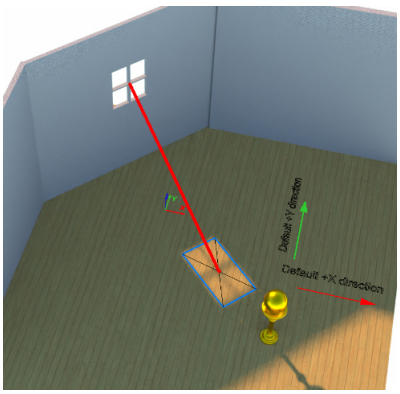
Therefore (figures rounded to 6 decimals)

X = Sin (0.96221398) * Cos (2.37865513) - - - = -0.593036

Y = Sin (0.96221398) * Sin (2.37865513) - - - = 0.566977

Z = Cos (0.96221398) - - - = 0.571705

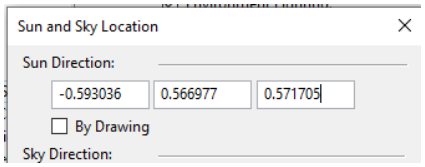
Fig 35



Now its time to enter the coordinates in the Sun luminance direction boxes. Fig 34.

Cross ones fingers and hope our calculations were correct.

Fig 34

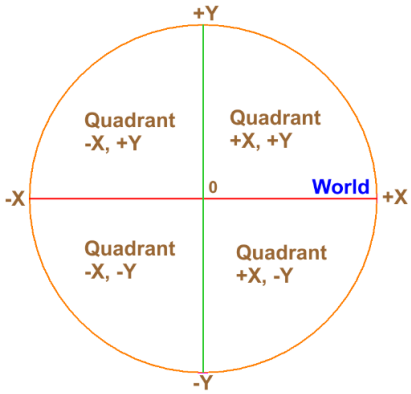


As can be seen from Fig 35. The sun shines down pretty good through our window, onto our estimated floor blue rectangle.

As previously stated, there are other ways to get the vectors required. And probably much easier ways. This is document is just three ways to get one started.

Obviously if one is using Geo-Location, then none of this is necessary. This is for occasions where you need the sun to always point in a specific direction, or at a specific location.

One last thing about quadrants. When deciding the positive or negative x and y signs. Fig 36. Simply shows the sign needed in each quadrant .

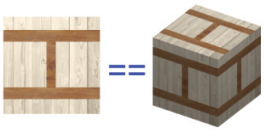


Disclaimer. I've tested this many times, but I have not tested every combination of X, Y, Z coordinates. Therefore cannot guarantee the methods will work in every scenario.

Andy

Appendix 2

Texture scale



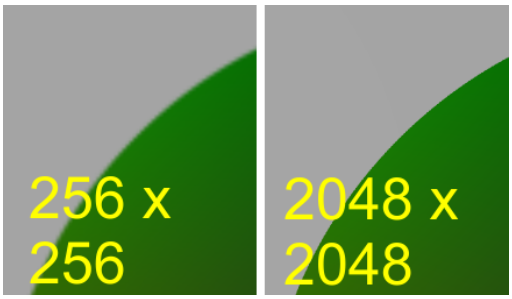
Page	
About scaling	*A2-1*
Offset	*A2-2*
Size texture to fit object	*A2-3*
Size Object to fit image	*A2-3*
Close enough is OK	*A2-3/4*
Tile Image	*A2-4*
Tile to real world sizes	*A2-5*
End notes	*A2-6*

Ideally the texture should be the same aspect ration of the object it is being wrapped onto. This is to prevent undue scaling of the texture in one direction.

And there are some situations where this is important.
A company logo, for example. If someone has spent hundreds of thousands of pounds, getting their perfect company logo. You cannot just stretch it out of all proportion.

But many times some scaling does not matter, because a it will not be unnoticeable, concrete for example. On many occasions though, you simply need to choose the best method to ensure the texture wraps without too much distortion.

Image size in pixels. Unless one is designing an image to be a perfect size for an object, Then this only affects the quality of the wrapping. If the original image is 256 x 256. It will almost never look as good as 2048 x 2048. No matter how you try to apply it to the object.

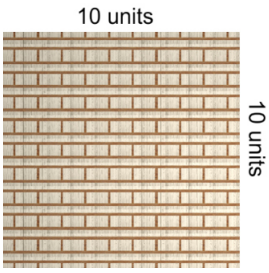


In the past, textures were often 512 x 512 for square textures. Nowadays, one can have 1K (1024 x 1024), 2K (2048), 4k (4096) and even 8K (8192) textures for download,

If the scene is lots of small objects, you could get away with 512. For larger objects, and where an image is being tiled. 1K or 2K, may be needed. Wrapping large objects with a single texture may require 4K. But I think it would be rare that you would need an 8K texture in TC. Except for a background image, or a HDRI image.

What happens during wrapping. In some rendering programs including RedSDK. One can think in terms of one wrap per unit. i.e. when using space units as mm, it is one texture wrap per mm. It doesn't matter what the units are, mm, inches, km, mile, or light years.

If we draw a 1mm square, with a wrapping scale of U and V = 1. It will wrap perfectly. If you draw a 10 mm x 10 mm square, at U, V scale of 1, it will place 10 images wide and 10 images high. Because it is still wrapping 1 texture per drawing unit.



If we wanted the single texture to wrap once on the 10mm square instead of 10 times, we need to amend the wrapping scale U and V. Our single texture is one tenth of the object. So 1 / 10 = 0.1 Therefore entering U = 0.1, and V = 0.1, we will get a single texture covering our 10mm x 10mm square. Remember this is drawing units. So if units were feet, a single texture would be 1ft x 1ft, An the calculation is 1 / object size in feet.

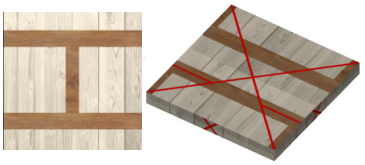
To sum up. For any size of object, if we want a single texture to be applied across the whole object face. We simply divide 1 by the objects width for U, and 1 divided by height for V. (see tiling for a variation).

NOTE, TC can wrap faces of an object differently. For example, on a cube, the top face may be wrapped differently than the side face. This is not a bug, just the ways the object mesh is unwrapped ready for the material to be applied. See last page for wrap on face.

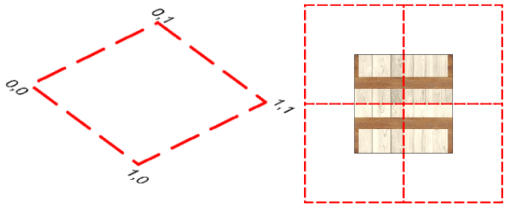
What is the offset boxes for. NOT tested in TurboLux

When a texture is wrapped, it isn't just dumped in the middle, and can be trial and error. For square / rectangular objects, it appears to wrap slightly different to non rectangular objects.

Looking at the far right images, we can see it has not wrapped in the centre of the square. This is often undesirable. And is where the offset comes in.

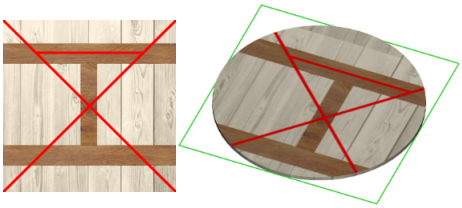


Offset can be thought of in terms of the image. '0, 0' being the bottom left, 1, 1 top right. In order to shift the image into position on the object, we need to move it over by ½ the image, in X and Y.



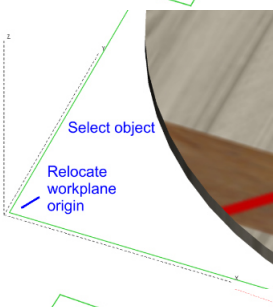
Therefore half way betweeo 0 , 0 and 1, 1 is 0.5, 0.5 .
we can type 0.5 in both the U and V boxes. To push the texture into the desired place. However if using tiling, for example two or more copies of the image on the same face, this changes things slightly.
As a general rule, tiling an odd number 1, 3, 5 etc. one would offset by 0.5.
Tiling even numbers 2, 4, 6 etc. offset = 0.

Now for a wrinkle. Non rectangular objects appear to be wrapped with reference to their own workplane not the object itself. To see this in action, the image right was wrapped with offset at '0'. As can be seen far right, the wrapping of not even centred in any way.

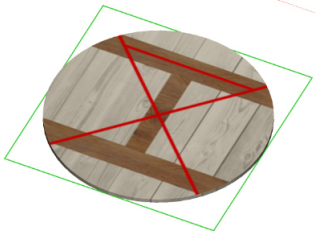


The green square is just a reference polyline.

To see the effects, we select the object and relocate its workplane origin. Snapping it to the bottom left corner of the green square.



Lastly, issue a 'place on workplane' command on the cylinder. As can be seen the texture now wraps correctly. With the image being wrapped from the objects workplane.



Of course, instead of messing with the workplane origin. One would normally just adjust the offset, until the image wrapped correctly.

IMPORTANT. I cannot say this is how TC is programmed. Just what appears to happen on my computer. Your experiments may show a different situation. Therefore the above is just a guide only.

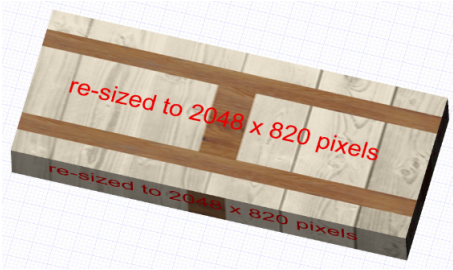
Texture scale is discussed as 5 options.

