

Cycles quick settings guide: light paths

1-02-2014

Once again I immersed myself into the wild waters of rendering engine. In the past I have written a similar [tutorial](#) covering some settings in the great YAFARAY raytracer. This time I decided to examine 'CYCLES' which I have grown very fond of lately. It has been around for some time now and it is a great FREE option for an unbiased path-tracer. It has been neatly implemented into the Blender. Now, together with the blender's node system it is even more powerful CG tool! (I love you guys). Sadly, the documentation is not covering everything or at least not in the scale any user would ideally prefer. This should serve (again) as a manual (or part of the manual) for decision making during setting up of cycles render.

THE LIGHT PATHS

It was the "Light paths" Rollout in the "Rendering tab" that I was going to examine this time. I suppose learning what this setting does not only helps understanding the path tracing algorithm in general, but can also greatly reduce the rendering times once wisely used.

First some theory. The theoretical grid behind the engine is visualized [here](#) .There are four basic types of rays:

Camera

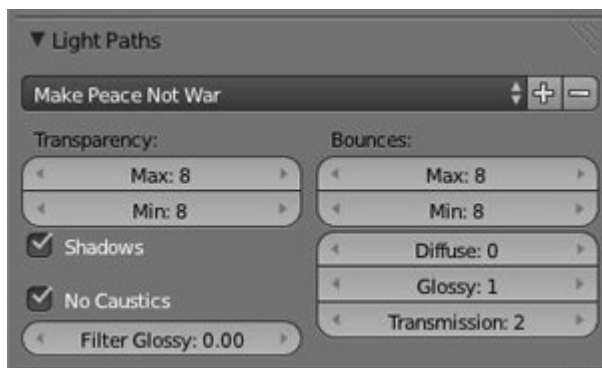
Reflection

Transmission

Shadow

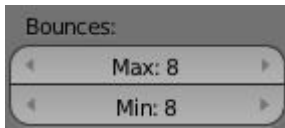
These rays are very closely connected to the type of material surface (or material's shaders). The shaders of a Material determine which rays would bounce off of the surface and the manner. *Camera rays* are those between the Object with the Material and the Camera. *Reflection rays* are those that hit the Material's surface and travel on. *Transmission rays* are rays that travel within materials that can transmit light (glass, translucent materials), and finally *Shadow rays* are called rays that examine areas that are blocked from a light source by another object.

The "Light paths" rollout control the number of bounces for certain types of rays. This is the default settings for direct lighting preset:



These values all control the behavior of the light rays in the scene. I will go through all of them separately.

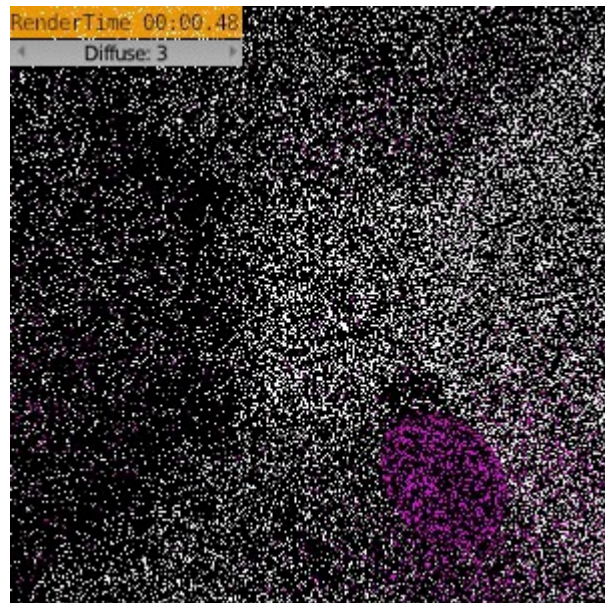
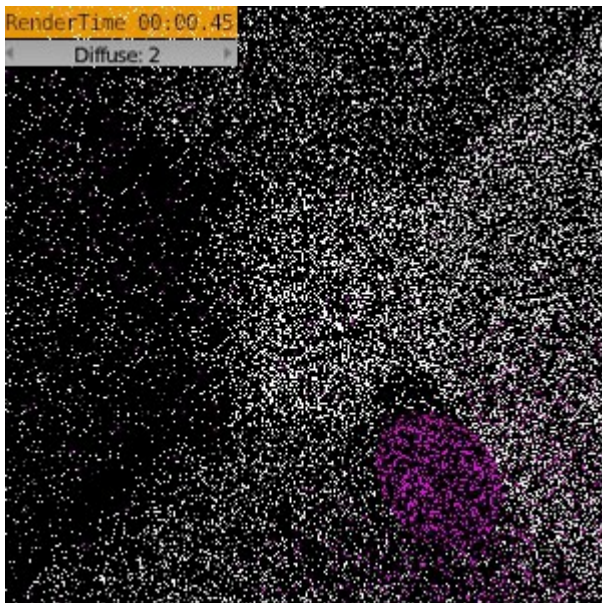
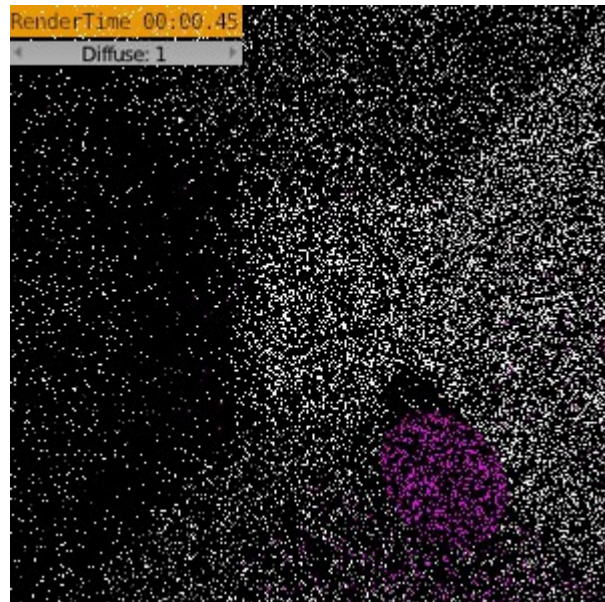
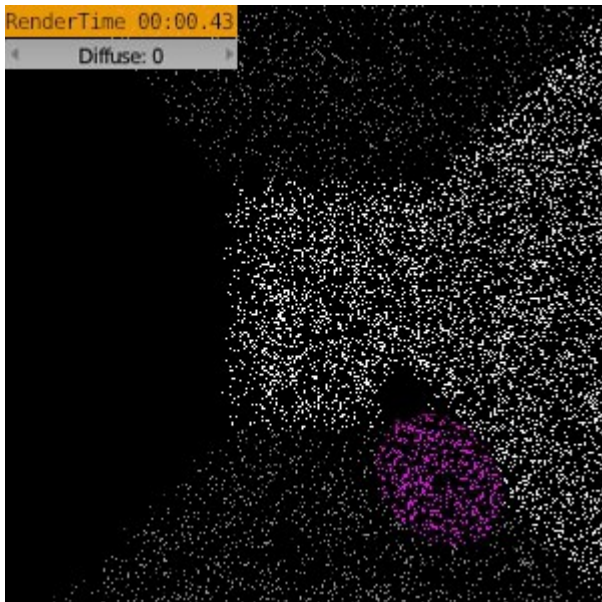
Bounces Max /Bounces Min(0-1024)

