2018

# OCTANE POSER PLUGIN USER GUIDE

Wim van de Bospoort

Version 1

7/9/2018

# Contents

# 1 Introduction

## *1.1 What is Octane?*

Octane is a highly efficient raytrace render engine which makes full use of CPU and GPU architectures and is available for a variety of devices. It is currently running on Windows, MacOS, IOS, Android and Linux.

In this guide I will go through the steps of setting up Octane and the Poser plugin, guide you through the conversion process, explain the different concepts of Octane and how they relate to Poser. Then I will go into more detail on how to get the best results in Octane with regards to lighting, materials and render settings. Sometimes I will refer to later in the guide because the explanation will make more sense after you have learned other features of the render engine.

The last part of the guide will explain some of the additional features Octane will provide and which will expand the possibilities you now have within your poser scene.

Although the guide is oriented towards the Poser user, it can be used in part for other plugins as well, but terminology used may be different.

I wrote this guide because many of the people who start to use the plugin always come up with the same questions and with the expected influx of new users with the new free version of Octane 4 I expect many more will have the same questions.

I know it will be hard, but do not fall into temptation of loading up a scene and running the plugin and expect to have a nice Octane render. Read at least the first few chapters of this guide and save yourself a lot of frustration. It is not hard or complex, you just need to understand the workflow and steps you need to take.

I have been using Octane since 2012 and went from version 0.9 up to version 4. The guide is not intended as a replacement for the reference manuals from Otoy or the author of the plugin but more as a quick way to get going by answering the most common questions.

This guide is intended to be a living document and will be updated with new and updated material. If you have comments, please contact me at wimvdb@euronet.nl.

Many thanks to Otoy and to Paul Kinnane (the author of the Poser and many other plugins for Octane) for this wonderful piece of software.

## *1.2 Requirements*

### 1.2.1 Hardware

The hardware requirements will change in Octane V4 (wider support of hardware) but you will need a PC or a Mac with a Nvidia GTX 600+ CUDA enabled card, with at least 8GB memory. For large scenes and good render performance you will need a GTX 900 series card or better with at least 4GB VRAM and 32GB of memory on your PC or Mac.

The OS and other software will use the GPU as well, so it may be a good idea to have a dedicated card for GPU rendering.

### 1.2.2 Software

Windows 7 or higher, Mac OS 10 or higher, Poser 9 or higher, Octane V3.08 and Poser plugin 3.8.087 or higher. CUDA and drivers, latest ones.

### 1.2.3 Skill set

To be able to follow this guide a reasonable understanding of Poser is needed. A general knowledge of CGI is helpful, but not required. Google the terms if you do not understand them.

# 2 Getting Started

In this chapter I will guide you through the installation of Octane, the plugin, user interface and go into the terminology I will use in this guide. Octane and the plugin integrate quite well, but you need to have things set up correctly and understand the concepts. That is what this chapter is about.

## *2.1 Installing Octane and the Poser Plugin*

After downloading Octane and the plugin, you need to install them on your system.
The provided installer of the Octane Standalone package will install in the programs area of your Operating system. No user input is required. The Poser plugin will do the same but place its software in the Poser program folder (which is an Addon requirement within Poser). It will also install several useful scripts in the python folder.

*Although not needed, it is recommended to install the Octane standalone as well. The Standalone installer will update the registry settings which is needed for GPU rendering. The Standalone version will also provide additional features which you might need later.*

There are a couple of additional settings you need to adjust to make things easier and more future proof (such as default folders) but I will go into that later in this chapter when you know where to find them.

## *2.2 The User Interface of the Plugin*

The Octane plugin for Poser is a Poser addon which makes the integration with Poser possible. It communicates with Poser via python, takes care of synchronizing the Poser scene with Octane, takes care of much of the conversion process and it provides tools/macros to improve the materials of the props and figures in the scene.

You start the plugin from the Render menu: OctaneRender.

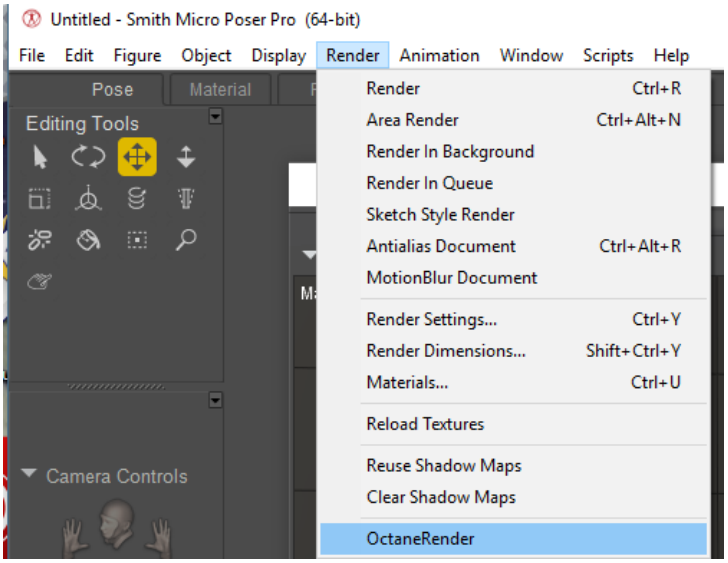



**Figure 1**

When the plugin is opened, it will authenticate itself at the Otoy server (you need both the license for the plugin and the standalone to authenticate – it is tied to your user account at Otoy.

When you start up a scene for the first time with the plugin, it will give a warning that no octane settings were found for the current scene. Click OK and the plugin will use the default values which are fine.

The plugin will now read the current scene. It is important to know that the plugin will not touch your original scene. It will keep all the octane settings and materials separate from Poser. It will however save all this information in the Poser scene. So, the next time you start octane with the same scene, it will load the octane settings and materials for you. It is stored in the PZ3 file as a separate data section which will not be used by Poser itself.

Once the scene has been loaded, the octane plugin window will appear:

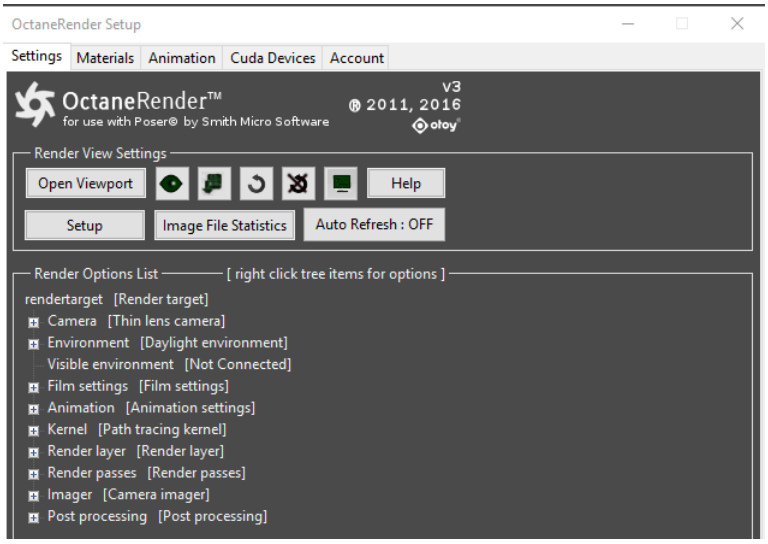



**Figure 2**

The ***Settings*** tab has all the settings for the scene wide parameters such as Render options, environment settings and image dimensions.

The ***Materials*** tab contains all the settings for all the materials used in the scene

In the Animation tab you can find all the settings for animating a scene.

The ***CUDA Devices*** tab allows you to configure which CUDA devices are going to be used and which priority they should have.

The ***Account*** tab finally, has information about the current version and allows you to activate or deactivate the software for your current PC. Activation is handled automatically, but for off line use, the activation here can be used. Please go to the Otoy website for more information on licenses and offline use if you need that information.

## 2.2.1 Settings tab

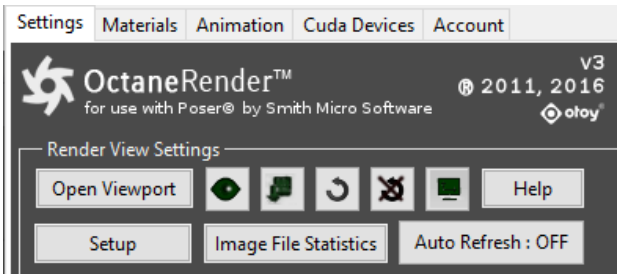



**Figure 3**

The top portion of the Settings tab are buttons or toggles. On the top left to right they are *Open Viewport*, *Set Octane focus from Poser camera focus* (), *Progressive Save* (), *Refresh* (), *Lock scene* (), *Render resolution* () and *Help*. On the bottom *Setup*, *Image File Statistics* and *Auto Refresh*. By hovering over the button, a popup will appear explaining the function.

### *2.2.1.1 Open Viewport*

This will open the Octane Viewport where the render will be shown. More about that later.

#### *2.2.1.2 Set Octane Focus to Poser Camera Focus*

Octane by default has set the focus to Autofocus. It focuses on the object in the centre of the viewport. You can set it explicitly to the Poser camera focus by pressing this button. This is also how you can get the camera keyframes in Poser to Octane.

There is a third way to set the focus which I will explain in the Octane Viewport section

#### *2.2.1.3 Progressive Saves*

Octane has the ability to make “in between” saves of a render. You can use this to compare renders during a render. Clicking the Left mouse button on this button will open a new window where you can set the filename, location and interval of the intermediate saves. Right Click makes an immediate Save and the middle button on the mouse will open de save folder.

You can compare the current render with the previous saved one directly in the Octane Viewport by right clicking in the viewport on the scene.

#### *2.2.1.4 Refresh*

The Refresh button will read the entire scene from Poser again. This is needed when Auto Refresh is not on and you change the poser scene. It will read the scene while keeping all the Octane materials you have changed. New objects in the scene will be converted.

In some case you need to do a reload, such as when you want to change the subdivision in Octane itself for an object.

#### *2.2.1.5 Lock Scene*

This will lock the scene from updating. This is to prevent accidental updates to the scene (such as camera move or light change) which would cause a restart of the render. Press it again to unlock the scene again.

#### *2.2.1.6 Render Resolution*

By default, the Viewport size is the Render size. You can overrule this by clicking this button. After you have done that, the Film Settings in the Render options part will allow you to set the exact resolution you want. Click it again to restore it to de viewport size.

### *2.2.1.7 Help*

Help will open the Otoy Documentation centre where the online documentation exists. It will have links to all manuals.

### *2.2.1.8 Setup*

The Setup window regulates the defaults for UI Settings, conversion settings, folder locations and some miscellaneous settings.

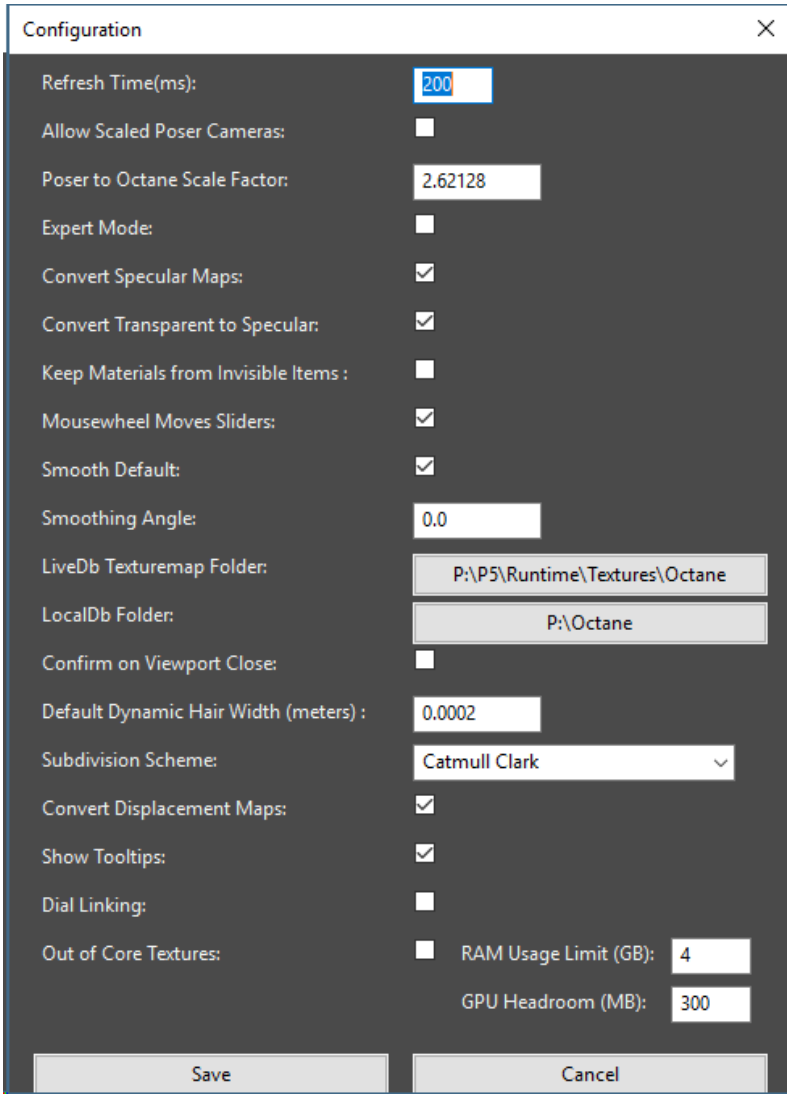



Figure 4

For now, we will leave most of these alone since they will make more sense when material conversion is discussed.

The folder location settings (**LiveDB Texturemap** *folder* and ***LocalDB folder***) are important to set right.

> The LiveDB texture folder is where the plugin stores the textures used by Octanes own material library (LiveDB, more about that later). If you move to another PC, it is easy to forget to copy the LiveDB texture folder since you never use it consciously. Having it in the Poser runtime makes it part of

the Poser runtime and is automatically moved with the runtime. If Octane cannot find the textures, it will ask you to locate each individual one. This is no fun.

The LocalDB (your own Octane materials) can be placed there as well, but this is a folder you actually select in most cases and is not stored in the scene files.

*Also note that Octane and the Plugin store all textures in absolute path format. This means that if you move to another PC, make sure that you use the same path structure (including drive letter).*

***Refresh time*** and ***Poser to Octane Scale factor*** should not be changed.

***Allow Scaled Poser Camera*** does not really work very well and should be avoided.

***Expert Mode***, ***Mouse wheel Sliders***, ***Confirm on Close***, ***Show Tooltips*** are UI conveniences which you can set to your liking.

All other settings will be discussed later.

#### *2.2.1.9 Image File Statistics*

This button will tell you how many color and greyscale textures are used and how much VRAM it will take. This is more or less an obsolete feature since Octane V3 does not have limits anymore on number of textures and is able to use out of core textures if it will not fit in VRAM.

#### *2.2.1.10 Auto Refresh*

Turning Auto Refresh on will synchronize changes in poser scene instantly with the Octane render. Although this works quite well, it does put a penalty in performance while working in Poser. I always have it off, except when finetuning transitions and rotations.

## 2.2.2 The Render Options List

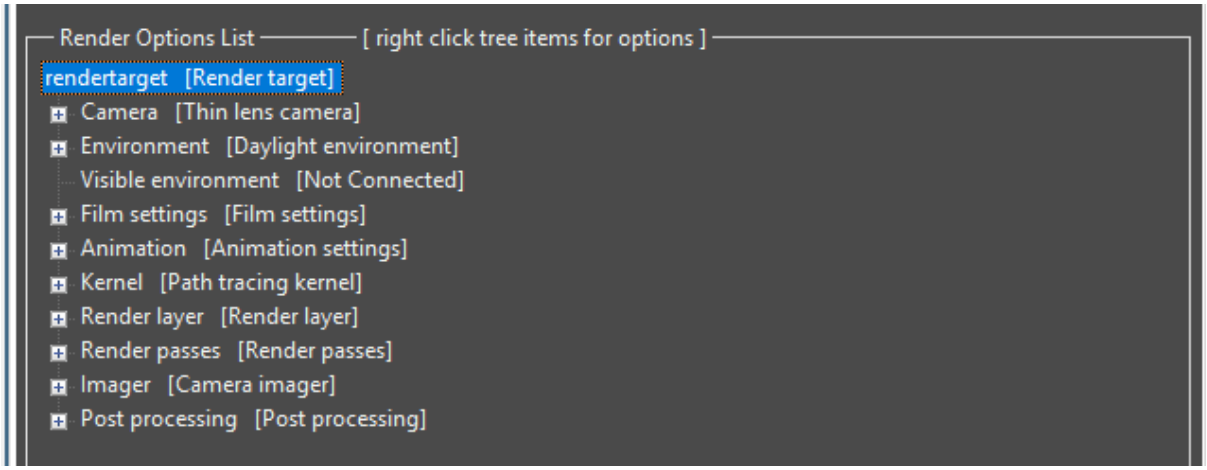



Figure 5

There are several groups for the Render options.

*Camera* deals with every to do with cameras: focal length, Aperture (depth of field), bokeh, VR settings and field of view.

*Environment* deals with the global environment: Sun, sky, haze and backplates.

*Film Settings* is where you define the render resolution.

*Animation* is where you can specify film speed for animations.

*Kernel* deals with all settings for the render engines itself.

*Render Layer* allows you to setup layers for rendering. In the material setting you can specify which objects belong to which layer.

*Render Passes* allows you to single out the different render passes such as diffuser, reflection, shadow and light passes.

*Imager* deals with photographic options such as exposure, gamma, film responsiveness, saturation vignetting, etc.

*Post processing* is the group where you can add lens flares, bloom, and spectral shift.

Below are the most important ones for now. Later chapters will deal with other options.

#### *2.2.2.1 Camera*

Most of the camera settings are controlled via the Poser camera (*focal length*, *field of view*, direction and *position* of the camera). The cannot be changed in the plugin directly.

*Autofocus* can be set to On or Off. When On, Octane will take the object in the centre of the viewport and sets the focus to that object. When Off, it is a user setting – either by taking it from Poser (S*et Focus* button on in the Setup window) or by clicking on an object in the Octane viewport.

*Aperture* is the depth of field setting. A value of 1 means an average depth of field (portrait style), a setting of close to 0 will decrease it or disable it. You can control the falloff with the *Aperture aspect ratio*.

#### *2.2.2.2 Environment*

There are 2 types of environment settings: *Daylight* and *Texture Environment*

*Daylight* uses the sun as primary light source. It will take the first infinite light in Poser and turns it into a Sun. The Poser infinite light direction defines the sunlight direction for Octane. The *Power* value defines the strength – 1 being normal sunlight.

The *Sky Color* is the sky color if no sky texture is used.

The *Sunset Color* is the color which the sun takes when the sun gets at a low angle, simulating a sunset.

The *Sky Texture* node is a spherical mapped sky texture or HDRI. It will be used for global lighting as well

The *Environment Texture* mode uses an HDRI to simulate global lighting. It is by itself sufficient enough to provide daylight. Make sure the HDRI is a proper HDR or EXR (not a converted JPG or PNG) since the HDR contains light intensity information which the other formats lack and will not properly do the global lighting. Also make sure the correct gamma is used for the HDR.

Use the Power setting to define the strength of the HDRI lighting.

The HDR will also act as a background environment, so use a high resolution HDR to get sharp backgrounds.

#### *2.2.2.3 Kernel*

There are several render kernels which Octane can use. For now, I will focus on the *Path Tracing Kernel*, the most common and powerful one.

*Max Samples* is a value where you – the user – thinks the render should stop rendering. The best value depends on the lighting, the scene, the materials and your personal preference. The render is finished when the renderer adds no more detail you can see. Sometimes this will be after a few 100 samples. Sometimes it will take 12000 samples (under very low light circumstances with lots of specular materials).

*Alpha Channel* is a setting where you can disable the background and where you can use it for postwork production.

*Ray Epsilon* is a threshold where the render will stop rendering shadows. This setting is primarily there to avoid self-shadowing. In some cases, you may want to lower this to prevent artefacts where meshes intersect with each other (like in a hair cap intersecting with the head).

#### *2.2.2.4 Imager*

There is one important setting you should know about: *Hotpixel Removal*

*Hotpixel removal* is a post processing option which will remove fireflies – these are white unwanted pixels in your render. They are sometimes happening with specular materials and specific lighting. You can render longer, but this may take considerable time and the Hotpixel removal is usually a good alternative. Just set the value of it to 0 and the fireflies will disappear.

These are the most important things in the settings menu which we need to know to render an image. All the other default settings will be ok.

## *2.3 The User Interface of the Octane Viewport*

Before we start to render, we need to understand the Octane Viewport.

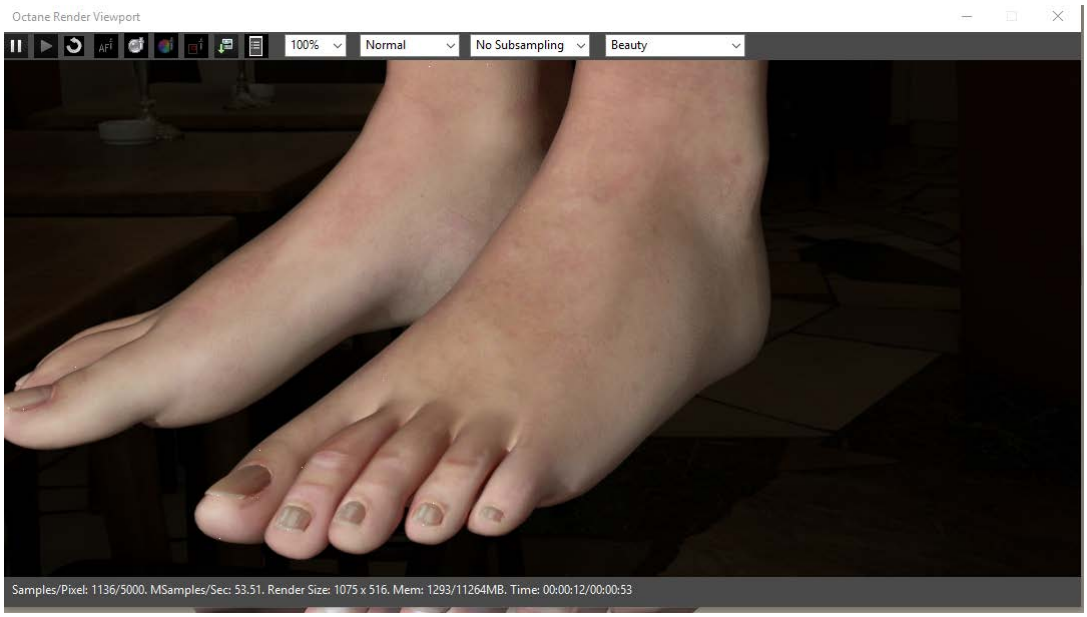



**Figure 6**

The Icons in the top line of the viewport are (from left to right): *Pause* (  ), *Start* (  ), *Refresh* (  ), *Set Focus* (  ), *Material Picker* (  ), *White Balance Picker* (  ), *Area Render* (  ), *Save Image* (  ), *Plugin window* (  ). The drop-down boxes next to it are *Render scale*, *Render Mode*, *Sub Sampling* and *Passes*.

### *2.3.1.1 Pause*

This speaks for itself: Pause the Rendering. You can resume it later if you made no changes in your materials or scene. There is no option to resume a render after you have closed the render window.

### *2.3.1.2 Start*

This will restart the render after it has been paused.

### *2.3.1.3 Refresh*

The Refresh or Reload button is the same one as in the plugin, it reloads the geometry.

#### *2.3.1.4 Set Focus*

This button toggles focus picking On or Off. When active, the mouse cursor acts as a picking tool onto which object the focus has to be set. Do not forget to turn off the Set Focus mode.

#### *2.3.1.5 Material Picker*

This will change the mouse cursor into a material picker. Click on any material in the Octane Viewport and in the plugin material list, this material will be selected.

#### *2.3.1.6 White Balance picker*

This will turn the mouse cursor into a color picker. You can use it to change where actual white in your image should be.

#### *2.3.1.7 Area Render*

With this button on, the mouse cursor will change into a region selector. Drag a rectangle and Octane will render only that portion. This is very convenient to test materials in the selected region. Turning it off again will restore the full render.

#### *2.3.1.8 Save Image*

This button is used to save the rendered image. You can use it at any time during a render. You can select PNG 8 bit (the standard PNG format used on the web), PNG 16 it (True color PNG), EXR (True color and intensity information), EXR Tonemapped and EXR all passes (all enabled render passes will be saved as separate layers).

#### *2.3.1.9 Plugin Window*

The plugin window will bring the plugin window to the front.

#### *2.3.1.10 Scale Render*

This allows you to set the scale of the render (to view an image larger as the viewport).

#### *2.3.1.11 Render mode*

This can be set to Normal, Clay or Colored Clay.

#### *2.3.1.12 Subsampling*

This setting allows you to improve the speed of the viewport while navigating the scene with camera or lights. It will return to normal once navigation has been done.

#### *2.3.1.13 Passes*

This will allow you to select the pass you want to view from the ones you have enabled in Render Passes.

### 2.3.2 Information line

The bottom line of the view port shows the *Actual* and *Maximum* number of samples for this render. The *speed* at which it renders, the *Render size*, the *actual* and *maximum* VRAM usage on your card, the *actual* and *expected* render time for the maximum number of samples.

## *2.4 The First Render*

In this section I will explain the steps you need to take to render a poser scene in Octane.

We will start off with the default scene with Andy.

Octane has its own sky, so there is no need for a sky dome or construct like enclosure (Poser 11) needed to provide the global lighting. Therefore, we need to disable those in the Poser scene for now (I will get back to it later for when you do want to use them).

So, delete or disable (ctrl-G) the sky dome or enclosure first.

We are going to use the Sun in Octane. The plugin uses the first available Infinite light in Poser. So, we need to be sure there is one. In Poser 11 there is no infinite light in the scene, so we need to create one or turn one into an infinite light. Create a new light and set its property to be an infinite light.

Point the infinite light to Andy if needed.

Now we can render this scene. Start the Octane Render from the Render menu: Render!OctaneRender.

The plugin will now authenticate and activate.

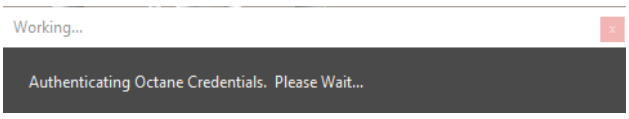



**Figure 7**

Next a warning will pop up. This will tell you that no rendertarget have been found for the scene. This is of course correct since we have done nothing with Octane in this scene yet.


Figure 8

Press Ok to continue.

Now Press the Open Viewport in the Plugin Window.

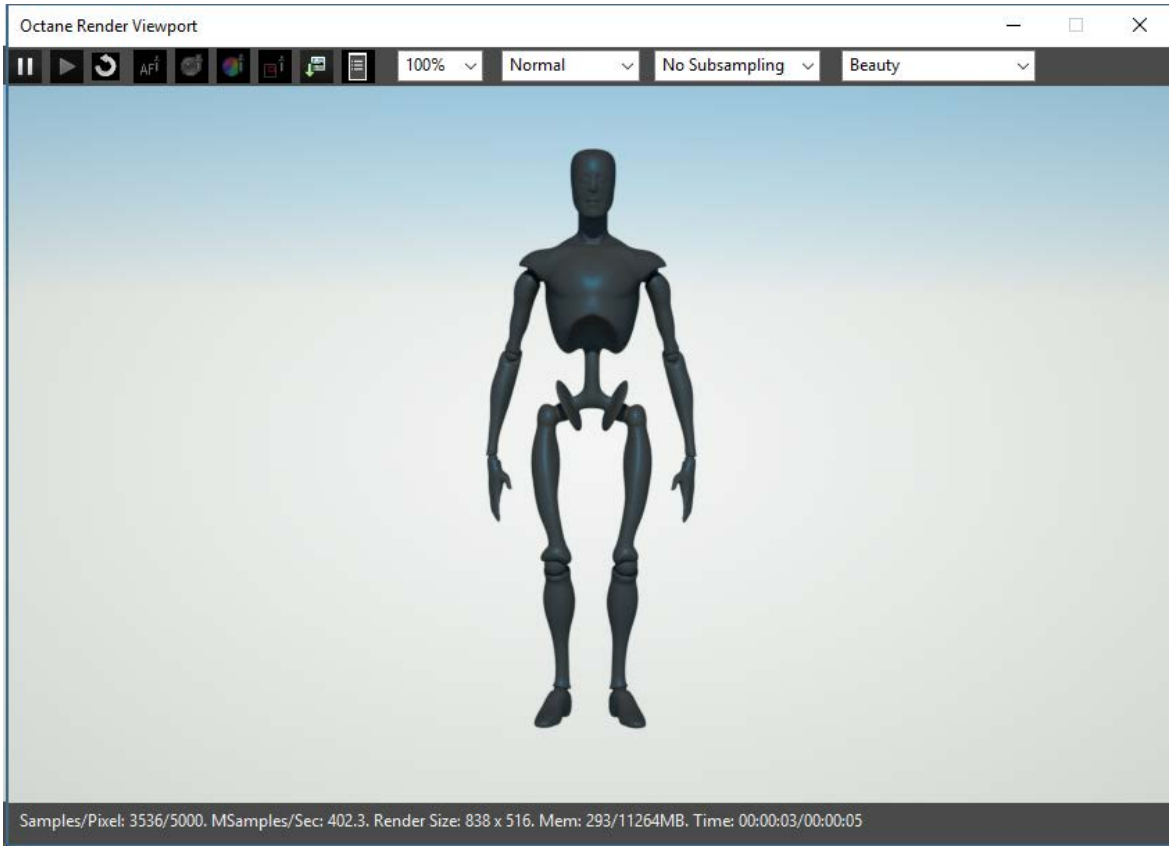



Figure 9

And we have our first render in Octane. You can rotate and move the camera and the render will use the new camera viewpoint. You can also rotate the infinite light and the sunlight in Octane will change.

Although this render is nothing exciting, we can now move forward and make things look good and interesting.

# 3 Poser Scene to Octane - Basics

## *3.1 Automatic Conversion and its limits*

Poser 1 has currently 2 built in Render engines: Firefly and Superfly. Through plugins it can also render in Lux (via pose2lux or Reality) and Octane (Poser plugin). With the use of Poser fusion Poser scenes can also be exported to a variety of Software where it can render in other Render engines.

Compatibility between all these render engines is limited. The Firefly/Superfly compatibility is pretty good, but not 100% either. This is where the conversion plugins come in.

There are a variety of reasons for this, but the most important ones are the differences in capability of these render engines and the use of different material definition languages and nodes.

The Poser plugin for Octane tries to do its best with the help of a set of user definable defaults to make the conversion as easy as possible.

The plugin uses the Firefly material root and analyses it and takes the textures from the node tree and connects them as Octane materials. It cannot convert the more complex materials mainly because there are no equivalent nodes in Octane or which make no sense in Octane. Having said that, in most cases the conversion goes well, and it is relatively easy to make additional changes if needed.

## *3.2 Lights (Simplified)*

Lights are another compatibility challenge. Poser has Infinite lights, Point Lights, Spot lights and Area lights. Octane has Daylight Environment (sun), mesh lights, portals (which are a special kind of lights) and Texture Environment (HDRI).

### 3.2.1 Daylight

Daylight in Octane can be compared to an Infinite light in Poser. There is however only one Daylight and it acts like a Sun. Like Infinite lights in Poser it has no falloff in strength and is one directional. It is the default lighting model and the plugin takes the first Infinite light in the Poser scene. If there is none, the plugin will give you a warning that there is no Daylight present. It will use a default light which you cannot move.

The plugin will rename that first Infinite light to Daylight and you will be able to rotate the Daylight within Poser and Octane will automatically rotate its own Daylight.

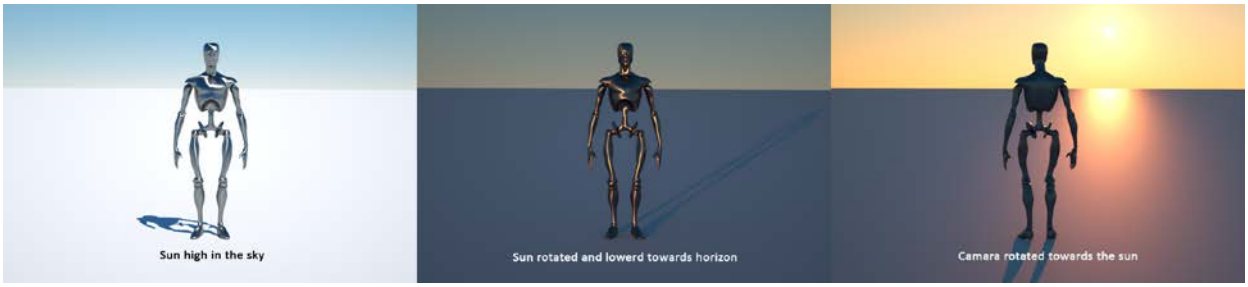



**Figure 10**

As you can see, the light dims and changed color towards the horizon. The Sky color and Sunset color and other parameters of Daylight environment can be changes in the properties of the day light. See the Light (Advanced) section for more information.

