otoy

2021 OTOY® Inc. All rights reserved.

OctaneRender® for Autodesk® 3DS Max® Plugin Manual

Version 2021 - Manual publication date: 28 April 2021

Cover image: "Neon Break" by Thomas Cheng, © 2020 Ambocc Studios LLC, All rights reserved.

All rights reserved. OctaneRender and OTOY and their logos are trademarks of OTOY, Incorporated.

Autodesk and AutoCAD are registered trademarks of Autodesk Inc and its subsidiaries and affiliates in the U.S. and other countries.

http://render.otoy.com

Contents

Installation Overview

This section outlines hardware, software, and driver requirements in addition to specifying the steps needed to install OctaneRender® for 3DS Max®.

Hardware Requirements

We recommend using your on-board graphics or a second graphics card for the Windows® display adapter, and dedicate a more powerful CUDA®-enabled card(s) for rendering.

OctaneRender® requires an NVIDIA® CUDA®-enabled video card. It runs on Kepler, Maxwell, Pascal, high-end GTX Titans, Volta, and Turing GPUs. Texture limits and differing power efficiency ratings also apply, depending on the **GPU**[1] microarchitecture. GPUs from the GeForce® line are clocked higher and render faster than the more expensive Quadro® and Tesla GPUs.

GeForce® cards are fast and cost-effective, but have less VRAM than Quadro® and Tesla cards. OctaneRender® scales well in a multi-GPU configuration, and can use different types of NVIDIA® cards at once, such as a GeForce® GTX 1080 combined with a Quadro® 6000. The official list of NVIDIA® CUDA®-enabled products is located at https://developer.nvidia.com/cuda-gpus.

OctaneRender® does not require RTX, but it does render some scenes much faster when RT Core hardware is present.

To use the engine's out-of-core features, we recommend using at least the following hardware:

- 8-Core CPU
- 16 GB RAM
- A CUDA®-enabled card with at least 2 GB VRAM

Looking To Buy a New GPU For OctaneRender®?

There are several things to consider when purchasing a new GPU. You'll want to purchase a video card with the largest amount of VRAM and the most CUDA® cores for your budget. Make sure your power supply can handle the new card as well.

---

[1]The GPU is responsible for displaying graphical elements on a computer display. The GPU plays a key role in the Octane rendering process as the CUDA cores are utilized during the rendering process.

To use the OctaneRender® denoiser features, you need additional memory to collect all necessary information. As an example, a 4k render requires around 5 GB, while an 8k render requires around 20 GB. High-definition renders require around 0.5 GB.

Memory is also required for geometry, textures, post-processing buffers, and for other 3D modeling software, so it's necessary to increase the system RAM along with about 450 MB VRAM on devices to run the denosier.

Use out-of-core features to move geometry and textures onto system memory to free up some space for the denoiser on the device.

Internet Access

Except for the demo versions, all OctaneRender® editions require authentication with its designated license key, and it requires internet access during the initial launch. Once you launch the program, OctaneRender® requests your OTOY credentials, and it attempts to retrieve an available license from the OctaneRender Live™ server.

Software Requirements

OctaneRender® for 3DS Max® requires:

- **3DS Max® Octane Plugin** - If you do not own the current plugin, you can purchase it on our purchasing page, or you can upgrade your existing license.
- **3DS Max® Host Application** - 3DS Max® host application versions 2013 and above for Windows®.
- Microsoft® Windows® 7 - current (64-bit).

NVIDIA® cuDNN Library File

OctaneRender® requires NVIDIA® CuDNN to run. You can download the cuDNN library file from here: https://render.otoy.com/downloads/e5/c5/f2/20/cudnn64_7.dll

The library file should be placed in the either of these folders:

```
C:\Users\[user]\AppData\Local\OctaneRender\thirdparty\cudnn_7_4_1\
or
C:\Program Files\OTOY\[OctaneRender Enterprise 2020.1.x]\
```

Drivers

OctaneRender® requires an NVIDIA® driver supporting at least CUDA® 9.1, and a graphics driver version 388.x or higher. Use NVIDIA Studio driver with version at least 435.80 for Windows to enable support for RTX hardware acceleration. You can find the download links for Windows® here.

Failure to install these driver versions may create instability. We cannot provide support to users of different driver versions.

The CUDA® driver is the part of the NVIDIA® driver stack that OctaneRender® uses. On Windows®, it is part of the NVIDIA® graphics driver.

Downgrading Drivers

Some drivers are released in a beta state and may cause errors when running OctaneRender®. You can revert to the previous driver by downgrading or installing an earlier but more suitable driver version.

After downloading a suitable driver from https://www.nvidia.com/Download/index.aspx, use DDU to remove the installed driver, then perform a clean installation of the preferred driver.

Installation

To install the OctaneRender® plugin:

1. Download the software from **Latest Additions To OctaneRender** and run the plugin installer. You must be logged in as admin to do this. While you are installing the program, you will be prompted to choose from a list of components. Install the components you need.

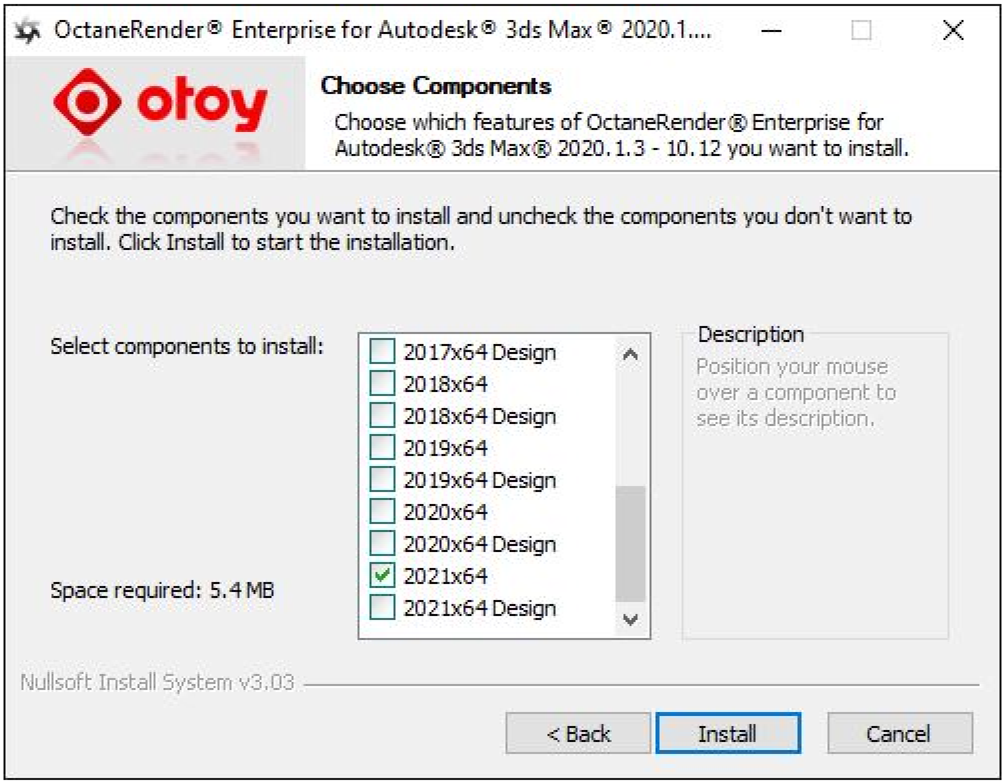
2. 

Figure 1: Choosing the installed versions of 3DS Max®

Setting Up The Plugin In 3DS Max®

Open 3DS Max® and click on the Render Setup menu by pressing F10 on the keyboard, or by pressing the Render Setup  button on the main toolbar.

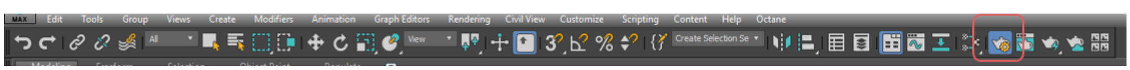


Figure 2: Accessing the Render Setup button from the main toolbar

Click on the **Renderer** dropdown and assign the OctaneRender® for 3DS Max® plugin.

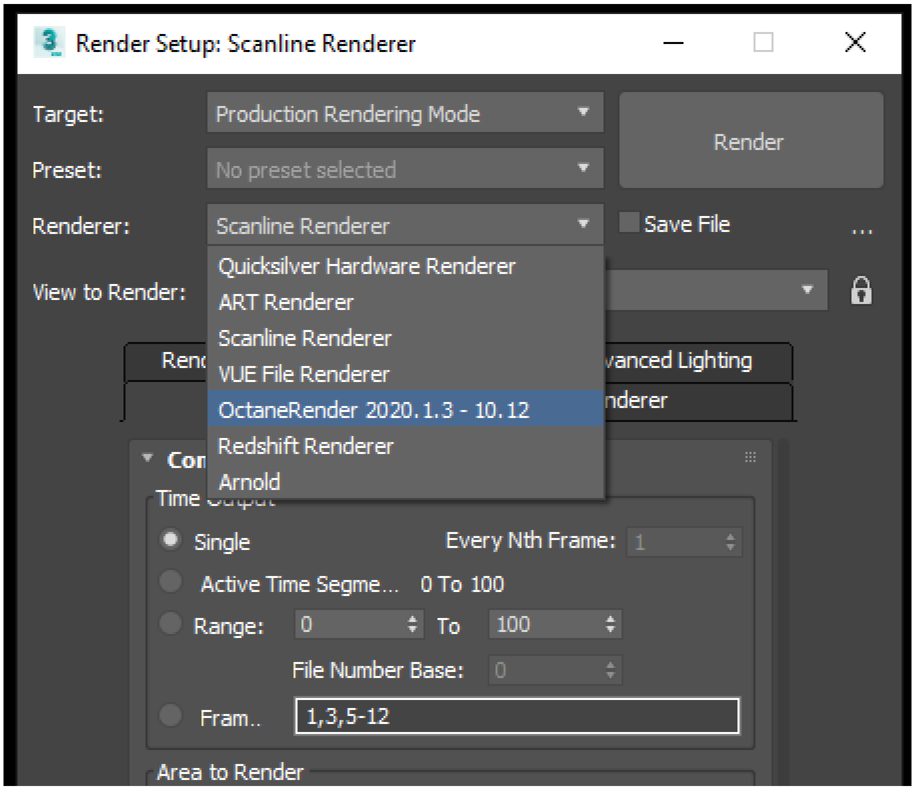


Figure 3: Choosing OctaneRender from the Renderer dropdown menu

Devices

OctaneRender® is a **GPU**[1]-based render engine. It is important to manage the GPUs in the system that are used for rendering. This is done from the **Devices** tab under **Render Settings**. Under this tab, the check-boxes for unsupported GPUs are not shown, and you can enable just the GPUs with a supported compute model.

[1] The GPU is responsible for displaying graphical elements on a computer display. The GPU plays a key role in the Octane rendering process as the CUDA cores are utilized during the rendering process.

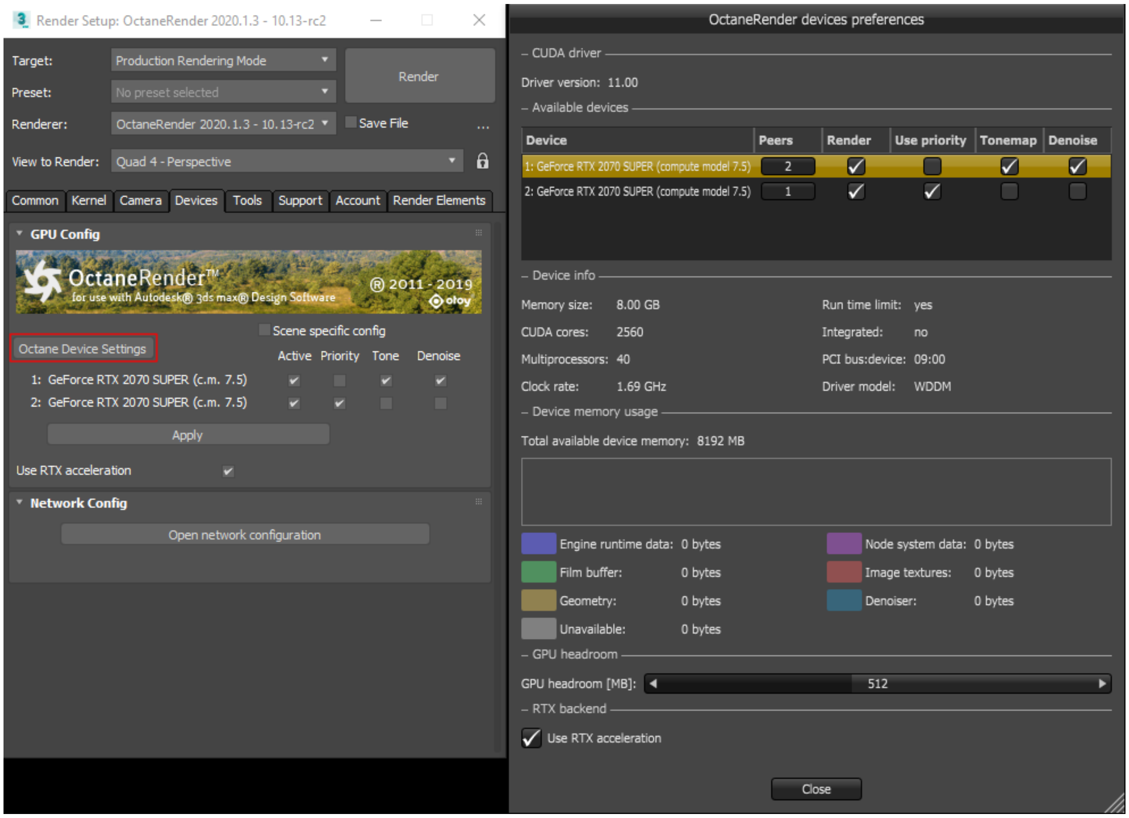


Figure 4: The Devices tab

Use Priority enables the **Priority** option, which sets the render priority for the active GPUs. The render priority is necessary when a GPU is not dedicated to rendering, but is also shared among different processes in one machine. For example, a machine with one GPU shares the GPU for processes across the whole system, including the operating system. You usually set the desktop GPU as Priority to avoid desktop hiccups and slowdown. From here, you can also specify what GPUs are used for denoising and which ones are used for tonemapping.