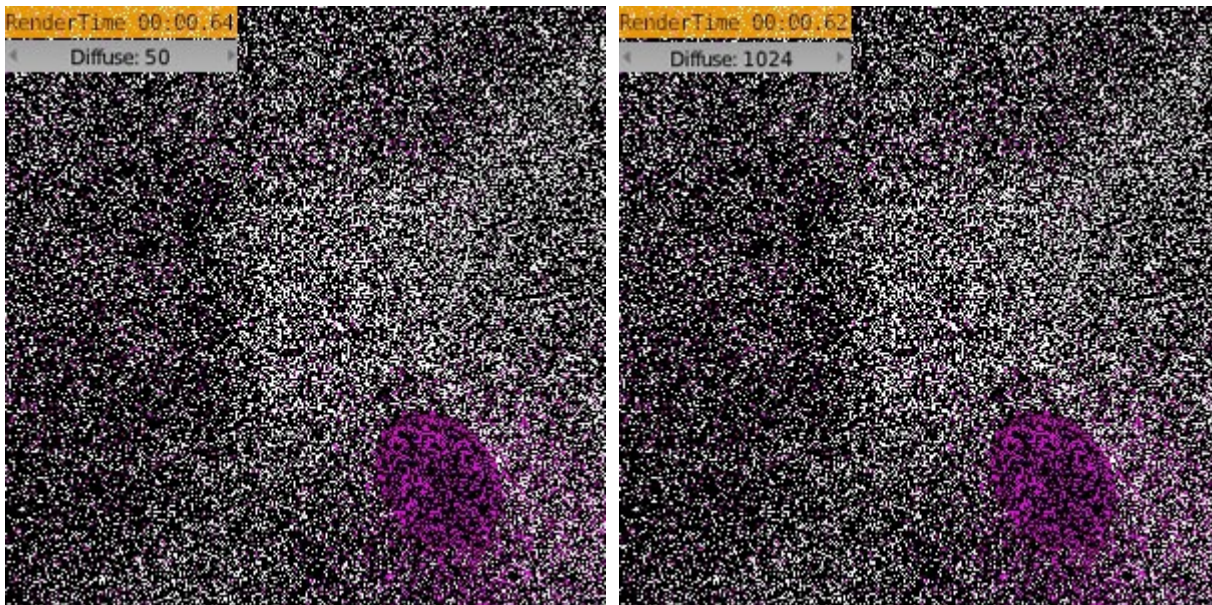


These values can be adjusted to limit the number of any kind of ray, be it diffuse, glossy or transmission. The Transparent rays are controlled separately. For example if the Max number is set to 10 there will be no more than 10 diffuse rays, 10 glossy rays and so on. The same applies for the minimum but there is not much to observe really in my humble opinion.

Diffuse (0-1024)

Diffuse is the basic color of a surface. The *Diffuse Ray* is carrying the information from the surface further to the environment. Obviously more distant objects would get less influence. The first image shows the result of the render with *0 Diffuse bounces*. In general this is referred to as *Direct Lighting*. That means there would be no bounced light occurring, no color bleeding whatsoever.