### 3.2.2 Mesh Light

Mesh lights in Octane are very powerful and can be used to simulate Point, Spot and Area lights. The Octane plugin has two python scripts to convert Spot and Point lights into a mesh light: Create Emitter from Current Light and Create Emitters from Lights. If you have an Area light, convert it to Spotlight first in Poser. You can find the scripts in *Scripts!OctaneRender* for Poser.

What these scripts do is to create a prop (one sided plane for spot and Ball for point light) and parent it to the light. You can now translate and rotate the lights as you like, and the prop will follow. The props will get an emitter material and will act as a light source.

Mesh lights emit light in the direction of the normal. So, a One-sided plane will only emit from one side as you can see in the render below.

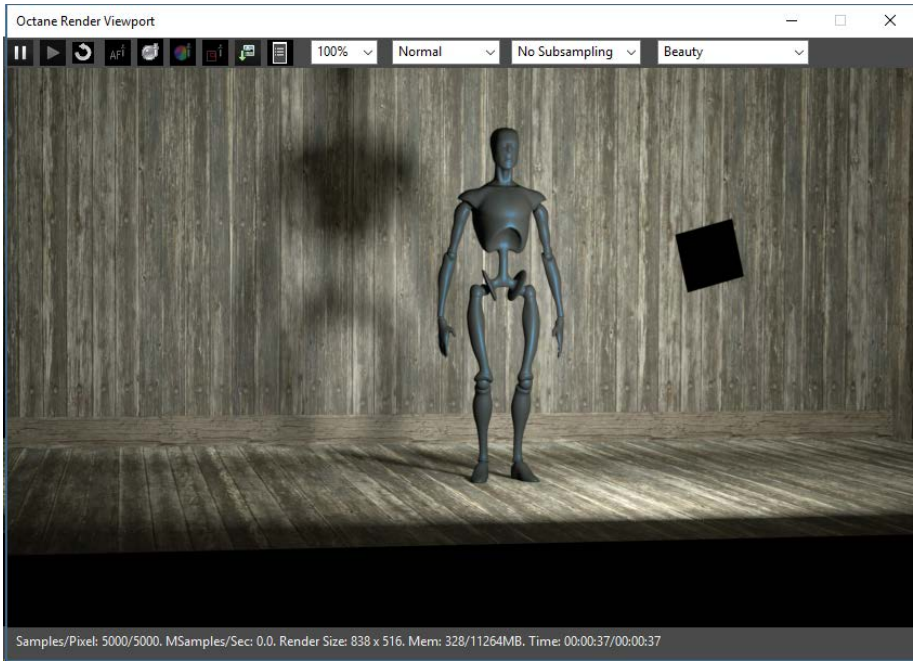



**Figure 11**

The emitter is visible now, but we can set the opacity to 0 (same as in setting transparency to 1 in Poser) or you can set the General Visibility to 0 in the mesh properties of the emitter prop in the plugin. It will continue to emit light, but you will not see the mesh itself.

The plugin will do an attempt to set the strength, but it depends on the distance and is not always accurate. So, you must adjust the strength of the light with the Power value. More about that and other settings later.

Shadow falloff depend on the size of the mesh light (larger is softer edges) and closeness of the mesh light to object or figure of which it casts shadows.

Area lights are not directly supported but are actually mesh lights with the same properties as spotlights. Converting it with Poser to a Spotlight and then converting it with the *Create Emitter From Current Light* python script will create that Area light.

A Point light will be converted in a ball which has the normals pointing in all directions and will act like a Point light. Look at it as a light bulb.

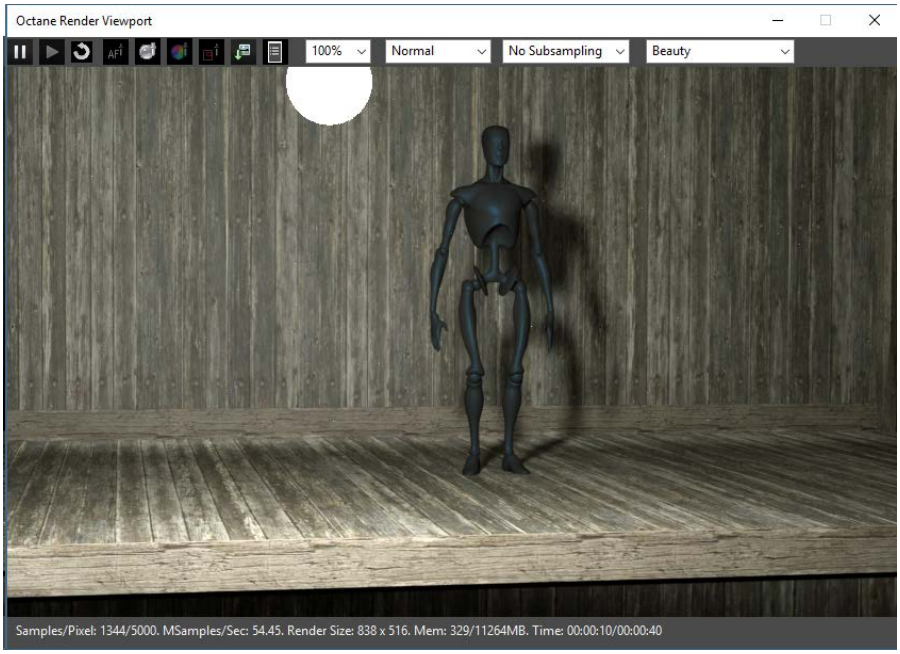



**Figure 12**

You can use any primitive or other prop as a mesh light, even a material on a prop. This means that if you have a ceiling with lights built in, you can make the light material an emitter and use it as the light without any additional mesh light added.

If the mesh light is small, you can increase the number of light rays emitted from the mesh light by increasing the sampling rate. This will give more attention to that particular light.

### 3.2.3 Portal

Please see the Portal Material section for information on Portals.

### 3.2.4 Texture Environment

Texture environment is the HDRI lighting environment of Octane. It disables the Day light and lets a color or HDRI provide the lighting. Think of this as Posers Indirect Lighting (IDL) on steroids. It is most suited for outside environments but can be used with interiors as well.


**Figure 13**

In the left image above, a white color is used as HDRI environment. You can see that it no reflected highlights, the reason is that all what is reflected has the same color (white) and does not really show. In the second image, an HDR texture is used and immediately highlights pop up and becomes more realistic. In the last image, I increased the Power to increase the strength of the lighting.

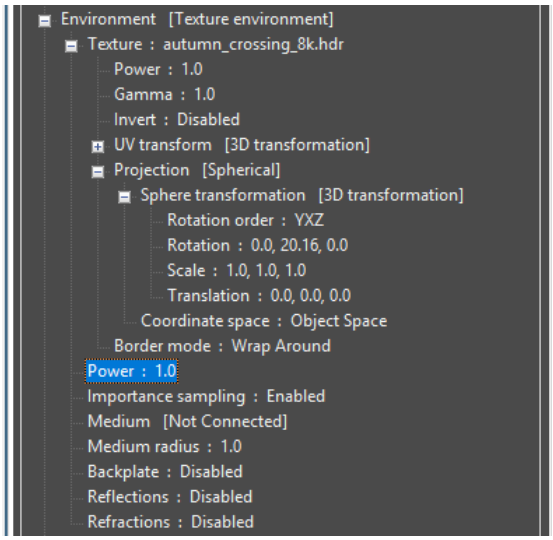



**Figure 14**

The interesting properties of the Texture Environment for now are the Texture node where you can load the HDR and the Power node where you set the strength. Almost all the HDR textures have a gamma of 1, be sure to set that to the right value.

I will describe the other properties in the Light Advanced section.

## *3.3 Materials*

In this section I will describe most of material nodes and how to use them. I will skip over some and refer to later sections, usually because they require understanding of other features of the Octane engine.

Octane uses a hierarchy for material assignments:

- Scene
    - Mesh
        - Geometry
            - Node Type
            - Properties
        - Material Type
            - Properties
                - Node Type
                    - Textures
                        - properties
                    - Values
                - Values
        - Material Type
    - Mesh

Each Material Type shares properties but has also unique properties for that type.

### 3.3.1 Geometry

Materials can be edited in the materials tab. All props and figures visible in Poser will be listed here. Hidden figures or props will not be listed.

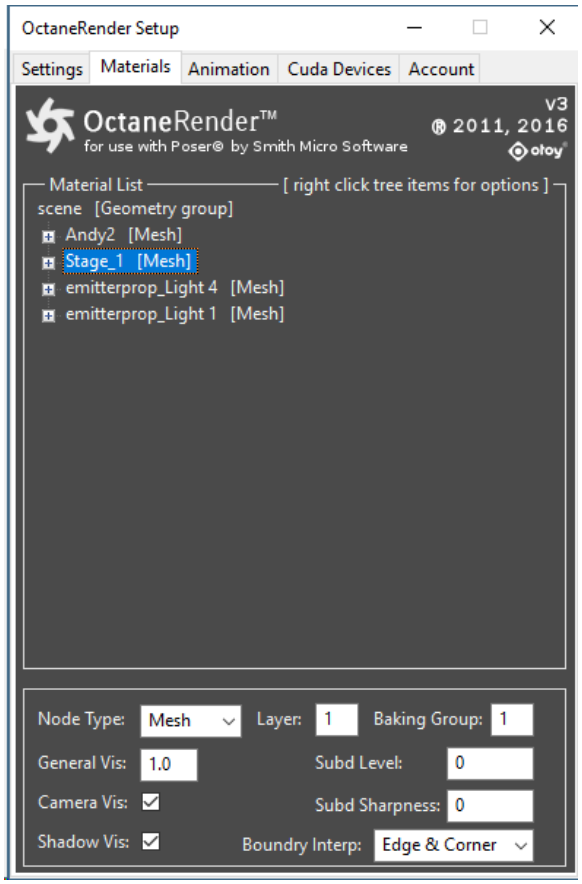



Figure 15

The bottom portion of the list is the property section. When a prop or figure is selected it will show the properties of the mesh.

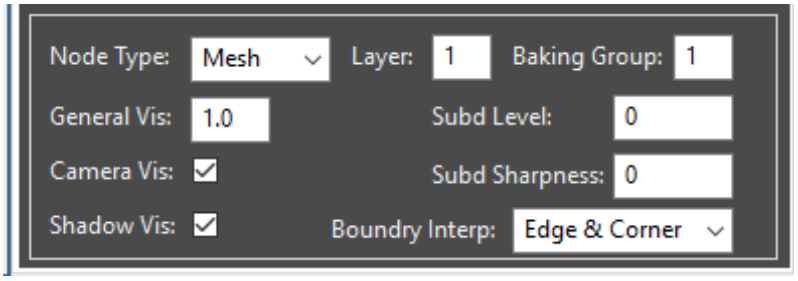



Figure 16

The ***Node*** *type* defines how you are going to use the mesh – either as mesh/prop/figure, or as scatter object (Instancing, more about that later) or a geometry out object (advanced feature).

The ***Layer*** value allows you to put different objects in different layers. You can render each layer separately.

The ***Baking group*** is used to identify a mesh for the purpose of a variable texture.

***General visibility*** allows you to make an object invisible for the camera and specular materials (reflections). Emitted light remains visible.

***Camera visibility*** is used to make a prop invisible to the camera while remaining visible for reflections (think of a wall in front of a camera).

***Shadows visibility*** turns off shadows for that prop.

Octane itself can do ***subdivision***. You can define the ***subdivision level***, the ***sharpness*** for curves and define the ***boundary*** for this: Edge/Corner, Edge Only, Always sharp, or none.

### 3.3.2 Material Type

When you expand the prop in the material list, it will show all the materials for that prop or figure:

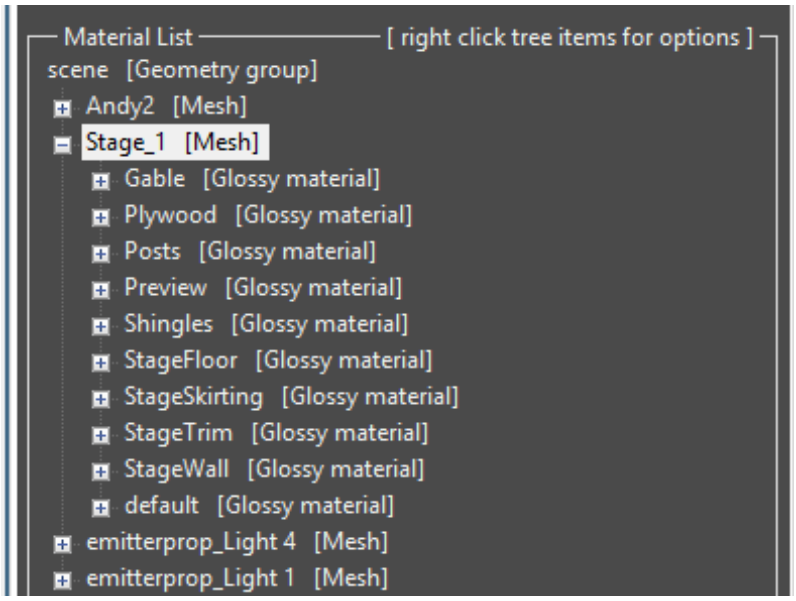



Figure 17

Materials in Octane are either ***Glossy***, ***Diffuse***, ***Specular***, ***Metal***, ***Portal*** or a ***Mixed* Material**.

Each of these material types has its own properties. Glossy materials reflect light, Metal materials are like Glossy materials, but has different properties. Diffuse materials absorb or emit light, Specular materials lets light travel through the material. You can create more complex materials with the Mixed Material where you can combine 2 different material types together. Portal is a special material type which collects light and passes it through the material unchanged (it can used for windows in interiors).

If you select the material the material type can be selected in the bottom part of the plugin window.

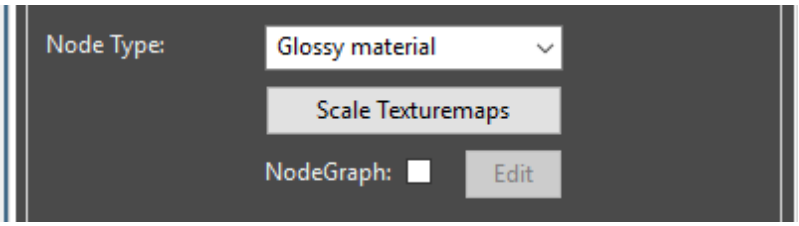



Figure 18

### *3.3.2.1 Glossy Material*

Glossy materials are the most common materials. Almost all materials have some shininess to them, so it is the default conversion for material found in Poser. The nodes define the properties of the material which give them their look. The input for these properties can be a node type (see later in this section), a value or a Boolean.

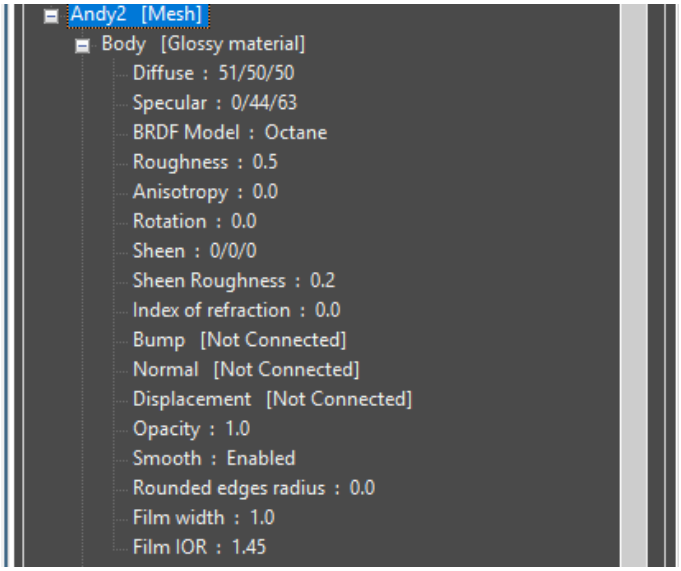



Figure 19

***Diffuse*** defines the base color (or Albedo) of the material, it is usually an RGB color or texture.

***Specular*** is usually a color, (specular) texture, or a value (0-1) and It defines the color or strength of the specular reflections.

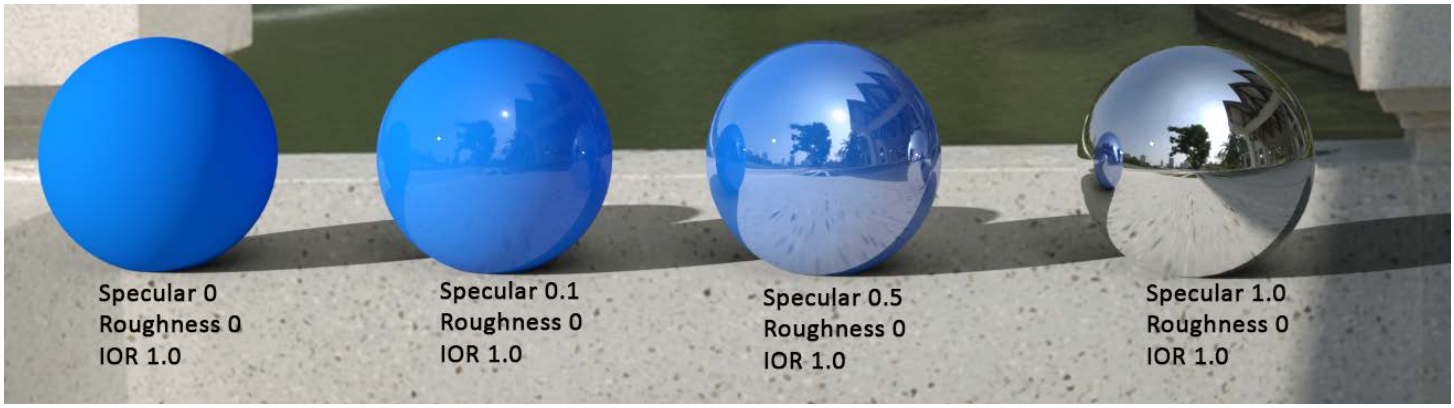



Figure 20

***BRDF Model*** is the type of specularity to be used. Octane has several implementations of the specular reflection: *Octane* (default and used up to Octane v3.07), *Beckman*, *GGX* and *Ward*. Beckman is used in many other render engines and is useful to get similar results, Ward is a faster version of Beckman and GGX is another approach. Other reason to use one of these BRDF models is to be able to render Anisotropy which is not support in the Octane BRDF model. For more information on this, see https://docs.otoy.com/StandaloneH_STA/StandaloneManual.htm#StandaloneDEV/BRDFModels.htm.

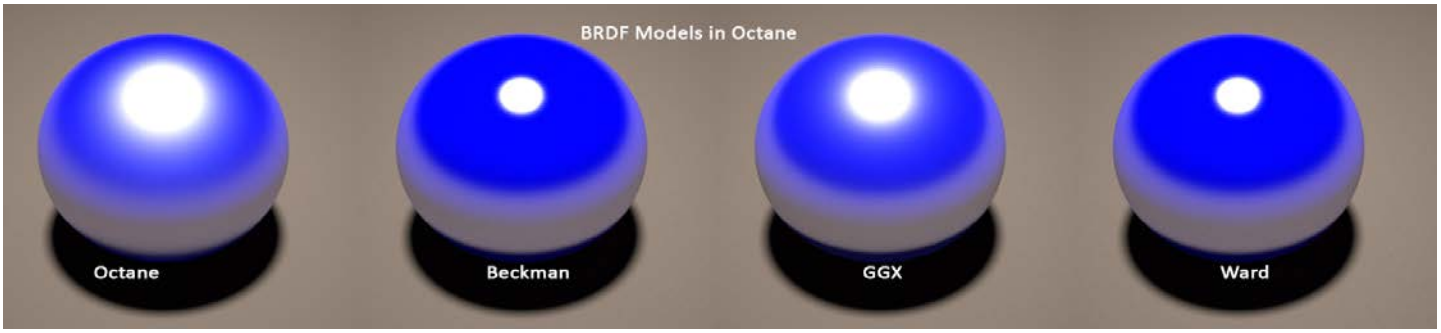



Figure 21

***Roughness*** is the size and sharpness of the highlight. A low value means sharp defined highlights and a high value will give blurred reflections.

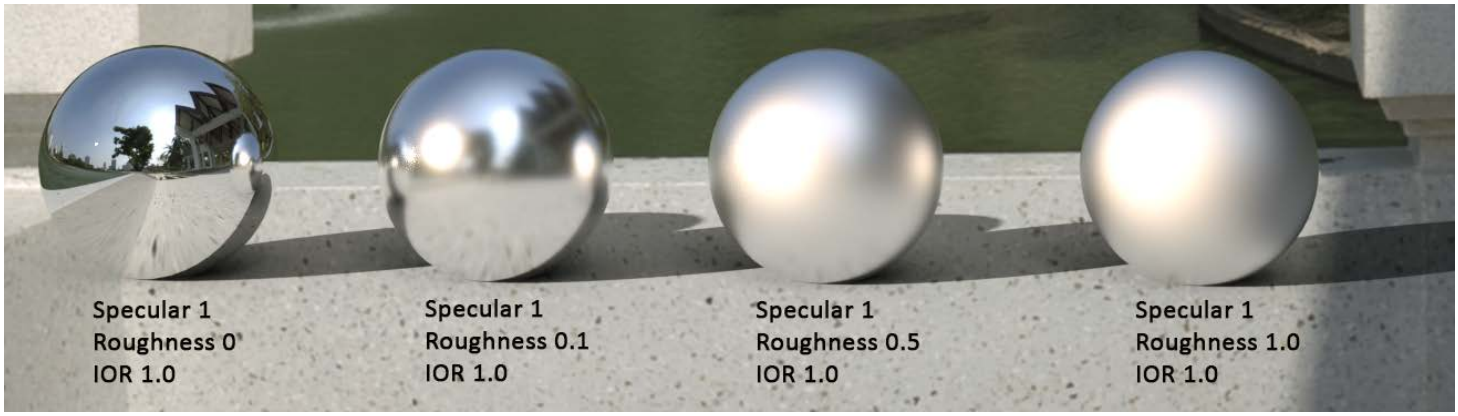



Figure 22

***Anisotropy*** is when the reflection follows the orientation or rotation of the surface. It only works with the Beckman, GGX and Ward BRDF models. Negative value is horizontal and positive value is vertical. A value of 0 means no anisotropy (isotropic).

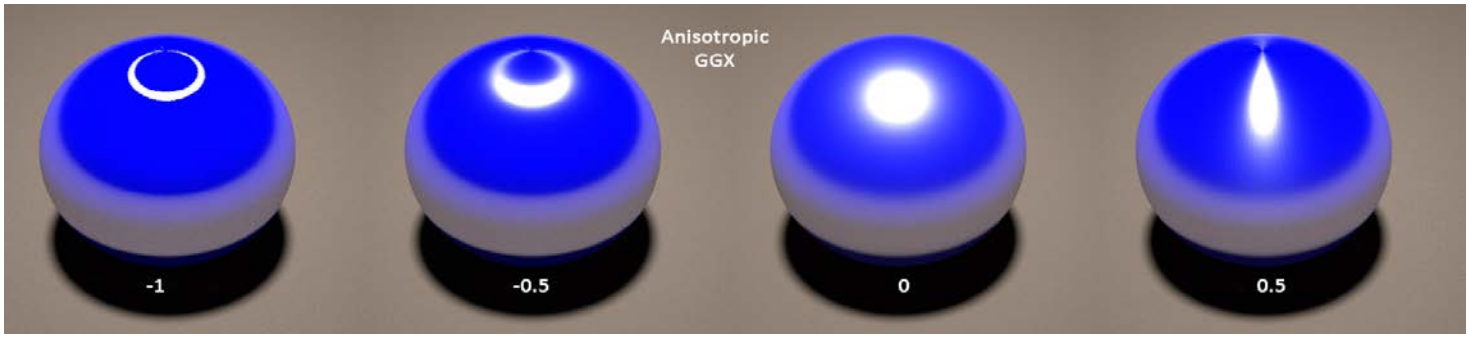



Figure 23

***Rotation*** rotates the reflection around the surface (anisotropic only). Its value ranges from 0 to 1.


Figure 24

***Sheen*** is like a very thin coating on a material, only visible from a high angle like a wax layer. Sheen will take a color or texture. The following examples show the differences between sheen and specular.


**Figure 25**


**Figure 26**


Figure 27

***Sheen Roughness*** defines the spread of the sheen.

***Index of Refraction*** (IOR) for glossy materials define the angle at which light is reflected (Fresnel). With values higher than 1 reflection will be the strongest on the surface turned away from the viewer. The higher it gets, it will reflect more towards the viewer. A value of 1 represents infinite and will reflect the complete surface (like a mirror).

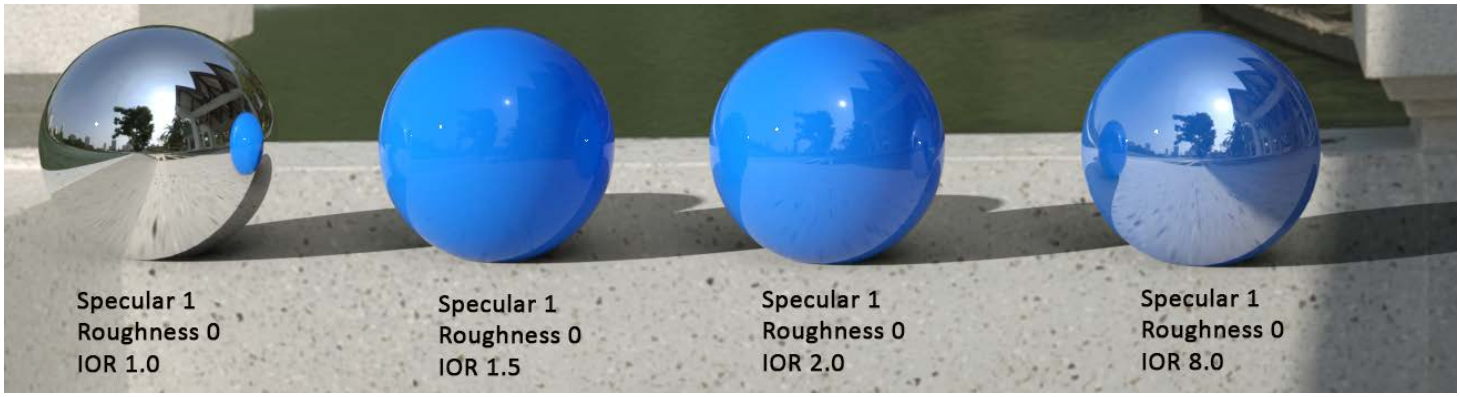
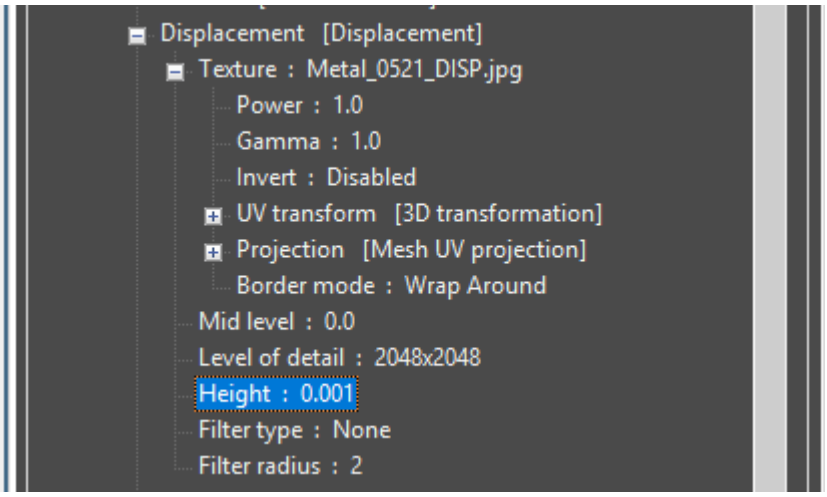



Figure 28

***Bump*** usually holds a bump map or some type of noise filter.

***Normal*** holds a normal map if present.

***Displacement*** holds the displacement map (Octane does support micro displacement). Units are in meters. It has several parameters which you need to be aware off. The conversion does a good job but cannot interpret what the actual displacement map is.




***Midlevel***

Standard Poser displacement maps have black as a zero level. However, many displacement maps now have mid-grey as zero level. In Poser we use a Subtract math node to correct this, in Octane we do this by setting the mid-level. Black zero level textures should have mid-level at 0, mid-grey should have it set at 0.5.

***Level of Detail*** is the texture size of the displacement map. You can set it lower but will get loss of detail. You might want to do it in case of the object being far away and to save memory usage.

***Height*** is the maximum height of the displacement. It will be multiplied with the displacement map. Units are meters.

***Filter type*** **and** ***Filter Radius*** can be used to smooth the transitions.

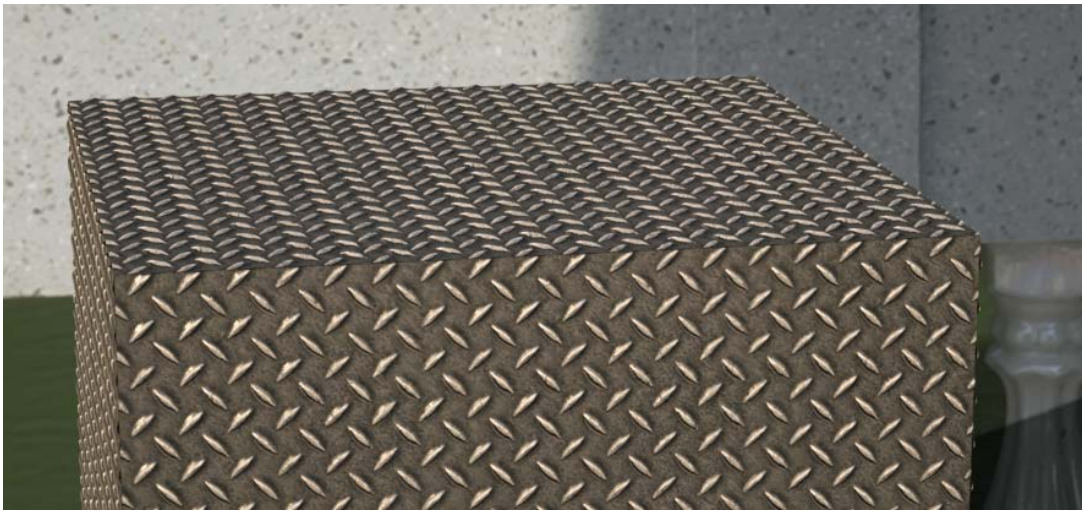


**Figure 29**

***Opacity*** holds a mask (inverted transparency map) or a value. 0 means completely transparent, 1 is opaque.

***Smooth*** is a Boolean whether Octane should use smoothing on the mesh material or not. Having smoothed on can cause issues with flat planes, the material will look distorted.

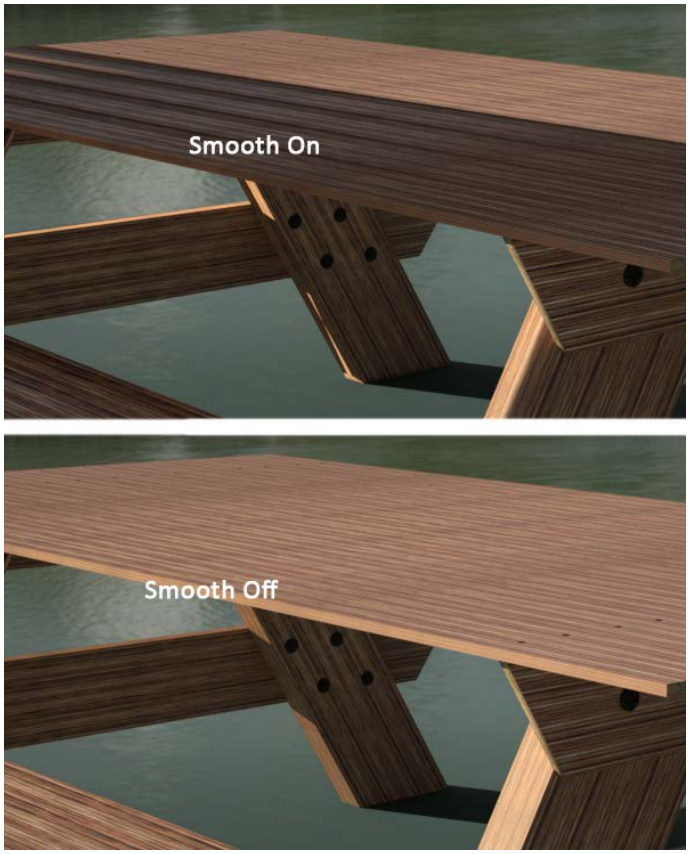



Figure 30

Turn smooth off when this happens. In general smooth on will give better results with curves. There is an option in the Configuration menu to get Smooth Off by default.