Authenticate And Internet Access

Signing In To The Octane Licensing System

You need an internet connection before starting OctaneRender® for the first time in order for it to communicate with the OctaneRender® licensing system. When you start the application, this sign-in screen appears.

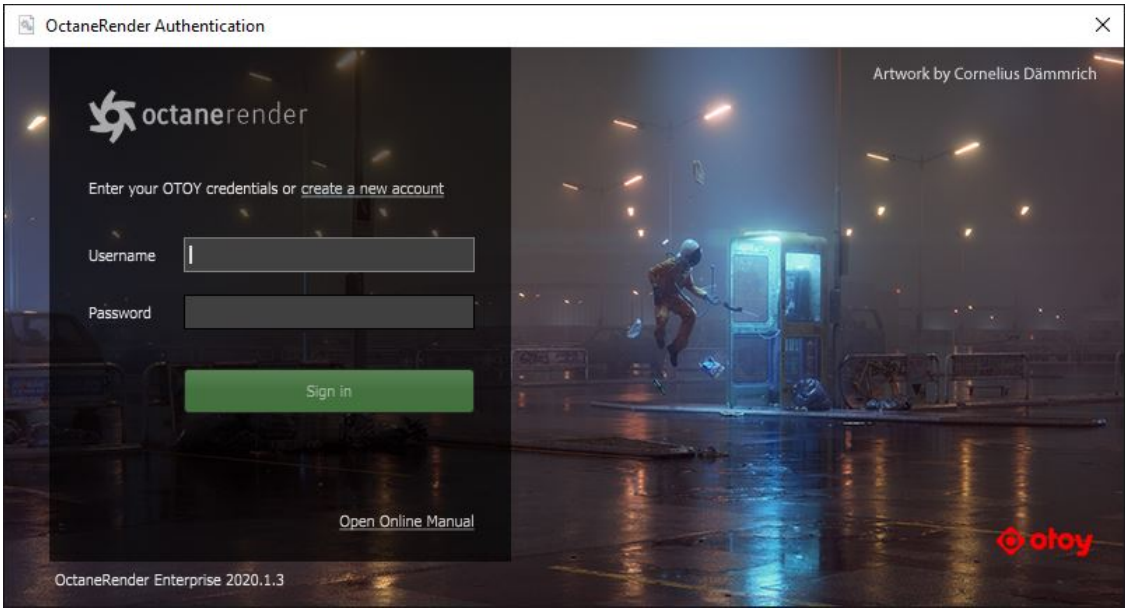


Figure 1: Signing in with OTOY® credentials

Enter your OTOY® account username and password, then click the Sign in button.

At this point, the single sign-on and licensing system pulls a valid license key from your account on OTOY's secure server.

If OctaneRender® detects a connection problem, make sure all communications use HTTPS (TCP port 443) for the following:

- Standalone edition
- Standalone edition daemon
- Your OctaneRender® plugin's host application, if you are using a plugin

The above may require updating your firewall settings. If the issue persists, check your proxy settings. Refer to the HTTP **Proxy**[1] Support topic in this manual for more information.

After signing in, OctaneRender® keeps a session alive as long you run the Standalone or the plugin application on a regular basis. In most cases, you should not have to sign in to the OctaneRender® licensing system again.

---

[1]An object saved as a separate file with the purpose of being reused in larger scenes. This is used to minimize any addition to the total polygon count in the scene, especially if the scene requires the same object to appear several times. If used in conjunction with instancing, Proxies help keep very large scenes from reaching polygon limits and also keeps the relative file size of the main project file manageable.

This session also lets you link your local installation to other OTOY® services like Octane Render Cloud® (ORC™).

Manually Signing Out Of The OctaneRender® Licensing System

To close a session, click on **Render Settings** -> **Account** tab -> **Activation / Deactivation dialog** -> **Sign out** button. This closes the current session and releases all licenses bound to the current machine. You must close all OctaneRender® applications before continuing to sign out.

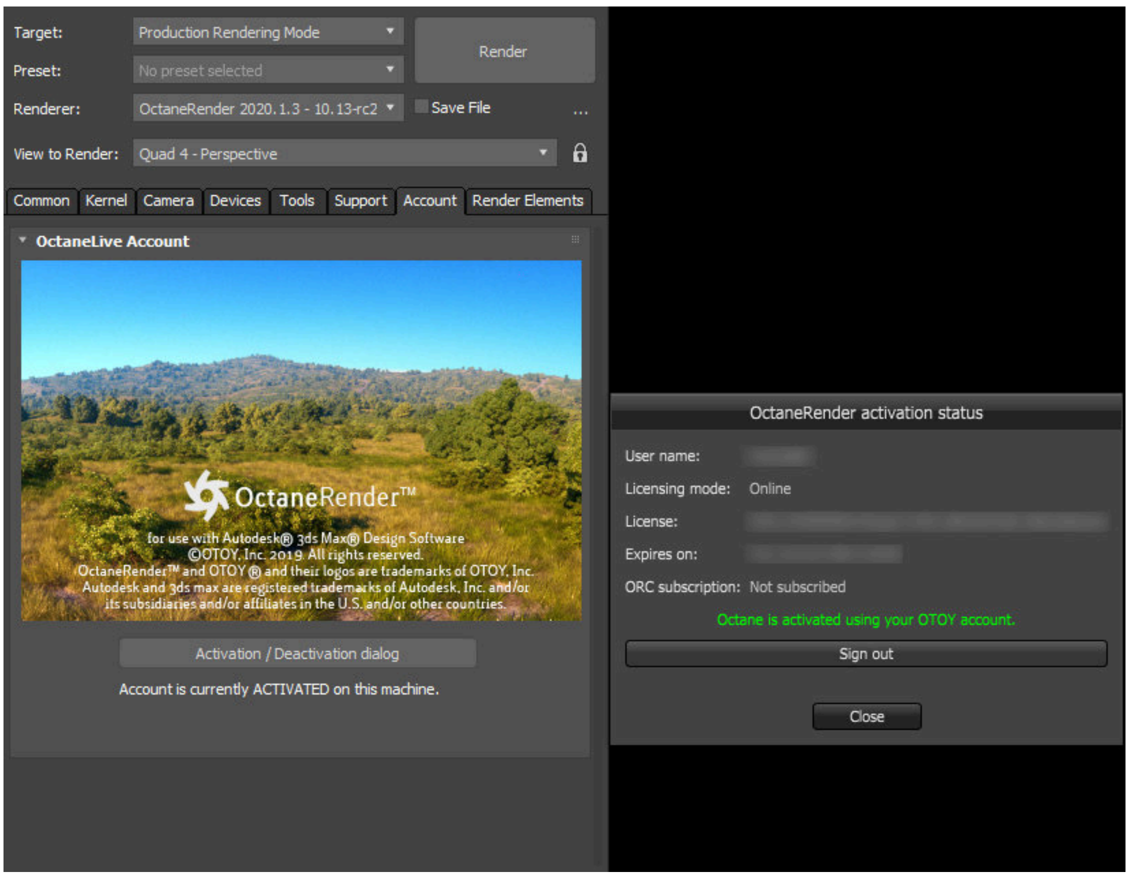


Figure 2: The OctaneRender Activation Status window

OctaneRender® License Management

OctaneRender® manages licenses several ways:

- If you have the Standalone edition, you must have one Standalone license to run the application.
- For each OctaneRender® plugin you have, you must have a Standalone license and one license for that plugin to run it.
- You must have one Standalone license for each machine you run OctaneRender® products on.
- You can run any number of Standalone and OctaneRender® instances on one machine.

Failsafe Web Deactivation (Unlocks)

Licenses deactivate when you shut down the OctaneRender® application. However, if your machine crashes and your application does not close properly, the license may still be active on your machine. In this case, you need to restart the application again to release the license. Other scenarios that could prevent the license from deactivating your license includes:

- The hard disk containing your OctaneRender® license data has crashed and you replace it.
- You erase the hard disk - for example, as part of a disk reformat or partition change.
- You change the network interface card and your network identification data changes for that machine.
- The OctaneRender® license data on your machine is corrupt.

In all of these situations, the OctaneRender® licensing system thinks your license is still active on that machine, but you and the licensing system are no longer able to access the license data. However, you can still deactivate your license using failsafe web deactivation.

You can deactivate or unlock a license (both are referred to as **Unlocks** on the licensing page) for a limited number of times over a given period. This number may change without notice in the future. You can see how many Unlocks you have remaining on your OTOY® account page.

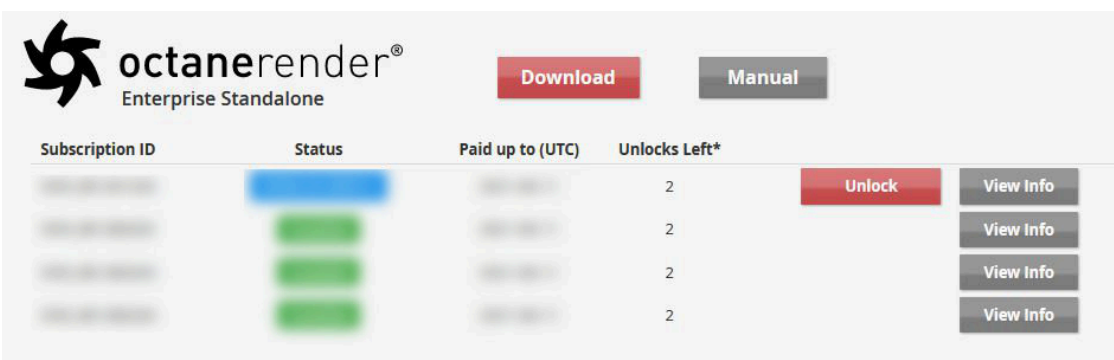


Figure 3: OctaneLive Accounts page

To deactivate and unlock the license, click the **Deactivate** or **Unlock** button, depending on the version number of the license. If you run out of unlocks, contact us at help@otoy.com to request an unlock.

OctaneRender Live™ IP/URL Whitelisting

These are the URLs that OctaneRender® requires access to as part of the activation process:

- Account.otoy.com
- Bridge.octanerender.com
- Live.octanerender.com

As the services above use dynamic load-balancing, the IPs of these servers often change. If you are whitelisting by IP, you need to redirect traffic through our gateway server at 52.1.219.88. You can do this by overriding the octanerender.com domain on your internal DNS server, or by setting the IP manually in your machine's hosts file.

Host file locations:

- Windows®:
  C:\Windows\System32\drivers\etc\hosts
- macOS®:
  /private/etc/hosts
- Linux:
  /etc/hosts

Host file entries:

```
52.1.219.88 account.otoy.com
52.1.219.88 bridge.octanerender.com
52.1.219.88 live.octanerender.com
```

If you have any problems, contact us here.

Unattended/Silent Authorization (Online Mode Only)

Another option for authorization is using an unattended authorization token. This allows OctaneRender® to activate in online licensing mode without requiring you to enter in a username and password. This works by saving an authorization file downloaded from our SSO system to every machine that runs OctaneRender® in a user-agnostic directory. This authorization file can be configured to only be valid for requests from a specific CIDR, preventing this file from being used outside your environment.

If the authorization file does not exist on a machine, you can still run OctaneRender®, but you need to enter the username (or login email address) and password.

As this feature has security implications for the licenses in your account, you will need to contact support to request that this feature be enabled for your account. As this feature is an extension of online licensing mode, it requires an active internet connection to function. To request this feature, contact us here.

HTTP Proxy Support

If you run OctaneRender® behind an existing proxy, OctaneRender® tries to find and use your current proxy setup. Workarounds to bypass the proxy settings are invalid.

If you are trying to setup your proxy for the first time or your proxy requires authentication, you may configure it either by using your operating system proxy settings or environment variables.

Proxy Server Configuration Via System Settings

This option allows OctaneRender® to retrieve your system settings. The configuration depends on your host operating system.

Windows®

OctaneRender® can obtain its proxy configuration several different ways.

From Internet Explorer's LAN Settings, this configuration applies only to the current user. To change IE proxy settings:

1. Press the **Win**+**R** keys.
2. Enter `inetcpl.cpl,4` and click **OK**. The **Internet Properties** window displays.
3. Click **LAN Settings**.
4. Select the **Use A Proxy Server[1] For Your LAN** checkbox.

---

[1]A Proxy Server, also known as an application-level gateway, is an intermediary server between the local network and the external servers from which a client is requesting a service. The external servers will only see the network proxy server's IP address thus providing some degree of security and privacy. There are various kinds of proxies, the most common are Web Proxies.

5. In the **Address** box, enter the proxy server's IP address.
6. In the **Port** box, enter the port number.

If you have a dedicated proxy for HTTPS traffic, click on **Advanced**, clear the **Use The Same Proxy**[1] **For All Protocols** checkbox, and specify the proxy address and port for the **Secure** server type.

From The WinHTTP Configuration

This configuration is system-wide, and stored in the registry. You can manage it using `netsh winhttp`. For more information, please check Windows HTTP documentation from Microsoft®. The proxy exceptions list is ignored.