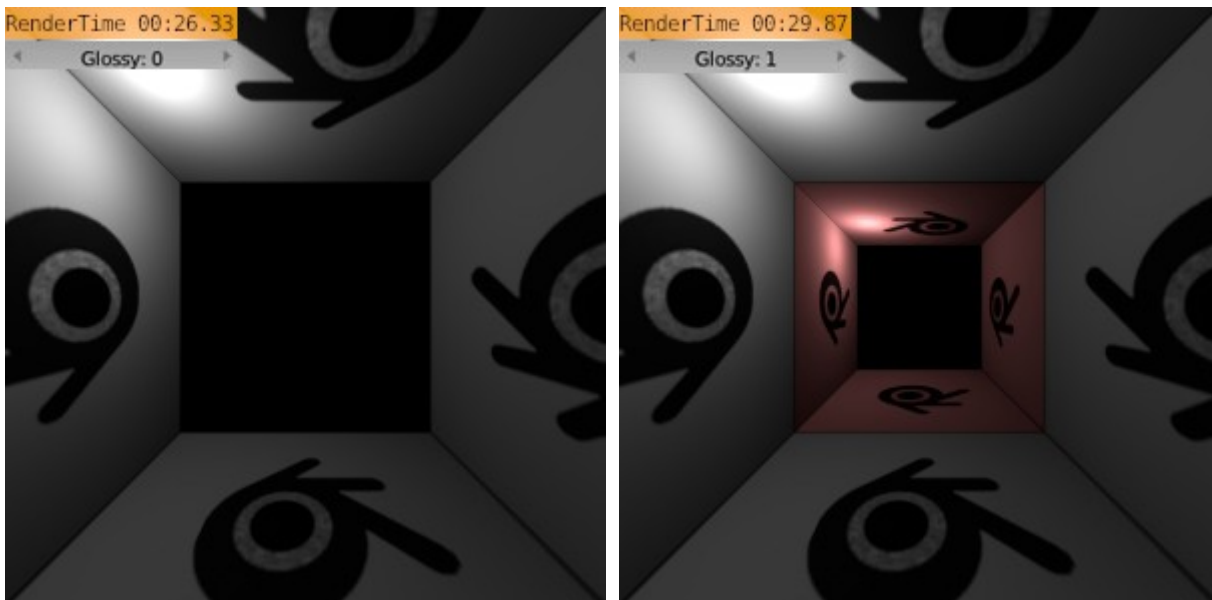


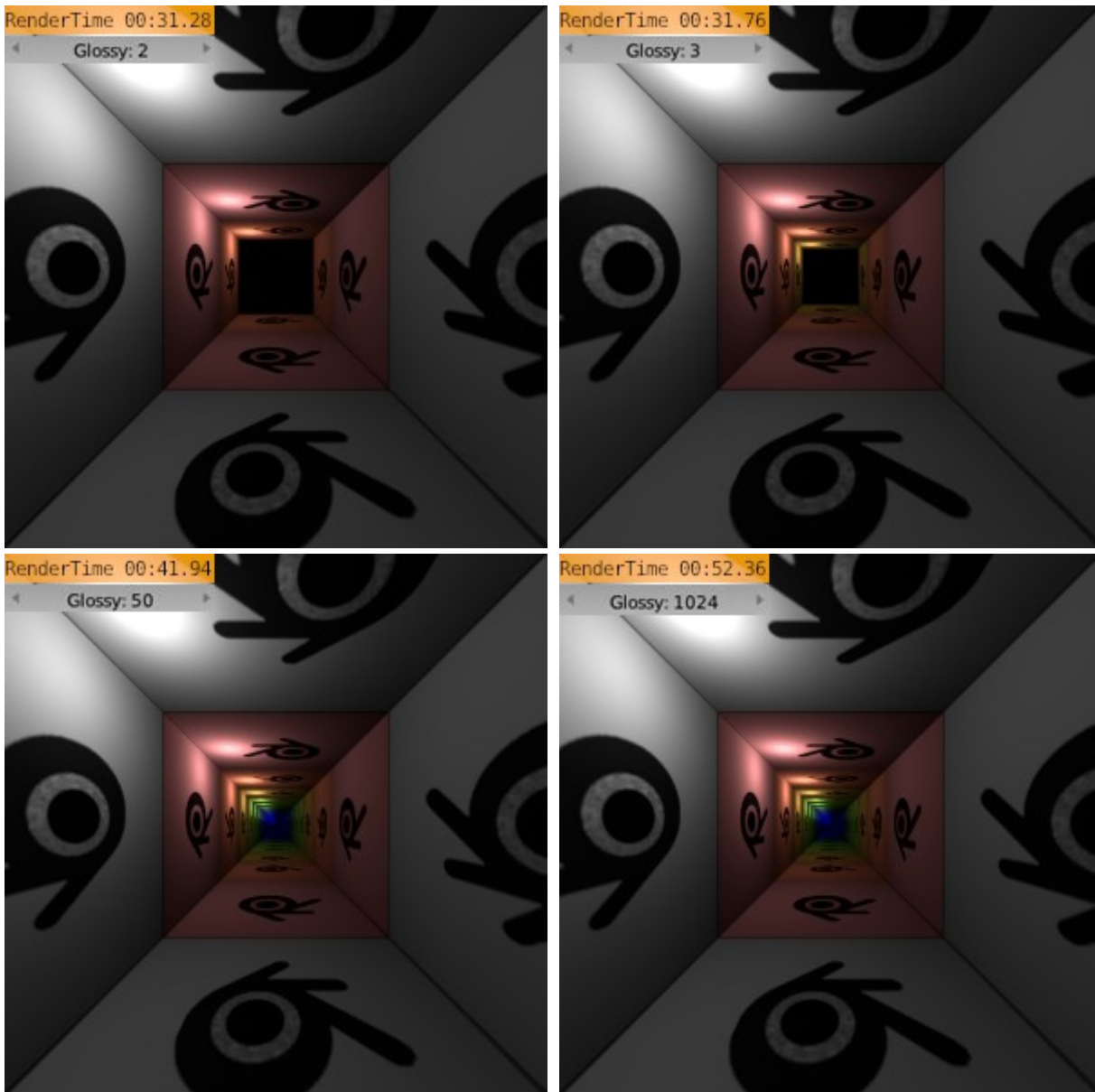


The Ray spurs from the Camera and hits the first surface which illuminates based on its parameters and position from the light source(s). But if this surface's material is based on diffuse shader only, the path is over. Any other case (*Diffuse bounces* ≥ 0) means the scene is being illuminated globally (GI). The more bounces of the light, the more precise the illumination is. But also less contrast there is and longer it takes to render. Also, for certain illumination interior light studies it might be very useful to keep the value low, because it produces contrasty scenes with greater difference between the light and the shadow areas.