***Rounded edges*** bevels sharp edges. This is useful to make edges less sharp without the need for geometry changes. Since poser's scale is very small, you need to have very low numbers for this to work correctly without artefacts. In the example below, I used 0.0027. Not all meshes are suited for this option.

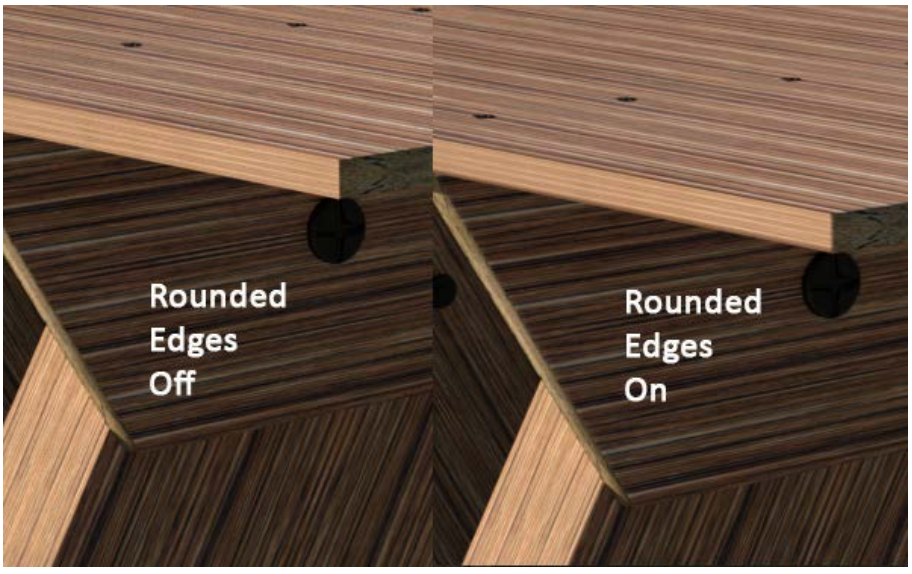



Figure 31

***Film Width* and *Film IOR***

This effect creates a thin film layer on top of the material. The width parameter defines the strength of the effect and the IOR defines the color it reflects. In the example below, I used a glossy ball (specular at 1) to show the film effect.

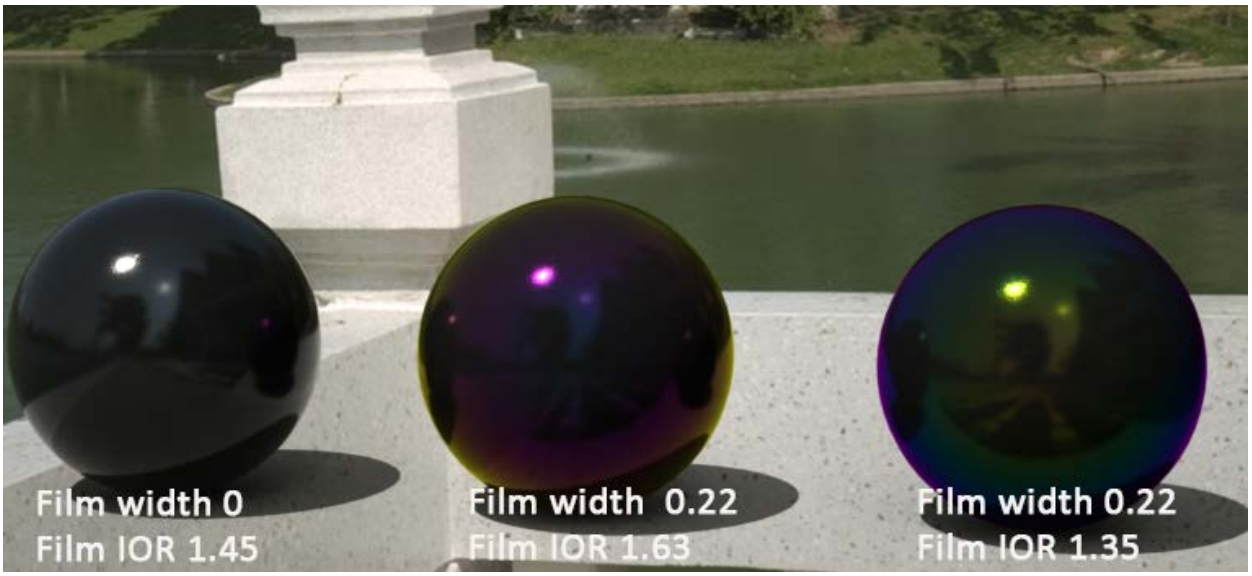



Figure 32

### *3.3.2.2 Metallic Material*

Metallic materials are suited for the Metal/Rough PBR workflow (for apps like Quixel and Substance Painter). The material type is similar to glossy but separates specular strength and color and uses a different method for refraction.

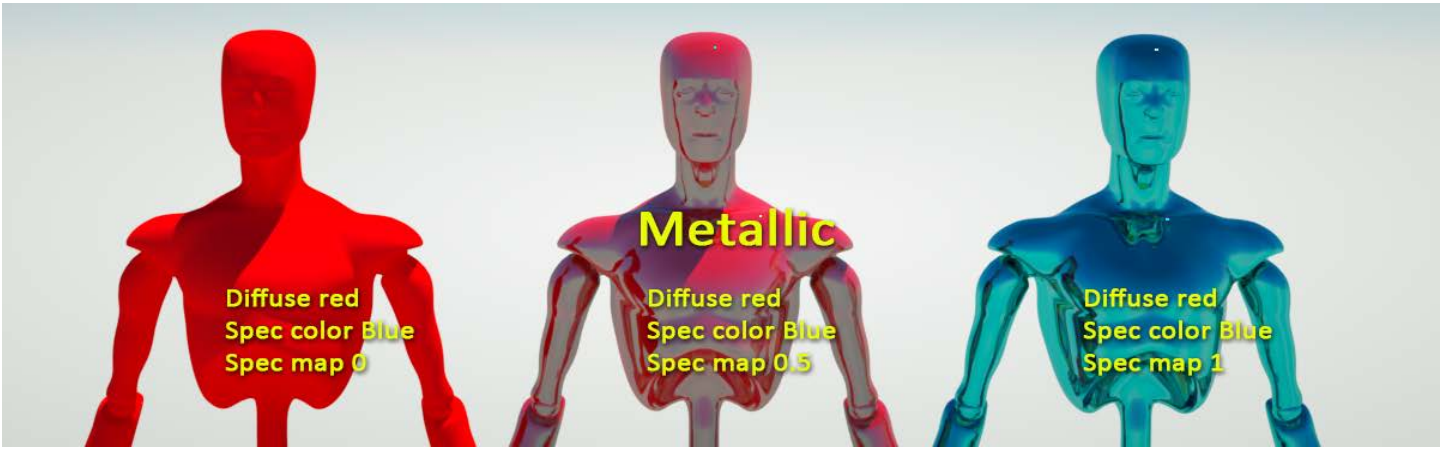



Figure 33

For comparison, here the same thing with glossy material (color multiplied by value).


Figure 34

The Metallic material nodes are very similar to the Glossy material nodes.

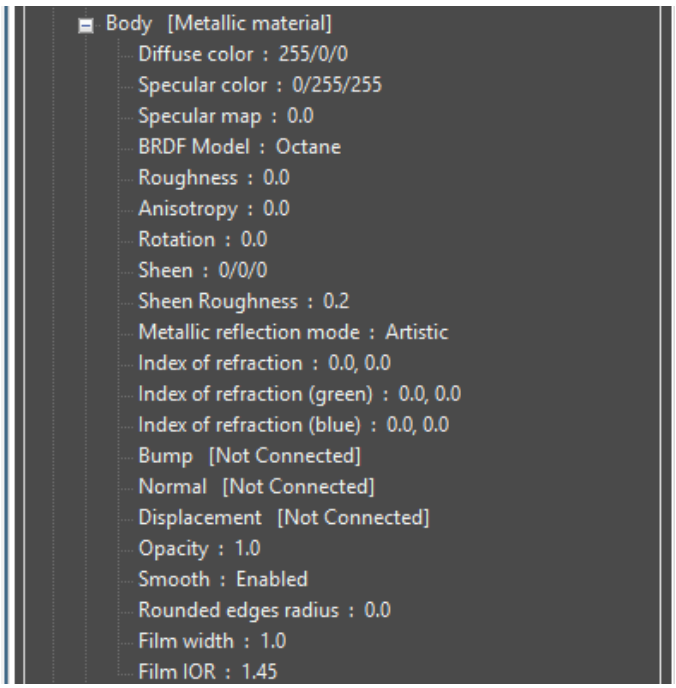



Figure 35

***Specular Map*** controls the blending between the diffuse and the specular map.

***Metallic Reflection mode*** defines how reflections are calculated.

***Artistic*** (Octanes default) uses only the specular color.

***IOR + Color*** uses the specular color and adjusts its brightness according to the IOR.

***RGB IOR*** calculates the reflection color for each of the RGB components (red 650 nm, green 550nm and blue 450 nm) and does not take the specular color into account. Each IOR value consists of *n* (refractive Index) and *k* (Extinction

coefficient) value. You can find these values in a table at this link: https://refractiveindex.info/?shelf=main&book=Au&page=Johnson.

It will reflect a more physically accurate color to the reflection.

The image below shows the differences. The Artistic and IOR Color have slightly different angles, but the color is identical. The RGB IOR has a slight color variation as well. Bear in mind that the gold you see in the real world is usually an alloy which will change the color as well.

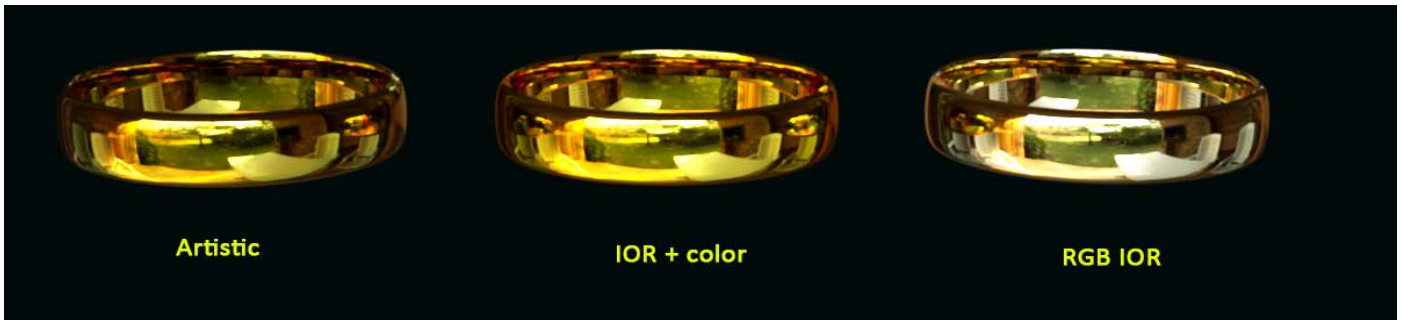



Figure 36

***Index of Refraction*** is not used for Artistic and is a high value for IOR. For RGB IOR is defines the k and n settings for the red color (650 nm).

**Index of Refraction green** is the k and n settings for the green color (550 nm) for RGB IOR.

**Index of Refraction blue** is the blue color setting for RGB IOR (450 nm).

All other nodes are the same as for the Glossy material.

### *3.3.2.3 Diffuse Material*

Diffuse materials are for plain non-reflecting materials, light emitting materials and other materials such as Scatter.

Most of the nodes like Diffuse, Bump, Normal, Displacement, Opacity, Smooth, Rounded Edge are identical to Glossy Material, so I will skip those.

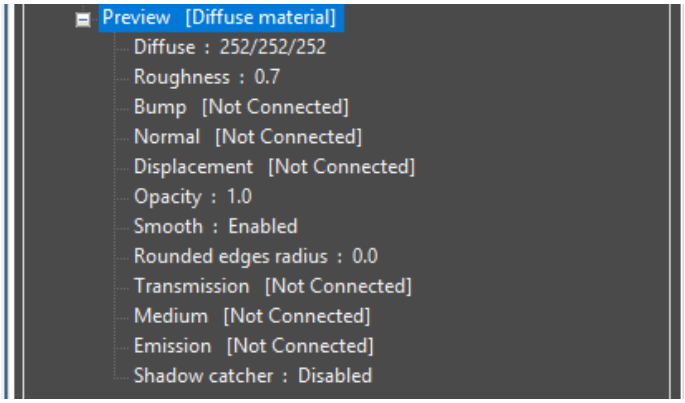



Figure 37

***Roughness*** defines how widespread the diffuse color is bounced back. If you set it to one, it will create a white sheen around the edges, a bit like velvet. Below on the left, roughness is set at 0, on the right at 1.

Figure 38

***Transmission*** is a color or texture which will be mixed with the diffuse color or texture. It determines how much light is passed through the material. This node is used for material like lampshades or scattering.

In the picture below, I placed a ball with an emitter material inside lamp.
The lampshade had its transmission channel set to white. For best result I increases the emitter samples to 1000.

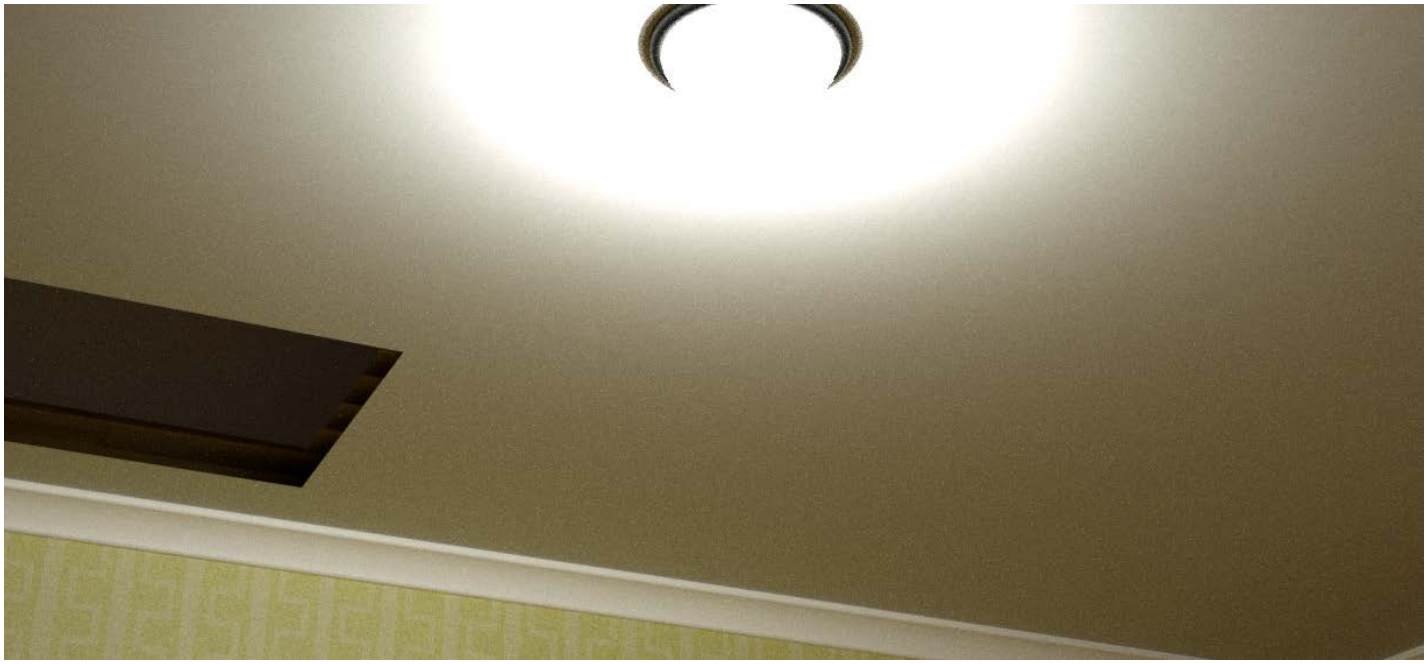


Figure 39

***Medium*** deals with what is inside an object and needs to be a closed volume (mesh). There are three types of medium: Scattering, Absorption and Volume.

***Scattering*** is where light is partially absorbed and partially bounced back (emitted). The color of the bounced light is defined in the *transmission* node, which can be a color or texture.

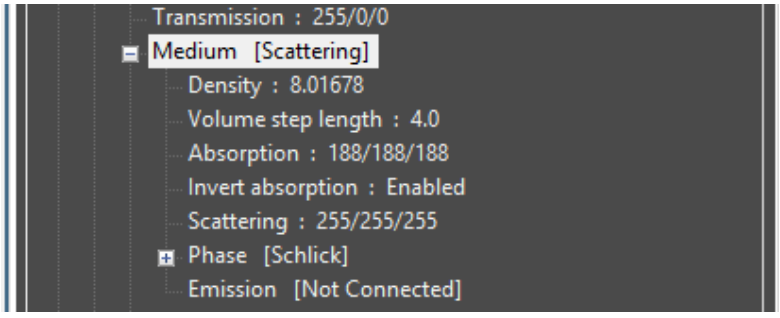



Figure 40

***Density*** is how thick/dense the material is. It is used as a scale factor for the Scattering node described below.

***Scattering*** is how far the light should travel before being bounced back. Together with the *Density* it defines how much scattering is taking place.

***Volume Step Length*** is the length of the step used (in meters). If the object is very small, the step size should be adjusted.

***Absorption*** defines how much light the color is absorbed. It can be a value or color.

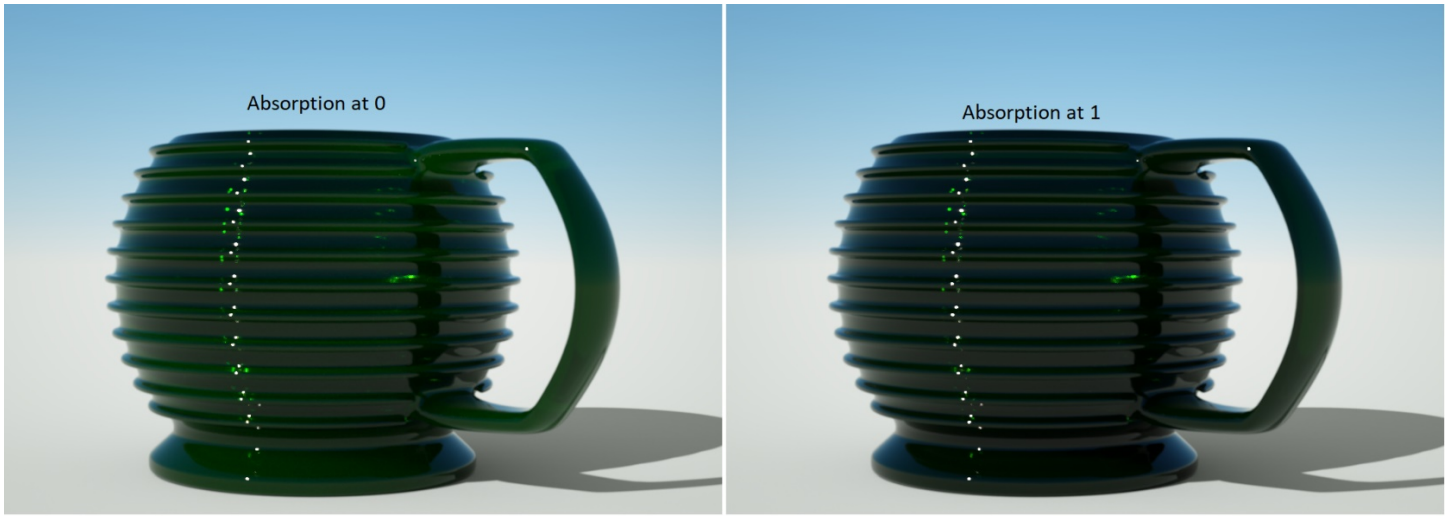



Figure 41

The effect can be very subtle as in the example above. In the left image I have no absorption at all, and on the right all lights are absorbed. To clearly see the effect it has, you can make the Scatter layer visible as shown below (more about render layers later):

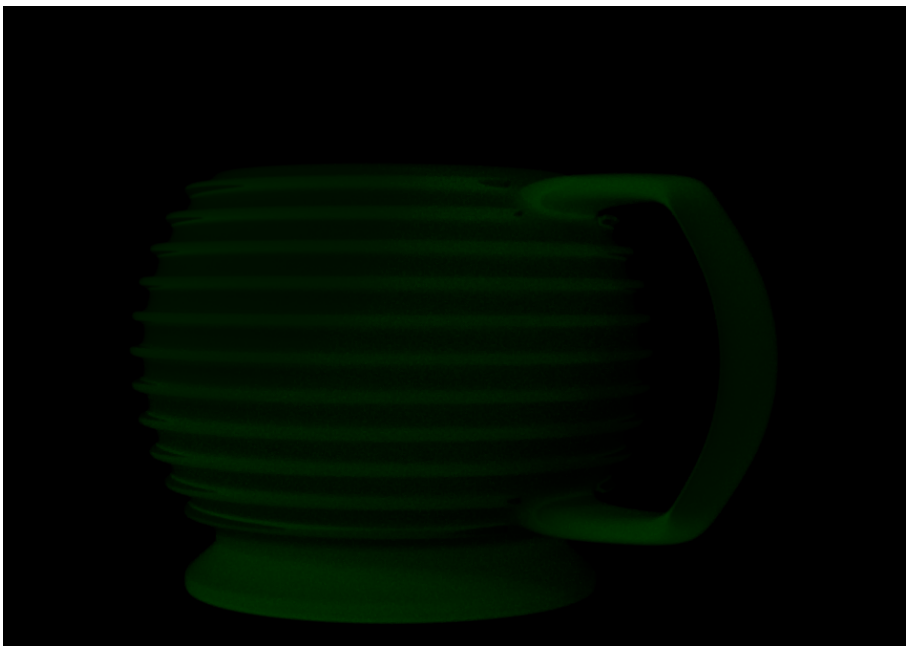


Figure 42

***Phase*** defines the direction of the scattered rays. 0 means in all directions, positive roughly in the direction they were going to, negative roughly back in the direction they were coming from.

***Emission in Medium*** can make the inside emit light. This is a different type of emission as the Emission described below. The emission in medium will be described in a later chapter.

***Emission*** is the node which makes a material emit light. The Emitted light can either Texture emission or Black body emission.

***Texture emission*** is the simplest one and uses an RGB node or texture to color the light. The plugin uses these light emitters when it converts the point and spot lights to Octane mesh lights.

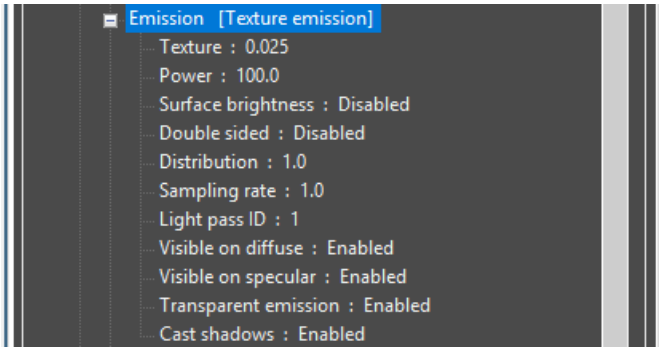



Figure 43

***Texture*** can be any material node, the most common one will be an RGB node.

***Power*** is the intensity of the light. It will be multiplied with the Texture node to give the final brightness. The plugin often has set the poser converted lights way too strong, so you probably need to lower this value after conversion.

***Surface brightness*** is a toggle whether the actual size of the surface should be taken into account or not.

***Double Sided*** is a toggle whether one sided surfaces should be treated as double sided materials. If so, it will emit lights in both directions. For directional lights it should be off.

***Distribution*** can be used to supply an IES light distribution map. More about this later.

***Sampling rate*** defines the importance of this light emitter. A higher value means it will get more samples.

***Light Pass ID*** is an identifier which can be used in the render passes.

***Visible on Diffuse*** toggles this light off for diffuse materials (essentially turning off lights). It then will only be rendered on glossy and specular materials.

***Visible on Specular*** toggles off the lights on specular and glossy materials.

***Transparent Emission*** will allow you to emit light from transparent materials.

***Cast Shadows*** is a toggle shadows on or off for this emitter.

*Shadow Catcher* makes the material transparent while keeping the shadows.

### *3.3.2.4 Specular Material*

Specular material is transparent material such as water and glass. It shares many of the properties of glossy materials but there are some additional properties.

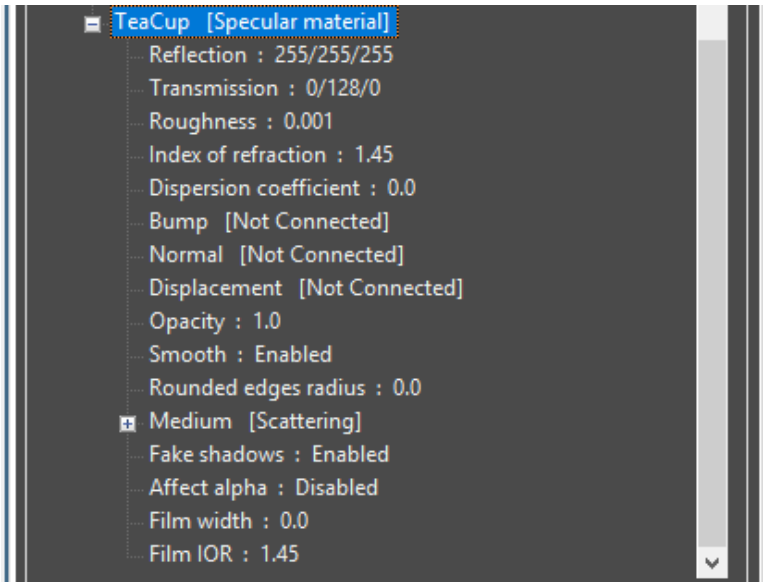



Figure 44

***Reflection*** is the strength of the reflections which are visible on the surface of the material. Its effects work together with the IOR and Roughness. At 1 there is full reflection. When you lower the value towards 0, the more light will pass through the surface and it will turn transparent.

***Transmission*** is the amount of light going through the surface. A value of 1 makes all light go through the surface making it completely transparent. It works closely together with the IOR which bends the light like water, glass, etc. This is not to be confused with Opacity which controls visibility.

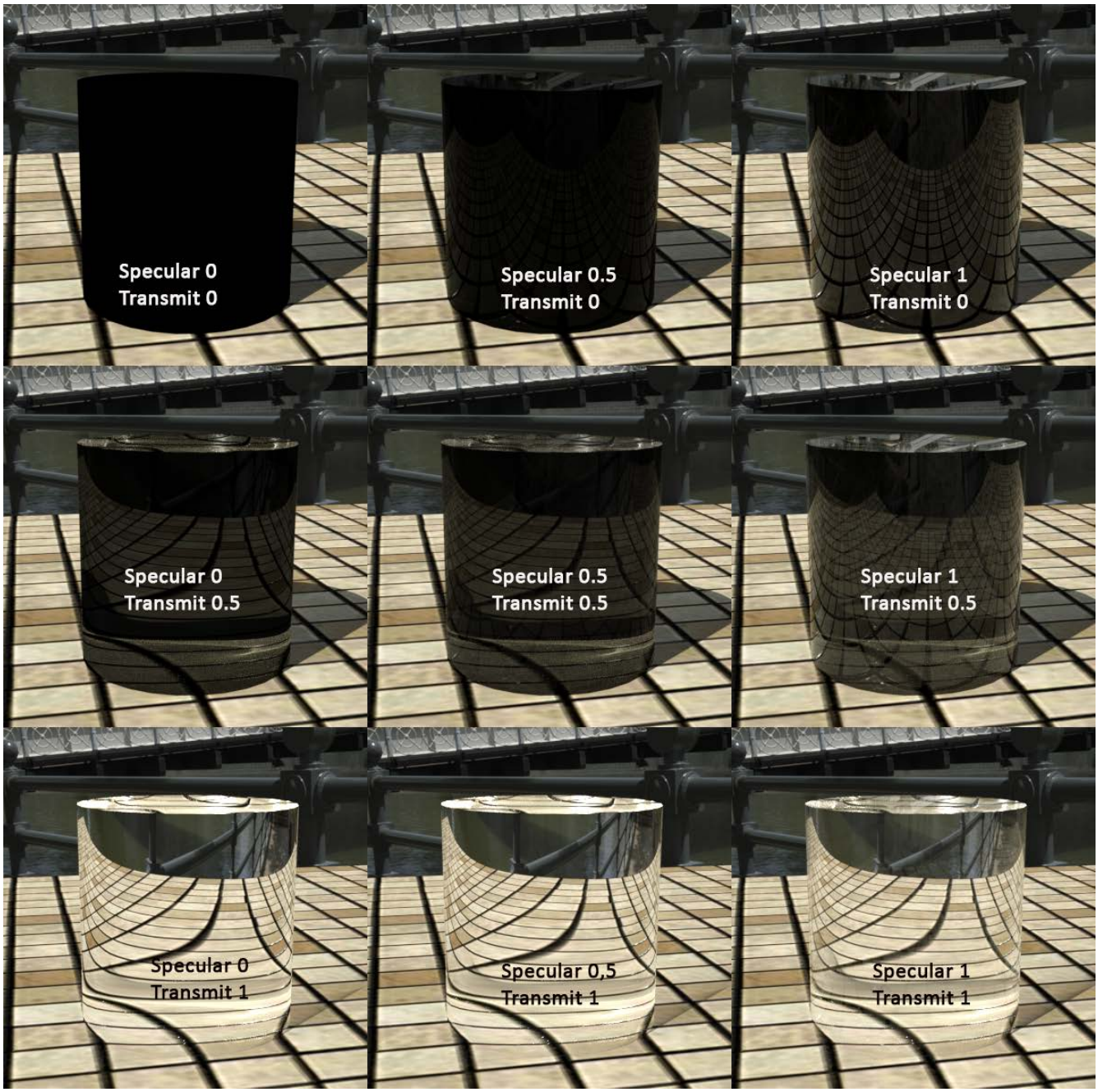



Figure 45

***Roughness*** defines how much reflections and transparency are blurred.

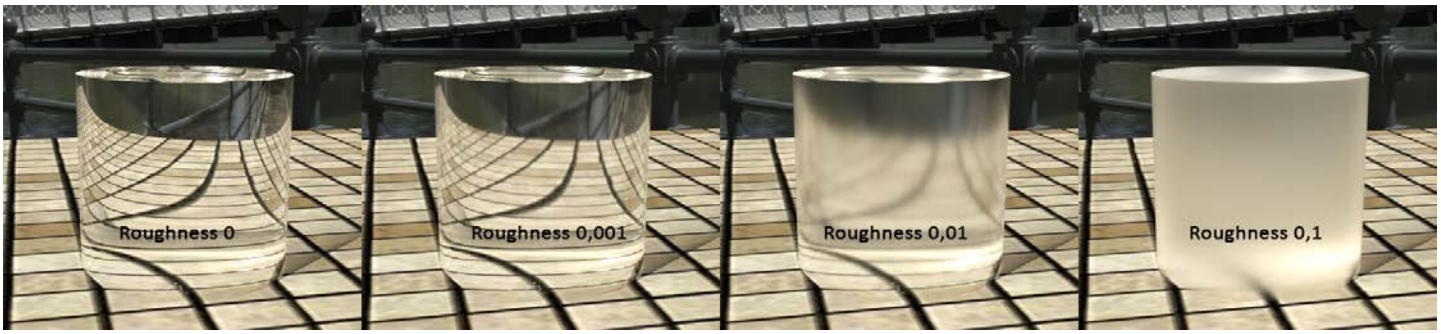



Figure 46

***Index of Refraction*** for specular materials controls how much light changes direction. A value of 1.33 represents water and a value of 1.5 represents glass.