Note: Environment variables are case sensitive, even on Windows®, for security reasons.

The accepted syntax for proxy environment variables is [protocol://][user:password@]proxyhost[:port]

For example, you may specify a proxy for HTTPS network traffic as https_proxy-y=johndoe:mypass@127.0.1.50. This tells OctaneRender® to use **127.0.1.50** as your proxy's address using the default port **80**, and authenticate as user **johndoe** with password **mypass**.

Hardware Options

The options for adding **GPU**[2] muscle to a computer depends on its available PCI-E slots.

OctaneRender® can handle 200 concurrent GPUs with enterprise all access as long as they're set up as CUDA® devices. It does not need to be SLI-enabled to detect additional GPUs in the machine, and it is not recommended for render engines - OctaneRender® runs much better without it because it can't differentiate GPUs accessible from the local area network or in the PC's PCI-E slots.

Single PCI-E Slot

If the computer has a single PCI-E slot, there are not many options to extend rendering performance. You could add a more powerful GPU, as long as the power supply can provide enough power for it. Dual-GPU,

---

[1] An object saved as a separate file with the purpose of being reused in larger scenes. This is used to minimize any addition to the total polygon count in the scene, especially if the scene requires the same object to appear several times. If used in conjunction with instancing, Proxies help keep very large scenes from reaching polygon limits and also keeps the relative file size of the main project file manageable.

[2] The GPU is responsible for displaying graphical elements on a computer display. The GPU plays a key role in the Octane rendering process as the CUDA cores are utilized during the rendering process.

single-slot card solutions can work, assuming that the power supply is sufficient to power the video card.

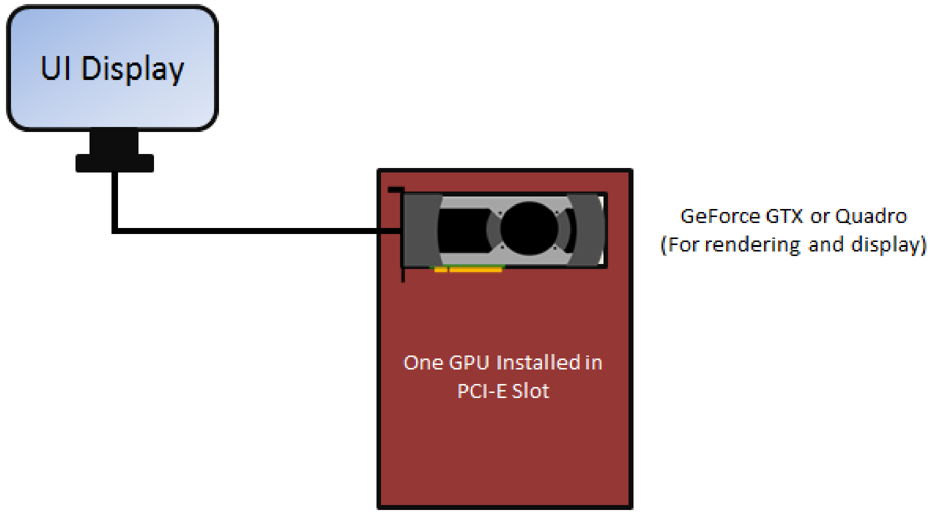


Figure 1: Single PCI-E slot configuration

Two Or More PCI-E Slot Motherboards

If the computer has two PCI-E slots, then you have more expansion options. If the power supply is sufficient, you can dedicate one GPU to the OS display, and then dedicate two or more GPUs for rendering. For the smoothest user experience with OctaneRender®, we recommend dedicating one GPU for the display and OS to avoid slow and jerky interaction and navigation. The dedicated video card could be a cheap, low-power card since it will not be used for rendering, and it should be disabled under **CUDA** devices in **Device Manager** > **Preferences**.

In this situation, we recommend matching the rendering GPUs in model and VRAM size. You can do multi-GPU rendering, but the OS interface may still be slow as all the GPU processing power is dedicated to the rendering process. In multi-GPU setups, the amount of RAM available to OctaneRender® is not equal to the sum of the VRAM on the GPUs, but it is restricted to the GPU with the smallest amount of VRAM. We recommend disabling GPUs that don’t have enough VRAM in order to render large scenes that can fit in the remaining GPU's VRAM.

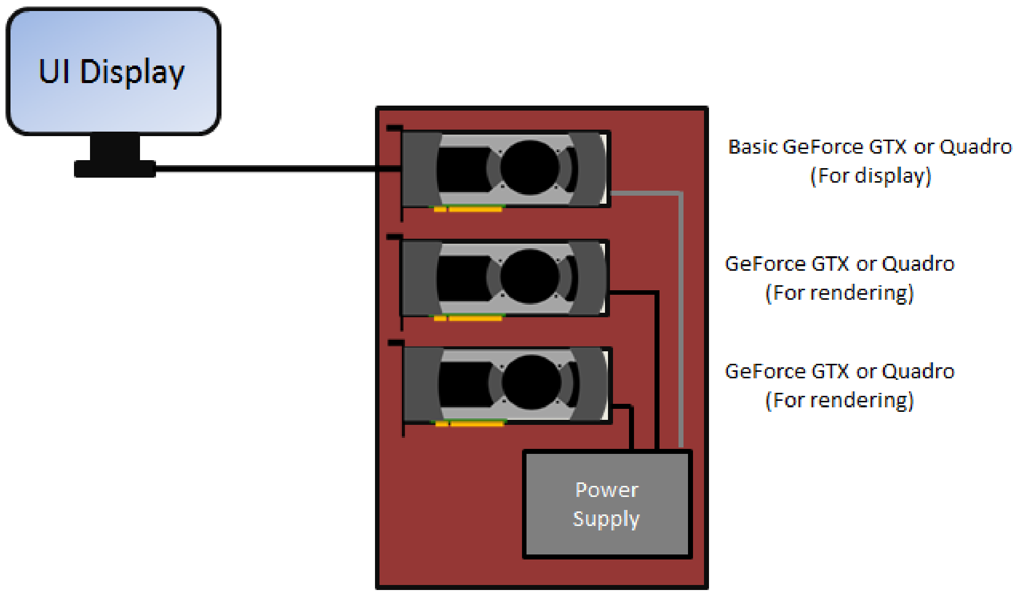


Figure 2: Multiple PCI-E slot configuration

Networked Primary Node And Render Node Machines

If a local area network is available, then you have many additional upgrade options. However, this requires each render node machine in the network to have its own designated OctaneRender® license.

Just like in a multi-GPU setup, it is best to have the GPUs match in model and VRAM size. You can do multi-GPU rendering, but the OS interface may still be slow as all the GPU processing power is dedicated to the rendering process. The amount of RAM available to OctaneRender® is not the total amount of VRAM from your GPUs, but it is the amount of VRAM from your smallest GPU. We recommend disabling GPUs that don’t have enough VRAM in order to render large scenes that can fit in the remaining GPU's VRAM.

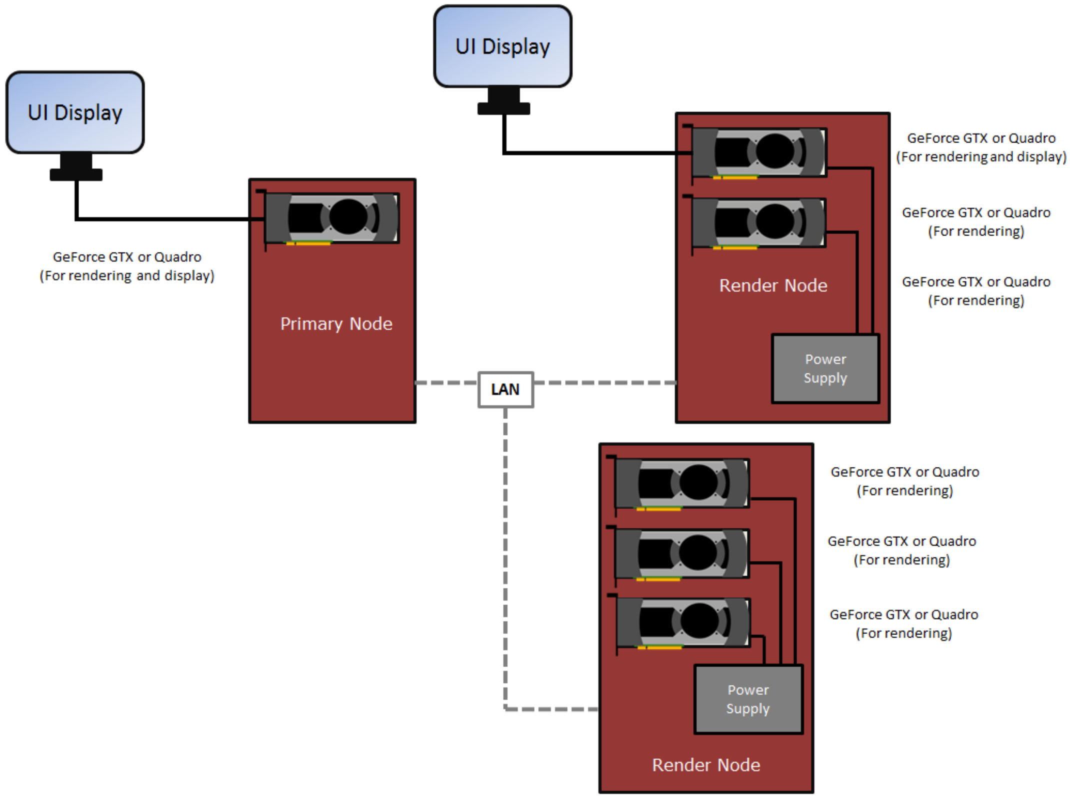


Figure 3: Networked GPU configuration

Multi-GPU Setups, Power Supply, And Energy Consumption Considerations

It is important to use a suitable power supply when using multiple GPUs. For more info on what power supply is best for your case, visit http://www.nvidia.com/object/slizone_build_psu.html.

The differences in the micro-architecture of the cards should also be considered. For instance, the Kepler cards have more memory and consume less power than Fermi GPUs, but are just as fast with OctaneRender®. Newer cards in the Maxwell and Pascal™ series are also more power-efficient.

NVLink® On Consumer GPUs

NVLink® lets you double your VRAM by combining two cards into one pool of fast shared (not mirrored) memory. NVLink® works with two cards, both of which need to be Quadro® or Geforce® RTX cards. Make sure that you use the bridge over your cards (Figure 1), otherwise you may experience a large performance drop.

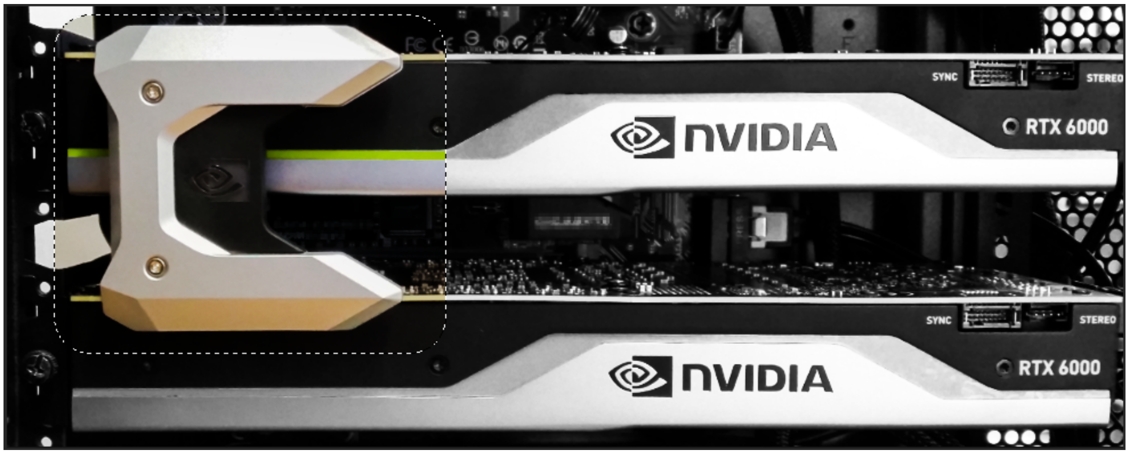


Figure 4: Example of a 3-slot NVLink® bridge connecting two Quadro® cards

To use NVLink® with non-Quadro® GPUs, enable SLI mode from the NVIDIA® Control Panel.

To use NVLink® with Quadro® GPUs, set the GPUs as Tesla Compute Cluster (TCC) devices. You can do this from the command line window with administrative privileges by running the `nvidia-smi` command within the NVSMI default folder (`C:\Program Files\NVIDIA Corporation\NVSMI`). The `nvidia-smi` command generates a table that displays your GPUs and what mode they are using (Figure 5).