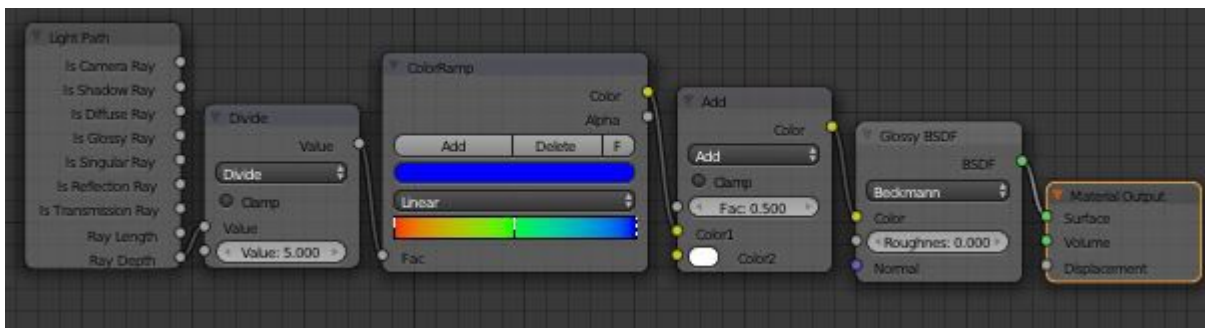
Glossy (0-1024)

The glossy is the ability of a material to reflect something that is in the surrounding of a stricken surface. To best illustrate this I used a box object as a simple room. For the two opposite sides (faces) that were in the direction of the camera I applied a full glossy material (a mirror). On the other surfaces I stitched blender logos as something to reflect and make reference. First thing to know is that any surface that has a glossy shader applied in its material, would render completely black when there are no glossy rays to hit it. The first image shows setting with 0 glossy bounces of any ray. And how it works? The ray starts its journey from the camera and when it hits the opposite glossy wall it stops and delivers the surface as completely black. When the setting is cranked up to 1 the rays continues on to make just one bounce and stops when hitting the opposite wall. And so on. Of course, interesting is to study the increase of the time.

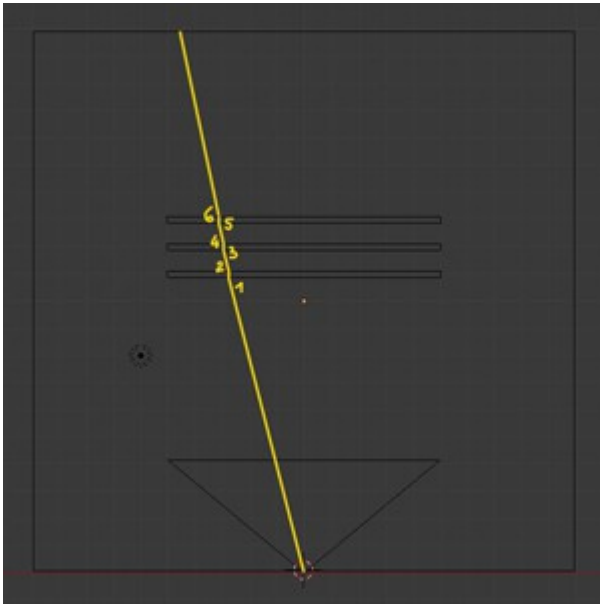




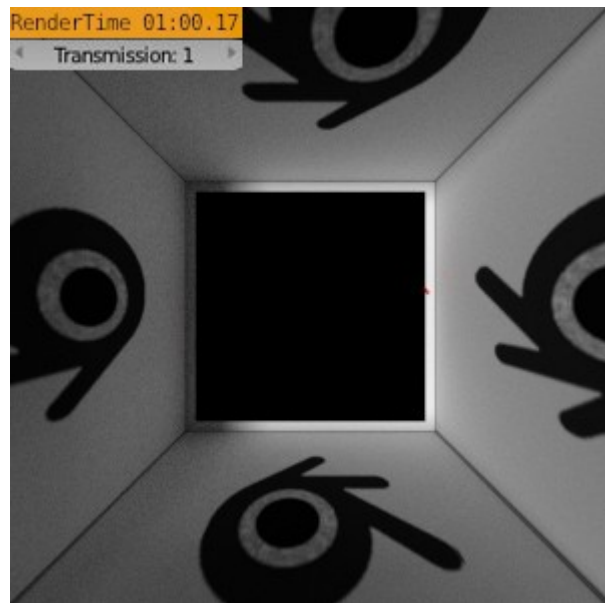
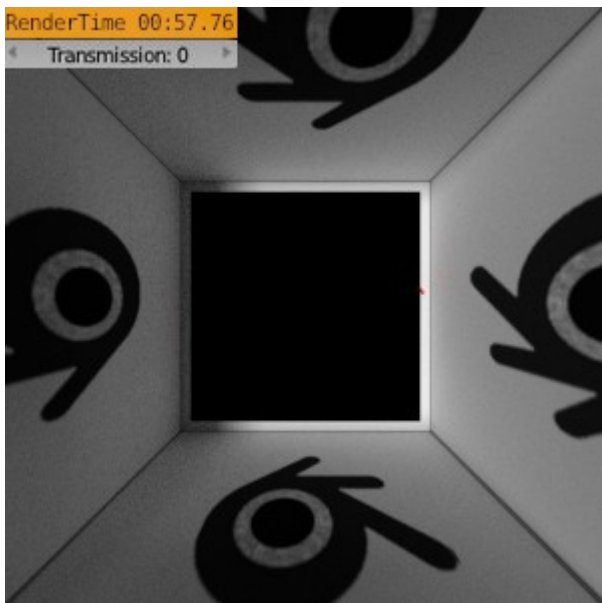
Note: Now you may have noticed all those rain-bow colors there in my tunnel. The node system includes nowadays an interesting feature. It is called a ray depth and it can be found on the Light path node. It is possible to separately access different bounces. I was going to test this on my example. For each of those reflections of the mirror I wanted a different color. So I added a color ramp node to map the numbers of the bounces to different colors. And tadaaa it worked! In order for the colors to be visible I had to add another Color correction node to make it lighter (Add).