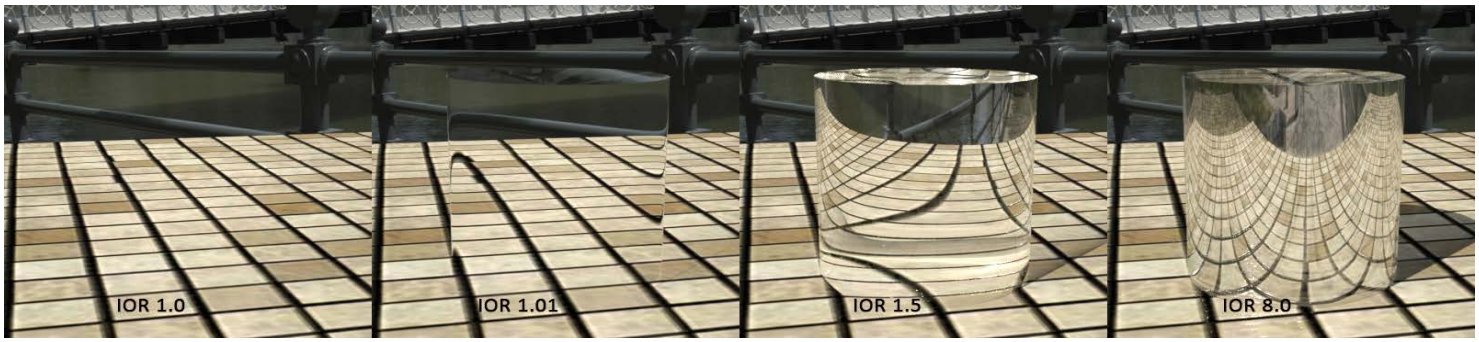



Figure 47

***Dispersion coefficient*** controls the change of color and how much diffusion takes place.

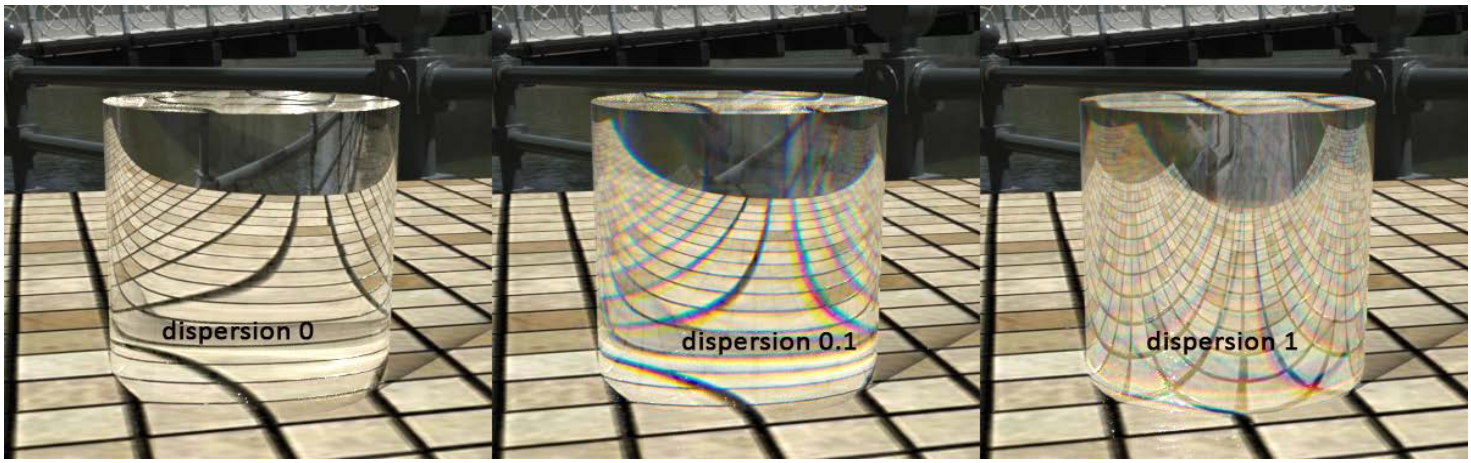



Figure 48

### *3.3.2.5 Material Mix*

Material Mix type is where you can blend different material types.

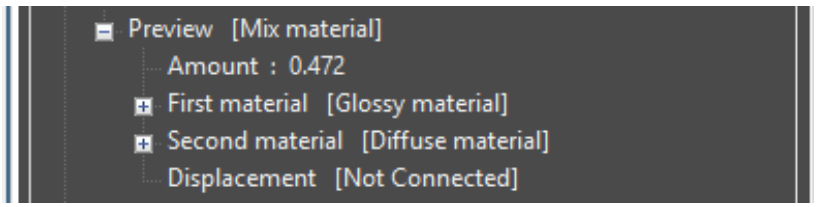



Figure 49

It is controlled by the Amount operator which can be any node type. First and second material can be any material type.

Figure 50

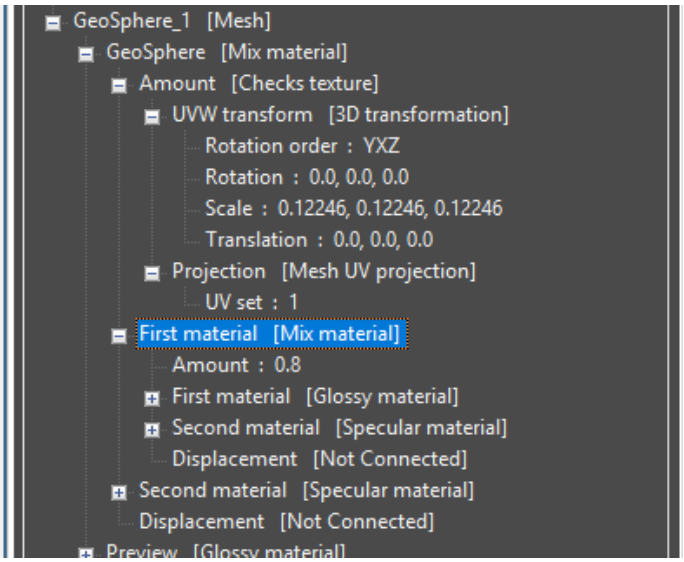



Figure 51

In the example above, I used a checkers texture as Amount operator, *First* material is a mix material (gold) which by itself is again a mix between a specular and gloss material. The *Second* material is a specular material (glass). *Displacement* can be applied to any single material or to all materials in the material mix.

### *3.3.2.6 Portal*

Portals are an aid to assist the render kernel in finding important light sources. You can set a portal up by providing a one-sided plane and assigning it the Portal Material. With the portal in place, the lighting quality will be improved and it will render faster.

One area where Portals are very useful is with windows where only outside light is present (daylight or HDRI lights) as shown in the example below.


Figure 52


Figure 53

## 3.3.3 Textures

A material often uses textures. They have several properties where you can manipulate them.

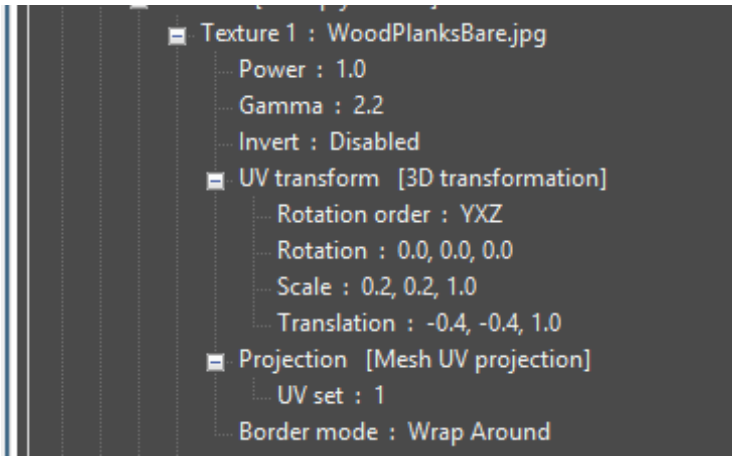



Figure 54

***Power*** is the texture strength.

***Gamma*** is the Gamma correction for the texture. 2.2 is usually the correct value for RGB textures and 1.0 for greyscale or HDR textures. In contrast to Poser, a texture map can have different gamma values for each occurrence.

***Invert*** creates a negative of the texture map. In Poser you would achieve that with a Color Math node where you subtract the texture from a white color.

***UV-Transform*** allows you to rotate, scale and offset (translate) a texture.

The offset is calculated different from poser when tiling is used. The plugin will make that calculation for you when the material is converted. The Scale Texture maps in the Material Type properties is a special macro which will rescale all textures in a material and calculate the offsets. Use the slider to scale the textures. For the more explanation on this and how to handle “shifting” textures, see the next chapter in this guide.

***Projection*** is the type of mapping which the texture should use. Normally this will be a UV Map in Poser, but it can be Box, Cylinder, Perspective, Spherical, Triplanar or XYZ to UVW too.

***UV set*** Octane can handle multiple UV sets, but those are not used in Poser.

***Border mode*** can be either *Wraparound* (tiling), *Black color* (remainder filled with black), *White color* (Filled with white), *Clamp* (cut off) and *Mirror* (repeated in reverse on opposite side).

## 3.3.4 Node Types

Octane has many node types, some are familiar to Poser users, and some Poser node types are lacking in Octane. In such a case Octane usually uses a different method to achieve the effect.

The following is a list of the most common ones in Octane and their purpose. The node type can be selected at the bottom part of the plugin window when a node is selected in the material.

For a complete explanation of all nodes, please refer to the Octane manual. Some of the nodes can be used for special effects which require knowledge beyond the scope of this User guide which is intended to get you started.

### *3.3.4.1 RGB Color*

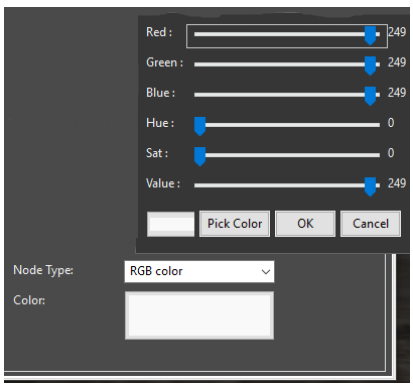



**Figure 55**

When an RGB Color node is selected you can select the color from the RGB slider, use the Poser color picker (click on color button in slider) or Pick a color from anywhere on screen: Click the Pick Color button, move the mouse over the color you want and press Space.

### *3.3.4.2 RGB Image*

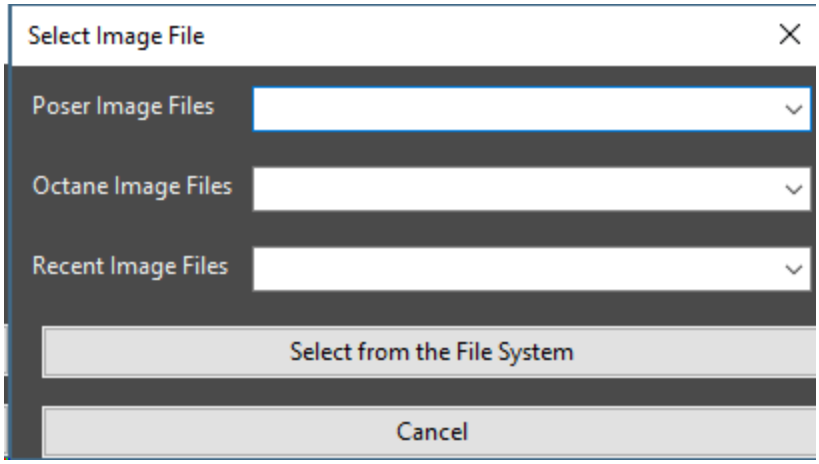



Figure 56

When you select the RGB image material node, a new window will open allowing you to select or browse for an image. Poser Image Files has a list of all the images in the Poser scene, even those not used in the octane materials.

Octane Image files contain a list of all the texture in the octane materials you have added.

Recent Image files contains a list of the last few images you have selected.

Select from File System will allow you to browse the file system for a new image.

### *3.3.4.3 GreyScale Color*

GreyScale Color is not actually a color, but a value between 0 and 1.

### *3.3.4.4 GreyScale Image*

Greyscale Image is essentially the same as Greyscale Color except that any texture loaded will be converted to greyscale if it is on color. This type is generally used for masks, bump and displacement maps.

#### *3.3.4.5 Mix Texture*

Mix Texture looks like the Mix Material type, but this one only acts on the texture or colors. When you select this node, it will change to three parameters: Value, First Texture and Second Texture. You can use a mask or a value in the value parameter. The First and Second textures are image files or colors or even other types like noise.

#### *3.3.4.6 Multiply texture*

Multiply texture is where you mix textures the traditional way. It will give you Texture1 and Texture2. All RGB binary values will be multiplied. Examples are adding a tattoo to a skin, changing the color of a texture.

#### *3.3.4.7 FallOff map*

This node has the same functionality as Edge Blend in Poser. It can change color depending on the camera angle. It is normally used as the value operator in a Mix Material or Mix node.

Depending on the angle, it will give a value of 0 (no angle) to 1 (maximum angle). This value is used to drive towards First or Second texture in the mix.

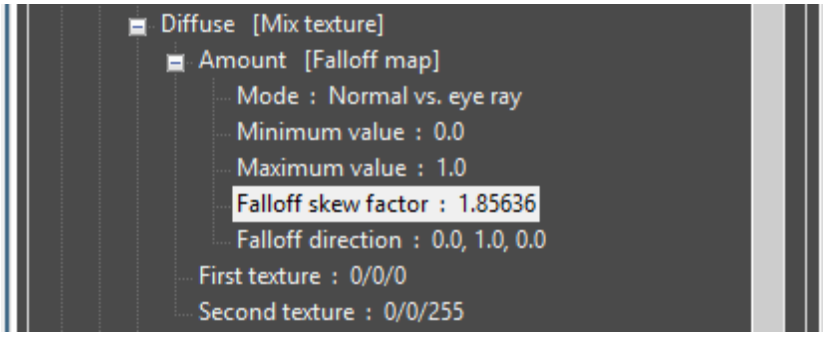



Figure 57

***Mode*** is default set at Norm vs Eye ray. This is how edge blend works.

***Minimum value*** defines the minimum amount it should start with.

***Maximum value*** defines the maximum.

***Falloff Skew Factor*** defines the strength of the effect (lower is more, higher is less).

***Falloff direction*** is only used for the other falloff modes. (See Octane manual for more info).

### *3.3.4.8 Noise texture*

The Noise node can generate a noise texture in different types.

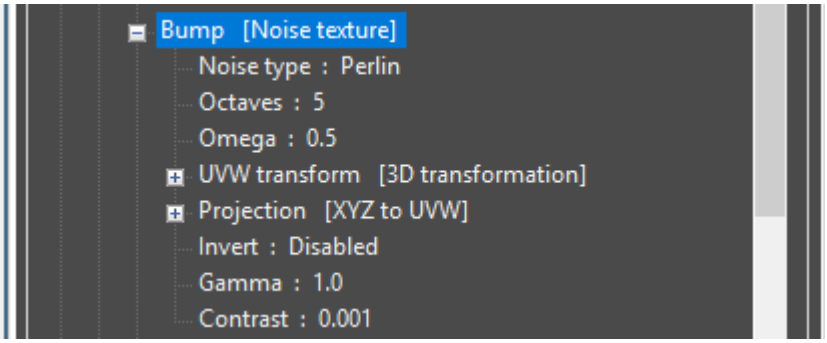



Figure 58

***Noise type*** can be Perlin, Turbulence, Circular or Chips:

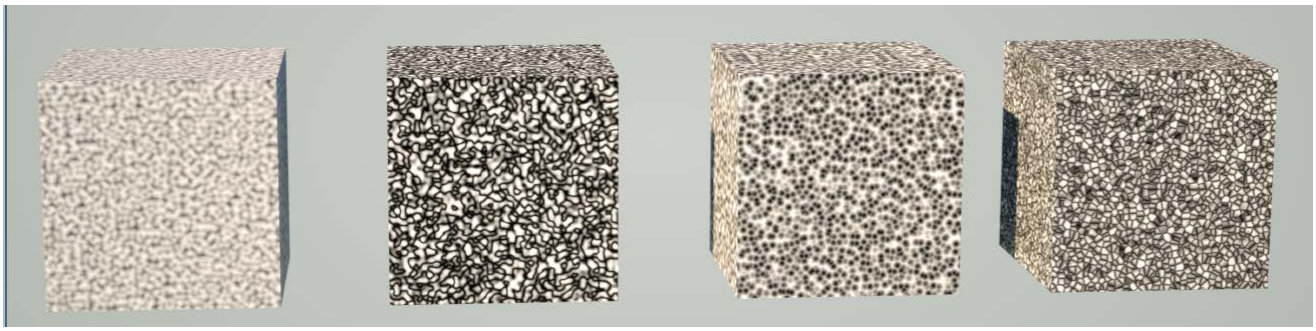


Figure 59

***Octaves*** the complexity of the noise

***Omega*** – variation within Octave (0-1)

The other parameters are the same as for a texture node.

### *3.3.4.9 Disconnect*

To remove and not replace a particular material node, use Disconnect.

Disconnecting is different as setting a value to 0.

# *3.4 Render Settings*

## 3.4.1 Kernel

There are 4 different render engines in Octane: Direct lightning, Info Channels, Path Tracing and PMC kernel. The most used one is the Path Tracing kernel, but the other kernels have their uses too.

### *3.4.1.1 Path Tracing Kernel*

The Path Tracing kernel is probably the most used kernel and gives the best photo realistic results.


**Figure 60**

***Max. samples*** is the maximum samples which will be rendered. You can stop the render any time when you think it is finished. When Max samples is reached, you can also opt for more samples by filling in a higher number. How many samples are needed depends on the scene and lighting.

***Diffuse depth*** is the maximum number of times a ray will bounce off of a diffuse surface. A minimum of 4 is suggested for outside environment and for internal environments with natural light (sun, sky) a minimum of 8 is suggested. Higher as 16 will not yield a better result. Too low number may result in render artefacts.

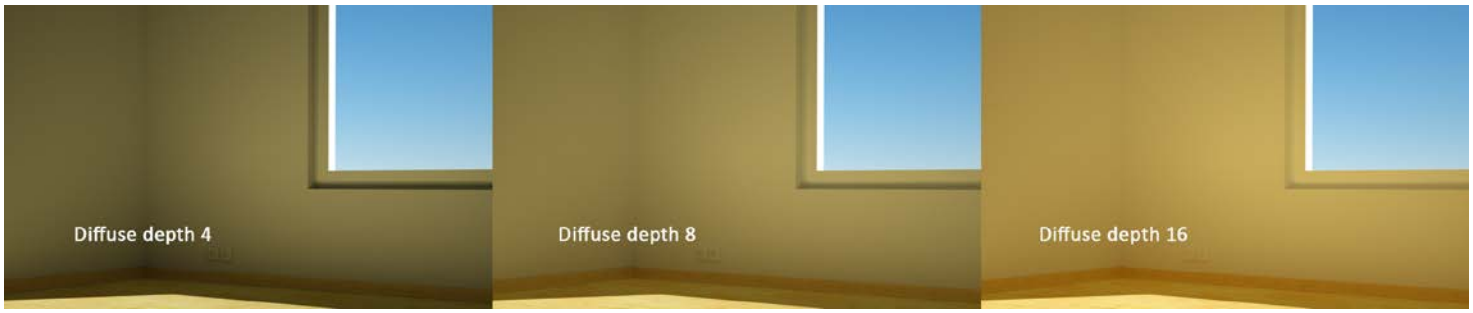



Figure 61

***Specular depth*** is the number of refractions of specular materials. Higher numbers will give better results but have higher render times. In the image below, you see that the default 24 value is not enough and even at 48 there are still refractions missing.
This type of image is better done with the PMC kernel who is better at caustics.

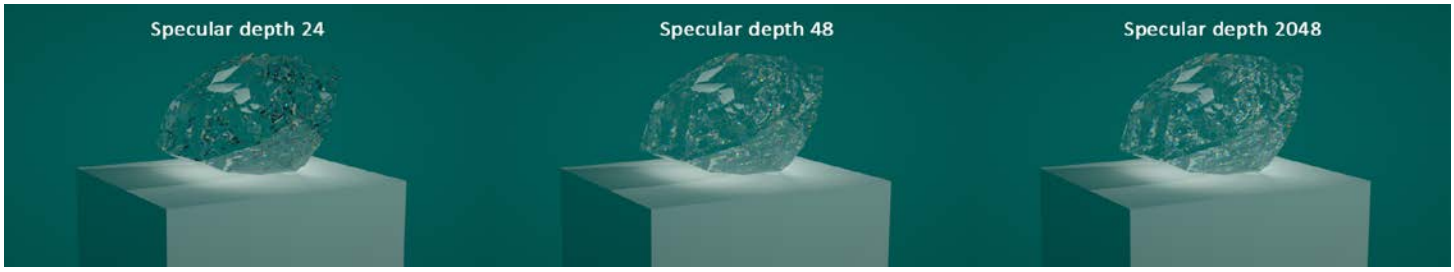



Figure 62

***Ray Epsilon*** is the distance where the render engine stops calculating shadows. A lower value means more accurate shadows, but longer render times. Lowering the value too much *may* cause self-shadowing where micro polygons cast shadows on its neighbours. It is similar to ray bias in firefly. For example on when to use it, see the *Troubleshooting Blacklines* section.

***Filter Size*** is the size of filter used for anti-aliasing. Setting it too high may cause blurring.

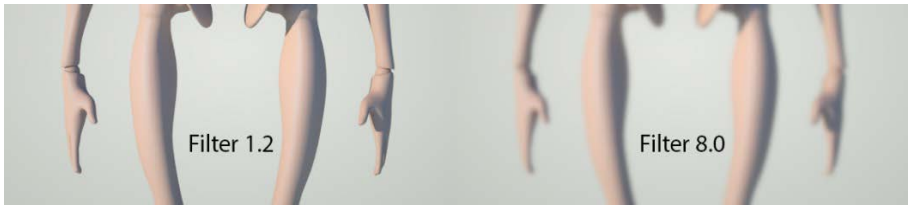



Figure 63

***Alpha Shadows*** Enabled causes the transparent parts not to cast shadows. If disable, the entire prop is treated as opaque for the shadow. In the example below a square is used with a transparency map.


Figure 64

***Caustic blur*** can be used to clear up noise faster, but a too high value will remove detail from the caustics. See separate section later in the guide for a chapter about Caustics.

***GI clamp*** is the max value for the amount of energy which is used for lights. In some cases, sparse but very strong light sources cause fireflies. Although they will clear up at some point, it may need a very high number of samples. By setting the GI clamp value a lot lower, fireflies will clear up sooner. For an explanation of fireflies, see the chapter in the Trouble shooting section.

***Irradiance mode*** this renders a single white bounce from each surface. Bump and displacement are disabled. This mode is used in Unity for baking materials.

***Alpha Channel*** removes the background images and sky colors while keeping its lighting effects. This is useful for compositing.

***Keep Environment*** keeps the background and sky color but creates an alpha channel so it can be used in compositing.

***Path term. Power*** is a tweak to improve the speed or rendering. A high value shortens the path and increases the number of samples/sec, but it may take longer to render dark areas (for noise reduction).

***Coherent ratio*** is another tweak where speed of the render is faster, but it will initially take longer to clear artefacts. It needs a few thousand samples to clear up.

***Static Noise*** is used in animations. It keeps the noise the same between frames. Without it you will get flicker in the animation.

***Parallel Samples*** defines the number of samples calculated in parallel. A higher number renders faster but requires more memory.

***Max tile samples*** is the buffer size for the samples calculation.

***Minimize Net traffic*** is used with network rendering.

***Adaptive sampling*** is a tweak where sample calculations will stop when a specified noise level is reached. This will allow you to set the number of samples very high.

***Noise threshold*** is the smallest relative noise level where adaptive sampling will kick in. Value should be in the range of 0.01-0.03. Changing this will not restart the render, so you can tweak this during the render.

***Min. adaptive samples*** is the minimum number of samples calculated before adaptive sampling will be applied. With a high noise threshold, you will need a higher number of samples here. Changing this will not restart the render.

***Pixel grouping*** is the size of the pixel group which will be evaluated for noise level in adaptive sampling. Only if all the pixels in this group have reached the threshold, adaptive sample will start. This will restart the render.

***Expected exposure*** should be approximately the same as exposure in the Imager section. Changing this will influence the adaptive sampling for dark areas. A setting of 0 will disable this (default).

***Deep image*** is used for Deep Compositing. It will add depth information to the image and you can use this in a compositing application to layer pixels depending on their depth. It has a much higher memory requirement as the usual images since it will contain the depth information. For more information on what Deep Compositing is, use this link: https://www.fxguide.com/featured/the-art-of-deep-compositing/.

***Max depth samples*** is the maximum number of dept samples which Octane will store.

***Depth tolerance*** is the threshold where depth samples are considered the same.

***Toon shadow ambient*** is the ambient modifier for Toon shadows.

#### *3.4.1.2 PMC kernel*

The PMC kernel uses a different algorithm as the Path tracing kernel. It is more geared to specular materials and caustics.

Most of the settings are the same as the path tracing kernel, but there are some differences.

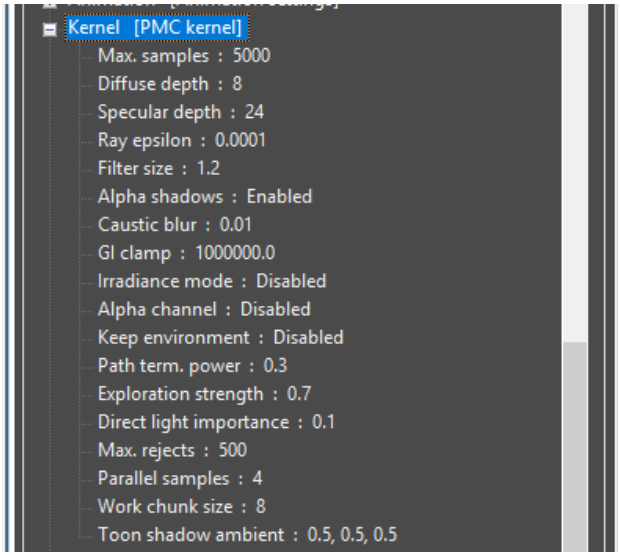



Figure 65

***Exploration strength*** defines how long the kernel should follow a good path. Lower values mean more noise, higher can create a splotchy image.

***Direct Light importance*** tells the render where to concentrate its efforts. A setting of 1 means it will concentrate on bright areas and lower values will give more importance to indirect light.

***Max. Rejects*** controls the bias of the render. A lower value will decrease render time but will make the render slightly more blurry.

***Work Chunk size*** is the size of the buffer, increasing it may increase the render speed, but will require more memory.

### *3.4.1.3 Direct Lighting Kernel*

See Direct Lighting.

### *3.4.1.4 Info Channel Kernel*

See InfoChannel Kernel.

# *3.5 Camera*

There are 3 different camera types: Thin Lens camera, Panoramic Camera and Baking Camera. The default is the Thin Lens camera which is a standard camera. The Panoramic camera can be used to render environments and the Baking camera can be sued to render textures from Octane materials.

## 3.5.1 Thin Lens Camera

This camera is controlled via the active Poser camera for position, scale, direction and Focal length. You can however only use the Main, Aux and Dolly cameras.

Figure 66

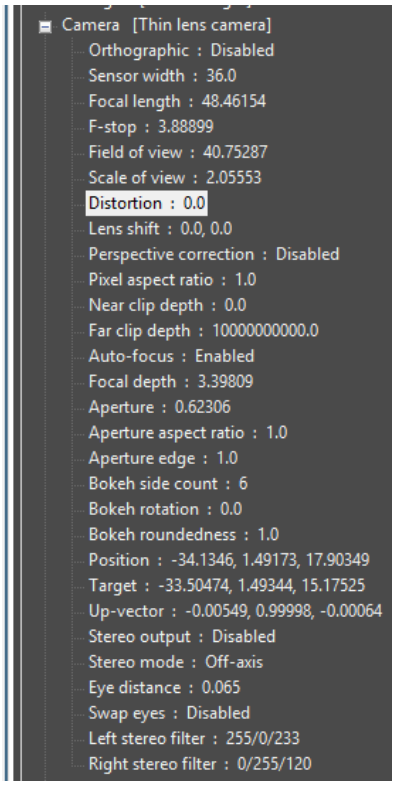



Figure 67

***Orthographic***, when enabled, will create an orthographic view of your scene. All objects will be projected on a single plane.

***F-Stop*** controls the aspect ratio between Focal length and Aperture. Use the Aperture to get full control over Depth of Field. A more in-depth explanation of Dept Of Field is in a later section.

***Distortion*** is the amount of spherical distortion.

***Lens Shift*** moves the camera left, right, up and down.

***Perspective Correction*** enabled will keep vertical lines parallel if *Up-Vector* is vertical.

***Pixel Aspect*** ratio allows to render non-square pixels.

***Near Clip Depth*** will hide objects from the camera up to the distance specified. Use it to hide objects blocking the view.

***Far clip depth*** defines the max distance which will be rendered.

***Auto Focus*** will enable or disable autofocus.

***Focal depth*** is the distance to the focal point.

***Aperture*** is the amount of depth of field to be rendered. setting it to 0 will turn it off.

***Aperture aspect*** ratio enables you to stretch and squash the depth of field sphere.

***Aperture edge*** makes the edges softer (high value) or sharper (low value).

***Bokeh side count*** are the number of edges for the bokeh shape.

***Bokeh rotation*** sets the bokeh shape orientation.

***Bokeh Roundness*** defines the roundness of each side of the bokeh shape.

***Up-Vector*** defines the up vector for the camera, default is Y axis (0,1,0).

***Stereo Output*** enables one of the following stereo outputs: Left, Right, Side by Side, Anaglyphic or Over-Under.

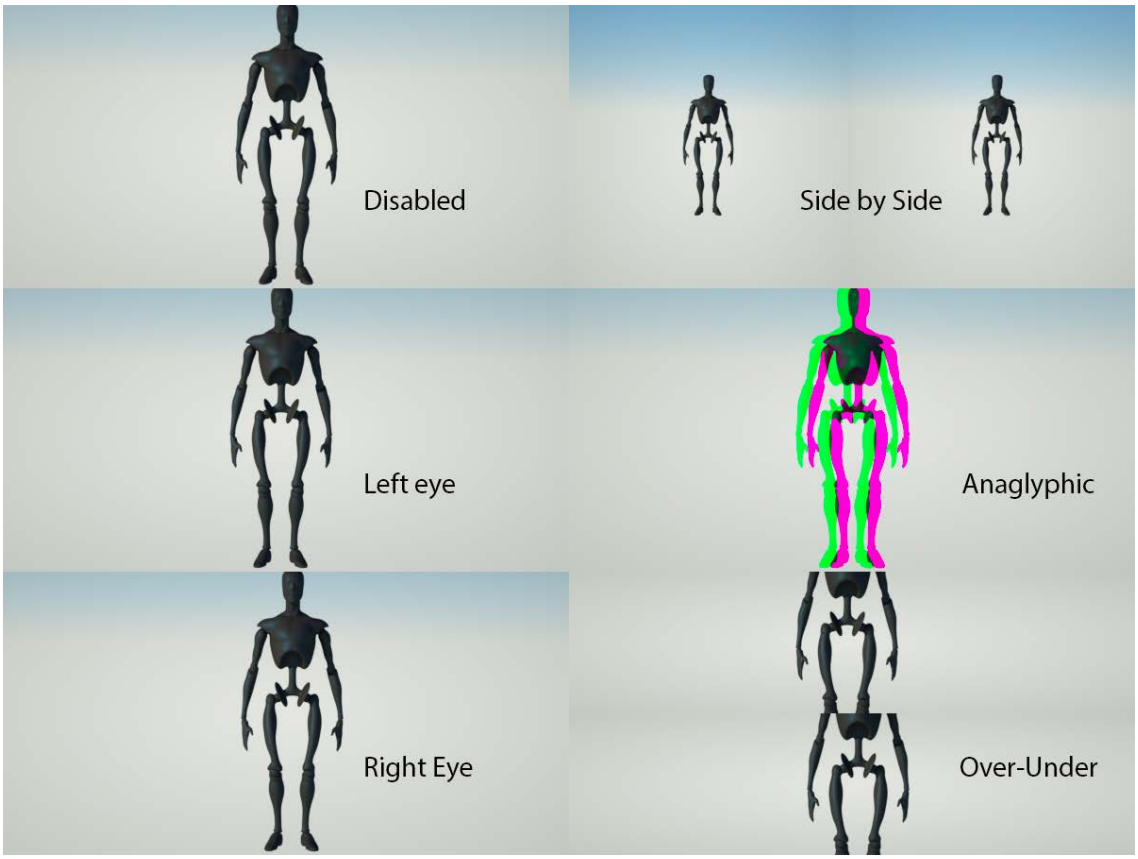



Figure 68

***Stereo Mode*** defines whether to use parallel cameras of off-axis camera views for each eye.

***Eye distance*** is the distance between left and right eye which Octane uses. Units are in meters.

***Swap Eyes*** swaps left and right eye camera

***Left and Right stereo filters*** tell octane what colors to use for the Anaglyphic stereo mode.

### 3.5.2 Panoramic Camera

The Panoramic Camera is controlled by field of view both in horizontal and vertical direction. It can be used to do wide screen projections.