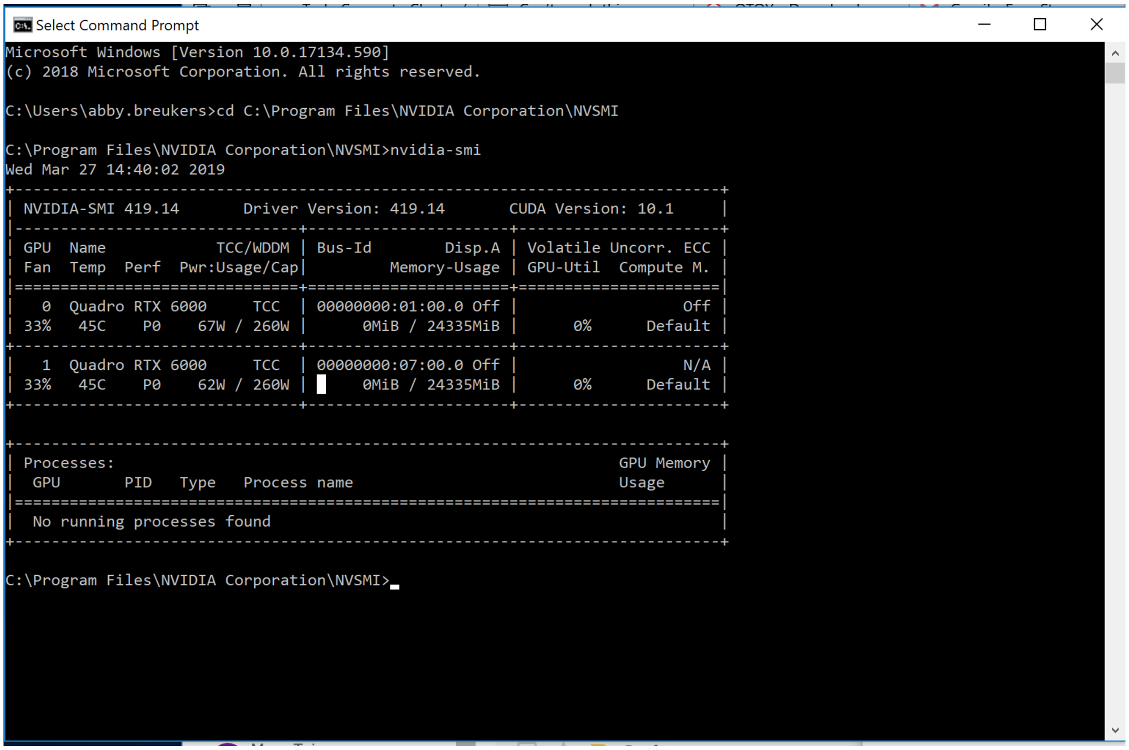


Figure 5: Table showing the devices in the machine, including the GPU ID and the mode of each device

To change the mode, use the following syntax in the command line:

```
nvidia-smi -g {GPU_ID} -dm {0|1}
```

Where, 0 = WDDM and 1 = TCC.

Examples

This command switches the first Quadro® GPU to WDDM mode:

C:\Program Files\NVIDIA Corporation\NVSMI**nvidia-smi -g 0 -dm 0**

This command switches the first Quadro® GPU to TCC mode:

C:\Program Files\NVIDIA Corporation\NVSMI**nvidia-smi -g 0 -dm 1**

When the devices are set and NVLink® is installed, you can combine p2p video memory. This is evident in the OctaneRender® **Devices** tab under **File** > **Preferences** (Figure 3). The device's Preferences window shows status info per device (Figure 4), not the total VRAM memory combined. OctaneRender® uses p2p when the the primary device's VRAM is maxed out (Figure 10).

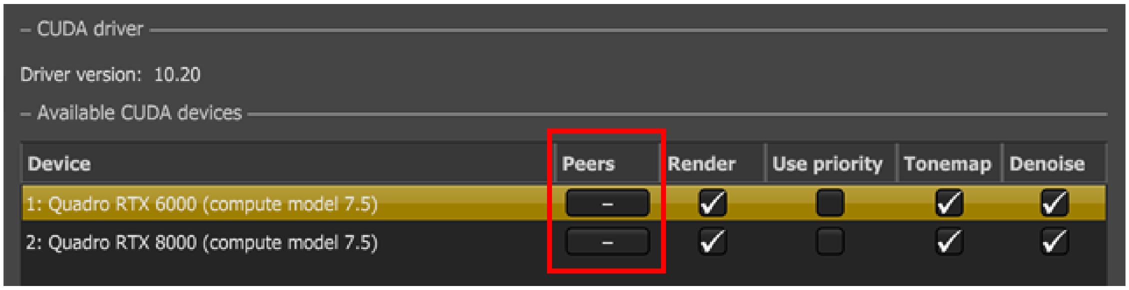


Figure 6: You can specify VRAM pooling peers for NVLink®-enabled devices

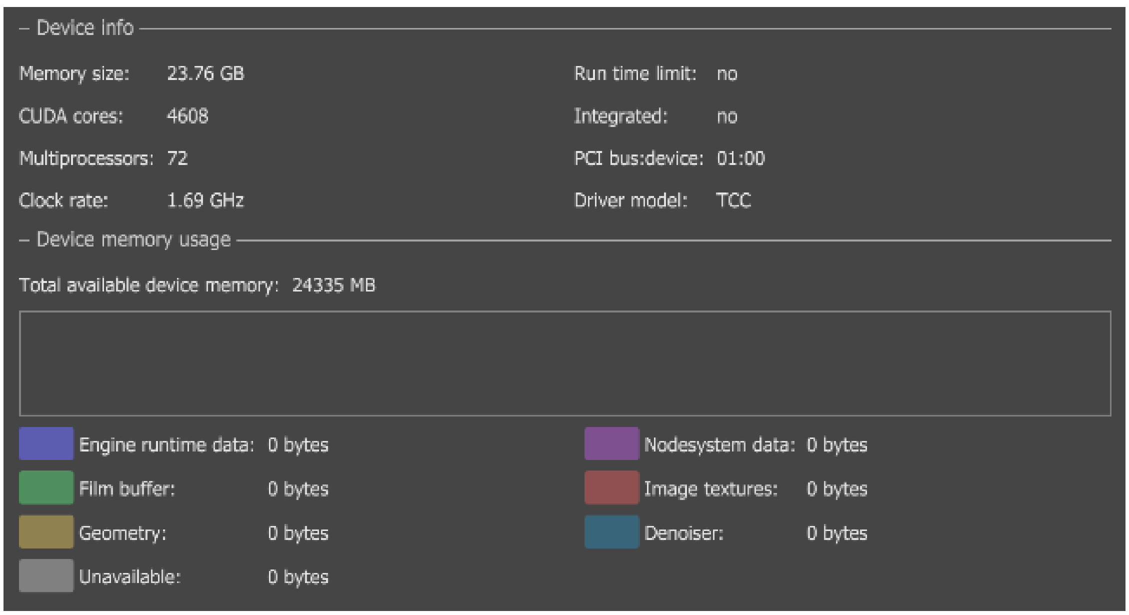


Figure 7: Device Info in the Preferences window shows info per device

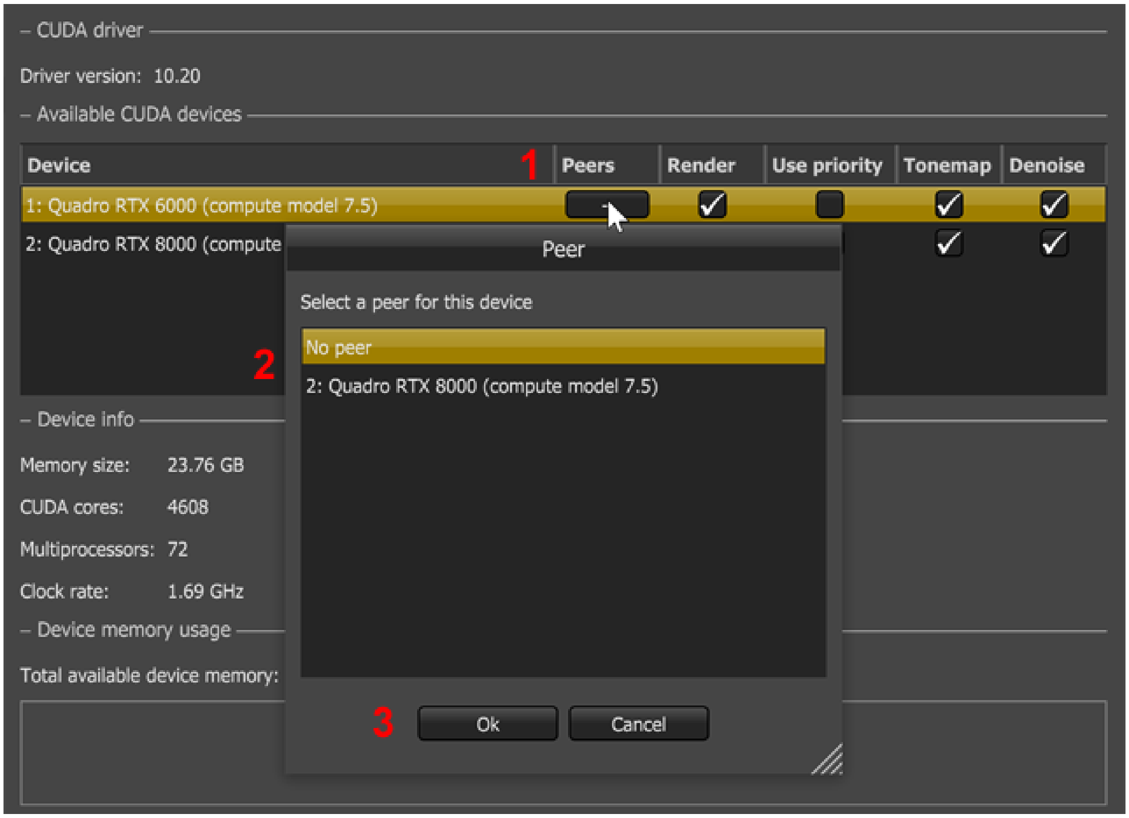


Figure 8: Setting the peers for the NVLink®-enabled devices

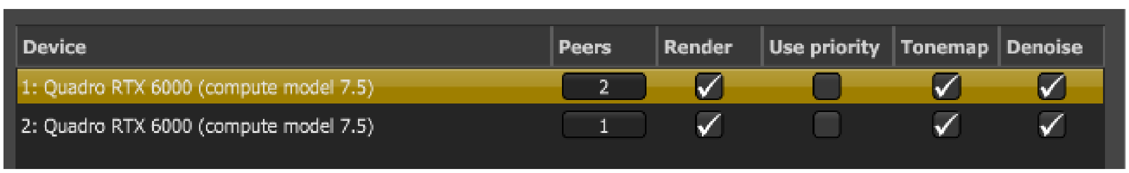


Figure 9: An example showing that device 1 is peered to device 2 and vice-versa

(Image Placeholder - Not implemented yet)

Figure 10: P2P is used when the primary device's VRAM is maxed out

Note: You can't connect a display or monitor to the GPU adapters when the underlying devices are running in TCC mode. This causes unpredictable behavior, and may result in having to reboot the entire system.

OctaneRender® Dongle Setup For Offline Mode

To set up OctaneRender® for offline use, you will need:

- A dongle
- DIT (Dongle Installer Tool)
- An internet-connected computer
- OctaneRender® Standalone v3.08.3 or newer

Once you have the items, follow these steps:

1. Add your license to your dongle. Your OctaneRender® standalone license is assigned to your dongle, pending activation. If you want to work offline with your OctaneRender® plugin, you also need to assign the plugin license to the dongle.
2. Navigate to the **Dongles** tab in your OTOY® account. Click **Add Licenses**, and select the plugin license. Next, click **Assign To Dongle**.

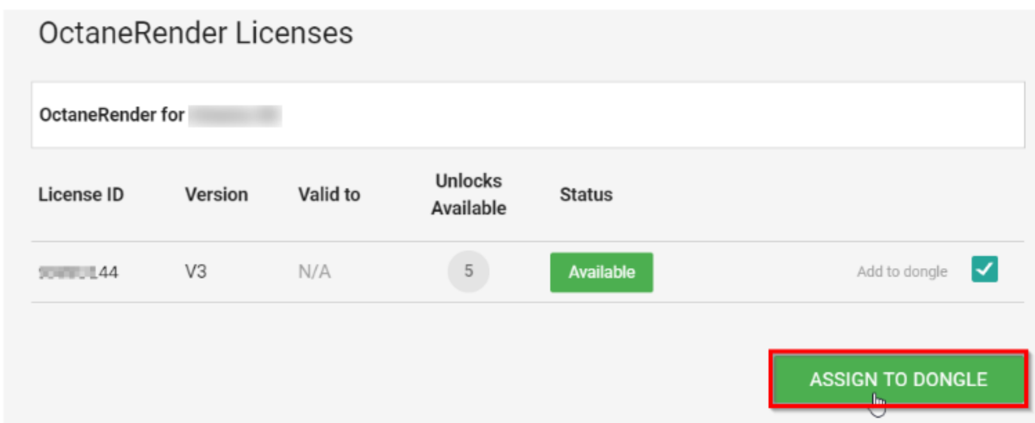


Figure 1: Information found in the your OTOY® account's Dongle tab

3. Download the [Dongle Installer Tool]() by clicking **Install Licenses**. You can select Windows®, macOS®, or Linux for the installer.

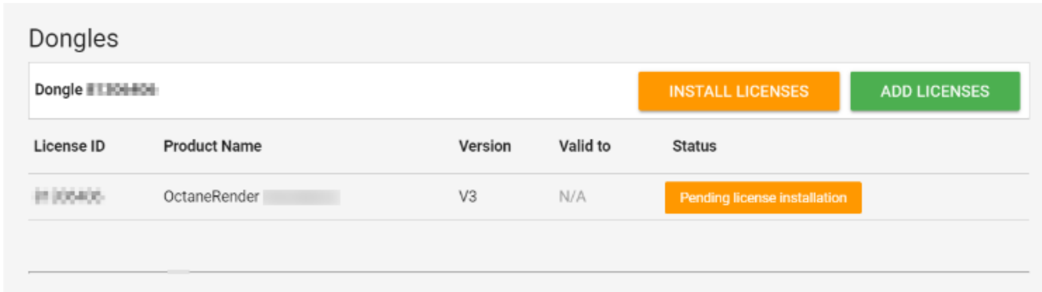


Figure 2: Adding licenses to the dongle

4. Launch the Dongle Installer Tool, then log in with your OTOY® credentials.
5. Click the checkbox in the **Select** column and click **Install Selected**. The installation may take a moment.