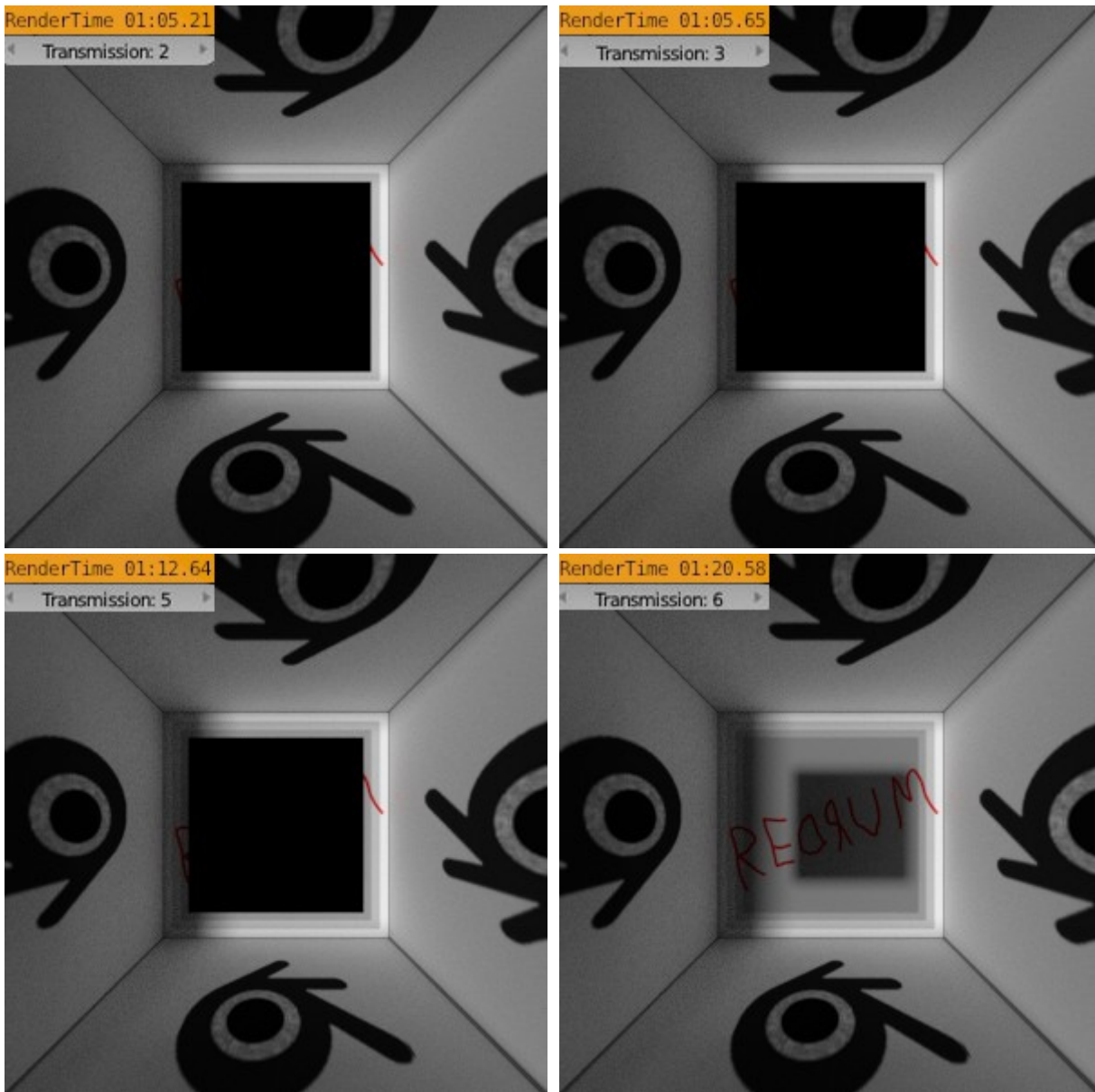


Transmission (0-1024)



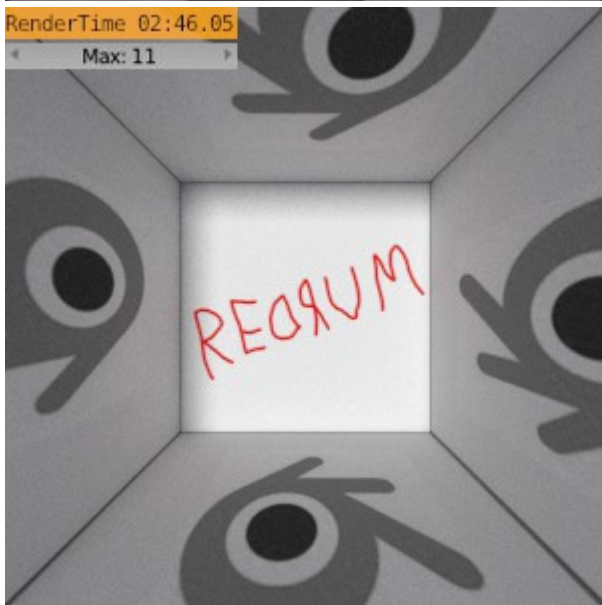
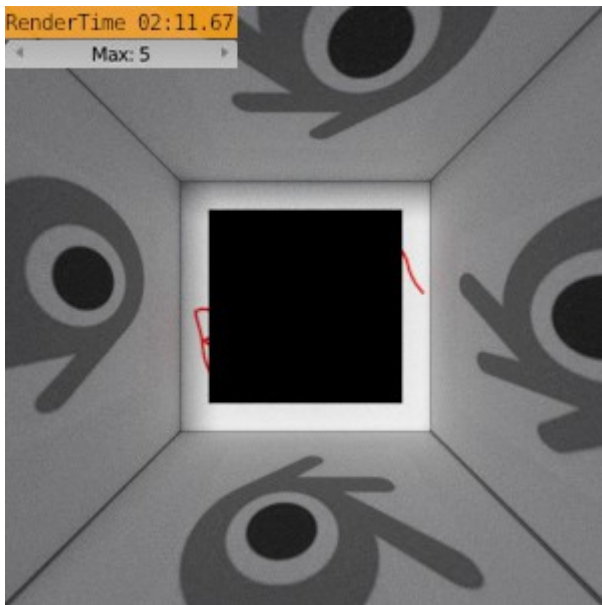
The transmission lever controls the overall amount of transmission bounces in the scene. Those include materials with glass, refraction and translucent shaders (not transparent shader - that is controlled via the transparent panel). For the example I selected a Halloween theme so I hope kids won't get too scared! I added three thin walls behind each other and added a glass material. When there is not enough needed *Transmission bounces* the Transmission Rays stop at the particular face and render it - well black. Since we have 3 walls each with 2 sides we need 6 bounces to make the back wall with the writing visible.