Most of the parameters for the Panoramic camera are identical to the Thin Lens Camera with some additions.

***Projection*** sets the projection mode for the camera: Spherical (equirectangular), Cylindrical or Cube map.


Figure 69

***Horizontal field of view*** defines how much of the horizontal view will be rendered (up to 360°

***Vertical field of view*** defines the vertical view (up to 180°)

***Pano blackout latitude*** is used in stereo rendering and defines the minimum latitude at which the rendering is "blacked out". This is to avoid eye strain, when people try to look up or down at the poles.

### 3.5.3 Baking Camera

To be provided later.

## *3.6 Imager*

The Imager section deals with the adjustments to the look of the render. They can be applied during or after the render and are non-destructive except for the change between sRGB and film formats which will cause a restart of the render.

Octane provides several methods to correct its output to emulate different media types. They are combined to create the final render.

***Exposure*** controls the exposure of the scene. Higher values mean more exposure. If you need to increase it beyond 3, you should increase the light intensity to avoid noise.


**Figure 70**


**Figure 71**

***Response curve*** allows you to choose how the render reacts to increased exposure. There are several popular camera film formats and other formats to choose from. Default is sRGB. For more info see: https://docs.otoy.com/StandaloneH_STA/StandaloneManual.htm#StandaloneSTA/CameraResponseCurves.htm.

The original image above was rendered with and Agfa film format. The same image with sRGB will give you a slightly different color palette. In general film formats will give a warmer result.


Figure 72

***Neutral response*** enabled will remove the color change (tint) in the response curve.

***Gamma defines*** the gamma correction which should be applied. A value of 1.0 is linear, no gamma correction. Gamma correction is traditionally used to correct the physical limitations of display devices. In general, you should not change this value since gamma correction is already applied when an image is saved depending on the format.

***Custom LUT*** can define a Look Up Table which is used on modern displays for custom color distribution.

***Order*** defines in which order Response, Gamma and Custom LUT should be applied.

***Highlight compression*** will soften burnout highlights. It is a very subtle effect and different from hot pixel correction.


Figure 73

***White point*** allows you to change the color balance by setting a true white point. You can also do this via the octane viewport by picking the color which should be white.

In the image below a slight yellow is used as white point. You can use this setting to remove a color cast caused by one of the lights.


Figure 74

***Vignetting*** defines the amount of darkening around the corners of the image. Default is at 0.3 which gives a realistic effect. You can remove it or set it higher for an artistic effect.


Figure 75

***Saturation*** adjust the amount of color saturation.

***Hot Pixels removal*** set at 0 will remove the render artefacts caused by unresolved render calculations. See Hot Pixels in the tips and tricks section.

***Dithering*** will add noise to image to reduce banding.

***Minimum display samples***. Set it to a multiple of GPU cards you have.

## *3.7 Render Layer*

Each mesh object has a layerID. If Render Layer is enabled, only the active layer will be rendered, or, all but the active layer are rendered.

You can read a more detail explanation here: RenderLayers.

## *3.8 Render Passes*

See: RenderPasses

# *3.9 Conversion walk through*

The Poser plugin for Octane uses the Firefly node setup in Poser for conversion.
It will pick up all the textures it can find and connects them to the correct Octane nodes. Most materials will be converted to Glossy materials since this is the most common material type. It does not try to interpret the complete firefly node tree but will analyse the tree to find the textures and the texture scale and offsets. The default for the glossy material is set to 0.05 with specular with a roughness of 0.7. These and other defaults can be changed, see Advanced Topics on how to customize Octane.

There are 3 types of conversions: Simple materials, Complex materials and PBR materials.

## 3.9.1 Simple Materials

The *simple materials* are materials which have only textures and colors applied to them. The default conversion will do a good job, only thing you might want to change is the glossiness of the material since the default is not suitable for all type of materials. Wool will be different from leather, stone different from plastic.


**Figure 76**

In the above example I used two approaches. In one I changed the specular value of the glossy material to 0.003 and in the other one I change the Index of Refraction to 1.5.

In many cases a cloth or prop has several material zones, but most of the material zones will use the same material. You can use the Copy/Paste macros (described later in the guide) in the plugin to copy your new material to those zones.

## 3.9.2 Complex Materials

*Complex materials* are materials where a node structure has been built in the material itself: blenders, procedural nodes and math nodes. The plugin does not convert those, it picks up the image nodes which are used and uses those to construct the Octane material.

Octane does not have the same procedurals, blend and math nodes and there is no 1 to 1 conversion possible. If the complex node is a procedural series of nodes to emulate a certain material, it is often easier to use a material from LiveDB and adjust scale if needed. Adjusting the scale can be done with the scale texture maps option in the material properties or more accurately in the scale parameters of the texture nodes.

But when this cannot be done because of UV mapping, we have to construct the complex material ourselves. In most cases it is a relatively simple thing to do.

*Blender* nodes in Poser are the same as Mix nodes in Octane. They have Color/Texture nodes as input and a value/amount node as blend or mask parameter. Like Poser, you can nest these blender nodes as deep as you want. Edge Blend in Poser can be achieved with the falloff node.

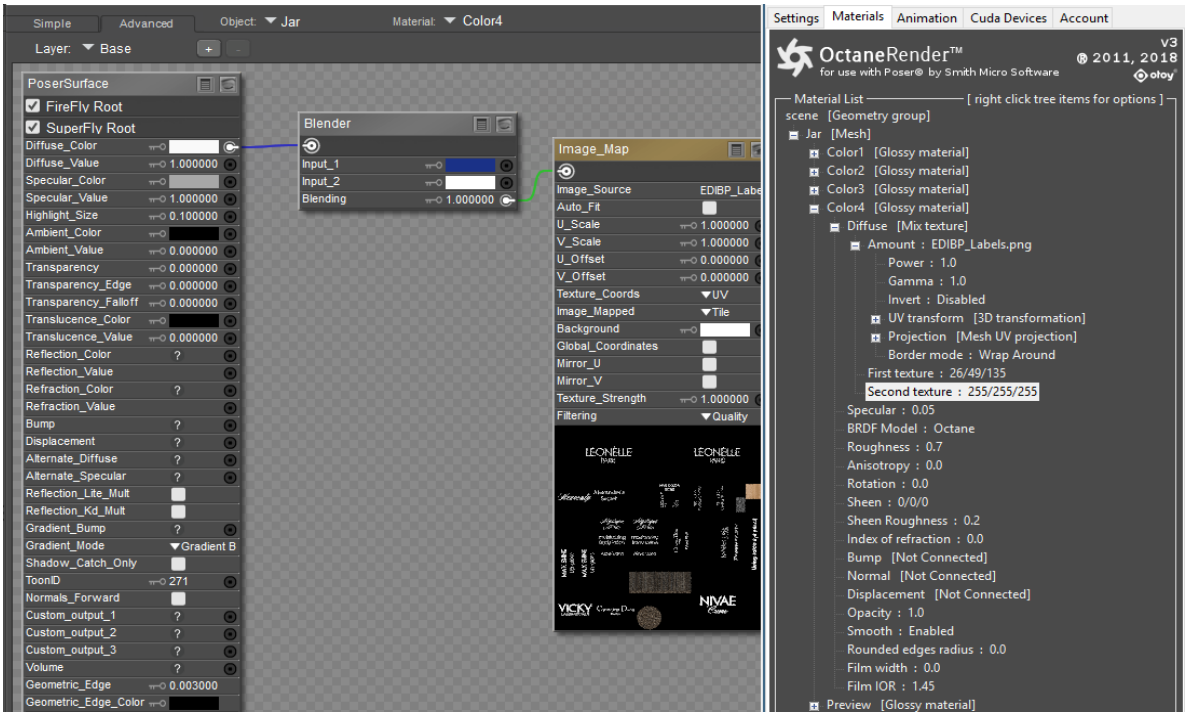



Figure 77

Poser has a lot of *procedural* materials, some of the most common ones can be found in Octane: Turbulence, noise, marble, Color Correction, gradient, color ramp but for others there is no direct solution. In some cases, using tiled textures is a workaround.
OSL (Open Shader Language) has been added recently and that will open up a range of already existing shaders to Octane.

The most used ***Color Math*** nodes have been implemented in Octane: Multiply, Add and Subtract Texture.

### 3.9.3 PBR Materials

Direct support for PBR (Physically Based Rendering) materials have been added in Octane 3.08 for the Metal-Rough workflow. This workflow is used in Superfly and iRay as well, so textures created with Substance Painter and Quixel van be used directly without any additional node structures.

The Metal Material type allows you to plugin the PBR textures in directly. Currently Octane does not support SSS in the metal-rough workflow, but that will be added in a next update as well as other features.

# 4 Advanced Topics

## *4.1 LiveDB*

LiveDB is the material library of Octane. This is a user contributed library and contains many useful materials. The quality differs and there are many duplicates, but they are an easy way to create your own materials.

LiveDB can be found if you select a material and right click. In the context menu select LiveDB and a new window will open.

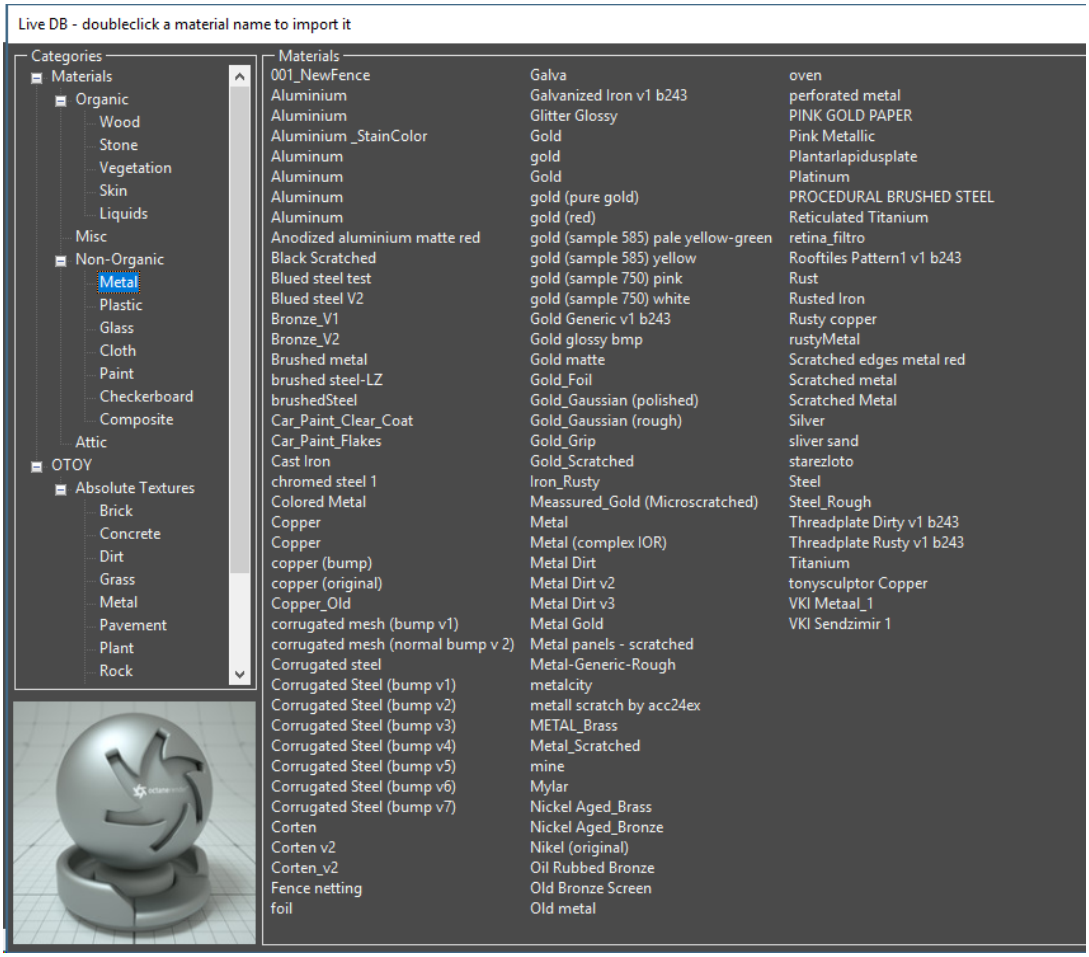



Figure 78

The tree on the left has the different categories. The uncategorized materials are in Misc. Select a category and on the right, you will see a list of materials. Click on one and a preview will be shown in the bottom left corner. To load the material, double-click it.

# *4.2 Lights (Advanced)*

## 4.2.1 Daylight Environment

Daylight can be combined with Texture environment by plugging the HDRI in the Sky Texture node. The advantage of doing this, is that you have more control over the shadows because the sun is active.


Figure 79

The following are the other parameter which you can use for control over the texture Environment:

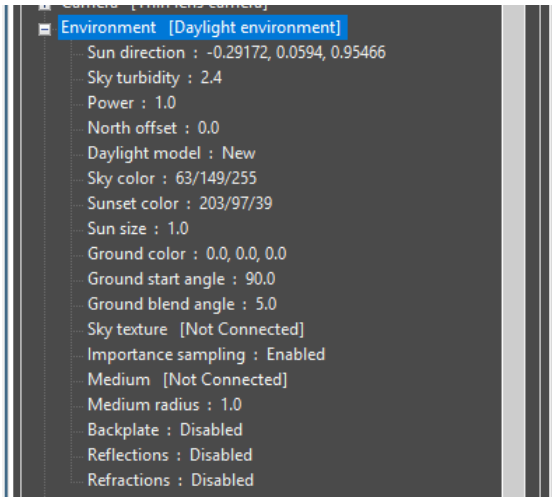



Figure 80

***Sun direction*** is controlled by the Poser Infinite light in Poser. I cannot be edited directly in the plugin.

***Sky turbity*** defines how much sun rays are scattered. A higher value will decrease the contrast between shadows and lit areas. It does not work in combination with a sky texture.


Figure 81

***Power*** is the strength of both the sun and the Sky texture.

***North Offset*** is not used by the plugin.

***Daylight model*** is used for compatibility with v2 of octane and older.

***Sky Color*** is the color of the sky in Daylight environment when Sky texture is not used. Sky color can be changed.

***Sunset Color*** is the color of the sunset (when sun is close to horizon. You can change this if you do not want it.

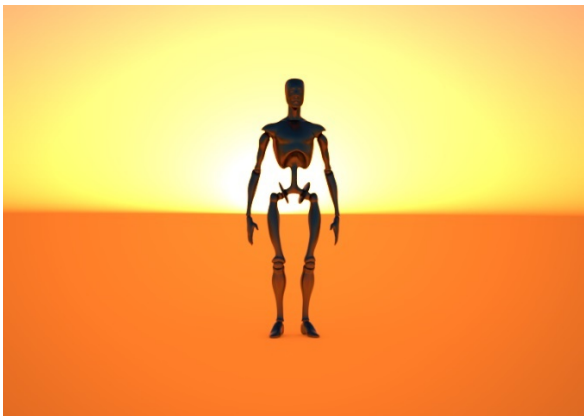


Figure 82

***Ground Color, Ground Start Angle and Ground Bland Angle*** control the color of the are below the horizon. I have not found a practical use for it.

***Sun size*** controls the size of the sun. By making it larger you soften the shadows like in the image below.


Figure 83

***Sky Texture*** will override Octane's sky and replace it with a HDRI. It will contribute to the lighting as well as provide a background. See *Texture environment* for more information.

***Importance Sampling*** enabled will give more attention to important areas in the sky texture (those who contribute more light) which will de-noise the render more quickly.

***Medium and Medium Radius -*** see at separate section on fog and volumes later in this guide.

***Backplate, Reflections, Refractions*** – see Visible Environment below.

## 4.2.2 Texture Environment

First a few words about HDRI. An HDR file contains not only color information, but also brightness information. This means that a standard image (such as JPG) converted to HDR format will not contain the brightness information and make it less suited to use as HDRI image. So, for best results always use an unclipped HDR image for lighting.
HDRI Haven (https://hdrihaven.com/) has many free unclipped HDR images at all kinds of resolutions.

HDRI mapping is spherical (equirectangular) which has a 2:1 aspect ratio.
The environment can be rotated by changing the Y rotation value in the sphere transformation.

The HDRI will be used for both lighting and background. With the Visible Environment (described below) you can override this.

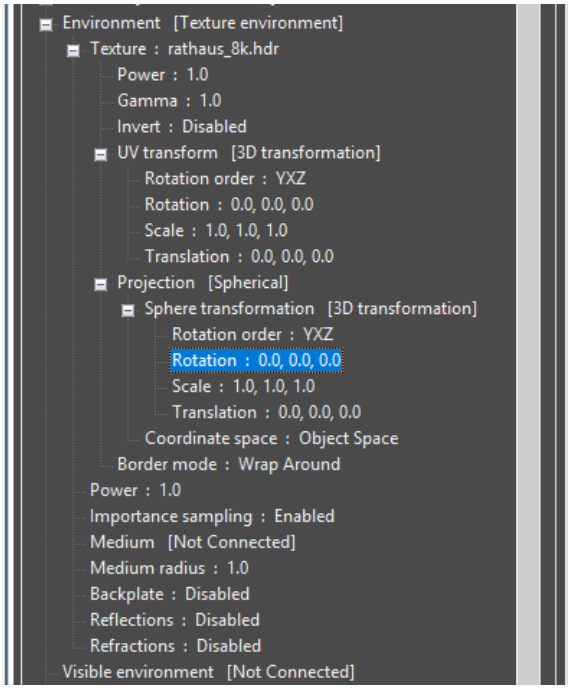



Figure 84

***Texture*** assigns the HDRI, make sure that gamma is set to 1.

***Power*** is the strength of the HDRI, you can use this to increase the light contribution of the HDRI.

The other nodes are identical to the Daylight Environment.

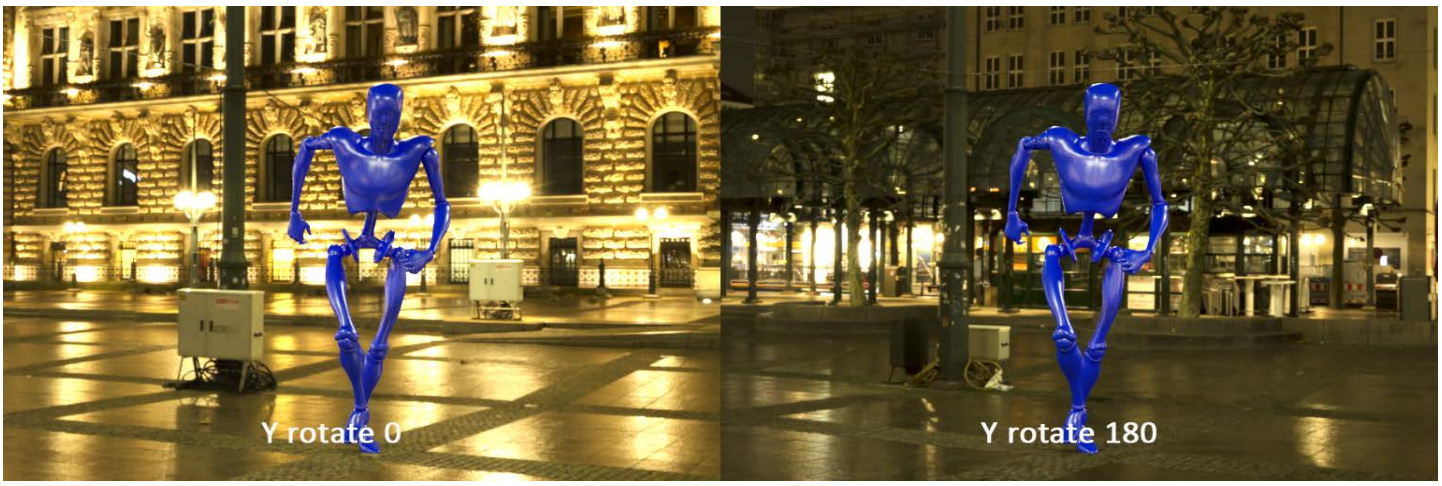



Figure 85

## 4.2.3 Visible Environment

With the visible environment you can separate the lighting from the background. For example, it will allow you to use a HDRI which shows a sunset and use another HDRI or Daylight to provide the lighting.

In the Visible environment you can choose either Texture Environment or Daylight Environment. The only difference is that now the *backplate*, *Reflections* and *Refractions* options can be used.

In the image below, I used a Daylight environment, a ball with a mirror reflection, and a cylinder with a glass material. In the Visible Environment a HDRI I attached an HDRI. All other options are turned off.

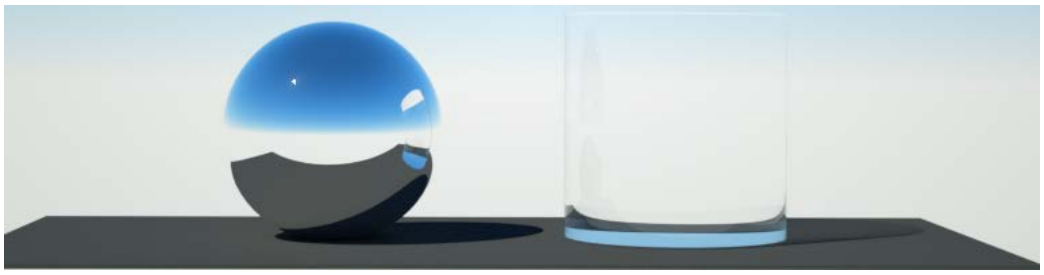


Figure 86

In the next picture I turned on *backplate*. The background has now changed to the HDRI, but reflections and refractions are still from the daylight environment. Daylight however still provides the light.

Figure 87

Now I turn on Reflections and the ball now shows the environment in its reflections.

Figure 88

When refractions is turned on the complete environment is complete but with light from the Daylight environment.

Figure 89

In the last render here I changed the sun direction, altitude and power.

Figure 90

## 4.2.4 Mesh Lights and IES distribution maps

IES distribution maps define how light behaves. It is an industry standard which is used in the real-world lighting industry and many 3d render engines have adopted this as well. You can find more information on how they are created on this link: http://www.cgarena.com/freestuff/tutorials/max/ieslights/. There are also may free IES files available on the web and at lighting manufacturers sites. One very useful tool to have is the free IES Viewer. It can generate a thumbnail of the actual light and this makes it easier to find the right one.

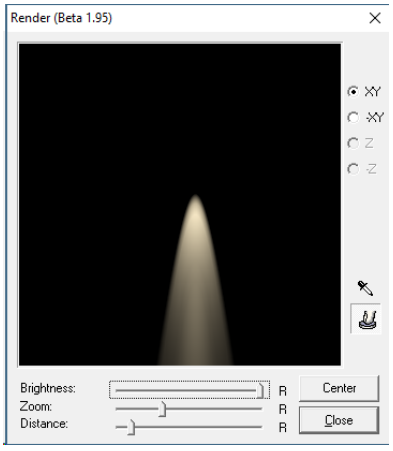



Figure 91

The thumbnail above represents one of them.

To use the IES file in Octane, we need to add them as a distribution map in the Emission node. Use a one-sided plane as mesh light (like the one produced with the Plugin macro to convert a spotlight in to a mesh light.

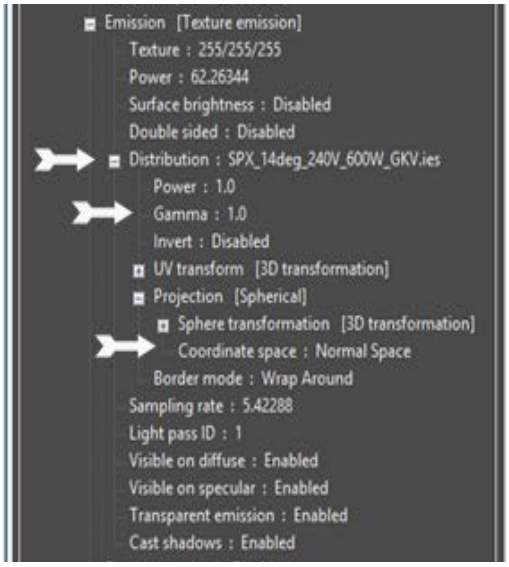



**Figure 92**

***Distribution*** contains the IES file, RGB texture and gamma must be set to 1.

***Coordinate Space*** in Projection must be set to *Normal Space* otherwise it will face downward.

**Figure 93**

To make shadows softer or harder change the size of the one-sided plane. Below the size of the plane has been reduced to 2%.

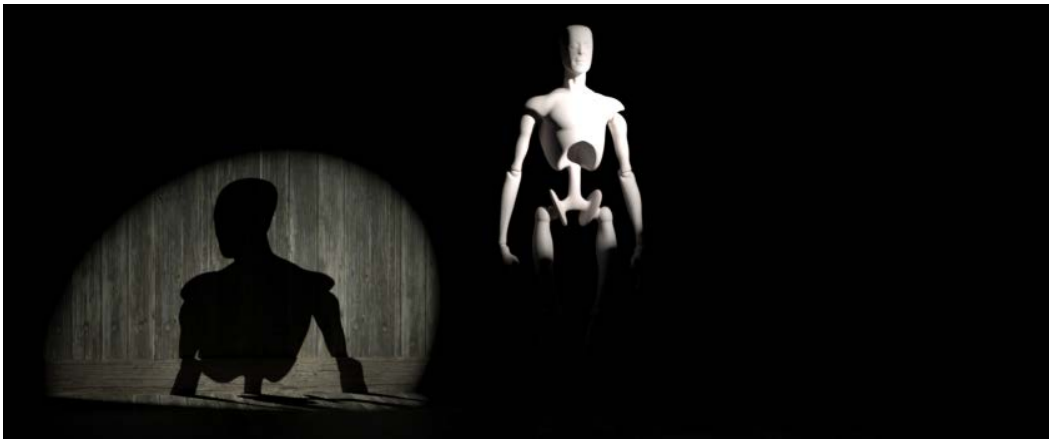


Figure 94

Mesh lights with IES distribution maps are the most flexible way to set up photographic lighting and can produce wonderful results.

## *4.3 Caustics*

Light caustics are sometimes difficult to do. It requires both a good mesh and accurate lighting to get good results, especially with glass where the light not only bounces around but also leaves the object with a color shift.

The color shift is determined by the dispersion value on the specular material. A higher value will increase the color shift. The following link has the dispersion values for many materials: https://refractiveindex.info/?shelf=3d&book=glass&page=BK7.

The following example has a closed room with a thin split in the wall. On the outside daylight is causing a light beam which falls onto a prism. This is the poser scene.

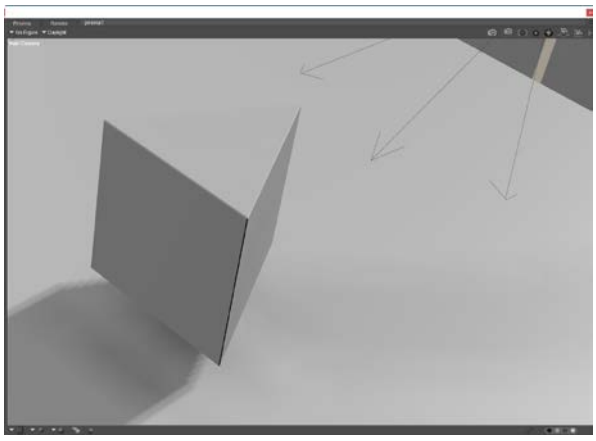


Figure 95

For the materials I used a specular material with an IOR of 1.509 and a dispersion coefficient of 0.062. The render settings have used a caustic blur of 0.02. A value of 0 does not show caustic shadows, so I set it to a very low value.

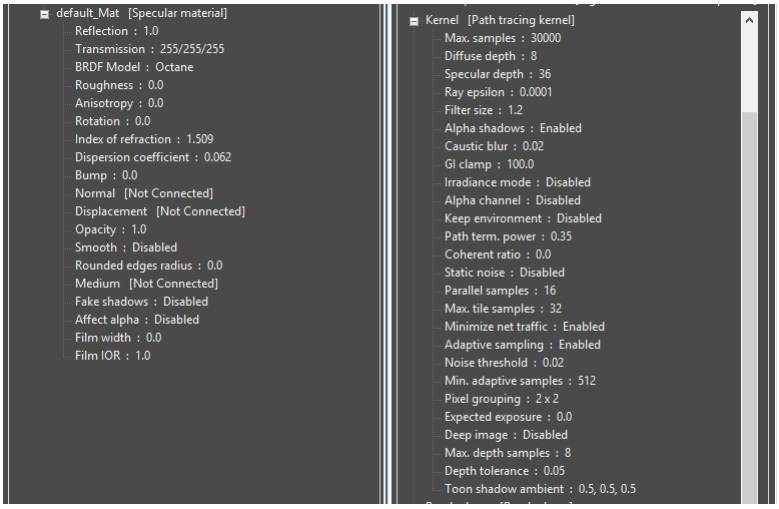



**Figure 96**

The following is the result with Path tracing.

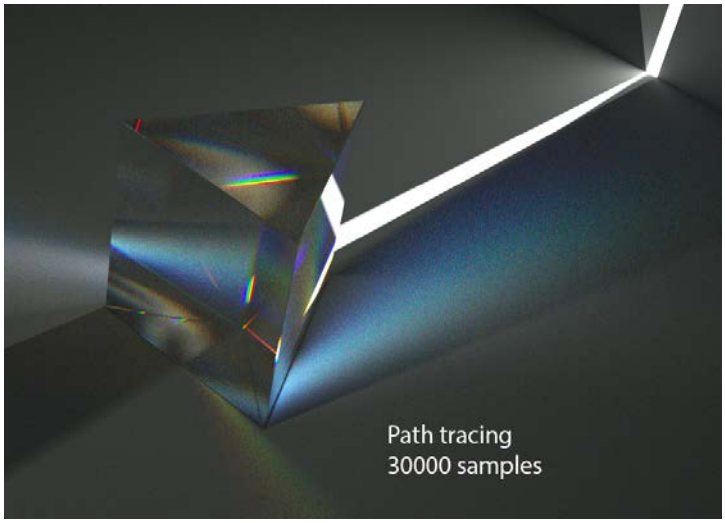



**Figure 97**

When a scene has a lot of specular nodes and has caustics, PMC can be used. It renders these a lot faster (clearing up noise).

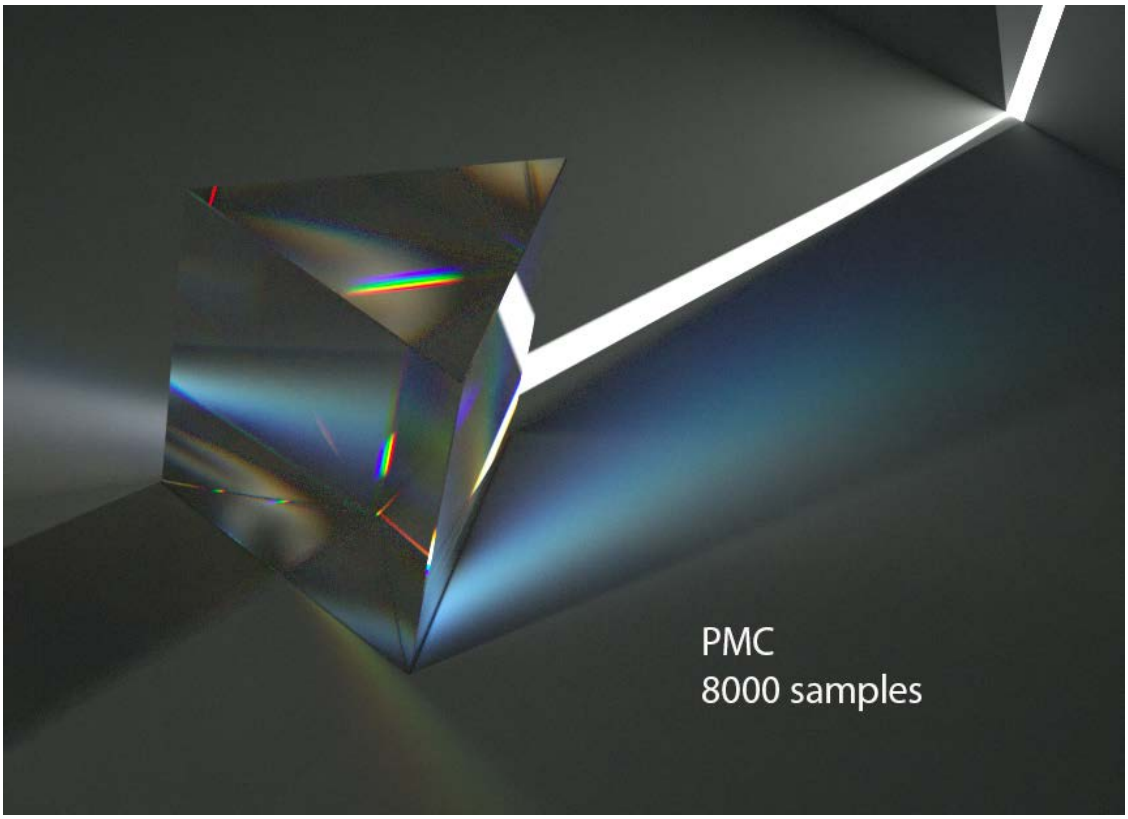



Figure 98

## *4.4 Import and Export*

### 4.4.1 Poser Scenes

You can import and export Poser scenes to Octane native format – OCS or ORBX.