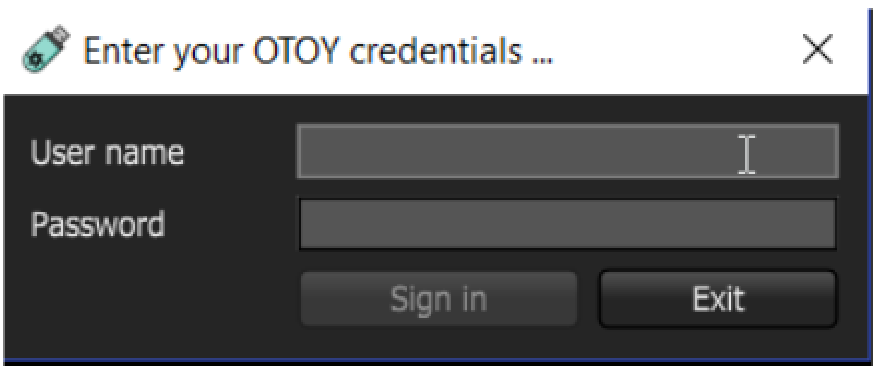


Figure 3: Entering your OTOY® credentials

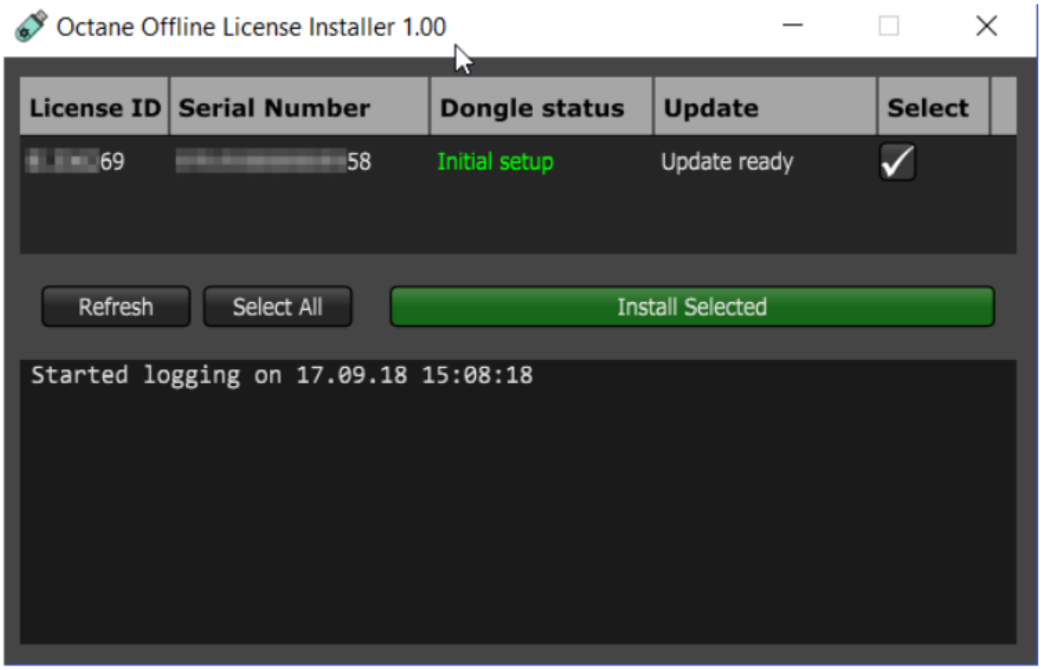


Figure 4: Installing the added licenses for offline use

6. You can now remove the dongle and use it with your designated offline machine. To make sure the dongle works, launch OctaneRender®. When you click on **File** > **Activation Status**, you should see **Octane is activated using an Octane dongle** at the bottom of the **Activation Status** window. You can add OctaneRender® plugin licenses to the dongle at a later time by repeating steps 1 and 2 for each license.

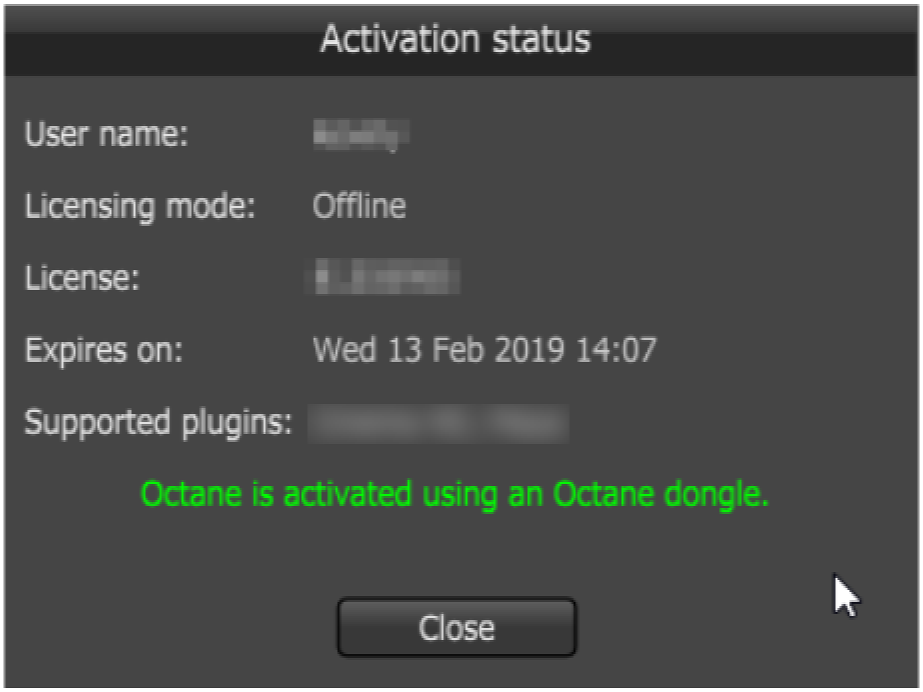


Figure 5: OctaneRender® activation status without an internet connection

7. You need to update dongles every four months with the Dongle Installer Tool. An error occurs during the update process if the licenses are still active so ensure to deactivate the licenses prior to updating the dongles. You can deactivate the license by **signing out** from the OctaneRender application's Activation Status dialog or by using the **Unlock** buttons from your online Otoy account. To update your dongle, run the Dongle Installer Tool, select the checkbox next to your dongle, then click **Install Selected**. Your dongle is now updated.

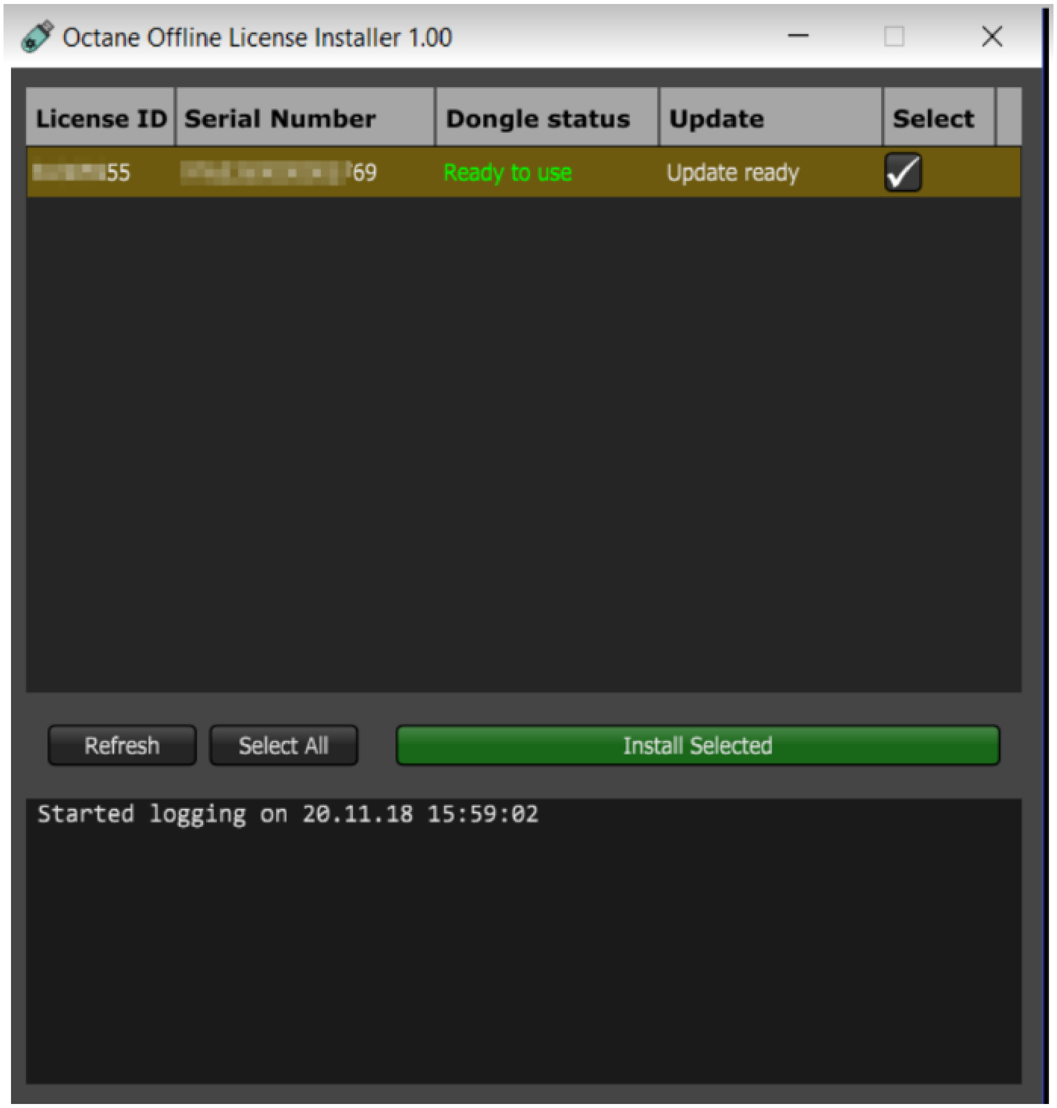


Figure 6: Updating a dongle

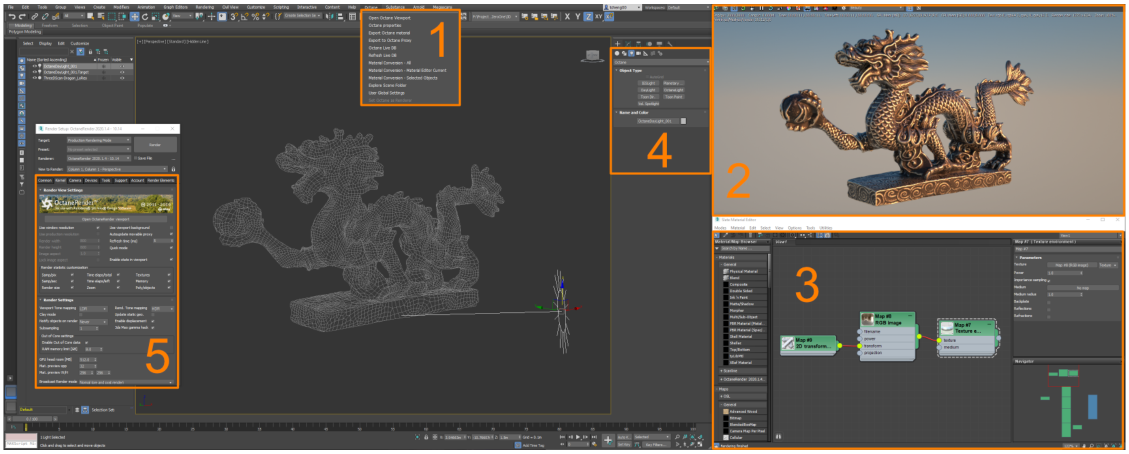
Note: By default, using the dongle in Linux systems requires `sudo` permissions. In order to get around this, copy the udev rules file, [`99-senselock.rules`](), into the udev rules directory. The udev rules directory is usually `/etc/udev/rules.d`, but this may vary across various Linux distributions. With the udev rules files in place, restart or force the udev daemon to reload the rules (`udevadm control --reload-rules`). See the Debian documentation at https://wiki.debian.org/udev for more information.

Octane for 3DS Max® Plugin Quick Overview

This document assumes you have a basic understanding of 3DS Max®. There are five main areas where you will control Octane for 3DS Max®. They are:

1. Octane Menu
2. OctaneRender® Viewport
3. **Material**[1] Editor
4. Lights and Cameras
5. Render Setup

[1]The representation of the surface or volume properties of an object.



Octane Menu

This is the quick access menu of common features used in Octane for 3DS Max®. By default, there are 12 options here. For beginners, keep in mind **Set Octane as Renderer** and **Open Octane Viewport**.

- **Set Octane as Render** - set Octane as the Render engine to use.
- **Open Octane Viewport** - start the OctaneRender® Viewport and display your renders.