1). Size the texture to fit the object. (aspect ratio).
Knowing the size of the object you have drawn, One can use a paint program to design or alter a texture, This is to ensure the same aspect ratio of the object. For example. You have drawn a box 500mm x 200mm. And using width divided by height, we get 500/200 = aspect ratio of 2.5 : 1.

If we were using a texture of 2048 pixels wide, we could say 2048 / 2.5 = 820 (round up) pixels high. Designing the texture to 2048 x 820, should (no guarantees), look the same on the object as it does in the texture. In other words, no stretching or compressing of the image.

From there it is a simple matter to get Scale U and Scale V values. Here we just want a single texture to wrap once on the object. Therefore, for the width, divide 1 (U value) by the object width. And height, divide 1 (V value) by the object height.

For example. Our object was 500mm x 200mm, The texture size is immaterial now as we have got the correct aspect rations. So. Scale U = 1 / 500 = 0.002
And Scale V = 1 / 200 = 0.005.
As can be seen, The text looks good with no stretching.



2). Size the object to fit the texture. (aspect ratio).
If one had a texture that cannot be resized in any way, without causing distortion. Then one may need to size the object so that the texture will fit, generally using the aspect ratio of the texture. A logo with text for example, may not be re-sizeable without distortion.

Example. We have a fancy logo that is 2500 pixels wide and 1200 pixels high. So, 2500 / 1200 = 2.0834 aspect (rounded up).

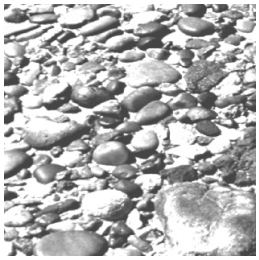
Our object is 800mm wide. This means we need our object to be, 800 / 2.0834 = 383.98 mm high (rounded down), to ensure correct aspect ration. Thus, the object and texture should fit together nicely.

Therefore our object is 800mm, so scale U = 1 / 800 = 0.00125 and scale V = 1 / 383.98 = 0.0026.

Minor variations will often be caused because of rounding of the calculations, as a cautionary measure, make the object height on the lower side. Hence rounding the object size down. If it was a touch out, we could change the scale V to 0.00259 for example

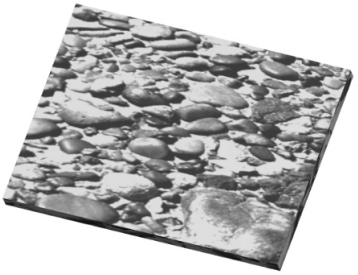
3). Close is good enough.
Many wrappings don't have to be exact, and whether the texture matches the object is of no concern. Marble for example. Has the impurities which give it the coloured effect. But assuming it is not man made, then every one is different. So it doesn't matter if the texture is stretched a bit.
Wood, stones, concrete, some leather, Brushed metal, rust, clouds, cork etc. All can be scales a small amount in any direction without too much detrimental effect.

Obviously sometimes there are limits, like scaling wood too much across the grain can look poor. But many times it is acceptable.



And for this method, the wrapping, is very simple as before.
For the width, divide 1 (U value) by the object width.
And height, divide 1 (V value) by the object height.

For example. One has an object 400mm x 300mm x 20mm, The stones texture is 512 x 512. So obviously it won't fit. But, as we are not bothered about whether it stretches a bit. The texture size doesn't matter.



The calculation is simply U = 1 / 400 = 0.0025 and V = 1 / 300 = 0.0033 (rounded)

4). Tiling (rough guess).
On many occasions image tiling is required. Many texture one creates, or downloads from the internet are 'tiled'. Which means the left side fits perfectly against the right , and same with top to bottom. Whilst this does mean repeating patterns, the same applies to a lot of things. Like wallpaper for example.

With many wrapped textures on large objects, it is not possible, or at least practical to avoid repeat patterns, There is little point in joining two images together in a paint program. Unless you specifically want to invert, or offset one of the images.

When deciding on tiling, there are two options. a). Decide on the amount of tiles you would like. Or. b). Use a fixed know element size within the texture. For 'b' see the next section - Scaling to a fixed element. For this section we will be scaling based on rough guess.

Tiling simply requires one to decide how many tiles are appropriate. For example, you have a 2000 x 1000 pixel image of wood grain (grain running the 2000px length). And wish to apply it to a 2500 mm round table top. Scaling the 2000 along the grain to 2500 is acceptable. But scaling the 1000mm up as far as 2500mm will unduly distort the image.

Luckily the top and bottom of the texture match, so it can be tiled without showing a join. But an actual answer as to now to proceed, is personal preference. And can sometimes be trial and error.

Here, as we are scaling the 2000 length, we can afford to scale the width a bit, so to avoid 3 joints, we will use two tiles. Effectively taking the 1000 to 1250.

For scaling the 2000 up to 2500. We simply use
U = 1 / 2500 = 0.0004
For 1000 width. We use V = 2 tiles / 2500 = 0.0008

It is totally up the the user now many tiles looks fine. Sometimes it may be necessary to find a starting point, like in this case 0.008, and adjust it either way until one is happy.

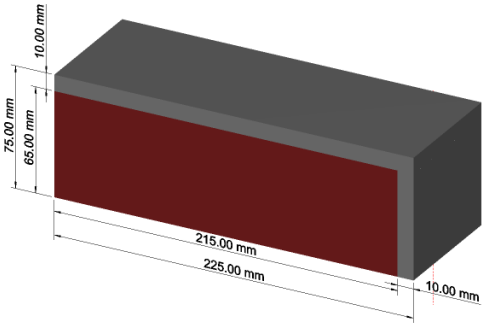
5). Tiling (fixed element - real world).

By fixed element it is meant a single known sized item, withing multiple items contained in a single image, For example bricks, tiles, chequer, grid etc. Taking bricks as an example, an single image may contain 7 bricks in width, and 20 bricks in the height, with mortar in between. Therefore a fixed element would be one brick + mortar.

This scenario doesn't rely on image size, nor object size. It relies on the real world dimensions of the element. **Note.** Using real world sizes, assumes the drawing is completed at a 1:1 scale in model space. If you are scaling model space, for example 1:10. Then the real world brick size, would also have to be scaled down by a factor of 10.

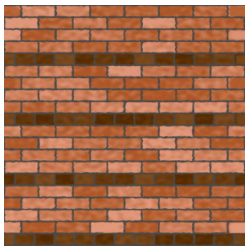
In this example, we will use a metric brick. Taking a rough dimensions of metric UK brick of 215mm x 65mm + 10mm mortar, and having our drawing space units as mm. We have 215 units wide, by 65 units tall.

Then allow for 10mm mortar, we get 225 x 75. Only include mortar once for top and one side.



Note. What is written below, may sound complicated at first. But it is very straightforward, and only takes a few seconds once you have done it a couple of times.

An important point. It will not be fully accurate, because we are guessing at the mortar size, and remember different textures may contain a different amount of brick wide and high.
For this example we will use the brick bond common texture. + don't forget to alter the bump map scaling as well as image map.

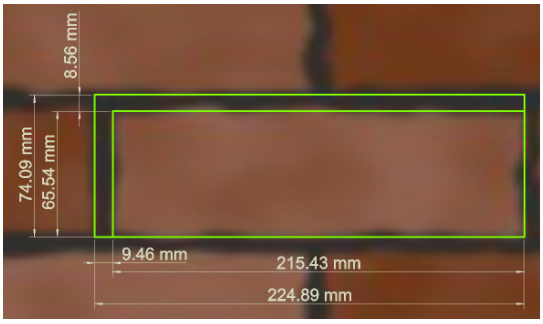


We know our texture has 7 bricks wide, therefore 225 units x 7 brick = 1575. And 75 units x 20 bricks high = 1500.

Using this information, we can say Scale U = 1 / 1575 = 0.0006349. And Scale V = 1 / 1500 = 0.000666.

We can easily check how close we were by measuring a brick on the wrapped object. As can be see I didn't estimate the mortar correctly. But its reasonably close.

Using this method ensures the bricks will be roughly the real world size irrespective of what size the wall or texture is. (assuming your wall is big enough to accommodate them).

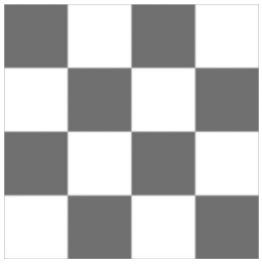


Using rough realistic brick texture, which has 8 bricks wide x 14 bricks high. And changing the space units. we have.

If the space units were Metres, it would be 0.225m x 8 = 1.8, And 1 / 1.8 = 0.555 in the scale U box.

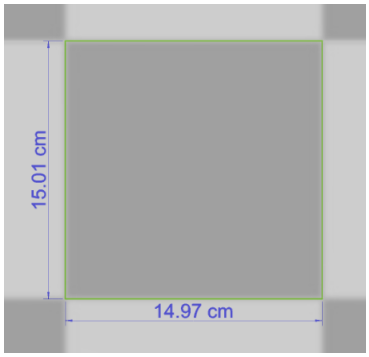
Using an template based on feet. and an American brick of 8 inch x 2 1/4 inch.
It would be 0.6666 ft and 0.1875 ft. + Mortar thickness of 0.03125 ft, gives 0.69785 x 0.21875. Therefore 0.69785 x 8 = 5.5825, Scale U = 1 / 5.5825 = 0.17912 and 0.21875 x 14 = 3.0625. Scale V = 1 / 3.0625 = 0.32653

A second much simpler example. Would be to use a checker pattern for lino floor (so no need for mortar).