***OCS*** contains only the settings for a scene, prop or material, it does not contain geometries, textures or other external files. Although small in size, file references are broken when those files are deleted or moved. Since Octane stores absolute path references, this may easily happen.

***ORBX*** is a package containing all the resources (settings, geometry, textures and external files). The package is self-contained and can move from one pc to another. Poser itself uses relative paths and uses a search algorithm to find geometry and textures, which Octane does not do. So, the preferred way to save scenes is in ORBX format.

### 4.4.2 Materials

Materials can be exported and imported from within the materials tab to either POE or POC. ***POE*** is the material for a single material zone and ***POC*** is the entire collection of materials within a geometry. Right click on a material and select Export Material to export it as a single POC material. Select the geometry and Right Click and select Export all Materials for the Figure/Prop to save it as a POC.

Import acts the same way: Select material to import a single material, select geometry and import as a material collection. See the Macros section for the extra information on this.

### 4.4.3 LocalDB

You can also export a single material as ORBX, this will store both the material settings, textures and external files. By default, it will be stored in the ***LocalDB*** folder defines in the setup screen.

### 4.4.4 Render Target settings

Settings for the render target (camera, environment, film settings, animation settings, Kernel, render layers and passes, imager and post processing) can also be exported and imported. Right select render target and choose ***Import or Export Rendertarget*** settings. You can also save a render target as default on startup of a new scene.

### 4.4.5 Scene settings

There is one special import and export method: ***Import and Export All Octane settings***.
The plugin saves all settings for Octane in the Poser scene file. So, upon load, all octane plugin settings are restored. The import and export All Octane settings allow you to save all the settings in a separate file. This is not only useful as a backup method might something happen to your scene file, but it also allows you to export the settings of one scene to another.
The scenes do not need to be identical, the import will only import the settings of the scene where the geometry and materials are the same. This is ideal for a series of scenes where you use the same props and figures. But it also allows you to save all the material settings for an entire scene after you converted it.

### 4.4.6 Load Background Image

This option allows you to import a background image into the Octane viewport.

### 4.4.7 Upload Frame to Octane Cloud

You can upload a frame to the Octane cloud instead of rendering it on your PC. You need a subscription to use this feature.

# *4.5 Macros in the Poser plugin*

The plugin provides a series of macros which will help you convert Poser scenes: Poser Scripts, accessible from the Scripts Menu in Poser (Octane Render for Poser) and Plugin Macros, which can be accessed by right clicking a mesh or material.

## 4.5.1 Poser scripts

### *4.5.1.1 Clear All Octane Data*

This will remove all Octane data from the Poser scene.

### *4.5.1.2 Compress Texture Maps*

Video RAM sometimes is precious and sometimes you will want to decrease the size of textures to load larger scenes. This script will open a new window where a list will appear with all the props in the current scene. Select a prop and then select the textures you want to convert. Then press the Compress button.

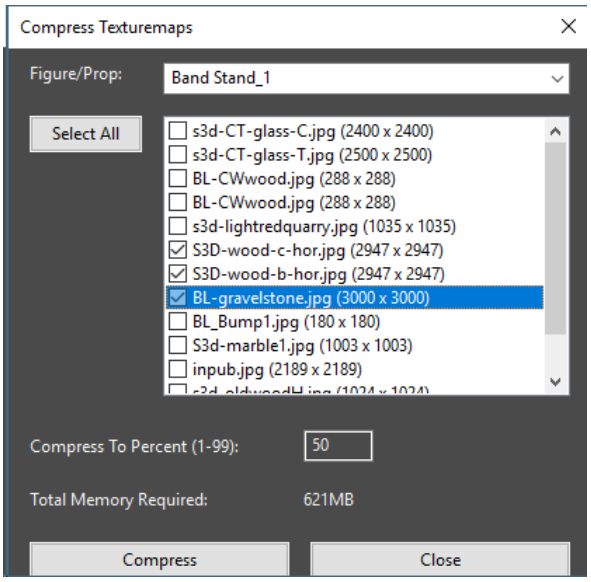



**Figure 99**

The texture size will be reduced by the specified amount (Compress to Percent) and stored in the Octane textures folder. The textures now will be replaced with the compressed ones in Poser. The original textures remain intact. Use this script before you start the plugin. If you already started it, use Reconvert from Poser for that prop.

The Out of Core option in Setup makes this script a bit obsolete, but it is still useful if you want to reduce texture size to increase render performance.

#### *4.5.1.3 Create Emitter from Current Light*

This will create an emitter for the current selected Poser light. This will only work for Point lights and spotlights. The script will ignore infinite lights and area lights. When the plugin is started for the first time and it will convert the first infinite light to an Octane Daylight. If a diffuse IBL light is found it will put the IBL image in the environment texture.

#### *4.5.1.4 Create Emitters from Lights*

All valid lights (see above) in the scene will be converted to emitters.

#### *4.5.1.5 Scene Texture Counter*

The script will produce a list with all the props, textures, texture size and total number of textures and total memory required to load them.

#### *4.5.1.6 Set All Hair Props to Populated*

When poser dynamic hair is used, the plugin required all the hair props to be populated (See Poser dynamic Hair section). This script will do that for you.

#### *4.5.1.7 Weld Selected Mesh*

This script will try to weld an unwelded mesh. Use this if mesh breaks occur when rendered in Octane.

### 4.5.2 Plugin Macros

All the plugin macros are accessed via the right mouse button click. The type of macros available depends on the active selection in the material tab: Scene, Figure/prop or Material.

#### *4.5.2.1 Scene*

***Reconvert all Materials from Poser*** will remove all octane materials in the scene and revert to the default conversion.

***Remove bump maps from the Whole Scene*** will remove all the bump maps from the scene to conserve video memory usage.

***Reload All Texture maps*** will update all the texture maps to the latest version. Use this if you have changed one or more texture maps in an external application.

***Reset all Hair widths to default*** will reset all the hair widths to default size.

#### *4.5.2.2 Figure/Prop*

***Save All Materials for this Figure/Prop as Poser Materials*** will write the Octane material in the Poser material tree. The nodes will not be connected, so do not use the Remove Orphans macro in Poser. When materials are saved this way, they will automatically be used and do not go through the default conversion. You can save the material in the library or on the figure as if they were Poser materials.

***Export all Materials for this Figure/Prop*** will save the octane materials for this figure or prop as POC.

***Import all Materials for this Figure/Prop ! Use Imported Image maps*** will load all materials for the figure or prop from a POC file. All textures, transmaps, bumpmaps and other maps will be replaced.

***Import all Materials for this Figure/Prop ! Retain Existing Image maps*** will load all materials for the figure or prop from a POC file but will keep the original images where present. All settings from the POC will be used and new images will only be added if the original material does not have them. This is a very powerful feature similar to EZSkin.

***Reconvert Materials for this Figure/Prop from Poser*** will revert to the default conversion for this figure or prop only.

***Remove Bump Maps from this Figure/Prop*** will remove the bumpmaps from the figure or prop.

***Reload All Texture maps*** will load all texture maps again.

***Paste <material node type> to all Materials for this Figure/Prop*** will *only* appear when a material has been copied to the clipboard. It will apply that material to all materials in the selected figure or prop.

***Paste <material node type> to all Skin Materials in this figure or prop***. This is a special version of the above where the target materials are member of the skin list as specified in OctaneDefaults.py (See the OctaneDefaults.py section). When it is applied it will keep the original textures, so torso, head and limb textures are retained.

#### *4.5.2.3 Material*

***LiveDB*** will open the LiveDB window where you can select an Octane material.

***Export Material*** will save a material as POE (single materials).

***Import Material ! Use Imported Image maps*** will load a POE material, replacing image maps.

***Import Material ! Retain Existing Image maps*** will load a POE material, keeping image maps.

***Save Octane Nodes to Poser Material*** writes the Octane nodes to the Poser material where it can be saved in a figure or prop or saved in the library.

***Load ORBX Material from LocalDB*** loads the octane material from LocalDB or other location.

***Save ORBX Material to LocalDB*** saves the Octane material to LocalDB or other location.

***Reconvert this Material from Poser*** replaces the current node with the default conversion.

***Reload al Texture Maps*** will load the updated texture maps.

***Paste <material node type> ! use Copied Image maps*** will *only* appear when a material has been copied to the clipboard. It will apply that material replacing image maps.

***Paste <material node type> ! Retain Image maps*** will *only* appear when a material has been copied to the clipboard. It will apply that material keeping existing image maps.

***Paste <material node type> to all Materials named <selected material name>*** will *only* appear when a material has been copied to the clipboard. It will apply that clipboard material to all materials with the same name as the selected node. This is a very powerful node when materials with the same name actually have the same material.

## *4.6 Texture shifting*

Texture offsets are not converted if the scale has not changed, you need to add those manually. If the scale has changed, the offsets are converted from Poser, ***unless*** the texture offsets are from other (math nodes), in this case use the following formula: (uScale/2)-1 + Poser offset, (vScale/2)-1 + Poser offset.

# *4.7 Fog and Volumes*

There are several ways Octane can produce fog like environments, each of them having different advantages.

## 4.7.1 Scattering Medium in Daylight and Texture Environment

This method can be used to get a fog like atmosphere in a scene. You select it in the medium node in the daylight or texture environment.

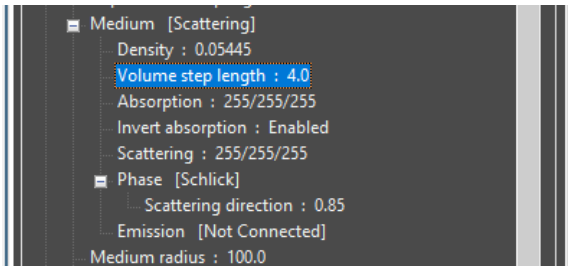



**Figure 100**

***Density*** controls how thick the atmosphere. It works together with the *Scattering* node and the *Medium radius*.

***Absorption*** controls how much light is absorbed, Black or zero means no absorption.

***Invert*** absorption is the inverse of the color.

***Scattering*** defines how much light is scattered.

***Phase*** is the direction of the scattered light.

***Emission*** is not used for fog.

***Medium Radius*** is the size of the fog.

You start off with the size of your scene in meters. Set this size as your *medium radius,* then set the scattering to white or 1, the *absorption* color to black or white with *invert absorption* enabled. Now you can control the amount of fog with the *Density* value, 0.05 would be a good value to start. The Phase should be set anywhere between 0.70 and 0.95. Setting it lower results in a too dark image because too little light is scattering in our direction.


Figure 101

The above image shows the influence of the phase with a density of 0.85 and a medium radius of 100 (size of the scene).

The image below shows the influence of the density with a phase of 0.85.


Figure 102

## *4.8 Customizing the Poser Plugin for Octane*

There are 2 places where the plugin behaviour in UI and defaults can be changed: The Setup Window and the OctaneDefaults.py file. You will need Admin privileges to change the OctaneDefaults.py script.

### 4.8.1 Setup Window


Figure 103

***Refresh time*** is how often the viewport will be updated during render.

***Allow Scaled Poser Camera*** is a setting where scaled cameras are allowed, however a scaled camera will have a different view in the Poser window and the Octane viewport. It is off by default so you do not get confused when something different is rendered as is shown in Poser.

***Poser to Octane Scale Factor*** is the ratio to match Octane scale to Poser scale. Do not change this.

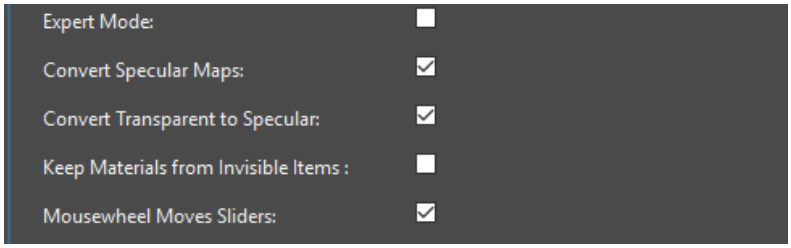



Figure 104

***Expert Mode*** on will suppress the confirmation dialogs and warnings. Only turn it on after you are familiar with the plugin.

***Convert Specular Maps*** allows you to turn off using the Poser specular maps. Often specular maps in firefly are derivatives of the diffuse or bump map and do not make much sense in octane due the difference in handling specular and glossy materials.

***Convert Transparent to Specular*** is by default On. So, it will convert glass to proper octane glass materials. The disadvantage is that completely transparent materials are also converted to specular material, like control handles or bones. You either need to make them invisible in Poser or set the opacity to 0 in the material itself. The threshold where the plugin decides whether something is a specular material or a transparent material, is by default set at 0.8 but can be changed in the OctaneDefaults configuration file.

***Keep materials from Invisible items*** is the option where octane materials are kept when the poser visibility of a mesh is set to off. If this option is turned off, octane materials are lost when you turn the mesh visibility off. This is really a personal preference thing. If the option is turned on, the invisible items will still appear in the item list, but do not render.

***Mousewheel moves Slider*** turns on the mouse wheel for moving the sliders in the plugin. The mouse wheel works fine in most cases but is not very accurate and only works on one slider in a two or three slider node.


Figure 105

***Smooth Default*** on means that in all materials smoothing will be set to on. In most cases this will give good results, but in some cases like flat rectangular surfaces with triangle polygons may not look good. Turn smoothing off for this material.

***Smoothing Angle*** has 2 purposes. If set to 0, Octane will use Poser smoothing algorithm. If it is set to an angle (degrees) it will its own smoothing algorithm. There is a slight difference and it will not have the triangle problem mentioned before, but in some cases, this type of smoothing may look different as in Poser.

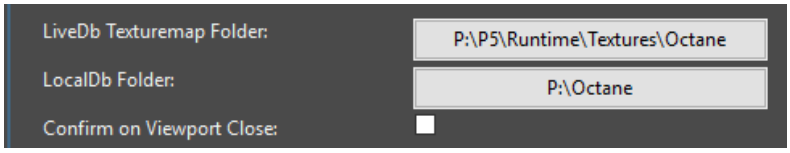



Figure 106

***LiveDB Texturemap Folder*** is the location where Octane textures will be stored which are part of LiveDB materials. I strongly suggest you create a folder in your runtime textures folder and point to this location. The default will store the textures in Posers Roaming folder and is often forgotten when you move to a new machine. Opening a scene which do use these textures will prompt you to a whole series of Locate dialogs to find each of them.

***LocalDB folder*** is the default location of any Octane material you create yourself and want to save it.

***Confirm on Viewport Close*** will give you a warning when you close the viewport (terminating your render).

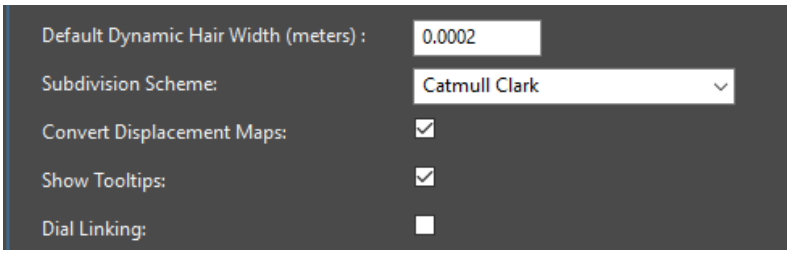



Figure 107

***Default Dynamic Hair Width (meters)*** defines the hair width which Octane will use to render the strands. See Poser Dynamic Hair section for an explanation on how to use this.

***Subdivision Scheme*** allows you to choose a subdivision scheme. For more details see the Subdivision chapter.

***Convert displacement maps*** enabled will transfer displacements maps from Poser to Octane. See Displacement section for more details.

***Show Tooltips*** enables or disables the tooltips which appear when you hoover over an item in the UI.

***Dial Linking*** is a feature where the plugin will read animated dials in Poser. Having this option on has an impact on the UI performance, so if you do not use it, turn it off. See Dial linking section for more details.


Figure 108

***Out of Core Textures*** allows you to let Octane store textures in main memory. This is particular useful on video cards which lack VRAM for large scenes. Be aware though that this memory is no longer available to other applications (non swappable).

## 4.8.2 OctaneDefaults.py

This file can be found in the Poser Plugin for Octane folder at c:\Program Files\Smith Micro\Poser 11\Runtime\Python\Addons\OctaneRender for PoserOctaneDefaults.py. You need administration privileges to edit the file since it is located in the Program Files folder structure of Windows. After making changes, either restart Poser or load a new scene. This will force the new defaults file to be used.

Many of the parameters are no longer used, but there are still some important defaults you can change.

```
# The maximum scene size that will load into the plugin.  Increase that if you have high-end equipment.  Increase this figure at your own risk!
# As a rule, if your PC has 4Gig on-board (CPU) RAM, the practical scene size limit with be 4Mil polygons
MAX_NUMBER_OF_SCENE_POLYGONS = 25000000
```

This is the setting for the maximum geometry size. Geometry is loaded in VRAM together with textures. Default is set at 4 million, but with the modern cards can be set higher. If you load a scene which exceeds this number, you will get a warning that it will not fit.

```
RESOLUTION_PRESETS = [ # Add your own customer resolutions to the end of this list.  Note, there must be a “ x “ between the 2 numbers
    “320 x 200”,
    “640 x 480”,
    “1024 x 768”,
    “1600 x 1200”
    “1920 x 1024”
    “1920 x 1200”
    “3840 x 2048”
```

```
    "18432 x 1536"
]
```

You can define your own resolutions here.

```
SAVE_PREVIOUS_RENDER_SAMPLES = 100  # The number of samples where the render is saved as
the "previous render" (to be displayed with the right mouse button).  Set to 0 to disable
previous render saving.
```

Default for previous render.

```
GLOSSY_DIFFUSE_RATIO = 0.2 # If total Specular_Color * Specular_Value *
GLOSSY_DIFFUSE_RATIO is greater than total diffuse color * value, then make the node
glossy, otherwise diffuse
# Set the above to 0 to force all materials to be glossy (and not diffuse).
# NOTE: If Alternate_Specular is "brighter" than Specular_Color * Specular_Value, it will
be used in the above calculation instead.
```

This has been depreciated.

```
SPECULAR_RATIO = 0.5 # If a Poser Material has a transparency greater than this figure, it
will be converted to an specular Octane material
```

The threshold to decide if a transparent material is a glass type of material (specular) or not. This can be overruled by the ***Convert Transparent to Specular*** setting in the setup window. This threshold is when no transparency maps are used.

```
SPECULAR_FILM_INDEX = 1.45 # The filmindex pin default value for glossy nodes
```

Default for Specular Film index.

```
# Below are default values for the glossy pins
GLOSSY_SPECULAR = 0.05 # The specular value for glossy materials converted from Poser
GLOSSY_ROUGHNESS = 0.7 # The roughness value for glossy materials converted from Poser
GLOSSY_BUMP_POWER = 0.05 # The bump value for glossy materials converted from Poser.  Only
applied if there is a bump map
GLOSSY_FILM_INDEX = 1.45 # The filmindex pin default value for glossy nodes
GLOSSY_INDEX = 0.0 # The index pin default value for glossy nodes
```

These are the default values used for Glossy materials.

```
#Daylight defaults
DEFAULT_DAYLIGHT_MODEL = 1 # 0 = Old, 1 = New
```

Legacy setting for Daylight. Keep at 1.

```
# Kernel defaults
DEFAULT_KERNEL = "pathtracing" # Opions are "pmc", "directlighting", "pathtracing",
"deep_channel_kernel"
DEFAULT_RAYEPSILON = 0.00001
DEFAULT_MAXDEPTH = 12
DEFAULT_MAXSAMPLES = 12000
DEFAULT_FILTERSIZE = 1.5
```

Default settings for Kernel. Setting a default rendertarget overrules this setting.

```
# Imager defaults
DEFAULT_EXPOSURE = 1
DEFAULT_GAMMA = 1
DEFAULT_CAMERA_RESPONSE = "Agfacolor HDC 100 plusCD" # Case sensitive
DEFAULT_VIGNETTING = 0.3
DEFAULT_MIN_DISPLAY_SAMPLES = 1
```

Default settings for the imager.

```
# If a texturemap is plugged into the Specular_Color and total Specular_Color *
Specular_Value > CONVERT_SPECULAR_MAPS_THRESHOLD the texturemap will be plugged into the
"specular" pin of the glossy
# If a texturemap is plugged into the Alternate_Specular and Alternate_Specular >
CONVERT_SPECULAR_MAPS_THRESHOLD the texturemap will be plugged into the "specular" pin of
the glossy
CONVERT_SPECULAR_MAPS_THRESHOLD = 0.8
```

The threshold to decide if a transparent material is a glass type of material (specular) or not. This can be overruled by the ***Convert Transparent to Specular*** setting in the setup window. This threshold is when Transparency maps are used.

```
USE_TEXTURE_EMISSION_FOR_EMITTERS = 1 # Set to 0 to set emission = "blackbody" for ambient
materials, or 1 for "texture emission".  If 1, then the ambient_color from the Poser
material will be used as the emitter color
```

Here you can set what type of emitters you get in the conversion by default.

```
ANIMATION_OUTPUT_DEFAULT_FOLDER = "" # "" defaults to the Poser Temp Files (set in the Misc
tab of the General Preferences)
# For example, you can set to a specific folder as
follows.....ANIMATION_OUTPUT_DEFAULT_FOLDER = "C:\\Users\\Public\\Documents"
```

This is the default save location for animation frames. You can always override this in the Animation tab.

```
PROGRESSIVE_SAVE_SAMPLES_INTERVAL = 500 # The number of samples between progressive save
images
```

Default for number of samples rendered for progressive save.

```
SkinNodeTemplate = [ # Always glossy
    ["diffuse", "power", 1.0],
    ["diffuse", "gamma", 2.2],
    ["specular", [0.0, 0.025, 0.05]], # Must be 3 floats = color
    ["roughness", 0.5],
    ["filmwidth", 1.0],
    ["bump", "power", 0.05],
    ["bump", "gamma", 1.0]
]
```

These are the default material settings for a figure which is converted from Poser. It will use these settings together with the textures found in the figure.

In the list below are all the names which are recognized as being part of a figure skin. This list is used when you use a plugin macro to apply the current material setting to all skin materials for the figure. You can add to this list if you have a new figure with a new material name for a skin type.

```
# Skin materials in the list below default to glossy.  Must be lowercase
SkinMaterials = [
    "skin",
    "skinbody",
    "skintorso",
    "skinneck",
    "skinhip",
    "skinfeet",
    "skinleg",
    "skinforearm",
    "skinarm",
    "skinhand",
    "skinhead",
    "skinscalp",
    "skinlips",
    "skinknee",
    "skinshin",
    "skinthigh",
    "nipples",
    "lips",
    "eyesocket",
    "nostrils",
    "head",
    "body",
    "legs",
```

```
        "arms",
        "skin_body",
        "skin_head",
        "skin_legs",
        "skin_arms",
        "1_skinface",
        "1_nostril",
        "1_lip",
        "2_skinhead",
        "2_skinneck",
        "2_skintorso",
        "2_nipple",
        "2_skinhip",
        "3_skinarm",
        "3_skinfoot",
        "3_skinforearm",
        "3_skinhand",
        "3_skinleg",
        "teeth",
        "tongue",
        "innermouth",
        "mouthinner",
        "gen_skin",
        "genitals",
        "foreskin",
        "lacrimals",
        "nailsfingers",
        "nailstoes",
        "ears", # V6
        "face", # V6
        "feet", # V6
        "forearms", # V6
        "hands", # V6
        "hips", # V6
        "neck", # V6
        "nipples", # V6
        "shoulders", # V6
        "torso" # V6
    ]
```

## *4.9 Displacement*

Octane uses micro-displacement like firefly. You can use the displacement maps directly, but you have to set some of the parameters yourself.

Poser Firefly uses black as a zero displacement. Both positive and negative values are allowed. A more common displacement map uses mid-grey (RGB 127,127,127) as zero displacement, where darker shades are negative and light shades are positive.

Octane can do both. It is defined by the Mid level node. At 0.0 it is like firefly – black = zero. At 0.5 it is like the mid-grey type of displacement maps and mid-grey is zero. You have check the displacement file itself to determine what method is used and set the appropriate mid-level.

Make sure you have the gamma set to 1.0 to get proper displacement.

You also have to set the level of detail. This is usually the resolution of the displacement map. To save memory you can use a lower resolution type if the level of displacement is not needed (if it is far away).

Height is defined in meters and should be converted by the plugin.

Displacement direction gives you extra control over how the displacement takes place: Follow vertex normal, follow Geometric normal or follow smoothed normal.

The picture below shows the displacement for each of them. The follow geometric normal causes unwanted artefacts because this is a low poly model. Both the other normal directions show good results.

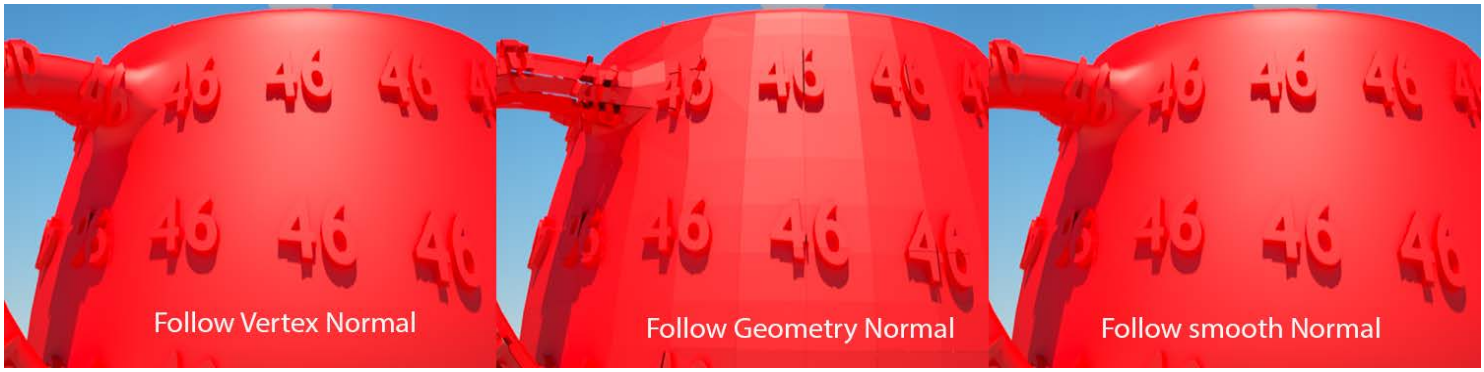



Figure 109

The filters allow you to smooth out transitions. The higher the value, the smoother the transition as shown in the following picture.


Figure 110

## 4.10 Smoothing

In the Poser plugin for Octane you can use 2 different methods of smoothing: The Poser smoothing algorithm and the Octane algorithm. In most cases it will not make much difference, but sometimes Poser smoothing leaves artefacts when rendered in Octane. Octane smoothing however has a single smoothing angle for all geometry while you can specify a separate angle in for each mesh in Poser. Unfortunately, you can only use one smoothing algorithm at a time.

The Poser smoothing is turned on when the Smoothing Angle is set to 0 in the Setup Window. If you choose an angle (like 80 degrees), it will use Octane smoothing.

In the picture below, you can see that Poser smoothing gives artefacts, while octane smoothing gives a much better result. However, the Octane version has a slight rounded edge in the pyramid.

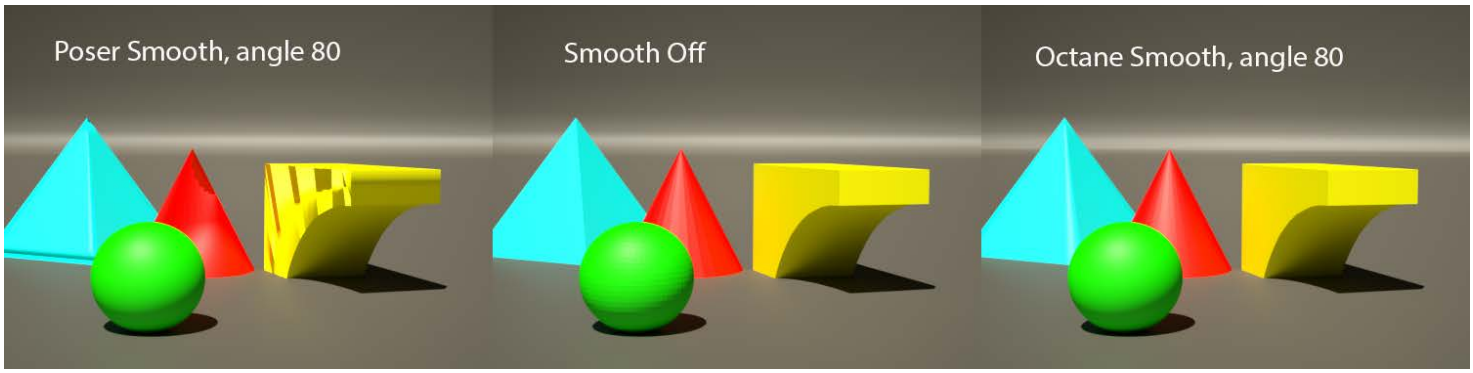



Figure 111

The Poser smoothing fails in Octane are usually cause by having too large polygons. Subdividing the prop first will usually clear the artefacts like in the following picture where I subdivided each prop once while keeping in mind the sharpness (see Subdivision section for more information).

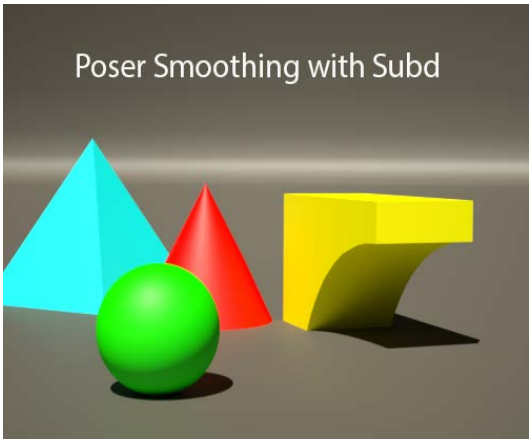



Figure 112

In most cases where artefacts appear with Poser smoothing, it is on large rectangles. In such a case, it is easier to turn off smoothing since it is a flat surface.

## *4.11 Subdivision*

There are several ways you can subdivide meshes in the Poser to Octane workflow. You can use the standard Poser method of subdividing or you can use one of Octanes subdivision algorithms.

If you use Poser subdivision, there is nothing you need to do within the plugin, it will use the subdivided mesh.

For Octane subdivision you can choose out of several subdivide implementations: Catmull-Clark, Loop and Bilinear.

***Catmull-Clark*** is most often used and the default in Octane. It is most suited for quad meshes which most Poser meshes use.

***Loop*** is most suited for triangular meshes, it subdivides by triangles.

***Bilinear*** is a subdivision algorithm which tries to keep the shape, keeps sharp edges sharp and smooths out smooth surfaces.

The subdivision method can be selected in the Setup window.

In the properties of each mesh you can set the subdivision parameters: The subdivision level, the subdivision Sharpness (0 -10 where 10 is sharpest) and Boundary Interpolation.