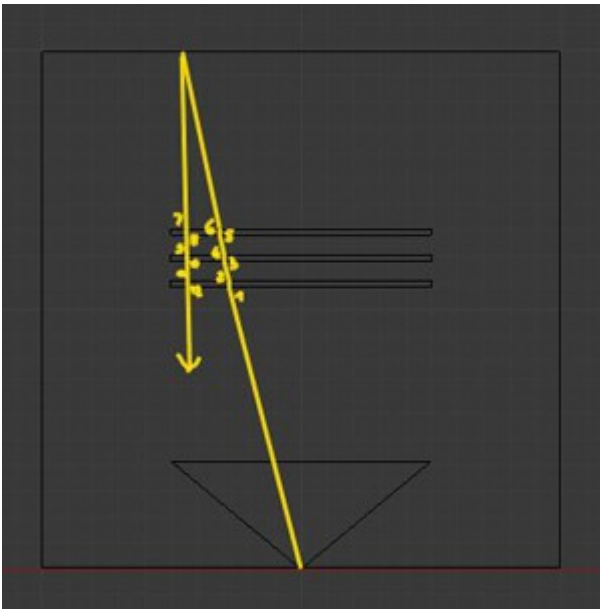


Transparency (0-1024)

Transparency panel takes control over the number of *transparent* bounces that take place in current scene. The principal is the same as at the Transmission (above) but this time it concerns the transparent shader. White color on a pure Transparent shader delivers completely transparent material. The change of the ray's direction or type does not take place in any way. The 'max' lever controls the maximum number of bounces through any surface that has a material with transparent shader component. So once again; if there are 3 walls, we have to set the bounces to 6 at least to see what is behind. Unfortunately I have not found out what the "min" button can be possibly used for. Apart from the decreased render time. I hope someone would tell me eventually!



Note: There is one more interesting thing to this. It is shown on the image with 6 maximum bounces. Why is there a shadowy shape on the wall when there are enough bounces to make all the obstacles transparent? It's because the transparent path bounces back off of the wall. It travels back, hits the first wall and since the maximum number of bounces is set to 6 it tells the renderer there is a black wall. But this time not for the camera rays but for the diffuse rays casting a shadow on the wall. Obviously the 'max' set to 11 makes the wall farther. 12 then is without obstacles already.