The checker pattern is 4 x 4, which makes it nice and simple. Our space units were set to Centimetres (cm). And we want each individual checker square to be 15cm x 15cm.

Because the chequer pattern consists of squares, the same value applies to both U and V scaling. Therefore it would be. 15 x 4 = 60. and 1 / 60 = 0.01666. For U and V.



Therefore we type in scale U and scale V boxes 0.01666, to get roughly the correct sized checker pattern. As can be seer right, Its pretty close when wrapped onto a box.

American users with a drawing in feet, would be.
6 inch squares = 0.5ft….. 0.5 x 4 = 2….. 1 / 2 = 0.5 in the scale U and scale V boxes.

The above should work with any image, that has something inside ones knows the 'real world' dimensions of. Roofing tiles, floorboards, Grid, Mesh, Tread (or chequer) plate etc. AND one is drawing 1:1 scale. Remember, If using your own scale, you would have to take that into account when checking real world dimensions. For example a real world wall may get 8 metres, and you draw it 1200mm. Using real world brick sizes, just won't work. You would have to scale down the size of a brick to the scale of ones drawing. Or just use a different wrapping method.

One note regarding RedSDK.
RedSDK has a material property called texture space. Texture space allows one lot of control, not just over texture scale, but also orientation, and preset wrap modes. Texture space also allows for the scale to be entered normally. i.e. larger numbers = larger wrapping. Therefore if using texture space mode, none one the calculations here are required.

Wrapping on a face.

All the above apply equally to wrapping on a single face of a 3d object. However if wrapping a material on say the side face of a cube, in a specific direction. It will generally be necessary to create a separate material, orientated to ones liking.

One cannot wrap the same material on different faces with differing sizes and orientation. For that one may be better using the UV mapping tool, if it is available in your tc version..

There are probably more options. But 5 is plenty for now.

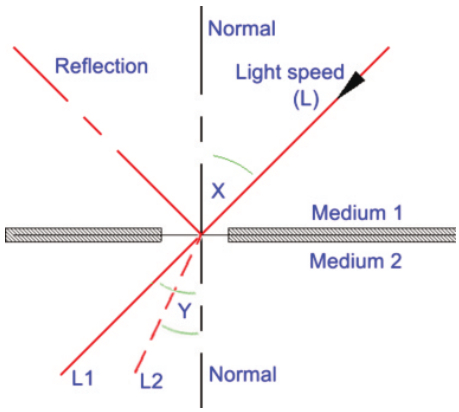
Appendix 3

IOR.

Important note - I am not a scientist, information about IOR is obtained from the internet and experiments. The information is included here as a guide only.

Refraction index is the difference in velocity between light passing in a vacuum, and passing in any other substance. This difference in index, can also cause light to bend when it strikes at an angle to the substance.

Fig 1



Referring to the diagram Fig 1. Imagine a single beam of light (L) passing through a vacuum (Medium 1). It strikes Medium 2 at an angle.

If the two mediums are the same refractive index, the light would pass though at equal angles with respect to the Normal. and exit at (L1). Medium 2 would appear not to be there.

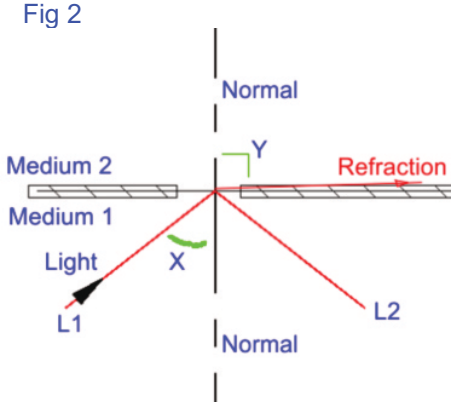
If Medium 1 was Air and Medium 2 was Water, and the light entered on the normal. There would be a slowing down of the light in terms of velocity, but no bending. As the angle of entry and exit would both be 90 degrees

If the light enters at an angle. Then as it is slowed down, there would be bending of the light (L2). This is due to water having a higher refractive index of 1.333, compared with Air 1.0003. Thus altering the angle (Y).

Different substances can have exactly the same refractive index, where the light will not be bent. There is an anomaly in the form of man-made materials, which have negative refractiion (not possible in nature).

These can be made to deliberately bend the light, in a predetermined way. If anyone wants to do some maths, to work out the refraction angle. The formulae is n (index) = Sine(Ø1) / Sine(Ø2). Or from the diagram n = Sine(x) / Sine(y). If x = 35 degrees. Sine(y) = sine(35°) / 1.33 (Index of water). Which translates to, Sine(y) = 0.57357644 radian / 1.33. Which leads to, Inverse Sine(0.38755165). And thus, Y = 22.205 degrees (after conversion). This is quite a bit less than the 35 degree light entry angle.

Fig 2



There is also something called the **critical angle.** This is where the light is passing through a denser medium, and exits into a less dense medium, (Example - going from water into air). There is a certain angle, which causes the light to become trapped inside its current medium. This is called total internal reflection, as it cannot pass into Medium 2. Think of optical fibres.

Each material has its own critical angle, for light passes through a dense material, into a less dense material.

Taking the picture on the right, Fig 2. Light passing from Medium 1 into the less dense Medium 2. Under normal circumstances, the light can exit. However when the light angle X, is such that the refracted angle, becomes greater than 90 degrees (Y). No more refraction can take place, and the light is reflected to L2.

This angle varies with materials but can be calculated.

The calculation is Sine(Øc) = Medium 2 / Medium 1. Or from the diagram Sine(Øc) = Y / X, (Y and X are the index of refraction for each Medium). As an example. Water to air (taken as 1.0003 and

1.333 index respectively). Transposing the calculation using Inverse Sine, becomes Øc = Inverse Sine(1.0003 / 1.333) = 0.851251 radians. Giving us 48.62613 degrees. Therefore if angle X is greater the 48.6 degrees, the light will reflect, instead of refract (assuming perfect conditions).

Of course light is never just a single beam, it is bouncing around all over the place. In the case of optical fibre it is specially manufactured to cause this total internal reflection.

A last note on objects like diamonds. The reflective index is so great (around 2.4), that it produces a critical angle of just 24.43 degrees, when compared against air. Therefore as light enters, very little will escape through, and most will be internally reflected. Until such time as it finds the angle to escape, (this can depend on how the diamond has been cut). It is the reason why diamonds appear differently, than other gems in the same light conditions. (*All the above information obtained from the internet).*

Sample IOR values.

Note. Figures obtained from various site on the internet.

Aluminium	1.39 - 1.44
Brass	
Bronze	1.18
Chromium	2.97
Cobalt	1.71 - 1.97
Copper	1.10 - 2.43
Gold	0.16 - 0.47
Iron	1.51
Lead	2.01
Mercury (liq)	1.62
Nickel	1.08
Platinum	2.33
Silver	0.18 - 1.35
Steel	2.5
Titanium	2.16
Tungsten	
Agate	1.544
Agate, Moss	1.540
Asphalt	1.635
Calcite	1.486
Chalk	1.510
Coral	1.486 - 1.658
Feldspar	1.525 - 1.539
Feldspar, Labradorite	1.565
Formica	1.47
Ivory	1.540
Mylar	1.65
Nylon	1.53
PET (g)	1.57 - 1.575
Plastic	1.460 - 1.55
Polycarbonate	1.584 - 1.586
Polystyrene	1.55
Rubber, Natural	1.5191
Silicon	4.01 - 4.24

Acrylic glass	1.490 - 1.492
Amber	1.539 - 1.546
Amethyst	1.532 - 1.554
Aquamarine	1.567-1.590
Crystal	2.00
Diamond	2.419
Diamond	2.417
Emerald	1.560 - 1.605
Garnet, Almandine	1.760
Glass	1.51714
Glass, Arsenic Trisulfide	2.04
Glass, Albite	1.4890
Glass Crown (impure)	1.485 - 1.755
Glass Crown (pure)	1.50 - 1.54
Glass Flint (impure)	1.523 - 1.925
Glass Flint (pure)	1.60 - 1.62
Glass, Fused Silica	1.459
Glass, Pyrex	1.474
Jade, Jadeite	1.64 - 1.667
Jade, Nephrite	1.600 - 1.641
Moonstone	1.518 - 1.526
Onyx	1.486
Opal	1.430 - 1.460
Pearl	1.530
Plexiglas	1.488 - 1.51
Quartz	1.544 - 1.553
Quartz, Fused	1.45843
Ruby	1.757 - 1.779
Sapphire	1.757 - 1.779
Sapphire, Star	1.760 - 1.773
Topaz	1.620 - 1.630
Turquoise	1.610
Zirconia, Cubic	2.170 - 2.21

Air	1.0003
Argon	1.000281
Carbon Dioxide (gas)	1.000449
Helium	1.000036
Hydrogen (gas)	1.000140
Hydrogen (liq)	1.0974
Oxygen (gas)	1.000276
Oxygen (liq)	1.221
Nitrogen (gas)	1.000297
Nitrogen (liq)	1.2053
Vacuum	1.0000
Acetone	1.36
Alcohol	1.329
Beer	1.345
Carbonated Beverages	1.34 - 1.356
Citrine	1.532 - 1.554
Chlorine (liq)	1.385
Ethanol	1.36
Fruit Juice	
Glycerin	1,367 - 1.473
Glycerol	1.4729
Honey,	1..484 - 1.504
Ice	1.309 - 1.31
Methanol	1.329
Milk	1.35
Oil of Wintergreen	1.536
Oil, Clove	1.535
Oil, Lemon	1.481
Oil, Neroli	1.482
Oil, Orange	1.473
Oil, Safflower	1.466
Oil, vegetable (50° C)	1.47
Rum, White	1.361
Vodka	1.363
Water (0° C)	1.33346
Water (20° C)	1.33283
Water (100° C)	1.31766
Whisky	1.356

Appendix 4

Caustics.