Boundary Interpolation is how edges are to be treated:

***None*** is bilinear interpolation – no smooth on edges and smooth where mesh is smooth.

***Corners Only*** sharpens only the corners by linear interpolation.

***Edge and Corners*** sharpens both the corners and the edges.

***Always Sharp*** sharpens all edges.

## *4.12 Nested Materials*

One of the things the octane plugin does not do is to convert complex materials.
The reason for this is that Octane uses a complete different render technique where you need a more realistic approach. One powerful technique is the Mix material node.
It looks a lot like Posers blender node but there is one big difference. The Mix Material splits one material into 2 independent materials and is controlled by a value operator which is usually a percentage, a falloff node or a mask. You can achieve the same effect in Firefly as well, but that involves a complex structure. You can also split up a mix material node into two other mix material nodes and you can go as deep as you want. This way you control a single material with a set of masks and control a variety of materials.

The following examples mixes a diffuse, a glossy and a specular material together, controlled by 2 masks.

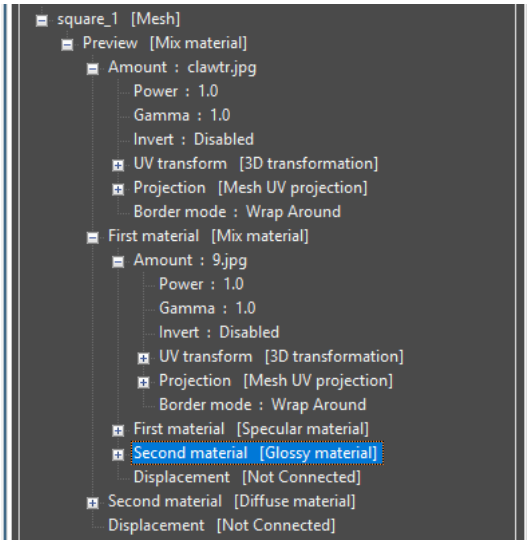



**Figure 113**

The following render shows the result. The single material on the one-sided plane now has 3 different materials.


**Figure 114**

## *4.13 Poser Dynamic Hair*

Poser dynamic hair can be rendered in Octane. Before you start the plugin, you need to populate the hair props. This can easily be done in the plugin by running the provided script (Scripts!OctaneRender for Poser!Set All Hair props to populated). After you have run the script enter the hair room to refresh the hair properties. Now you can start the plugin.

The Setup Window allows you to set the thickness of the hair, I used 0.0001 in the example below. With a low number of hair strands you may need a thicker hair or you

need to increase the number of hair strands. Refresh the scene if you change it, so the plugin can make the change in the import.

Figure 115

Octane does not have a hair node, but you can use one of the hair shaders from liveDB. In the above picture I used a brown hair shaders, but modified the colors with a hair texture and modifying the gamma of it on the specular node.

There is not a lot of dynamic hair out there for the new figures, but you can easily adapt hair from older figures. Use the morphbrush and fit the skull cap to the new figure. Since the hairs frown out of the cap, these will be automatically adjusted to the new figure. In the example above and below I have done that.

Figure 116

## *4.14 OpenVDB*

OpenVDB is an open source implementation for efficient storage and manipulation of volumetric data. Think of clouds, fire and smoke. OpenVDB is based on a Volumetric Dynamic grid that shares characteristics with B+ trees. For full information see http://www.openvdb.org/.

The workflow for using vdb files is a bit different from usual. The first thing you need to do is create the scene where you want to place the vdb files. We need to do this because we cannot manipulate vdb files directly in the plugin or poser. So, we need the scene for reference and scale. So, once you placed the objects in the scene and added a daylight, open the plugin and open the viewport. We need to open the viewport, so Octane can save our scene. Right click on render target and choose Export scene to OCS, then select ORBX format. Now select a place where you can find it and press Save. Quit the plugin now.

We are now going to create an OpenVDB volume. To do that we need 2 things – an OpenVDB file and we need the Octane Standalone application. If you do not have it, download it and install it. The Poser plugin requires a valid Octane license besides the plugin license, so the stand-alone program will be in your download section at Otoy.

OpenVDB files can be found on the internet, for this example I use a free one from Brandon Young at Gumroad (https://gumroad.com/l/xxeyl).

Start the Octane standalone engine and do a File Open and select the ORBX file you saved earlier.

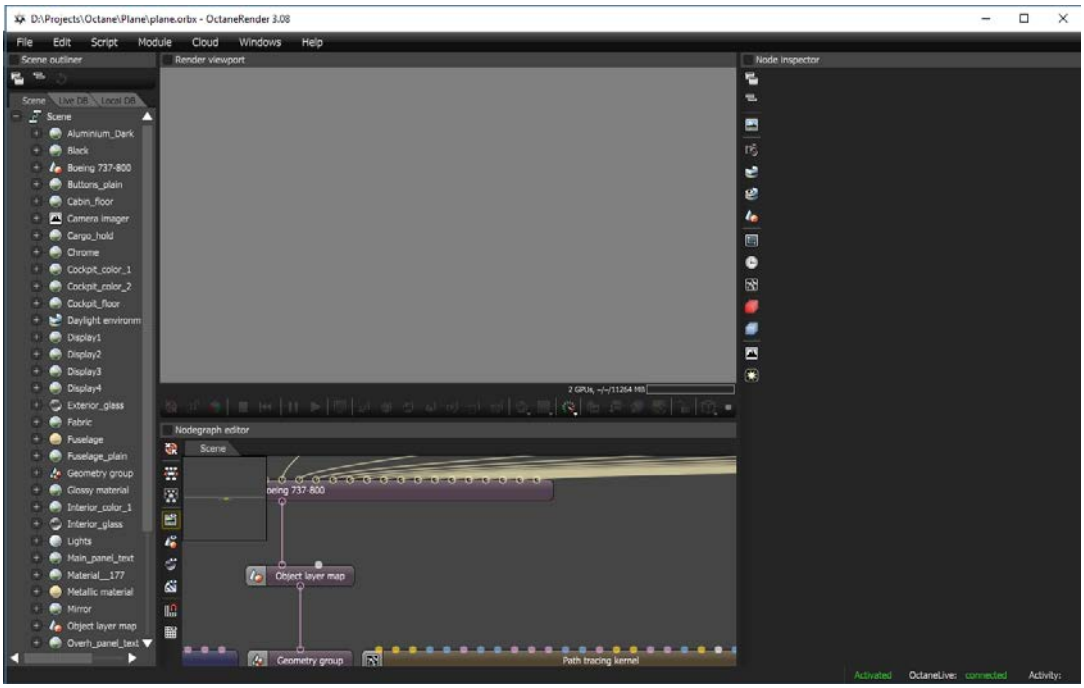



Figure 117

On the left you have all the scene elements: geometry, material, render target settings. In the middle you have the render viewport on top and the node editor at the bottom. On the right you will be able to see the content of the nodes.

For now, select rendertarget in the tree on the left. Wait a few seconds and Octane will start to render. On the right all the parameters of the render target are now visible.

It is out of the scope of this manual to fully describe the Stand Alone UI, so use this link Octane Standalone Manual if you want to familiarize yourself with it. I will only describe the actual nodes we use in this section.

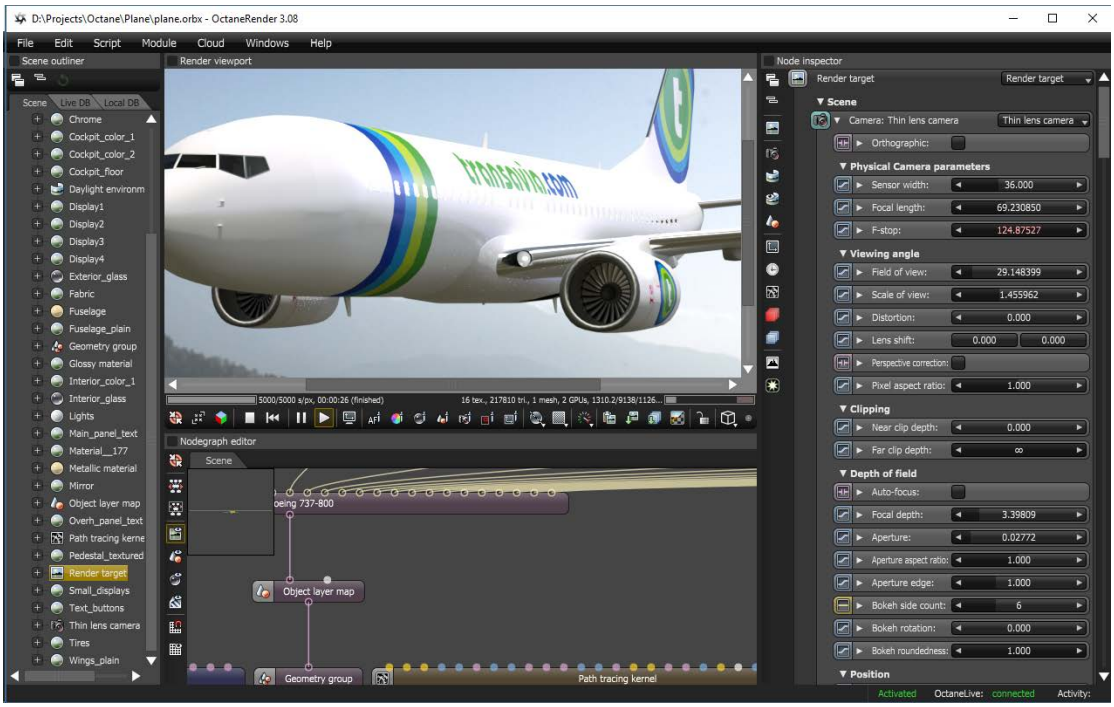



Figure 118

Pan the node editor and find a free area (you can pan by holding right button and dragging around) close to the geometry group.

Now right click on the free area and select from the Geometry menu the Volume item. And click it.

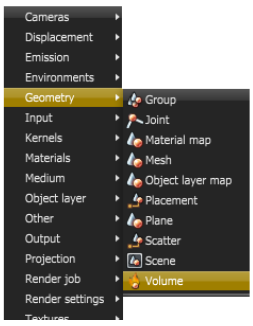



Figure 119

Octane now asks you to select a vdb file, select the free vdb file you have chosen earlier and press Open. The node is not connected to anything yet, so we do not see anything happening in the viewport. Now we add a placement node (Geometry!Placement). Left-click the mouse on the bottom connector of the volume node and drag it to the right connector on top of the placement node. Now we need to connect the Placement node to the geometry group.

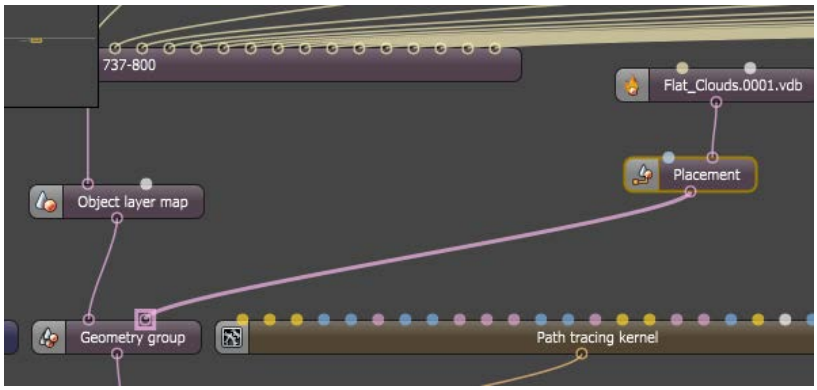



Figure 120

There probably is no connector available, so select the geometry group and on the right panel you can now select Add Input. Press that button. Now connect the Placement node to the Geometry group on the new input connector. Now you should see something appear in the render.


Figure 121

As you can see it is too small and black, so we need to fix that, and we want to position it as well.

To change scale and position, select the placement node and on the right, you can see the transformation channels. Use those to position the object.

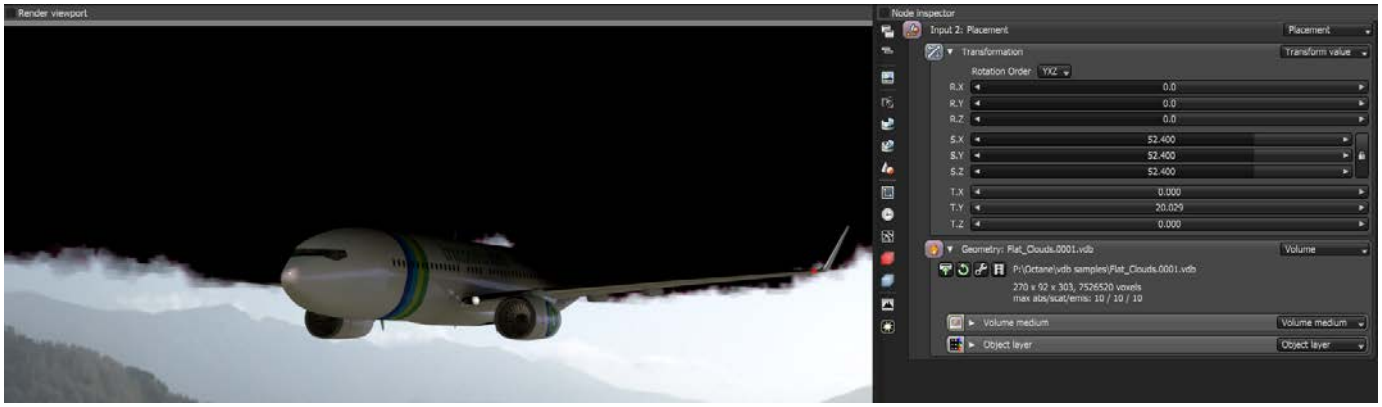



Figure 122

Now we need to make it a cloud. So, select the volume node in the node editor and start adjusting the density, scatter color and all other nodes to get the density and color you want as explained in the Fog and Volumes sections in this guide.

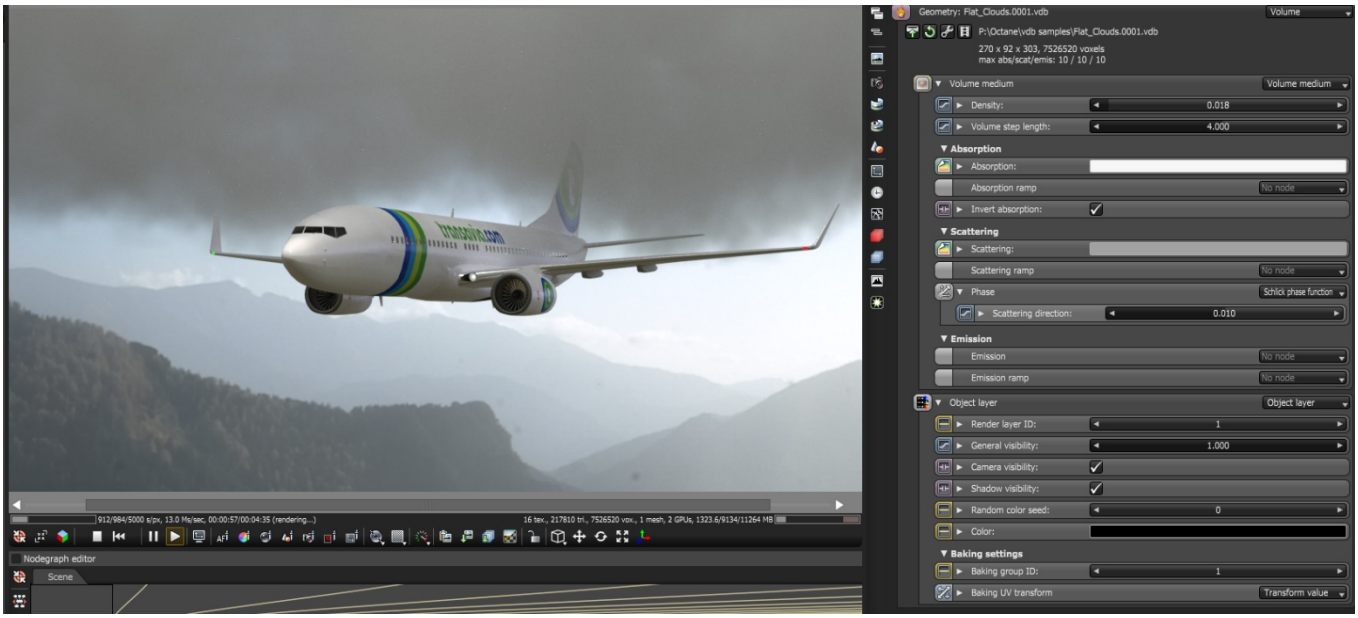



Figure 123

You can add more volumes and then group them together in its own geometry group. Then connect the new group to the original geometry group.

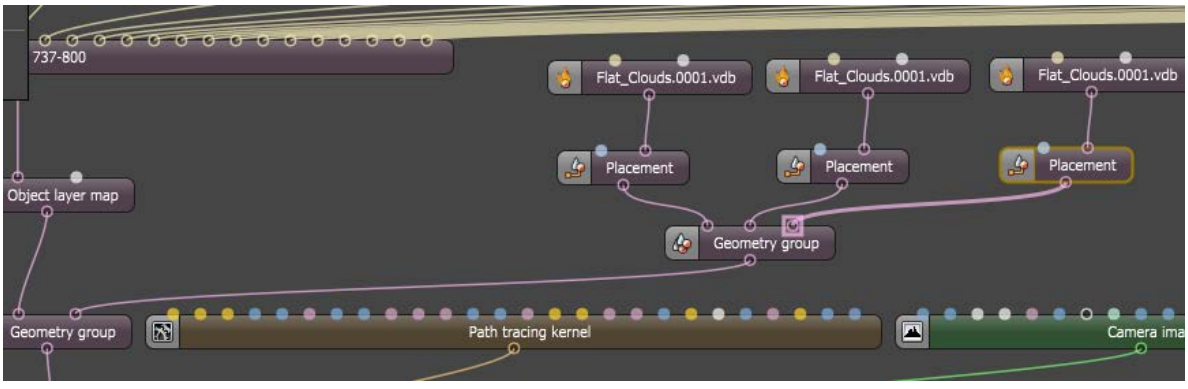



Figure 124

Then vary the position and scale of each and you will get a real cloud formation.

Figure 125

Now it is almost ready to export and make ORBX which the Poser plugin can use. To do that, we need to make one addition change and add a Geometry Out node (Output!Geometry Out). Save the scene now as ORBX, you might revisit this to make changes to the placement and materials. To save the ORBX for Poser, use the control key to select all of the new nodes you created and right click on one of the selected nodes and select Save.

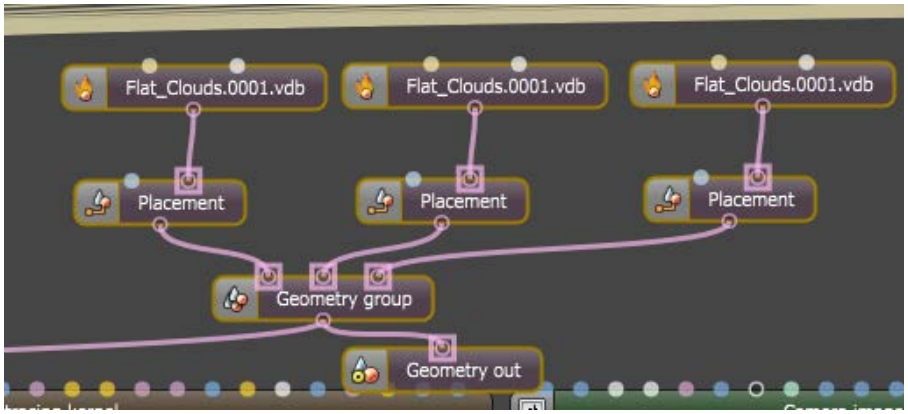



Figure 126

A new input box will pop up. Select browse and give it a filename. Then in the input box give it a friendly name and, in the description, fill in for what scene it is (not necessary, but recommended).

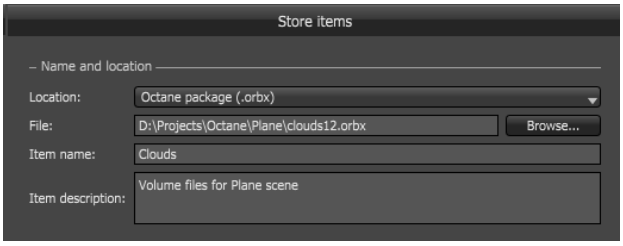



Figure 127

Quit Octane and return to Poser.

To use OpenVDB volumes, we need to place a proxy in the scene, I usually take a sphere. The proxy will be replaced by the actual volume. The location does not matter since the placement is done in Octane. Once you start the plugin you can change the properties of the sphere from mesh to geometry out. Once this has been changed you can load the new ORBX file with the button below the node type.

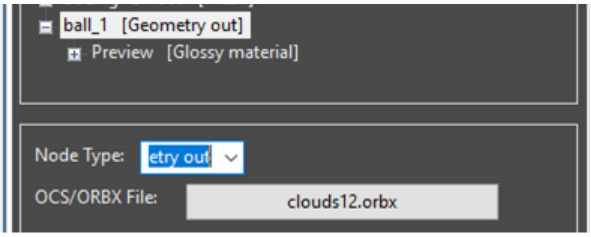



Figure 128

And that is it! Open the viewport and render. If you want to change the placement or material, go and load the scene in Octane Standalone and update it there. Save the nodes again and quit Standalone and in Poser reload the scene so the new ORBX will be used.

Figure 129

## *4.15 Instancing*

Instancing in Octane is the ability to replicate a prop or figure without the cost of geometry of textures. Octane does this by using a density map. The density map is a greyscale map where the greyscale defines the probability a replica will appear.

To use instancing, you need a density map, a prop or figure to replicate and a prop or figure to place the replicas on.

In the following example I use the highres ground plane, a tree, grass and weed props.

Select the prop to replicate and change its nod type to Scatter.

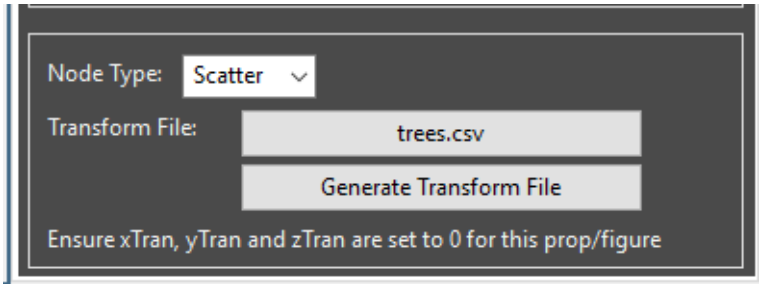



Figure 130

Now press the Generate Transform file button and a new window will open.

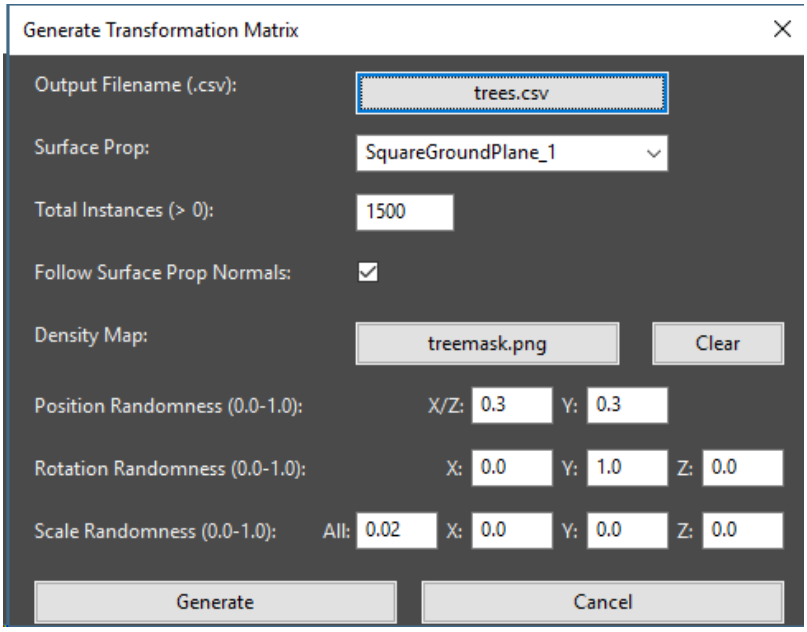



Figure 131

***Output filename*** is the name of the transform file (stored in CSV format). Give it a name.

***Surface Prop*** is the name of prop where the replicas are placed on – like the ground plane.

***Total Instances*** is the number of replicas to create.

***Follow Surface Prop Normals*** tells Octane whether to follow the direction of the normal to place the replica.

***Density map*** is the greyscale image we have to create in an outside paint program. If the prop where the replicas are placed on is UV mapped, use the UV map template as a starting point. Resolution is not too important.

I created this map for this example:

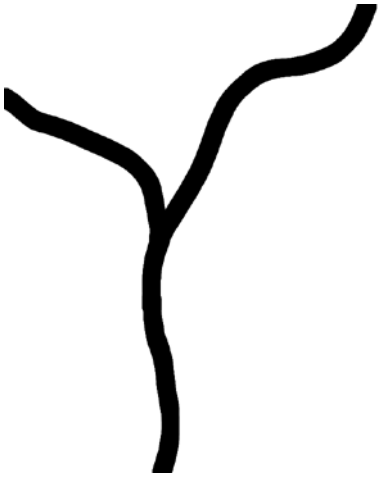


Figure 132

***Position Randomness*** is how the replica is positioned within a polygon. From centre (0.0) to completely random (1.0). X/Z is on the surface itself and Y is in height. Be aware that the polygon size comes into account here. If the number of replicas comes close or over the polygon count, having it set to centre will create a pattern which you may want to avoid. Randomness does not always solve the problem because replicas may now be too close to each other. Increasing the polygon fixes this problem.

***Rotation Randomness*** defines the range for random rotation (0 is fixed, 1 is 360 degrees). Rotate around Y axis would be most common.

***Scale Randomness*** defines how scale variation is.

***Generate*** will now create the transform file.

You can define multiple props to replicate and use the same or different density maps for them.

There is one gotcha here. The props to replicate will need to be at zero position for the density map to work. So multiple props will load at the same position. Although the originals do not render, it a bit inconvenient. I always move the camera somewhere away from the zero position, so those props do not obstruct my preview. Here is the finished render with my instanced props.

Figure 133

## *4.16 Node Editor*

The node editor can be used to create materials in the Material editor of Octane itself. Although almost all features can be done in the plugin. One example of something you cannot do directly in the plugin is a color ramp with multiple colors.

To enter the node editor, check the NodeGraph option in the material type and press the Edit button.

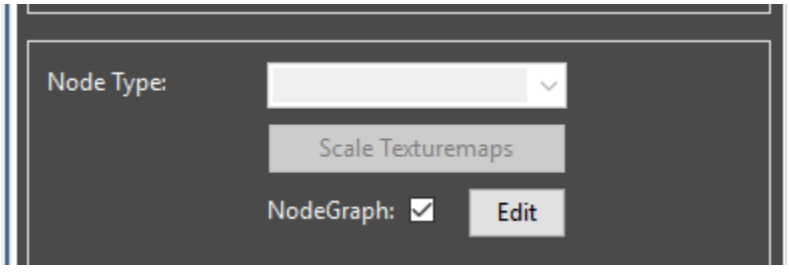



Figure 134

Octanes Node editor will pop up and you can now add nodes. How to add nodes can be found in the Octane standalone manual here: https://docs.otoy.com/StandaloneH_STA/StandaloneManual.htm#StandaloneSTA/TheGraphEditor.htm.

The end node always has to be a Material Out node.

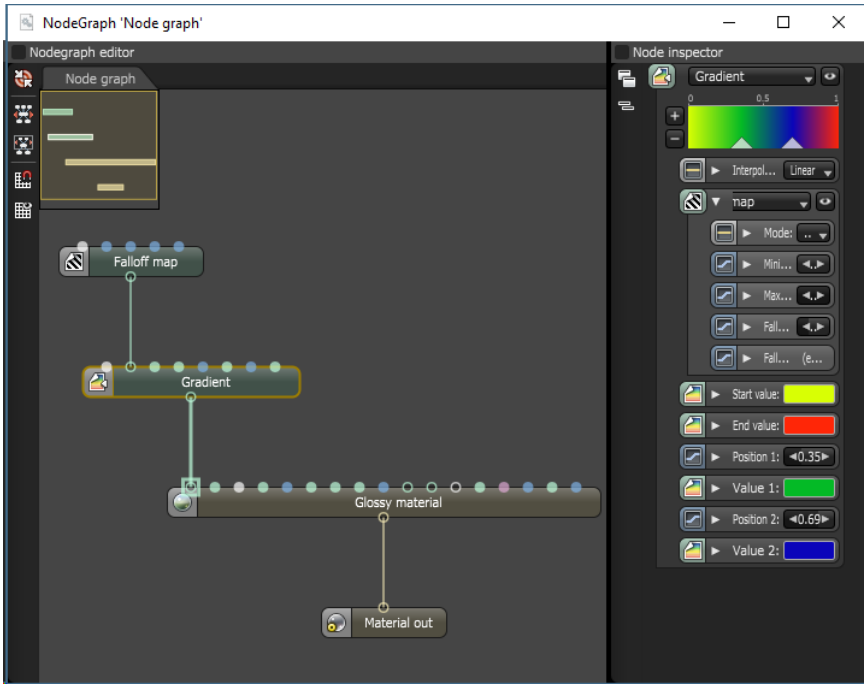



Figure 135

In the above example I plugged in a Falloff node into a gradient node. The gradient node in this editor allows you to add multiple colors for the gradient.

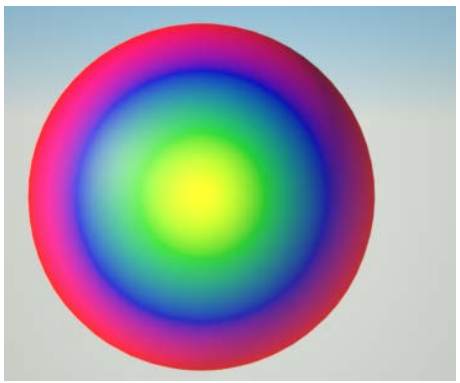


Figure 136

The above picture shows the result.

## *4.17 Depth of Field and Bokeh*

Depth of Field is generally used by photographers to draw attention to the subject which is in focus and blurring the background. Bokeh is the method how the blurring is achieved.

Depth of field in Octane is controlled by the focal length – the lens used, the focal depth – the distance from camera to focus plane and f-stop/aperture control the amount.

All examples below have the focus set at the butterfly on the left.


Figure 137

Bokeh is an effect which changes how the blur in Depth of Field is created. It changed the shape of the highlight. There are 3 parameters which control this. Side count defines the shape, rotation handles the orientation of that shape and Roundedness defines the hardness of the edges. It is a subtle effect which needs reflected lights or HDRI to show up.


Figure 138

The last option for Depth of Field allows you to stretch or squash the Depth of Field sphere as shown in the following example:


Figure 139

## *4.18 Post Processing*

There are 3 Post processing affects within Octane: Bloom, Glare and Spectral effects.

To enable these effects ***Enable*** them in Poser processing section of the rendertarget section.

***Bloom*** creates a halo around the sun and other light source as well as reflected lights.
***Bloom Power*** defines its strength.

In the following examples I use the Octane sun. Here the sun without post processing:

Figure 140

With Bloom at 10, no Glare, no Spectral effect:

Figure 141

Now Bloom at 10 and Glare set to 1, Ray amount at 3, Rotation at 15 and Blur at 0.6:

Figure 142

Finally add Spectral intensity at 0.6 and Shift at 5:

Figure 143

In the final example a full scene with no sun but with other light sources:

Figure 144

## *4.19 SSS skin*

There are many articles written on how Sub Surface Scattering should be done and probably just as many opinions on what is the correct way of doing it. One reason for this, is that subsurface scattering tells how light behaves when it enters the skin. The problem is however that our meshes do not really have anything to bounce against except the outer shell. So, all SSS solutions are approximations on how it should behave. Most solutions however depend on the skin textures, light it receives and on top of that, most people have different opinions on what looks best.

In this example I take a simple approach. I use the ear to define the scattering. If you shine a bright light from behind the light, it will let through some light at the thin portions of the ear. I take that to be the values I need for the skin depth and use if the SSS. You can use this starting point to define a skin shader to your own taste.

The setup I use is as follows:

A high-quality skin for Victoria 4 (I used one from Danae), a bright spotlight behind the figure, the camera at the front and a spot light from the front right. The spotlights are IES lights to get soft shadows which make the skin look better (sunlight is very harsh).