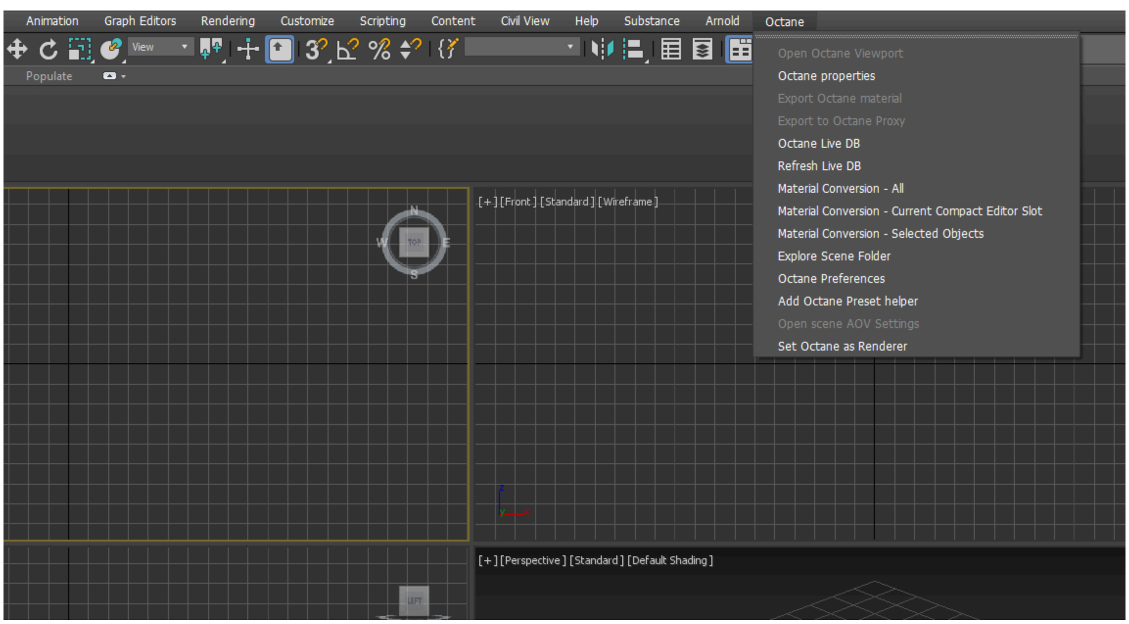


Figure 1: The Octane menu

Using the OctaneRender® Viewport

The OctaneRender® Viewport window is where you will do all your render work. The view will dynamic change to follow your active 3DS Max® Viewport. You can save your final still renders from here. For animation, use the standard 3DS Max® render window. For beginners, the main tools to understand are the **Save Image** and **Lock Camera** button.

-  **Save Image** - Saves the current render from the OctaneRender® Viewport to disk.
-  **Lock Camera** - Locks the camera for the 3DS Max® Viewport that you want to render. If you disable this function, 3DS Max® restarts the render each time you select a different 3DS Max® Viewport.

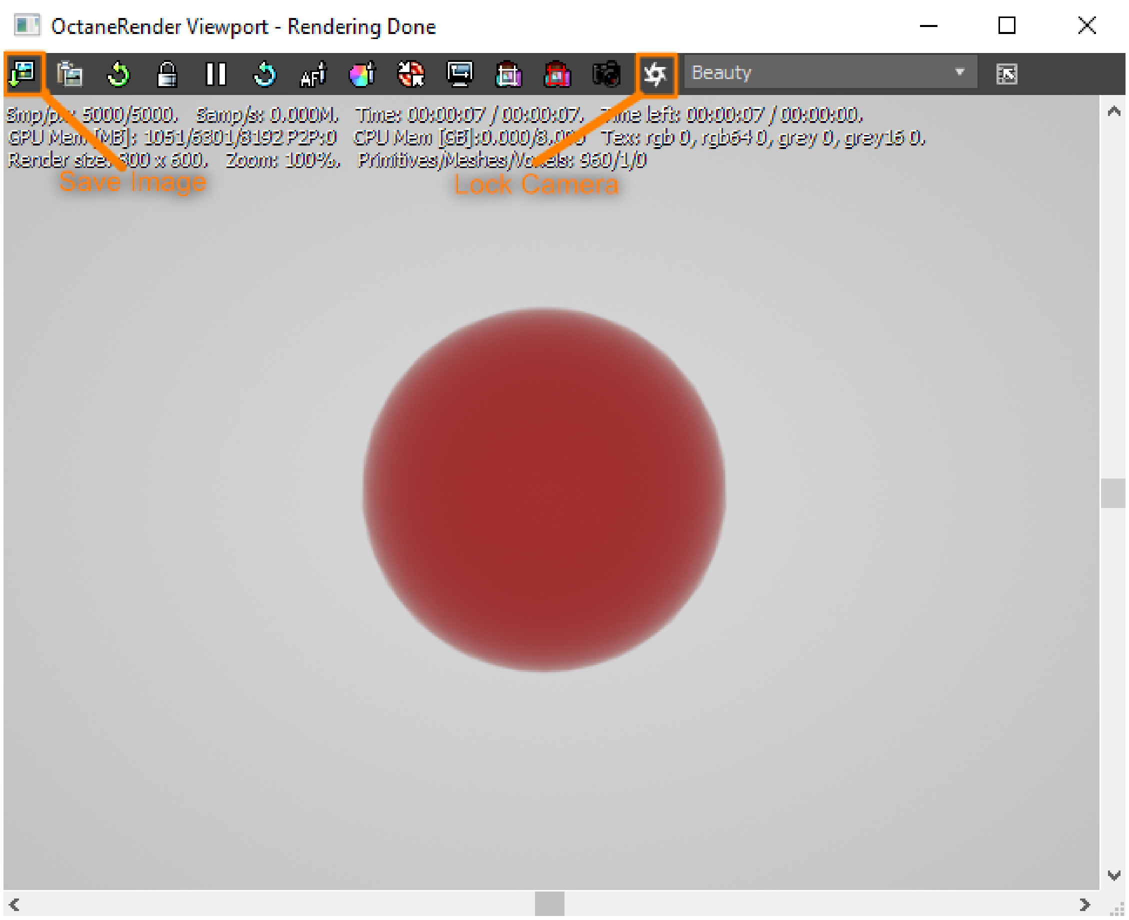


Figure 1: The OctaneRender® Viewport window

Material Editor

You use either the Slate or Compact **Material**[1] Editor to work with Octane materials and maps. For this example, we will focus on the Slate Material Editor. The Octane materials and maps are under the "OctaneRender..." groups in the Material/Map Browser. For beginners, focus on the **Universal Material**, **RGB Image,** and **2D transformation** nodes.

- **Universal Material** - integrates well with **PBR**[2] (Physically Based Rendering) workflows like Substance Designer.
- **RGB Image** - imports external texture maps to any material parameters that accept a texture map.
- **2D Transformation** - sets the rotation, scale, and translation of the texture.

[1] The representation of the surface or volume properties of an object.

[2] A contemporary shading and rendering process that seeks to simplify shading characteristics while providing a more accurate representation of lighting in the real world.

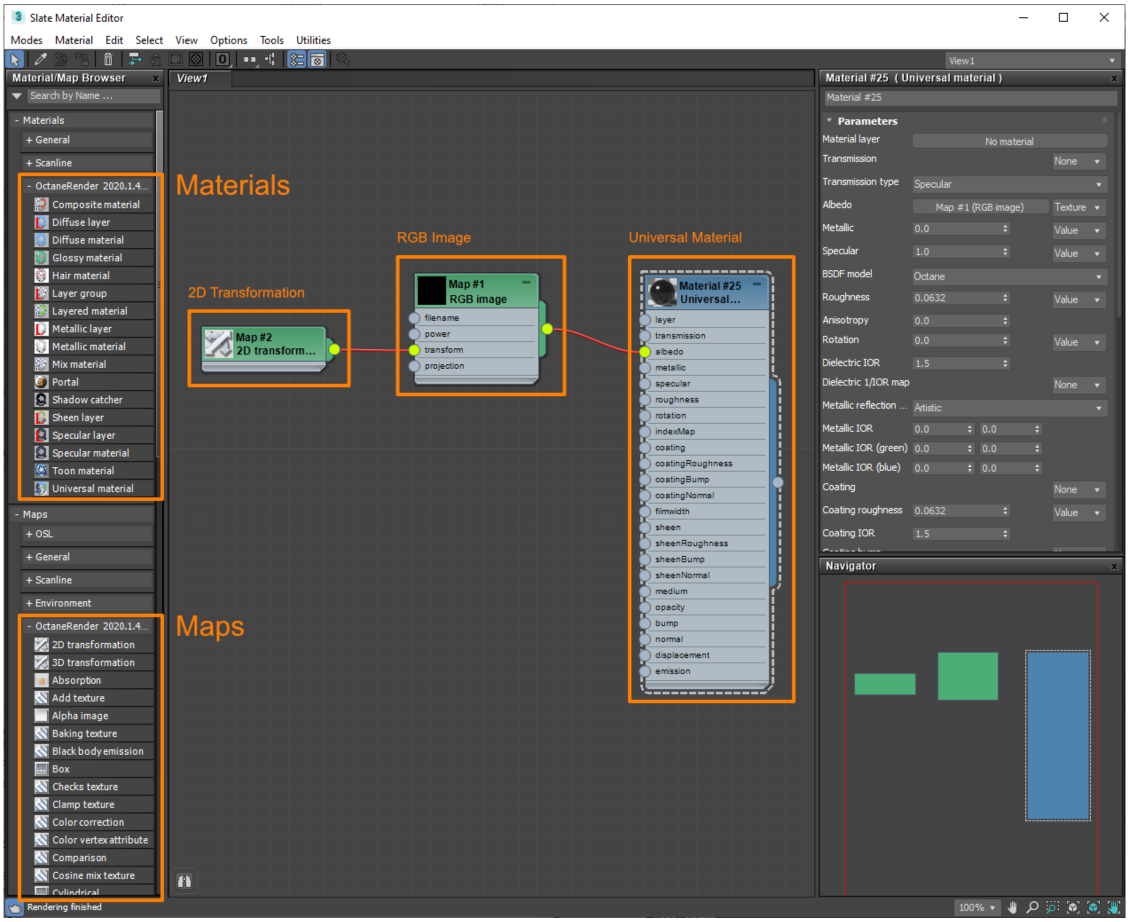


Figure 1: Slate Material Editor

Lights And Cameras

Use the **Create Panel** to setup Octane lights and cameras. For beginners, focus on **OctaneLight** and **Universal Camera.**

- **Octane Light** - provides a light source object to use for fast lighting setups without adding new geometry and materials.
- **Universal camera** is a full-featured camera, with support for three different camera types:
  - **Thin lens**
  - **Fisheye**
  - **Aberration And Distortion**

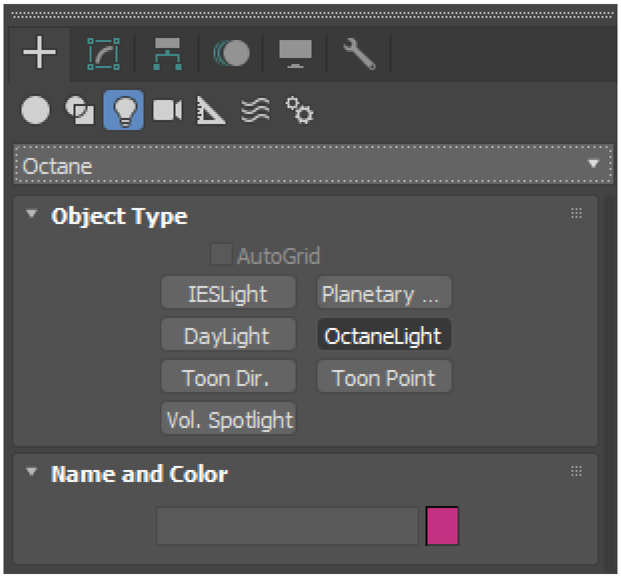


Figure 1: Octane Lights

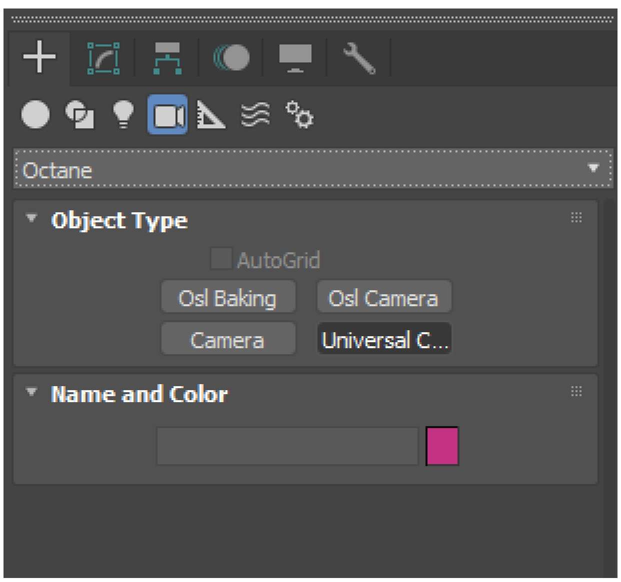


Figure 2: Octane Cameras

Render Setup

To make changes to the Octane render settings, open the 3DS Max® **Render Setup** window.

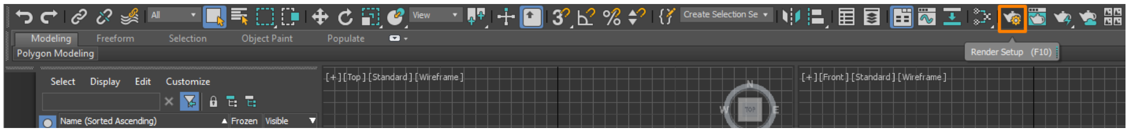


Figure 1: Render Setup Button

For the most part, you can leave the settings at default. For beginners, you should pay special attention to some settings in the **Kernel** and **Devices** Tabs. Specifically:

Kernel Tab

- **Render width** (see Fig. 3) - Width of the rendered image in OctaneRender® Viewport
- **Render height** (see Fig. 3) - Height of the rendered image in OctaneRender® Viewport
- **Max. Samples** (see Fig. 4) - Lower samples render faster, but with lower quality. Higher samples render slower, but with higher quality.