Shadow checkbox

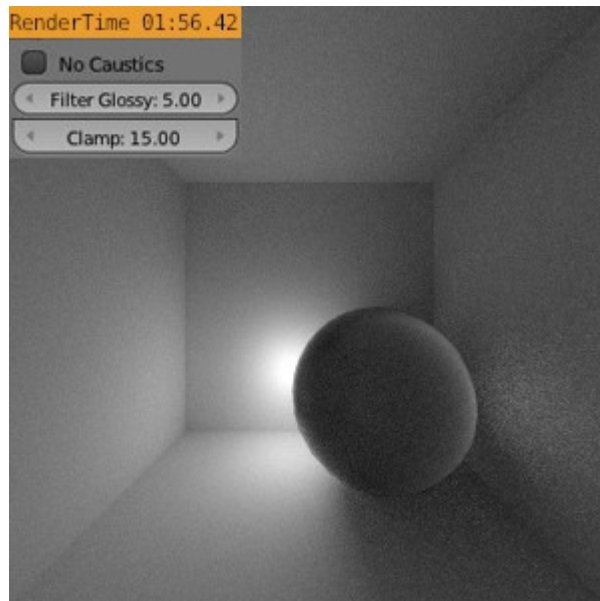
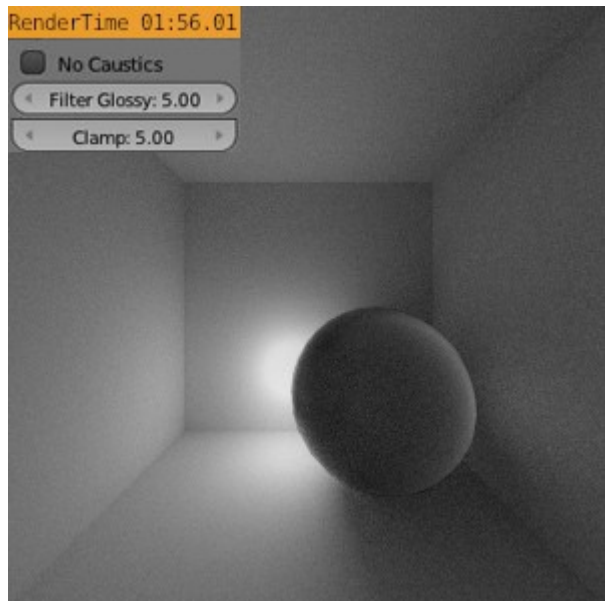
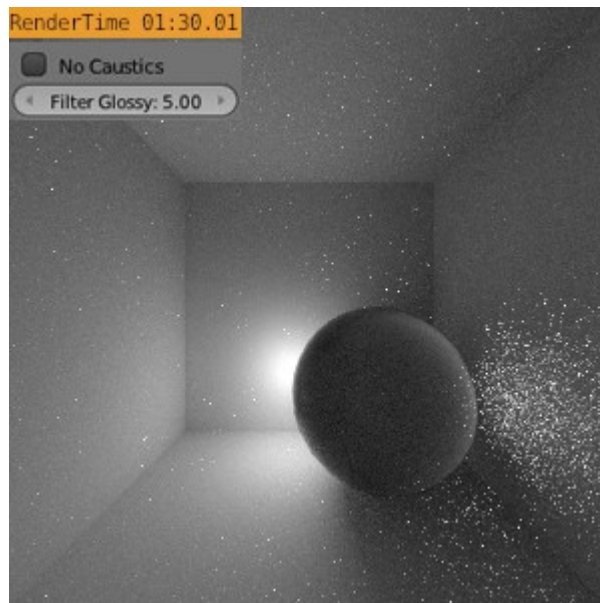
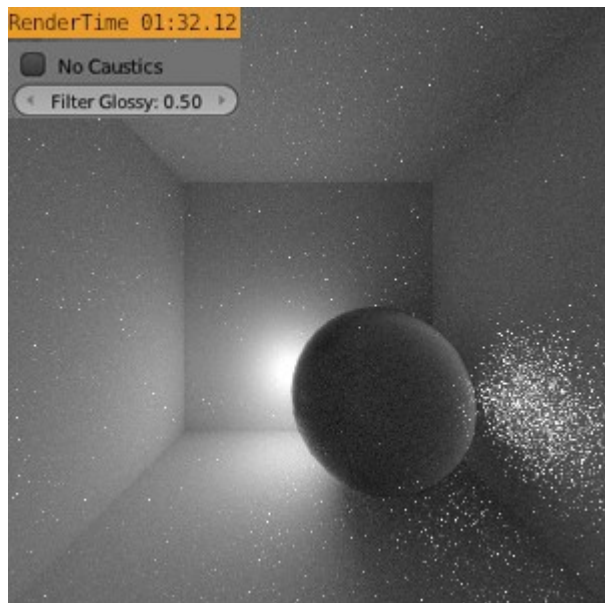
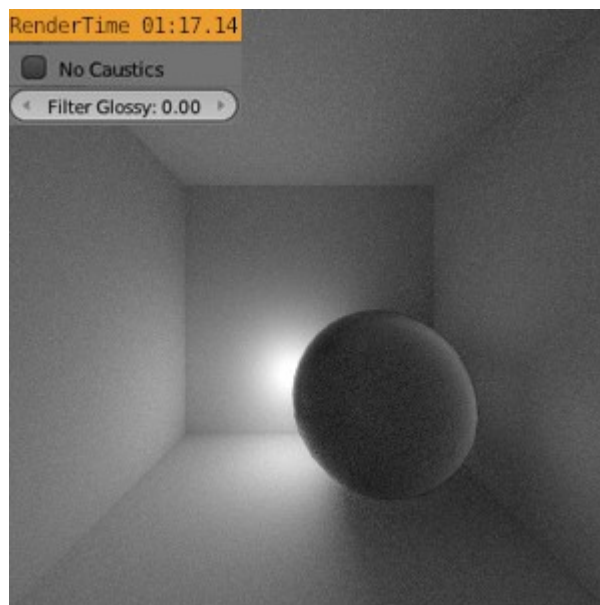
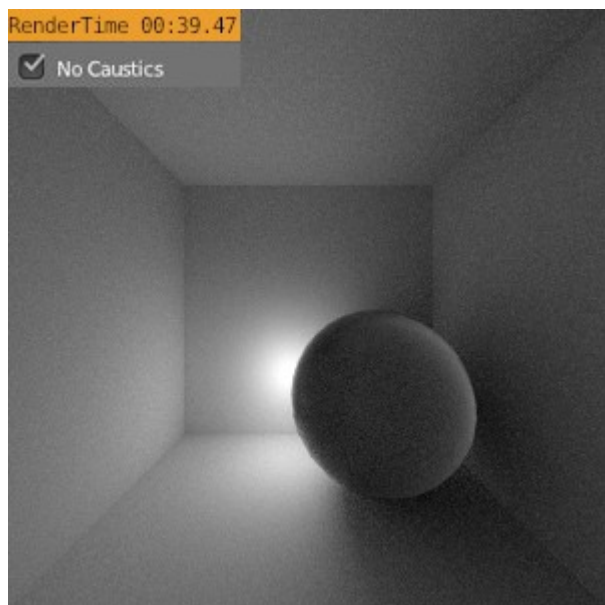
When shadow checkbox on it traces the shadows based on the transparency material settings (the *Shadow rays* go through) . When it is off it computes the shadow based on the geometry only.



No Caustics checkbox and Filter glossy (0-10)

Use this to disable the illumination through specular highlights and glossy reflections. The caustics effect can be generated by a glossy material(glass is also a glossy material). When the Diffuse rays accidentally find a shiny spot somewhere it jerkily adds brightness to a pixel. Which results in noise. And this button is primarily there to get rid off the noise in the scene. But as the life gives and takes it is not that easy. Sometimes we want the caustics effect in our scene. There is a very nice [tip](#) posted by *Sebastian Koenig* on the BlenderCookie using the Light path node. It takes advantage of the fact that we can exclude the *Glossy rays* from computation and use rather Diffuse rays instead. This allows us to keep the bounced light but cut the noise. For any other cases where we need the real caustics effect, we have to stick to noise reduction tricks and tips. There is a [wiki page](#) undertaking this subject.