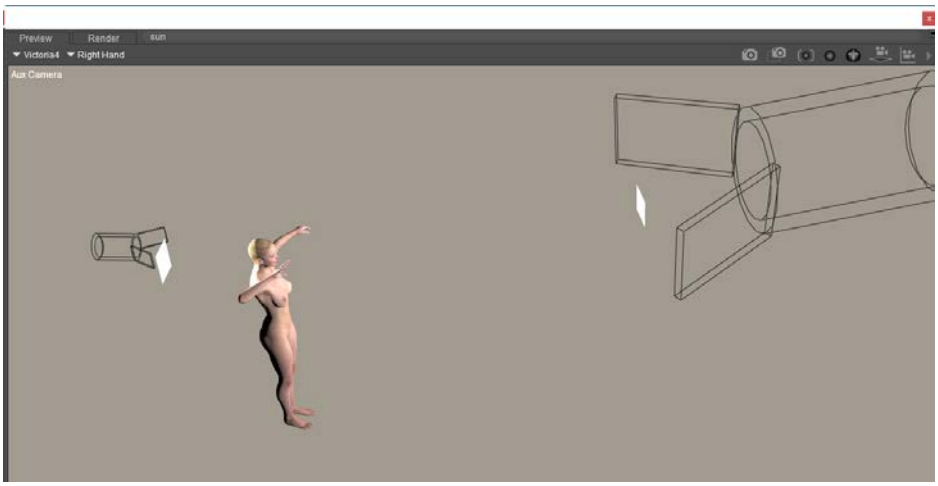



Figure 145

I use the base skin textures of the character because Octane Scatter will replace the Poser SSS nodes. The default conversion for the skin is a good starting point to add SSS.

Zoom the main camera in on the head and convert the spotlights to Octane mesh lights with the Poser macro, then start Octane and add distribution maps for a spotlight to the mesh emitters. I use the following setup for these:

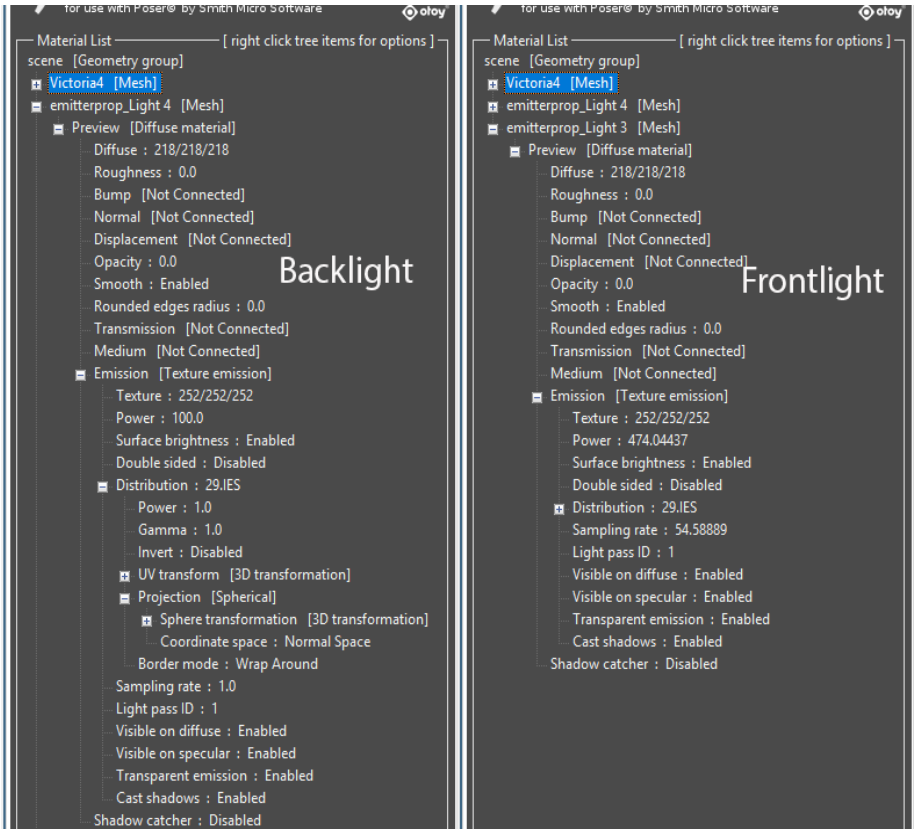



Figure 146

Next are the skin materials. The ear is located in the 2_SkinHead material, so select this and copy that material to the clipboard. Now change the Material type to Mixed Material and paste the original material from the clipboard both in First material and in Second material. The Second material will be the SSS portion of the skin material, so change it Material type to Diffuse material. The Amount will be the distribution between the two, for now set it to 0 so we only look at the Second material.

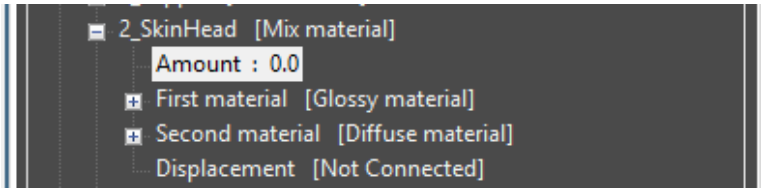



Figure 147

There are two nodes which we need to change to get Scattering: Transmission and Medium. The Transmission Node is the color we see when scattering is happening, usually red. Change the Transmission node to Multiply and then Copy the original diffuse texture into Texture 1 and change Texture 2 into an RGB type and make it reddish. The resulting color with be a reddish version of the texture.

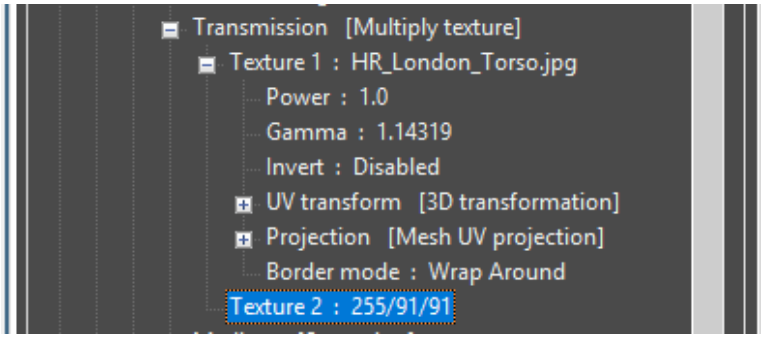



Figure 148

Next thing which needs to be changed is the Medium Node. Select it and choose Scatter. In this node change the Absorption to a Red color and check the Invert Absorption. This will absorb the complement (inverse) color of red. Set the Scattering node to 1 or white.
The density node is the one to experiment with.

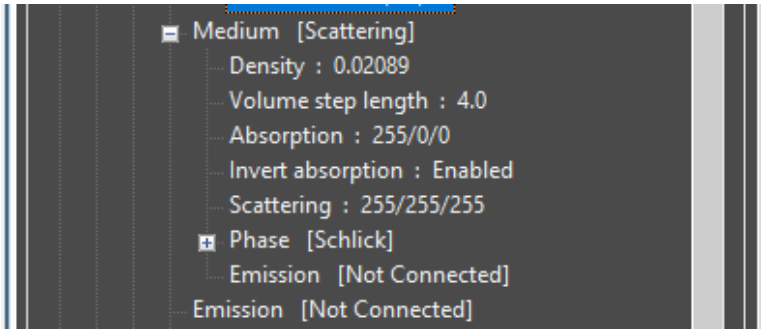



Figure 149

Now it is time to experiment. Change the Daylight power to 0 and set Emission power of the Frontlight to 0 as well. Render the scene. Now lower the Density in the Scatter medium so you can see the red appear in the ears. The optimal value is where the thick part of the ears is darker as the thin part of the ear which will be red in appearance.

Figure 150

Once you found the sweet spot we can finish the skin material. To do that we need to change the Amount parameter. The best result is achieved if you use SS masks. It is basically a mask which defines the strength of the SSS occurring on the skin. If you do not have a mask, you can use a value of 0.7 for the Amount and adjust it to your liking. In the example below, I use an SSS mask.

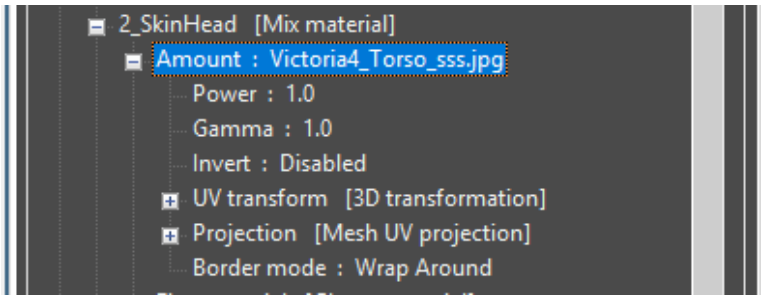



Figure 151

The next thing to do is to copy the new 2_SkinHead material to the clipboard and paste it onto the other Torso material zones (Nipple, 2_SkinHip, 2_SkinNeck, 2_SkinTorso). Paste with the *Use Copied image files* option. Now past this material to the 1_SkinFace material, but now choose *While retaining Image maps*. The Amount and Transmission nodes were not part of the original material, so adjust these to the proper maps belonging to that zone. Once you have done that, copy it again and paste it to the other face materials. Do the same procedure for the Limbs material zones.

Turn on the Front light emission and low power for the Daylight (0.1) and render.

Figure 152

If this looks all correct check the rest of the body of there are no reddish tints where there should not be. If so, adjust the Medium density or the value in Amount in the SSS material.

Figure 153

Now you can adjust the skin material to your liking.

RedSpec has a commercial package for SSS Skin in Octane for the popular figures. It is a more complex shader, you can find it here: RedSpec.

## 4.20 Animation

You can render a Poser animation as is, just start the plugin, set up your render option as for stills, convert your materials and go the Animation tab.

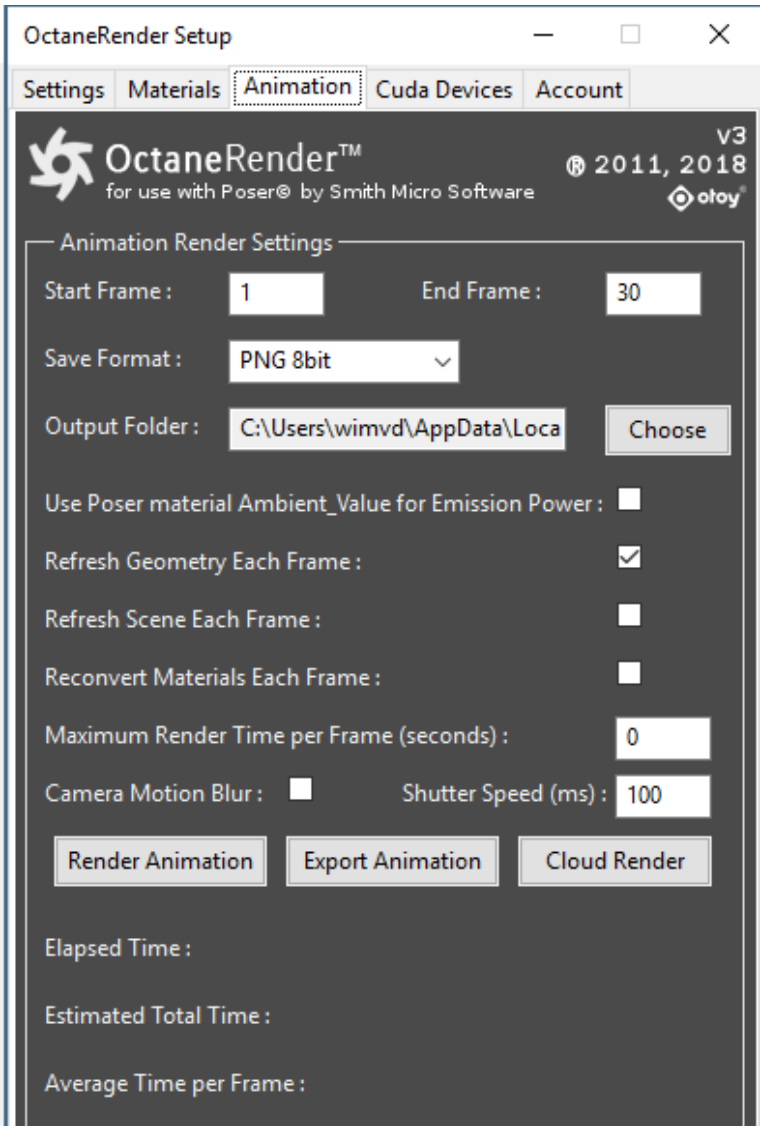



Figure 154

***Start & End Frame*** are the same ones as for Poser.

***Save Format*** is the type of image files you want for the animation (PNG 8 bit, PNG16 bit, EXR, EXR Tonemapped and EXR All passes). Use the last one for Animations which you want to post work.

***Output folder*** is the location where the animation image files are stored. By default they are stored in the users AppData/Local/Temp/Poser Pro/11 folder. Rendered frames all have the name of Framexxxxx where xxxxx is the frame number. So, make it habit to change the output folder name for every animation since older frames will be overwritten.

***User Poser material Ambient_Value for Emission power*** must be checked if you animate the Ambient_Value in poser.

***Refresh Geometry Each Frame*** needs to be enabled when geometry is changed (morphs, location, rotation, etc). If only a camera moves or light changes are used, this can be disabled.

***Refresh Scene Each frame*** is sometimes faster as enabling each of the options separately. It will read the entire scene each frame.

***Reconvert Materials Each time*** should be enabled when materials are animated in the animation, this includes dial linking.

***Maximum Render Time per Frame (seconds)*** defines how much render time Octane should spend on each frame. This excludes the time needed to load geometry or material changes. If this setting is 0, all the samples specified in the kernel are rendered. Do some test frames to see what samples or time is needed for the quality you want to achieve and used that for the render time.

***Camera Motion Blur*** enabled creates a camera blur by averaging each subsequent frame. This option only applies to Camera motions. Shutter speed defines the time the camera records defining the blurriness.

Render Animation will render the animation as image files.

Export Animation will create an ORBX file which can be used in Octane to render.

Cloud Render will export the animation to the Octane Cloud Service.

## *4.21 Vertex Motion Blur*

For Vertex Motion Blur (where geometries move) you must export the animation to Octane Standalone. Since Octane Standalone uses a batch script to render animations, it is not possible for the plugin to render Vertex Motion Blur directly. To render the animation, load the exported animation and select render target. Now you can render individual frames with motion blur or run the batch script for the entire animation.

One thing to keep in mind is to use real world measurements to correctly estimate motion blurring. A shutter time of 0.1 second means 0.1 meters if an object is travelling at 36km/sec. So, at close range that will be a significant blur.

Below a sample of Vertex Motion Blur rendered in Octane Standalone.

Figure 155

## *4.22 VR Rendering*

Rendering for Virtual Reality is essentially nothing more then rendering to a VR compatible format (resolution and filetype) with a special camera: Panoramic camera with Cube Projection and stereo output in Side-by-Side.

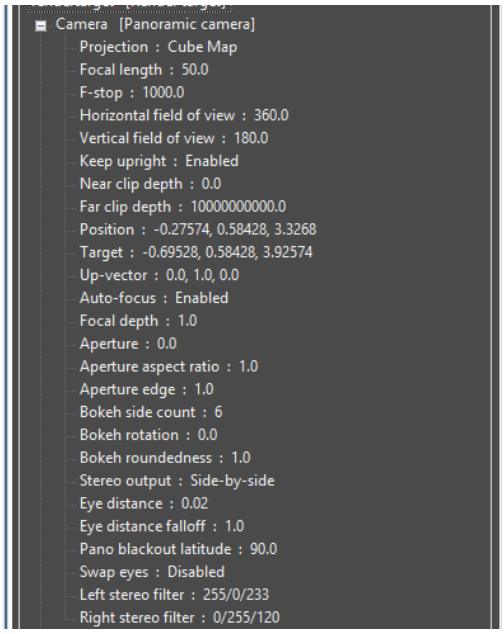



Figure 156

The focal length is determined by the horizontal field of view which is 360°.

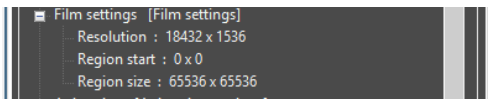



Figure 157

Output resolution depends on your target VR platform, 18432x1536 is the one used by Oculus and the Samsung Gear.

The scene itself must be viewable from every angle, so an HDRI is most helpful to fill in the gaps. Use a dolly camera and position it in the centre of the scene. With material selection, make sure that texture tiling is not causing an issue with visible repeats.

## *4.23 Photoshop Plugin*

The Octane Render Compositing Extension for Photoshop will allow you to import all render passes in one go with the layers in the correct order with the right blending.

You can get the plugin here: Octane Render Extension for Photoshop.

For a full description what it does, go here: Photoshop plugin manual.

# 5 Trouble shooting and fixes

## *5.1 Python error running macro*

This occasionally happens when the script code encounters an invalid input. Usually quitting and restarting the plugin will fix it.

## *5.2 Poke through in Octane and not In Poser*

Check your materials in that area for ***displacement*** nodes. These should be gamma 1.0 and the offset should be either 0 or 0.5 depending on the displacement map. Try the other.

## *5.3 Cracks in the forehead*

This is most likely caused by a hair cap which intersects with the figure mesh. Octane does not like that.

There are 2 ways to fix it:

Use the morph brush and remove the intersection. Switch to wireframe mode and you will see the intersection between cap and head. Select the morph brush, select the tighten fit, set the figure as target, margin at 0.0004 and paint over the intersection. It will disappear, and the cracks will have gone when rendering it.

The other method is to set ray epsilon lower (or even to 0) in the kernel settings. But setting this too low may cause other artefacts.

Figure 158

## *5.4 Seams at body parts*

If you get seams at body part borders, change the figure skinning method to Unimesh. This will remove the seams.

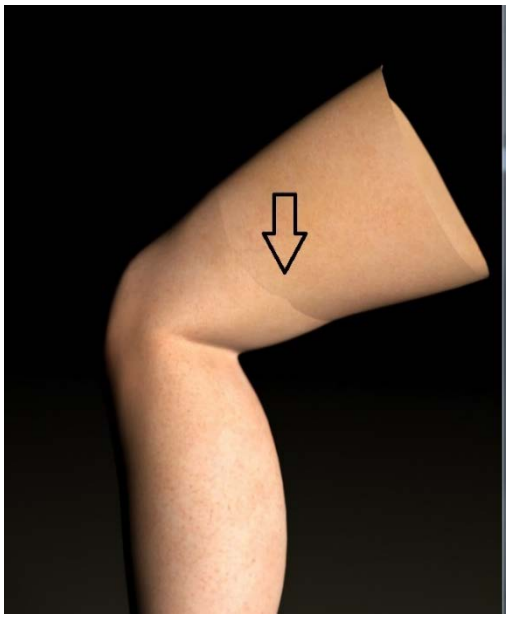


Figure 159

## *5.5 Black/dark artefacts*

This could be caused by self-shadowing. Shadows are calculated before smoothing takes place and can give a different shadow for neighbouring polygons depending on the light direction.

Usually there are 2 ways to fix this:

- Change the light direction slightly
- Add subdivision to the prop or figure

In some cases, this will not work due to the geometry intersecting.

In the following example this is the case:

Figure 160

Black artefacts are very prominent and are caused by the content geometry intersecting with the glass outside. Ray epsilon cannot solve this, and subdividing will not help.

There is however a way to fix this in Poser.

In Poser, select the prop and start the group editor. If the content does not have its own group, create a new group and name it, then add all the content material to this group.

With this group selected, choose create Prop. Close the group editor. Select the new prop and decrease its size slightly (98-99%). Start the plugin and copy the material from the original content material to the content material in the new prop. Then set the opacity of the original material to 0. Then render. If there are still artefacts, reduce the size a little further.

**Figure 161**

## *5.6 IES lights do not work*

Check whether the coordinate space setting is correct, it should be set to Normal Space.
Also make sure you have the gamma setting at 1.0. Be aware that the intensity of the IES light is usually lower as the non IES lights.

## *5.7 Noise*

It is important to make a distinction between noise and fireflies. Noise is not a render artefact, but sign of an unfinished render and will clear up with more samples, fireflies are render artefacts and may not clear up with more samples (see next section).

There are several ways to clear up noise faster. It usually is most prominent in shadowed areas and when light travels through refracting surfaces (like glass).

Use ***adaptive sampling***. This will smooth out noise by looking at surrounding pixels and stop sampling rendered pixels when threshold is reached.

Increase ***emission samples***. This will give more important to that emitter.

Lower ***Path Termination Power***. If the path is terminated too soon, it may leave noise.

Use ***low polygon geometry for emitters***. If you use scene materials as emitters, consider adding a low poly emitter.

Increase ***Caustic blur*** if you have highlight noise in specular or glossy materials.

Use ***PMC kernel*** if your scene contains many specular and glossy materials and you need caustics.

Avoid ***emitters behind specular materials*** if you can. Make the specular material transparent or place the emitter in front of it, if possible.

## *5.8 Fireflies*

Firefly are very small bright spots in your render. They are caused by numerical instability during calculation of the light rays. Fireflies are different from noise – that will usually clear up with more samples.

Fireflies usually occur when you have small bright lights in combination with glossy or specular materials, especially if the light itself goes off in all directions such as a sphere or tube emitter. Unfortunately increasing the render samples will not always work.

There are several tactics which may fix this problem, but the best one depends on the scene and lighting.

***Increase the number of samples in the light emitter***. This is the first thing you should try since it does not change anything in the scene but increase the number of calculation of that light emitter, giving it more importance.

***Set GI Clamp lower***. This will shorten the path light has to travel and decrease the number of calculations. It also reduces the overall size over which the light travels.

***Hot Pixels***. This option will clamp the brightness of the pixels. It does make the image with a lot of fireflies a bit less sharp.

***Add a new emitter***. If the emitters themselves are not visible in camera or in reflection, this is probable the best solution. Add a one sided square or sphere, make it an emitter and set the opacity to 0 and optionally turn reflections for it off. Turn the old emitter off or set it to a low value. Though this is technically a cheat, it makes the render much faster and you have more control over the lighting.

## *5.9 Scene does not load (too large)*

OctaneDefaults.py has an entry which sets the maximum scene size (`MAX_NUMBER_OF_SCENE_POLYGONS = 5000000).` The default is conservative and modern video cards can handle a lot more. I have set it to 25 million and have not had any problems on a GTX 1080ti with 11GB VRAM.

## *5.10 Mismatched geometry*

Occasionally this error will occur when a scene is loaded and rendered in the Octane plugin. The error is caused by vertices going out of bounds on a UV map. Poser ignores this, but Octane finds a mismatch and the error will occur. Contact the vendor and ask for an update explaining the problem or use UV Mapper or other program to correct the error yourself.

## *5.11 Flat surface does not look right in render*

If a flat surface does not look right (single color breaking up in more colors), it is probably due to smoothing.

In the picture below the surface on the prop behind is a mirror like material, but it looks completely wrong.

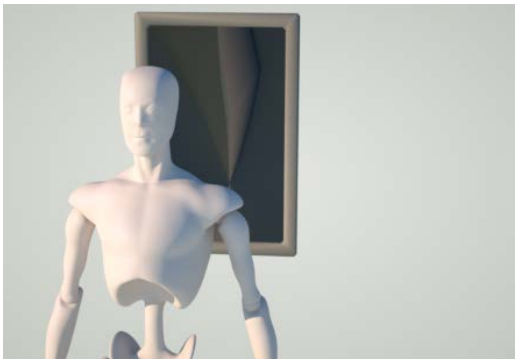


**Figure 162**

If you turn off the smoothing on the surface material, you correct the problem and get a good reflection.

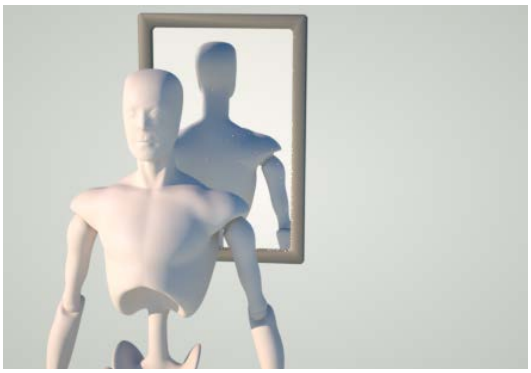


Figure 163

The problem often happens if a flat surface is build up out of triangles.  You can also try to subdivide the mesh. Some scene props from Stonemason have this problem when converted from DAZ Studio.

## *5.12 Cartoon like render*

The render below has a cartoon like render, the skin has a large shadow and the colors do not look right. See image below.

Figure 164

This is caused by the Normal node having a value of 0.0. Change the normal node to Disconnected and the problem will disappear.

Some older POC and POE files have the normal set to 0.0 and need to be changed. To correct this, change the normal to Disconnected and save the POC and POE again. Use copy/paste macros to speed things up for the various body parts which have the material.

## *5.13 HDRI looks too dark*

If an HDRI looks too dark, change the gamma to 1. If still too dark, change its power to 2 or 3 in Texture Environment and change power of Sky texture node to 2 or 3 in Daylight environment.

## *5.14 How to turn Off Depth of Field (DOF)*

Set aperture to 0 in the camera settings of the plugin.

## *5.15 Red Eyes*

If you use a very strong Scatter on the skin materials you may get a red edge around the eyes. This is caused by the eye surface refracting rays from the back of the head. 3d figures are empty inside and eye surface acts like a curved glass surface showing the inside of the back of the head. You can fix this by changing the IOR of the eyesurface or by blocking the refraction on the inside of the head with a primitive.

## *5.16 Face, Pose, Hand camera do not work.*

Octane only supports the Main, Auxiliary and dolly cameras.

## *5.17 Time Out errors in Video card.*

Time out errors (CUDA errors) are caused by not having the registry updated for Octane. Install Octane Stand Alone and it will set the registry for you.

# 6 Tips and tricks

## *6.1 Speeding things up*

There are many things you can do to speed things up for rendering. But every tweak you make may have unwanted consequences, since the settings are there for a reason.
In the following list I will go through several tweaks and what you should watch out for.

### 6.1.1 Number of Samples

The actual number of samples needed depends on the scene. Some scenes need a few 100 samples, others need 10,000 or more. Look at your scene and compare renders regularly to see if anything is changing. If there are no more changes, just stop the render. Trust your own eyes (or image editor).

### 6.1.2 GI Clamp

Reducing GI Clamp (to 100m or below) is a fairly safe tweak you can do to speed the render up. It will limit how far the light will travel and therefore reduce the number of calculations. It also reduces fireflies because it will have less unresolved calculations. Disadvantages are that it is less accurate since not all calculations are done and if you do have a very large scene, you actually may need a high value.

### 6.1.3 Parallel Samples

Increasing parallel samples and increasing max tile samples is a safe way to speed up render time. Increasing it will make more efficient use of the GPU cores which are available. Downside is that it takes more video memory, so be careful if your VRAM is low.

### 6.1.4 Hot Pixels

Setting Hotpixel to 0 will greatly reduce or remove fireflies, but it will also make the image slightly more blurry if a lot of fireflies are in the scene.

### 6.1.5 Adaptive Sampling

Turning on adaptive sampling will look at the surrounding pixels and favour those for noise reduction and stop rendering those pixels. It will however be noisier in the first few 100 samples, but the end result should be faster.

### 6.1.6 Path Termination Power

Increasing this setting will reduce calculations. But a too high setting will remove AO shadows or may cause noise.

### 6.1.7 Increase size of your emitters

Small emitters cause noise and may take long to clear up. Increasing the size of the emitter (not power) will improve render times. If not possible, consider adding extra (large) emitters and turning off reflection for them. Or add a good HDIR which matches the environment. If possible avoid those small emitters behind refractive materials.

## *6.2 Reduce Render scale during setup*

Set the render scale to 50% or lower during setup of your scene. For positioning and lighting you do not need a large view port and you keep your render resolution.

## *6.3 Save the material as Poser material*

If you save the octane material as Poser material, and then save the prop or figure in the library, the Plugin will automatically load the octane material.

It is also useful to do this if you want to duplicate props.

## *6.4 Create special light prop*

If you use a one sided square or a sphere as light prop, the material zone will be called Preview. If you use just this one prop as an emitter, that is no problem. However, if you have a series of props acting as emitters and you want change one emitter and paste the nodes to all of them, you have a problem. Since almost all props have a preview material (often not used), it will paste those nodes to all of them.

There is a simple solution to this. Once you have loaded the prop, enter the group editor in Poser and create a new group called emitter and add all of the polygons to it. Then give it the material name of Emitter (or other logical name) and quit the group editor. Save this prop to the library or duplicate it for as many times as you need it.

Once you are in the plugin, the props will now have its unique material names which you can use to copy nodes to.

## *6.5 Getting Material conversion from one scene to another.*

There are several ways to save and re-use material conversions.

- Save the material directly in LocalDB as ORBX.
- Save the entire figure or prop as ORBX.
- Save the octane material as Poser material and save it to the library.
- Export the Octane settings.

And there is an undocumented trick: Load the scene which has the converted material, start the plugin. Now quite the plugin and load the new scene with the non-converted materials. Start the plugin again. Now all the materials which were in the previous scene will have the conversion automatically done.

Caveat here: The new scene should not have been loaded with the plugin before. In this case it will not work. You can however remove all the octane settings with a script in Poser (remove all Octane settings from the scene), save the scene, restart Poser, and then perform the trick: load the converted scene, start plugin, quit the plugin, load the new scene, start the plugin.

## *6.6 Auto Refresh*

While I do recommend having auto refresh off by default because of performance reasons, there are situations where you might turn it on. One of those is when you want to carefully position props or figures and light and shadows are an issue. Poser lighting is very different from Octane lighting and auto refresh will give you the feedback you need.

## *6.7 Getting into narrow spaces*

If you want to position the camera into narrow spaces and have walls blocking your view, use ***Near clip depth***. It will remove anything in front of the camera for the specified distance. The removed geometry will still be taken into account in the render calculations, so it will still cast light, shadows and reflections.

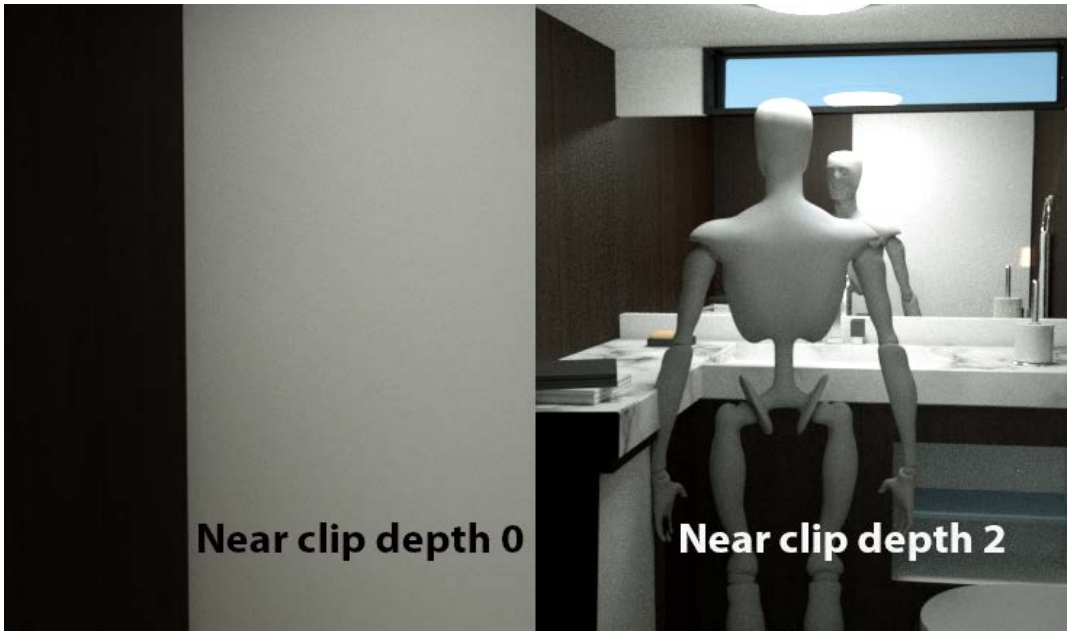



**Figure 165**

# 7 Afterword

Many thanks go to the people asking questions. It was a great help in deciding what to put in this user guide. Without them it would have been a lot shorter.

Octane is a great render engine. The incredible speed and great output gives you the time to make small changes to make your renders better without a real time penalty. For me that made a world of difference.

And many thanks to Otoy and its developers and Paul Kinnane for his great work on the many plugins he has made.

And finally, allow me to share the very first real render I made back in 2012 which convinced me to start using Octane, probably nothing special, but a huge jump from Firefly at that time. And Octane got a lot better since then.

Figure 166