Devices Tab

- **Active** (see Fig. 5) - Select the **Active** **GPU**[1] device(s) used for Octane, then apply the settings.

[1]The GPU is responsible for displaying graphical elements on a computer display. The GPU plays a key role in the Octane rendering process as the CUDA cores are utilized during the rendering process.

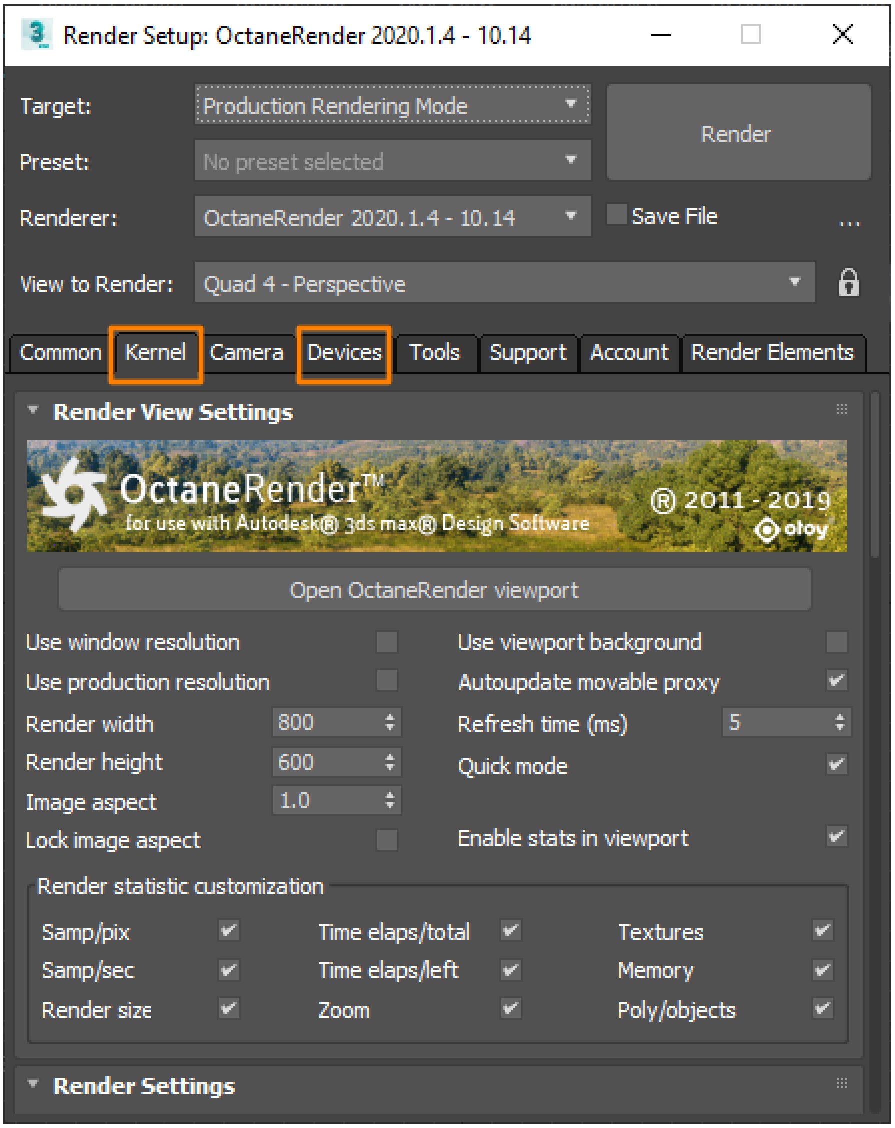


Figure 2: Render Setup Window

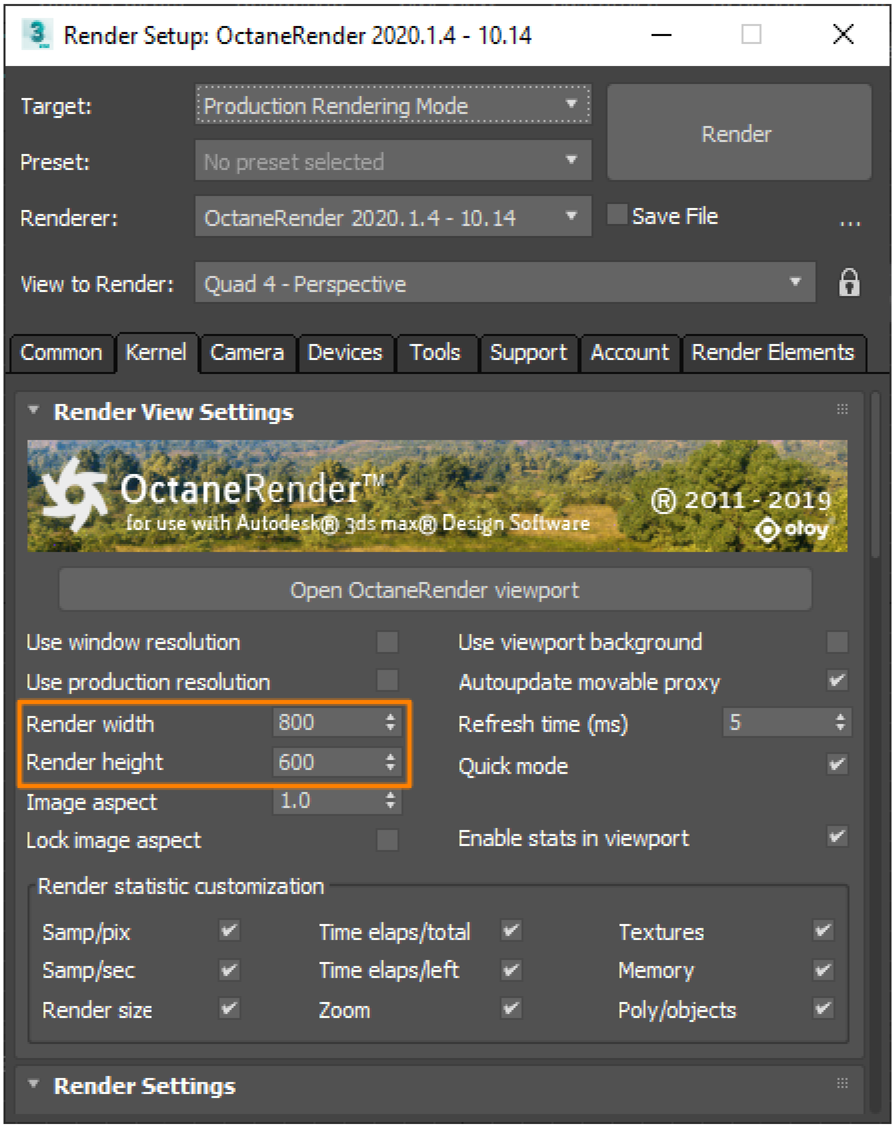


Figure 3: Render Width and Height

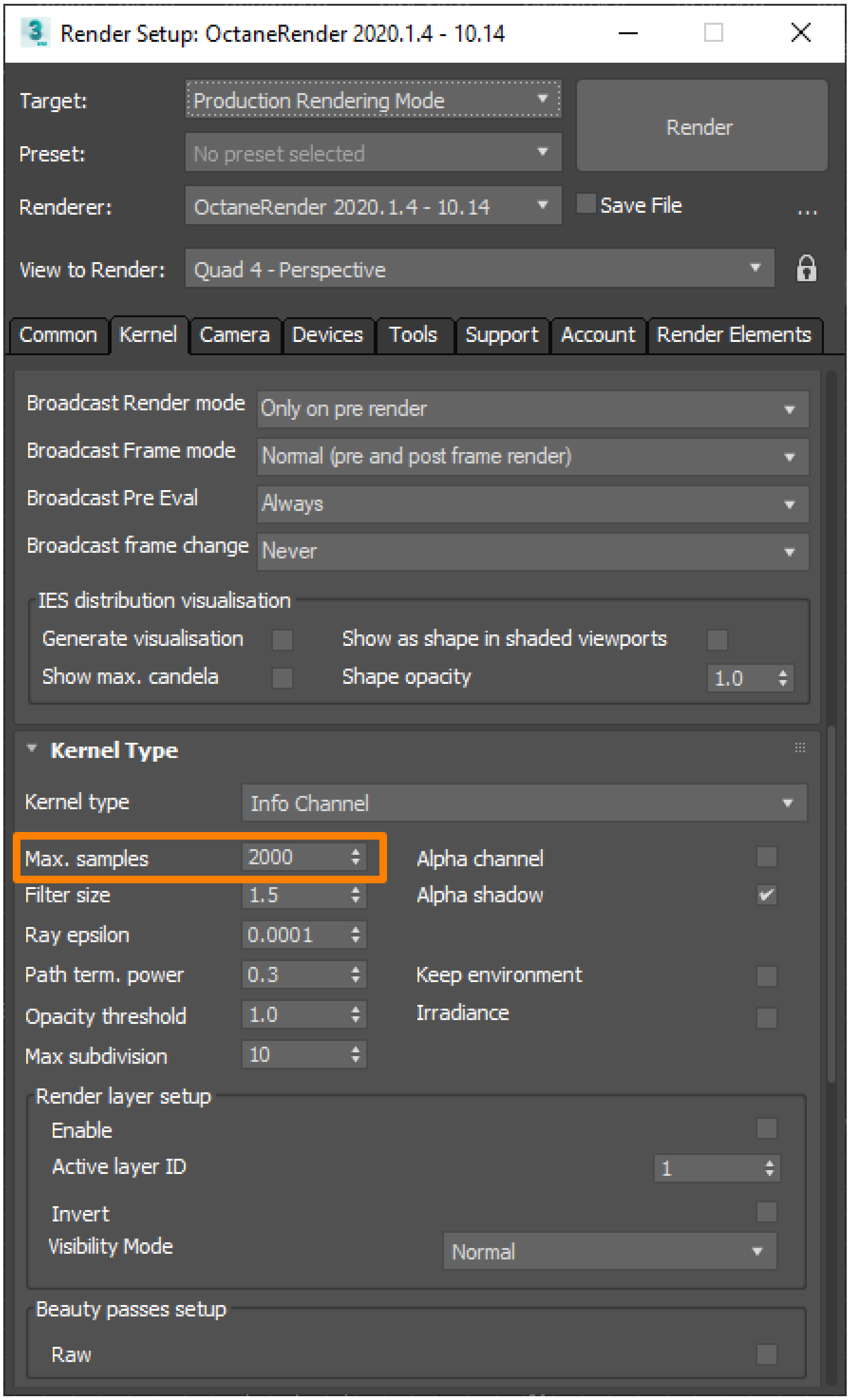

Render Setup: OctaneRender 2020.1.4 - 10.14
Target:
Production Rendering Mode
Render
Preset:
No preset selected
Renderer:
OctaneRender 2020.1.4 - 10.14
Save File
View to Render:
Quad 4 - Perspective
Common
Kernel
Camera
Devices
Tools
Support
Account
Render Elements
Broadcast Render mode
Only on pre render
Broadcast Frame mode
Normal (pre and post frame render)
Broadcast Pre Eval
Always
Broadcast frame change
Never
IES distribution visualisation
Generate visualisation
Show as shape in shaded viewports
Show max. candela
Shape opacity
1.0
Kernel Type
Kernel type
Info Channel
Max. samples
2000
Alpha channel
Filter size
1.5
Alpha shadow
Ray epsilon
0.0001
Path term. power
0.3
Keep environment
Opacity threshold
1.0
Irradiance
Max subdivision
10
Render layer setup
Enable
Active layer ID
1
Invert
Visibility Mode
Normal
Beauty passes setup
Raw

Figure 4: Max. samples

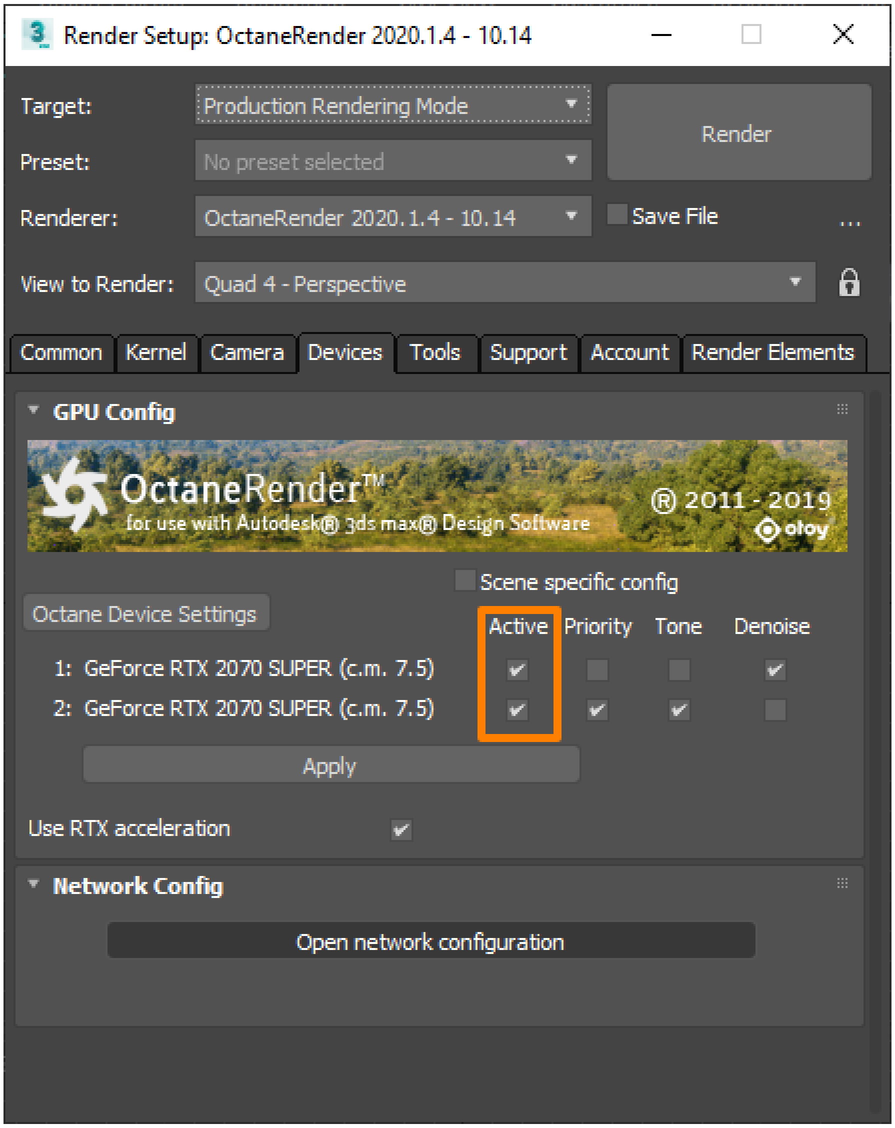


Figure 5: Active Devices

The Octane Menu

The **Octane** menu provides access to many of the key features and tools available in OctaneRender® for 3DS Max® (Figure 1).