To show these settings I used a box with a glass ball inside. Behind the ball I put a point lamp to produce some caustics on the wall to play with.

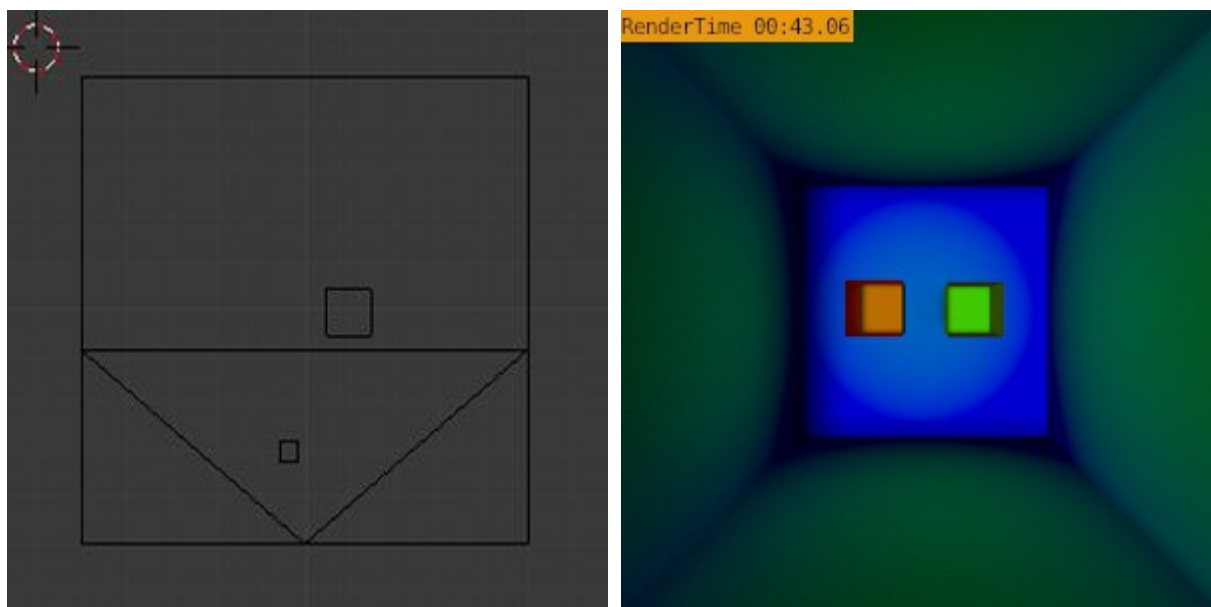


On the first image -- almost like a total eclipse of the sun!;) The caustics effect can be clearly seen on the second image above. The Filter Glossy blurring did not do a very good job this time. To lower the number of fireflies I had to Clamp the highlights for computing of each pixel. With some tweaking a pretty good results can be achieved.

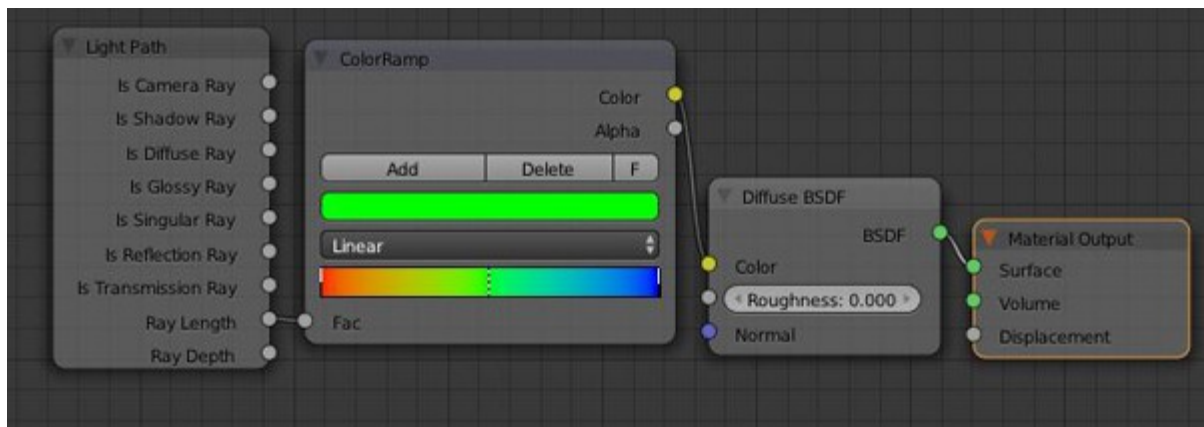
It came to my mind suddenly: why the hack there is so many fireflies right by the ball on the wall and much less elsewhere? Well... from any pixel in the scene there would be a glossy reflection somewhere on the ball of the light behind it. Unlike the computation of contribution to the pixel's brightness coming from the lights, this glossy spot is hit randomly by scattered rays. But there is a place, right on the wall where this bright spot is not reflected but magnified by the "lens" of the glass ball. Also the ball is much closer in this area and covers a huge part of the world for those poor tiny pixels on the wall. There is much bigger chance therefore that a ray hits the splotch on th ball. That is why there are so many fireflies.

Note: Light path length node tip

When I was playing around with the Light path depth node I was also curious how does the the Light path length node work too? And what is the outcome of the node? I placed a cube of size 1BU. And inside I added two smaller boxes and placed them in different distances from the camera(as shown on the image bellow).



For the material on all boxes I used the color ramp node plugged into a diffuse shader again to map some colors to values and visualize thus the node. The material color now shows the ray lengths. Look what I got! Awesome, isn't it? Looks almost like the predator's vision! :)



Well, that would be all for now. Hope this would make sense for some of you. Have fun and keep blending.