Caustics are focused particles of light, due to reflections and refraction's in transparent / translucent materials. Clear Glass is the obvious example. This is apparently one of TLux (Luxcore's) strengths. The actual effect can be quite varied. From the normal white light caustic effect, to rainbow effect when used with laser.

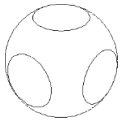
Although this is not actually necessary, the general advice from Luxcore is to enable the metropolis engine (Options menu - Turbolux engine). And to use Bidirectional path render in Camera properties - quality render.

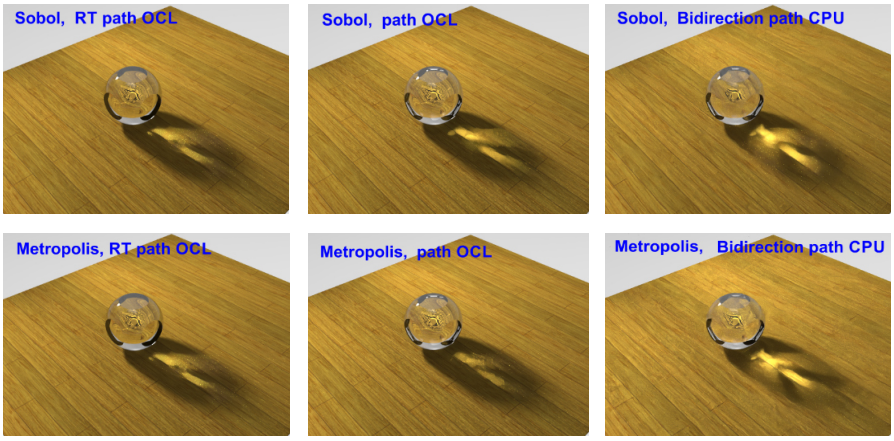
As can be seen below. Bidirectional does produce a more balanced look to the caustics.

For the images right.
The caustics are generated though the glass object.

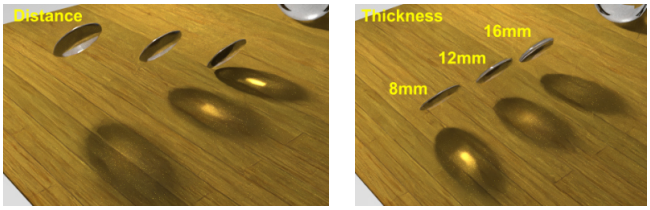
The light source, comes from an emission texture, applied to an object, (object not shown in the images).

The glass object is a sphere, with 6 flat faces cut away.

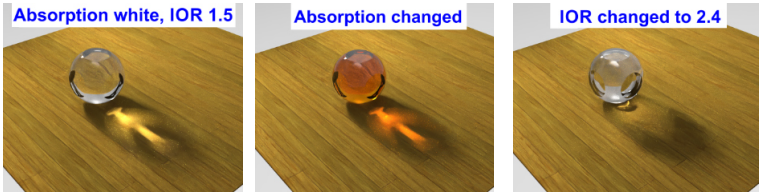




Just like real life, distance makes a difference to the effect. As does curvature and thickness of the object.



Caustics are affected by many settings, Including Absorption colour, IOR, scattering etc. (these settings are altered for the glass object).



Obviously the shape of the object has a great influence on how the caustics look. There is no one setting works for all occasions. It is simply trial and error until one gets the desired effect.

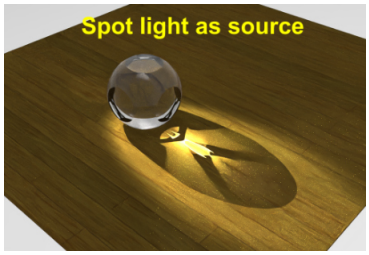
In RedSDK caustics are turned on / off, using a simple tick box, TurboLux does not have such a control (at the time of writing). It is on by default. Therefore.

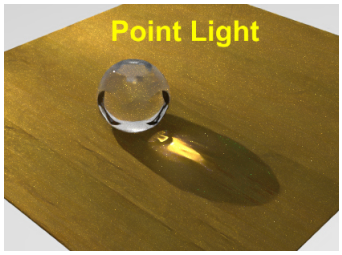
What if you don't want caustics.

Whilst one can play with IOR, Absorption, etc. the simplest way is to turn down shadow transparency, in the material common properties. Even just turning a tough away from black, will normally shut off caustics.

This has not been tested on every setting. But it works with many setting.

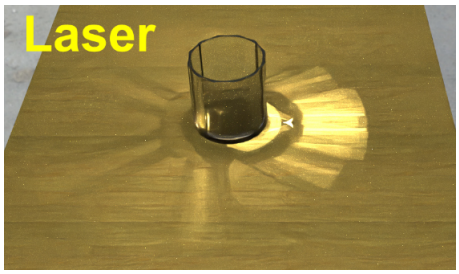
Although the previous page's images used a material emission as the light source (not shown in the images). The caustic effect works well with other luminance.





Lasers can also be used to create an effect

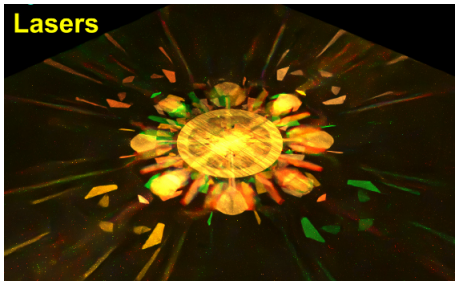
The image right uses a single laser.



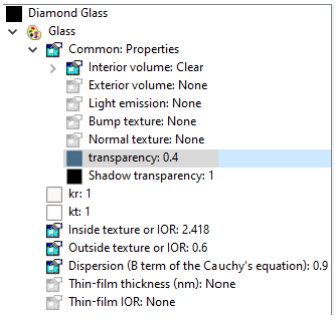
This image uses 5 lasers, four at 90 degree, and one aimed straight down.

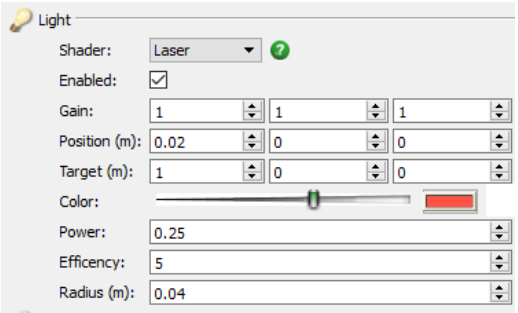
The object is a diamond, from TC 2022 samples folder.

Four lasers at 90 degrees around the diamond, and are are red, green, blue and yellow. The laser heading down is white.



The diamond and one of the laser light setups are shown here.





In blender one can also use light path tracing, but at the time of writing, it is not included in TurboLux (yet).

Appendix 5

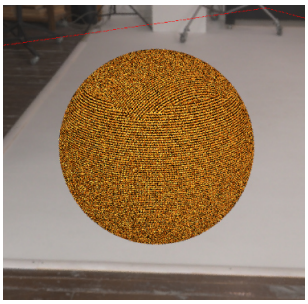
UV wrapping.

TC's UV wrapping can be used with TurboLux. The setup is pretty much the same as previous UV wrapping.

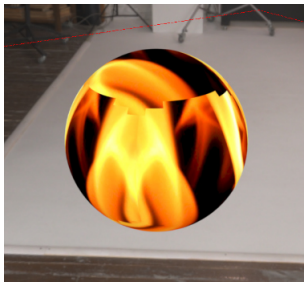
Because the scaling of the map is handled by the UV mapping tools, this simplifies the imagemap properties. In that the u and v scaling can be set to 1 (or -1).

NOTE. This is not a UV mapping tutorial, see the help files, or internet for that.

First a comparison image. This uses the traditional wrapped image (kd). And a u v scale of 1, in properties. As one can see at scale 1 there are lots of tiny wrapped images.

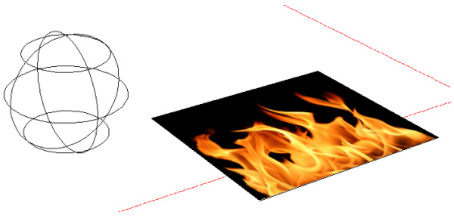


Altering the u v scale to make the wrap large (scale u & v = 0.003). Is trial and error. And doesn't really wrap good on a spherical object, This is a limitation of TurboLux material wrapping.



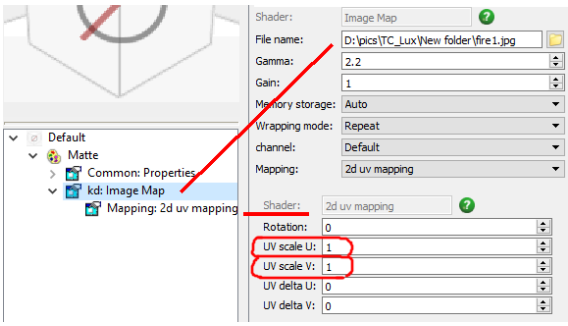
For the UV mapping tools.

First we will insert an image into the image list, and place in drawing.



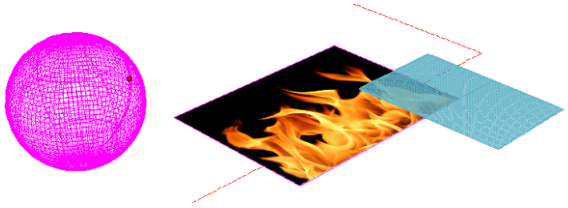
Whilst this can be done later, or using the UV wizard.

I created a default material and assigned the flame image. This was used in the first two images above. The U and V scales were set to 1.

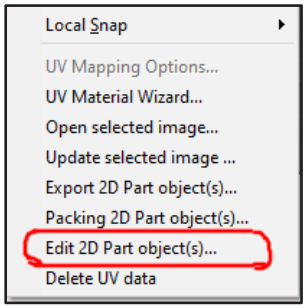


The UV mapping tool was selected.

Selecting the image, then the object, will display the blue uv map.

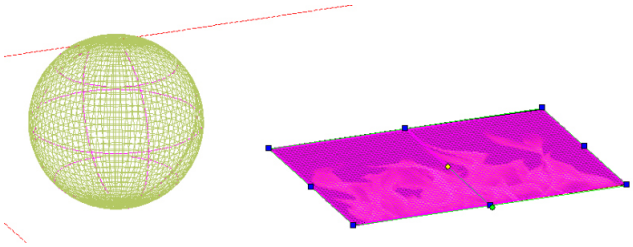


Edit 2d part was selected so the map can be positioned.

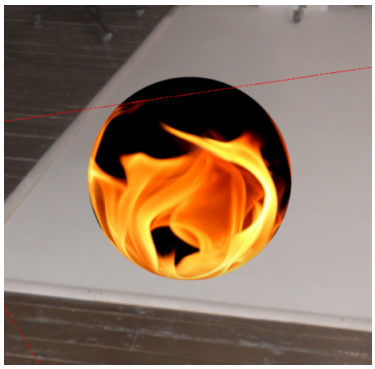


The map was edited to fit the size of the image.

And Finish edit 2d part object selected

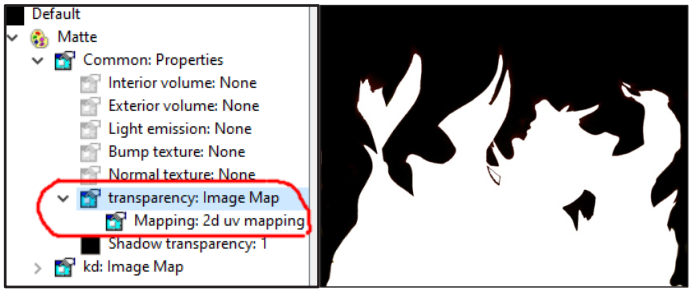


When rendered the flame wraps around the sphere. And will look a lot better than the standard imagemap wrapping.



It is also possible to add other effects, including adding a transparency map.

Transparency uses black as its transparent colour, therefore e create a black and white image, of what we wish the eliminate.



When rendered, the black parts of the transparency map are removed. Leaving just the flames in the finished render.

Appendix 6

Colour codes

Colour codes can be displayed in multiple formats.RGS, sRGB, HSV (HSB), HSL, CYMK, Hex etc.

Add to these are full decimal, 0 to 16777215. Sometimes used with animation colour adjustments, among other uses.

Also there is RGB 0 to 1. Used In some 3d rendering inputs, in Luxcore, Blender etc. These are generally approximated by dividing the RGB channel by 255. Each RGB channel is 256 colours, assigned from 0 to 255. 0 = Black, and 255 = white.

Often in TurboLux one can change the material property drop down box to show an RGB colour selector instead of having to enter numbers, but some things are pre-programmed, to accept 0 to 1. and do not have a colour selector, Fresnel colour for instance.

The following colour codes are not an extensive list, Also some programs may display the colours differently, and computer monitors may display colours differently.

The RGB colours are standard browser colours, except for the ones marked TC, these are from the TC colour palette.

One point to note, is that some colours have the same name, but are different colours. For example, Firebrick, Dark Green, blue violet etc.

Disclaimer. Whilst effort has been taken to ensure accuracy. Errors can occur. So the following is supplied as a guide only, with the best intentions.

C	Name	R,G,B (Decimal)			Luxcore (0 - 1)			Full decimal
		R	G	B	R	G	B	
TC	black	0	0	0	0.000000	0.000000	0.000000	0
TC	dim grey	84	84	84	0.329412	0.329412	0.329412	5526612
	dim grey	105	105	105	0.411765	0.411765	0.411765	6908265
	grey	128	128	128	0.501961	0.501961	0.501961	8421504
TC	dark grey	168	168	168	0.658824	0.658824	0.658824	11053224
TC	light grey / silver	191	191	191	0.749020	0.749020	0.749020	12566463
	light grey	211	211	211	0.827451	0.827451	0.827451	13882323
	gainsboro	220	220	220	0.862745	0.862745	0.862745	14474460
	white smoke	245	245	245	0.960784	0.960784	0.960784	16119285
TC	white	255	255	255	1.000000	1.000000	1.000000	16777215
	maroon	128	0	0	0.501961	0.000000	0.000000	128
TC	firebrick	142	35	35	0.556863	0.137255	0.137255	2302862
TC	brown	165	42	42	0.647059	0.164706	0.164706	2763429
	firebrick	178	34	34	0.698039	0.133333	0.133333	2237106
TC	orange	204	49	49	0.800000	0.192157	0.192157	3224012
	crimson	220	20	60	0.862745	0.078431	0.235294	3937500
	indian red	205	92	92	0.803922	0.360784	0.360784	6053069
TC	red	255	0	0	1.000000	0.000000	0.000000	255
	orange red	255	69	0	1.000000	0.270588	0.000000	17919
	light coral	240	128	128	0.941176	0.501961	0.501961	8421616
	salmon	250	128	114	0.980392	0.501961	0.447059	7504122
	light salmon	255	160	122	1.000000	0.627451	0.478431	8036607
	dark salmon	233	150	122	0.913725	0.588235	0.478431	8034025
TC	coral	255	126	0	1.000000	0.494118	0.000000	32511
	orange	255	165	0	1.000000	0.647059	0.000000	42495
TC	gold	204	126	49	0.800000	0.494118	0.192157	3243724
	dark golden rod	184	134	11	0.721569	0.525490	0.043137	755384
	golden rod	218	165	32	0.854902	0.647059	0.125490	2139610
	gold	255	215	0	1.000000	0.843137	0.000000	55295
TC	yellow	255	255	0	1.000000	1.000000	0.000000	65535
TC	khaki	158	158	94	0.619608	0.619608	0.368627	6200990
	olive	128	128	0	0.501961	0.501961	0.000000	32896
	dark khaki	189	183	107	0.741176	0.717647	0.419608	7059389
TC	golden rod	219	219	112	0.858824	0.858824	0.439216	7396315
TC	Medium goldenrod	233	233	172	0.913725	0.913725	0.674510	11332073
	khaki	240	230	140	0.941176	0.901961	0.549020	9234160
TC	dark green	47	79	47	0.184314	0.309804	0.184314	3100463
TC	medium sea green	66	110	66	0.258824	0.431373	0.258824	4353602
	dark green	0	100	0	0.000000	0.392157	0.000000	25600
	green	0	128	0	0.000000	0.501961	0.000000	32768
TC	forest green	35	142	35	0.137255	0.556863	0.137255	2330147
TC	sea green	35	142	107	0.137255	0.556863	0.419608	7048739
	medium sea green	60	179	113	0.235294	0.701961	0.443137	7451452

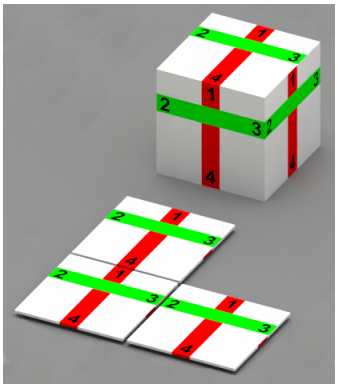
C	Name	R,G,B (Decimal)			Luxcore (0 - 1)			Full decimal
		R	G	B	R	G	B	
TC	dark olive green	79	79	47	0.309804	0.309804	0.184314	3100495
	dark olive green	85	107	47	0.333333	0.419608	0.184314	3107669
TC	medium forest green	107	142	35	0.419608	0.556863	0.137255	2330219
TC	yellow green	153	204	49	0.600000	0.800000	0.192157	3263641
	pale green	152	251	152	0.596078	0.984314	0.596078	10025880
	dark sea green	143	188	143	0.560784	0.737255	0.560784	9419919
	green yellow	14	219	112	0.054902	0.858824	0.439216	7396110
	light green	144	238	144	0.564706	0.933333	0.564706	9498256
TC	pale green	142	188	142	0.556863	0.737255	0.556863	9354382
TC	lime green	49	204	49	0.192157	0.800000	0.192157	3263537
TC	green	0	255	0	0.000000	1.000000	0.000000	65280
TC	medium spring green	126	255	0	0.494118	1.000000	0.000000	65406
	green yellow	173	255	47	0.678431	1.000000	0.184314	3145645
	medium spring green	0	250	154	0.000000	0.980392	0.603922	10156544
TC	spring green	0	255	126	0.000000	1.000000	0.494118	8322816
TC	medium aquamarine	49	204	153	0.192157	0.800000	0.600000	10079281
	medium aquamarine	102	205	170	0.400000	0.803922	0.666667	11193702
TC	aqua marine	112	219	147	0.439216	0.858824	0.576471	9689968
	aqua marine	127	255	212	0.498039	1.000000	0.831373	13959039
TC	dark slate gray	47	79	79	0.184314	0.309804	0.309804	5197615
	teal	0	128	128	0.000000	0.501961	0.501961	8421376
TC	cadet blue	94	158	158	0.368627	0.619608	0.619608	10395230
	light sea green	32	178	170	0.125490	0.698039	0.666667	11186720
	medium turquoise	72	209	204	0.282353	0.819608	0.800000	13422920
	turquoise	64	224	208	0.250980	0.878431	0.815686	13688896
TC	medium turquoise	112	219	219	0.439216	0.858824	0.858824	14408560
TC	turquoise	172	233	233	0.674510	0.913725	0.913725	15329708
	powder blue	176	224	230	0.690196	0.878431	0.901961	15130800
	light cyan	224	255	255	0.878431	1.000000	1.000000	16777184
TC	cyan	0	255	255	0.000000	1.000000	1.000000	16776960
TC	midnight blue	47	47	79	0.184314	0.184314	0.309804	5189423
TC	corn flower blue	66	66	110	0.258824	0.258824	0.431373	7225922
	midnight blue	25	25	112	0.098039	0.098039	0.439216	7346457
TC	navy	35	35	142	0.137255	0.137255	0.556863	9315107
	navy	0	0	128	0.000000	0.000000	0.501961	8388608
TC	medium blue	49	49	204	0.192157	0.192157	0.800000	13381937
TC	blue	0	0	255	0.000000	0.000000	1.000000	16711680
	royal blue	65	105	225	0.254902	0.411765	0.882353	14772545
TC	slate blue	0	126	255	0.000000	0.494118	1.000000	16743936
	dodger blue	30	144	255	0.117647	0.564706	1.000000	16748574
TC	dark turquoise	112	147	219	0.439216	0.576471	0.858824	14390128
	corn flower blue	100	149	237	0.392157	0.584314	0.929412	15570276
	light steel blue	176	196	222	0.690196	0.768627	0.870588	14599344

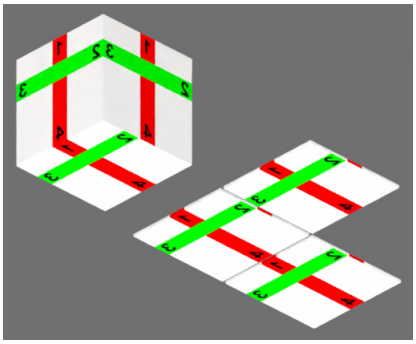
C	Name	R,G,B (Decimal)			Luxcore (0 - 1)			Full decimal
		R	G	B	R	G	B	
	deep sky blue	0	191	255	0.000000	0.749020	1.000000	16760576
	sky blue	135	206	235	0.529412	0.807843	0.921569	15453831
TC	light blue	191	216	216	0.749020	0.847059	0.847059	14211263
TC	steel blue	35	107	142	0.137255	0.419608	0.556863	9333539
TC	light steel blue	142	142	188	0.556863	0.556863	0.737255	12357262
	slate gray	112	128	144	0.439216	0.501961	0.564706	9470064
	light slate gray	119	136	153	0.466667	0.533333	0.600000	10061943
	steel blue	70	130	180	0.274510	0.509804	0.705882	11829830
TC	sky blue	49	153	204	0.192157	0.600000	0.800000	13408561
TC	violet	79	47	79	0.309804	0.184314	0.309804	5189455
	dark slate blue	72	61	139	0.282353	0.239216	0.545098	9125192
	indigo	75	0	130	0.294118	0.000000	0.509804	8519755
	slate blue	106	90	205	0.415686	0.352941	0.803922	13458026
	medium slate blue	123	104	238	0.482353	0.407843	0.933333	15624315
TC	dark slate blue	107	35	142	0.419608	0.137255	0.556863	9315179
TC	Maroon	142	35	107	0.556863	0.137255	0.419608	7021454
	dark magenta	139	0	139	0.545098	0.000000	0.545098	9109643
TC	blue violet	158	94	158	0.619608	0.368627	0.619608	10378910
TC	medium slate blue	126	0	255	0.494118	0.000000	1.000000	16711806
	blue violet	138	43	226	0.541176	0.168627	0.886275	14822282
	dark violet	148	0	211	0.580392	0.000000	0.827451	13828244
TC	dark orchid	153	49	204	0.600000	0.192157	0.800000	13382041
	medium orchid	186	85	211	0.729412	0.333333	0.827451	13850042
TC	medium orchid	147	112	219	0.576471	0.439216	0.858824	14381203
TC	magenta / fuchsia	255	0	255	1.000000	0.000000	1.000000	16711935
TC	orange red	255	0	126	1.000000	0.000000	0.494118	8257791
	deep pink	255	20	147	1.000000	0.078431	0.576471	9639167
	hot pink	255	105	180	1.000000	0.411765	0.705882	11823615
TC	orchid	219	112	219	0.858824	0.439216	0.858824	14381275
	violet	238	130	238	0.933333	0.509804	0.933333	15631086
TC	plum	233	172	233	0.913725	0.674510	0.913725	15314153
TC	thistle	216	191	216	0.847059	0.749020	0.847059	14204888
	medium violet red	199	21	133	0.780392	0.082353	0.521569	8721863
TC	violet red	204	49	153	0.800000	0.192157	0.600000	10039756
TC	medium violet red	219	112	147	0.858824	0.439216	0.576471	9662683
	light pink	255	182	193	1.000000	0.713725	0.756863	12695295
	pink	255	192	203	1.000000	0.752941	0.796078	13353215
TC	pink	188	142	142	0.737255	0.556863	0.556863	9342652
TC	sienna	142	107	35	0.556863	0.419608	0.137255	2321294
TC	Indian Red	79	47	47	0.309804	0.184314	0.184314	3092303
TC	Salmon	110	66	66	0.431373	0.258824	0.258824	4342382
	saddle brown	139	69	19	0.545098	0.270588	0.074510	1262987

C	Name	R,G,B (Decimal)			Luxcore (0 - 1)			Full decimal
		R	G	B	R	G	B	
	sienna	160	82	45	0.627451	0.321569	0.176471	2970272
	chocolate	210	105	30	0.823529	0.411765	0.117647	1993170
	rosy brown	188	143	143	0.737255	0.560784	0.560784	9408444
	peru	205	133	63	0.803922	0.521569	0.247059	4163021
TC	tan	219	147	112	0.858824	0.576471	0.439216	7377883
	sandy brown	244	164	96	0.956863	0.643137	0.376471	6333684
	burly wood	222	184	135	0.870588	0.721569	0.529412	8894686
	tan	210	180	140	0.823529	0.705882	0.549020	9221330
	moccasin	255	228	181	1.000000	0.894118	0.709804	11920639
	peach puff	255	218	185	1.000000	0.854902	0.725490	12180223
	wheat	245	222	179	0.960784	0.870588	0.701961	11788021
	blanched almond	255	235	205	1.000000	0.921569	0.803922	13495295
	antique white	250	235	215	0.980392	0.921569	0.843137	14150650
	linen	250	240	230	0.980392	0.941176	0.901961	15134970
	floral white	255	250	240	1.000000	0.980392	0.941176	15792895
TC	wheat	216	216	191	0.847059	0.847059	0.749020	12572888
	beige	245	245	220	0.960784	0.960784	0.862745	14480885
	corn silk	255	248	220	1.000000	0.972549	0.862745	14481663
	light goldenrod	250	250	210	0.980392	0.980392	0.823529	13826810
	light yellow	255	255	224	1.000000	1.000000	0.878431	14745599
	ivory	255	255	240	1.000000	1.000000	0.941176	15794175
	honeydew	240	255	240	0.941176	1.000000	0.941176	15794160
	azure	240	255	255	0.941176	1.000000	1.000000	16777200
	misty rose	255	228	225	1.000000	0.894118	0.882353	14804223
	lavender blush	255	240	245	1.000000	0.941176	0.960784	16118015
	sea shell	255	245	238	1.000000	0.960784	0.933333	15660543
	snow	255	250	250	1.000000	0.980392	0.980392	16448255
	ghost white	248	248	255	0.972549	0.972549	1.000000	16775416
	alice blue	240	248	255	0.941176	0.972549	1.000000	12357262
	lavender	230	230	250	0.901961	0.901961	0.980392	16443110

Appendix 7

Seeding 3d object.





Material wrapping, depends on how the object was drawn, and on what work plane it had when it was drawn. This has nothing to do with the current work plane one may have set up since the object was drawn.

IMPORTANT…. I have no idea how TC was programmed.

There are a few ways software can wrap an image onto a 3d box, One way is to use a cross, just like wrapping up a gift. This is generally the way cube maps work.

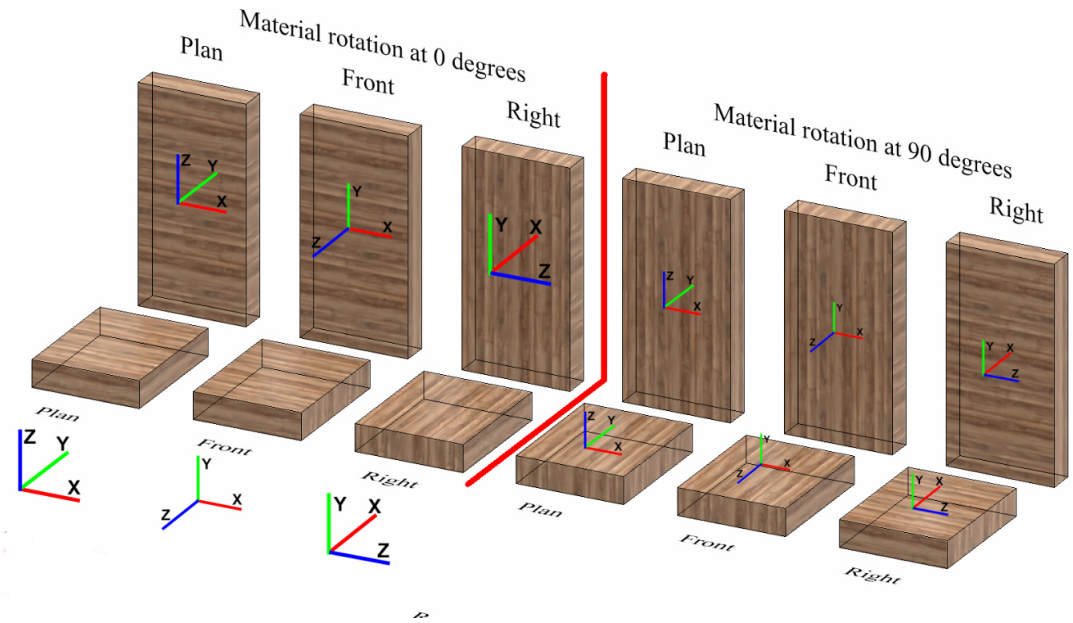
In TurboLux it appears to wrap three faces of a box as positive images, Top, Front, Right (first image on right). And the other three faces are a negative image of the first three (Bottom, Back, Left). Far right image.

This is not new, and appears to be the default way TC wraps. This is possibly why, in RedSDK and Lightworks there were different wrapping modes. To counteract this effect.

At the time of writing, TurboLux only has the one default mode. What this means is, if one intends to add a texture which contains text, one needs to use TC's UV mapping tools. And not texture wrap.

As stated previously, wrapping depends on how one made the object. The image below shows the same texture applied to the same objects, which differ because of the workplane they were drawn on. As can be seen using the x, y, z, axes indicators. None have been rotated, they were simply drawn differently.

As can be seen, none of them have the correct grain orientation, in a situation where one is drawn flat on the workplane, and one is drawn vertically to the workplane. Example. The upright rotated plan object, has good front and side direction. But the flat rotated plan object, has an incorrect side grain direction.



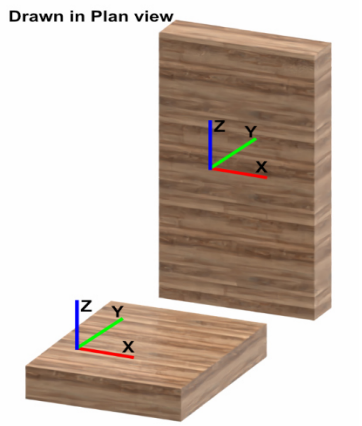
An Ideal situation is to draw everything flat on the world work plane, and rotate the objects to suit. Just like one would do in real life, when using plywood for example. This will ensure one only has to use one texture wrap to cope with all situations.

Life isn't always that simple, it is easy to forget and draw using the most convenient method. In these situations two texture s may be required.

Textures. My own personal preference, is to have the texture running top to bottom, as in the first image right.

For the next test we will assume the texture is running left to right (X direction). Image far right.

Scenario.

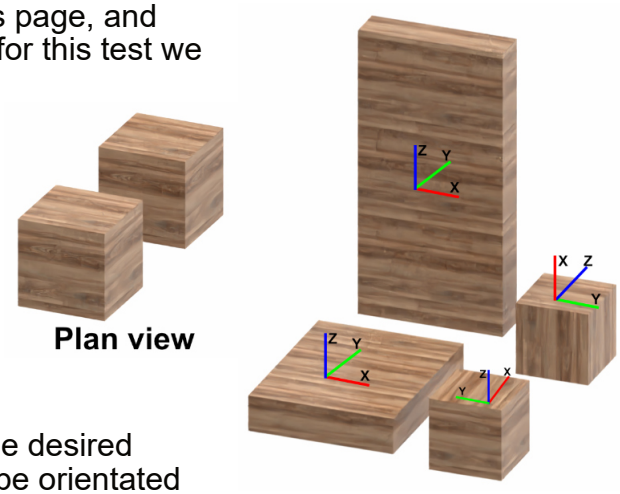


We have drawn lots of panels, and find just two of them incorrect. We know from a previous image, we can rotate the texture which will solve the vertical object, but we would have to create a third texture to also solve the flat object. Which we don't want to do.

For this we can alleviate all problems by seeding the object to force the internal plane reference, to the direction we choose, thus no extra textures are required.

To get both direction correct we need two seeds orientated differently. One for the vertical object, and one for the flat object.

We could just look at the images on the previous page, and orientate the seeds to the same directions. But for this test we will use boxes.

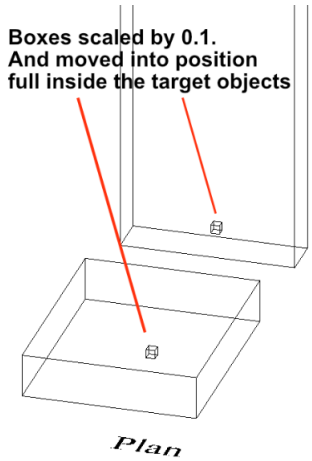


If we look at the two boxes (right). We can simply re-orientate them to the directions we want the grain to run.

Whilst in real life, a solid pieced of wood would have one of them end grain. When laminating (chipboard for example), we generally have long grain on all edges.

Far right, we have orientated the two boxes in the desired pattern. As can be seen the two boxes need to be orientated differently for each panel.

The boxes are the scaled down by 0.1 in all three scale boxes. And the boxes are them moved so that there completely inside the target object.



The last step is to add the box and target together. But first , ensure that options menu - advance preferences. 'Transfer material from operand on its facet' is NOT ticked.

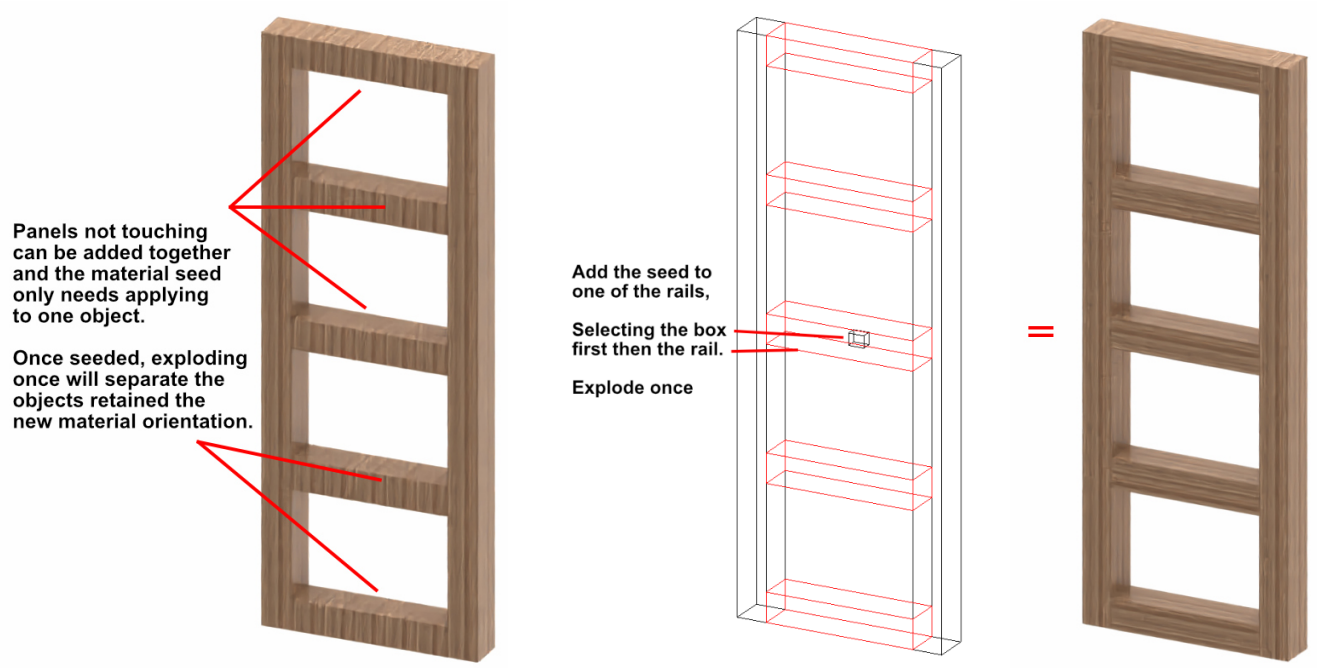
Select the 3D add tool. Select the box first, then select the target object.

The material x, y, z axes of the target, is changed to that of the box.

The final render is shown in the image right.

The same method can be used to orientate any object where the material direction is not what is required.

There may be occasions where many panels have incorrectly orientated materials. Providing two objects are NOT touching each other, one can boolean add the objects, apply the seed, then explode into separate parts.



Appendix 8

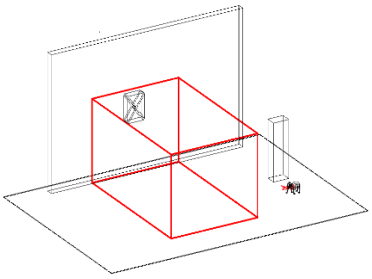
Volumetric.

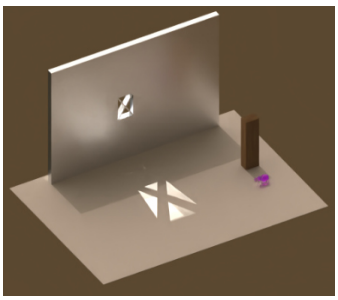
When one thinks of a volume effect, it would normally be done via options like clear volume, where the volume is inside a solid object like glass. And can change the absorption colour.

Volumetric effects can also be particles, like dust, mist, smoke etc. For this one would normally use the ‘Null’ shader, which avoids problems with reflection or specular affecting the results.

This first example is simple light entering through a hole, like light entering a dusty room.

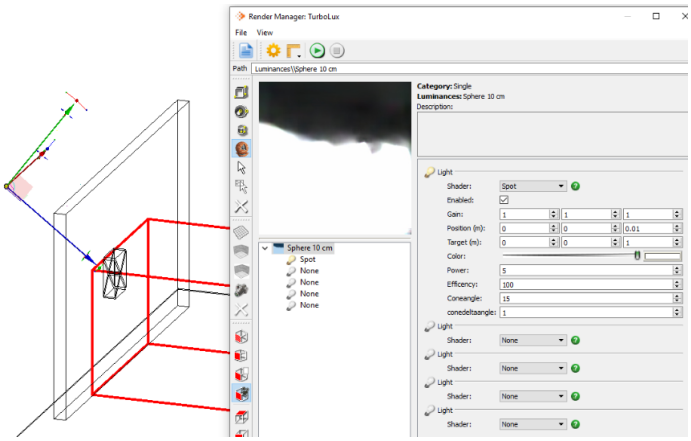
A box to capture the volume (shown red in the first image), is place into position between the hole and the target area. The target is shown in the second image, which is test rendered, before the box was inserted.





The lighting for this setup, is a spot light. Positioned outside the hole (window).

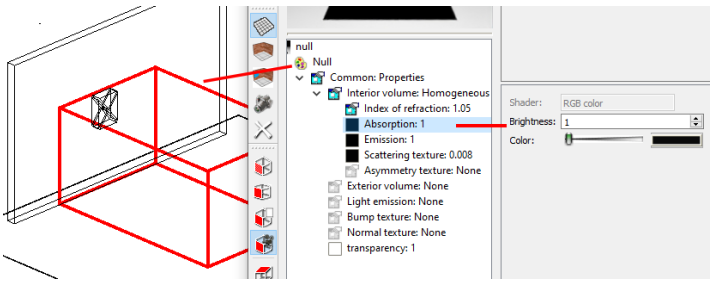
The spot light has quite a narrow beam, so it doesn’t shine over the wall.



The material is applied as Null. The internal volume is set to Homogenous, so that we have a scattering component.

Experimentation is required to get the desired effect, in any given scene. As other lighting may also affect the volume.

To my knowledge, there is no way to assign the effect, just to a particular light.

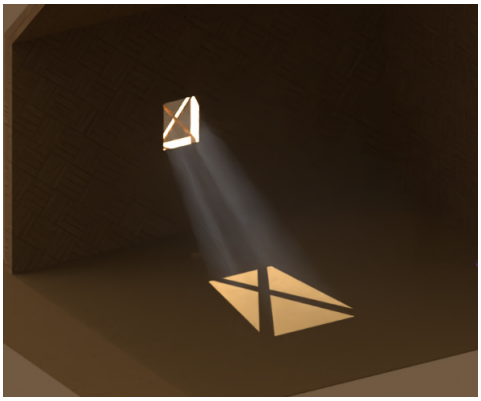


That said. Generally the effect in real life, would not occur, or be less pronounced, if there are multiple lights, of equal power.

An initial render of the scene will show if further adjustment is required.

For the render we use path tracing, (set in camera properties - quality render). It can produces a better effect than RT Path in some circumstances.

For the effect, Absorption is set to black. Scattering was set to 0.01. IOR = 1.05

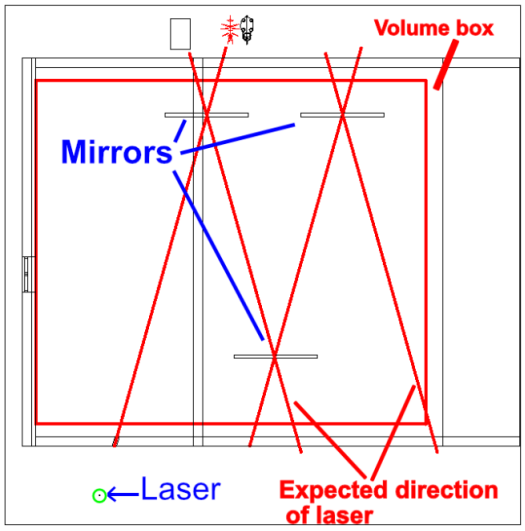


For the second test, this uses a laser reflected off some mirrors.

For this to work we need to use Bi-Directional CPU render mode.

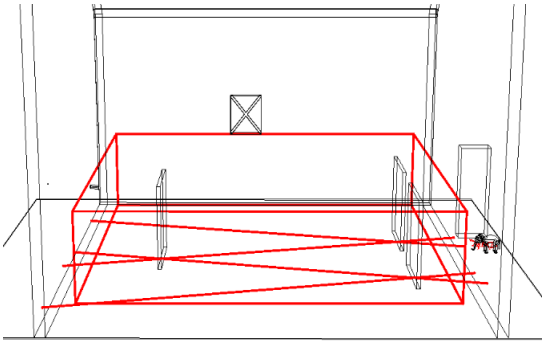
The scene uses 3 mirrors to reflect the laser beam. An environment map is used for overall luminance, With a single laser luminance providing the beam.

The volume is contained in a box object. And the whole thing enclosed in a housing. This was done simply to exclude some of the environment light.



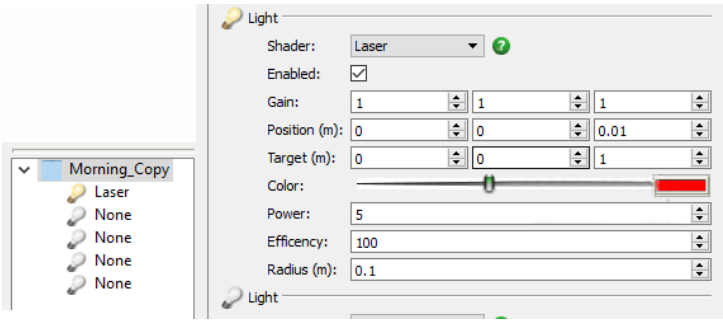
For the volume box (null material), The scattering is set to 0.005. IOR 1.05. And scattering to black.

The front view simply shows the setup from a different angle.



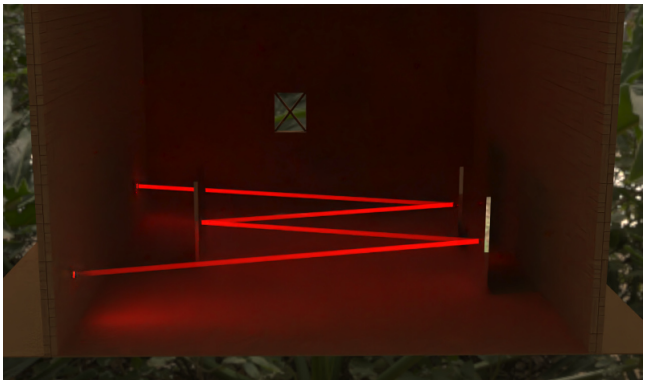
For the laser. It is a matter of ensuring the position and target are set correctly, with respect to the source object. A 20mm diam sphere with the +Z direction pointing at the first mirror.

Colour, Power, Efficency and Radius are personal preference.



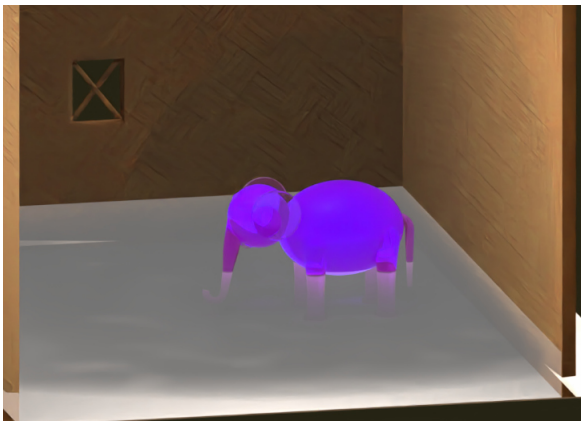
The finished render is shown on the right.

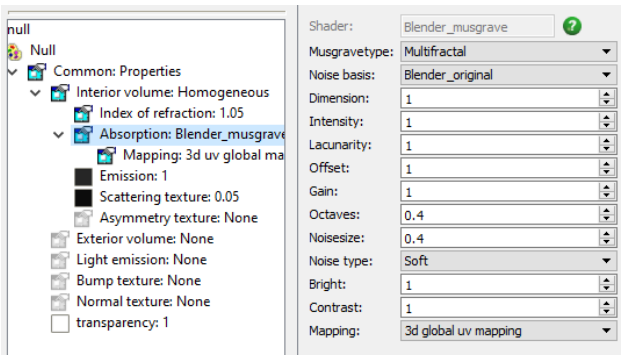
The laser can clearly be seen bouncing off the mirrors.



This Next test is simply lo level smoke inside a room.

The setup is similar to the others. The light was changed to a sphere light, and placed inside the room.



One main difference with the material setup, is the inclusion of Blender musgrave as the absorption.

This was done in an attempt to stop the absorption from simply being a solid look.