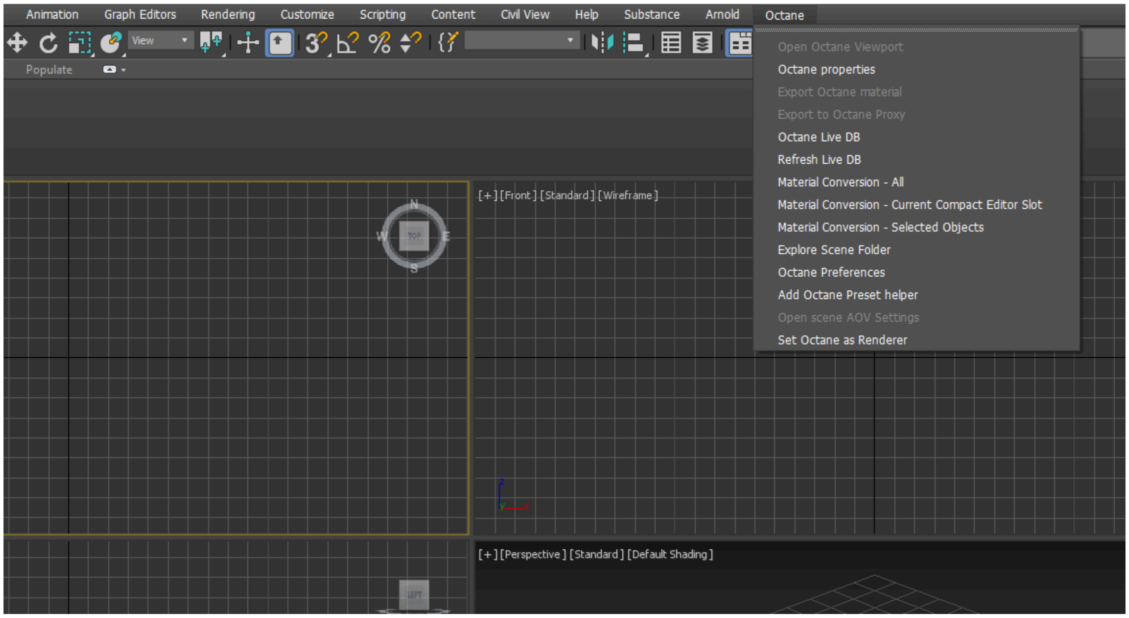


Figure 1: The Octane menu

The Octane menu is only visible in the default workspace. To make the Octane menu visible in a new workspace, go to **Rendering** > **Render Setup**. From the Render Setup window, go to **Tools** tab > **User Global Preferences** section, then click the **Reset Octane Menu** button (Figure 2).

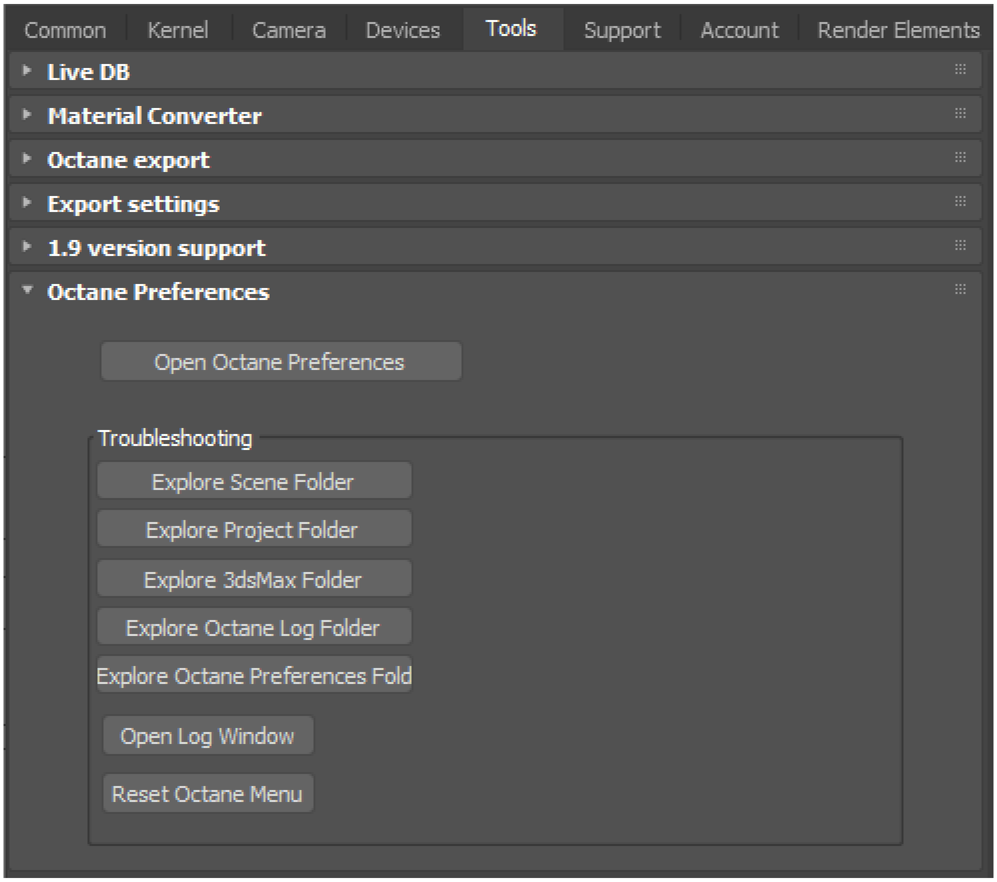


Figure 2: The Reset Octane menu button

Open Octane Viewport

This quick access option opens the **OctaneRender® Viewport**. See **OctaneRender® Viewport Section** for more information.

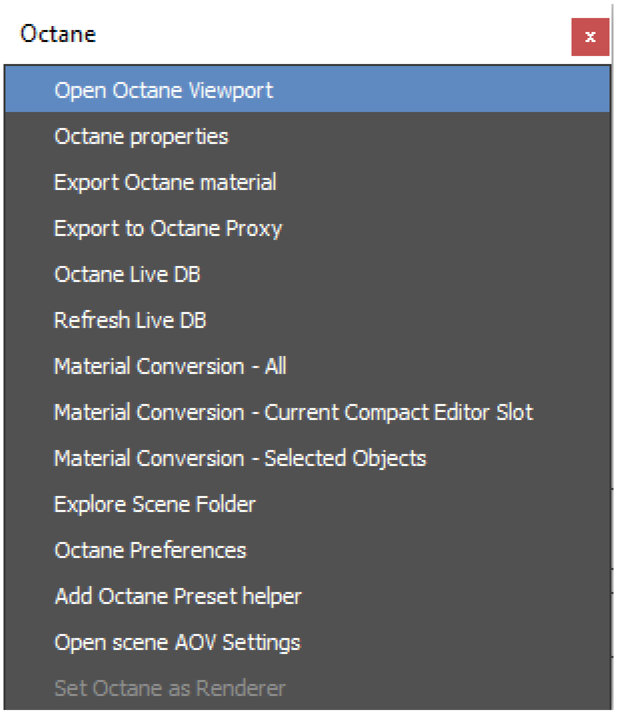


Figure 1: Accessing the OctaneRender® Viewport from the Octane Menu

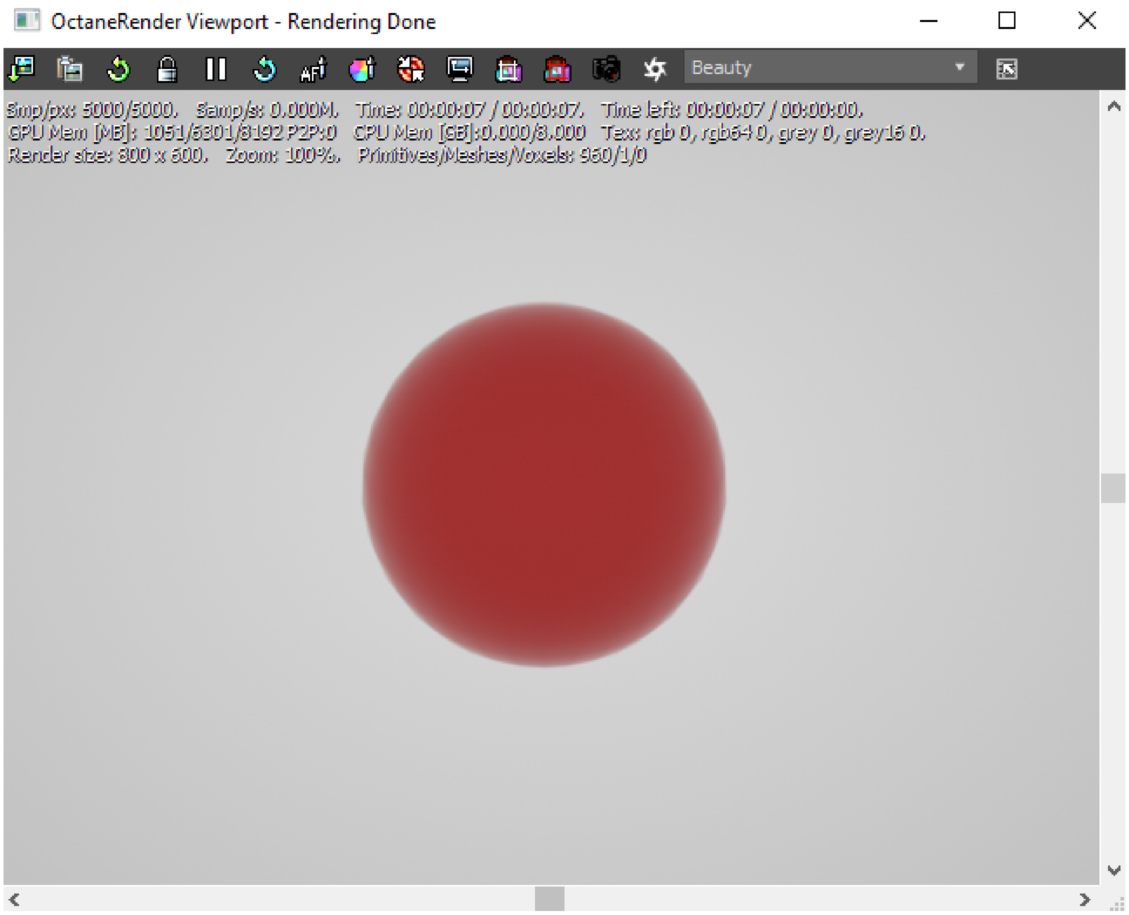


Figure 2: The OctaneRender® Viewport

Octane Properties

This option opens the **Octane Object Properties** window. You can also open the Octane Object Properties window by right-clicking on an object in the scene.

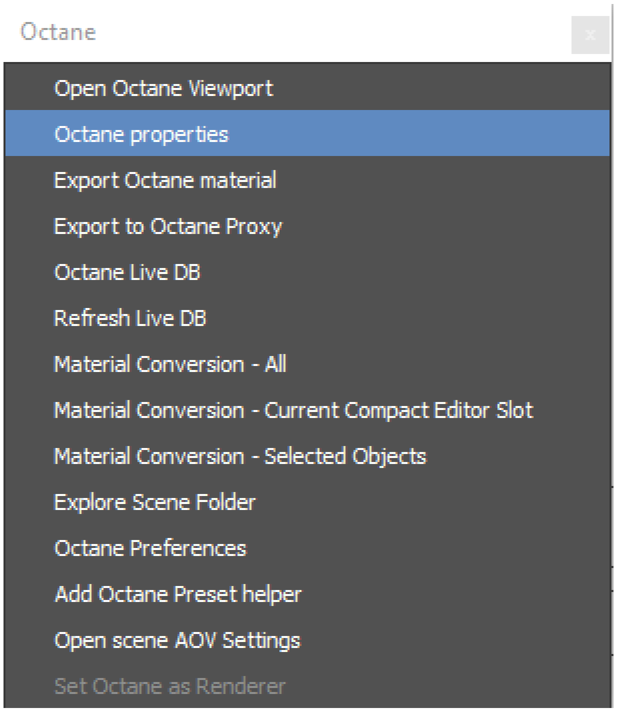


Figure 1: Accessing the Octane Properties window from the Octane menu

The Octane Object Properties window provides a list of all **Objects** in a scene along with a set of parameters that you can edit for each scene Object.

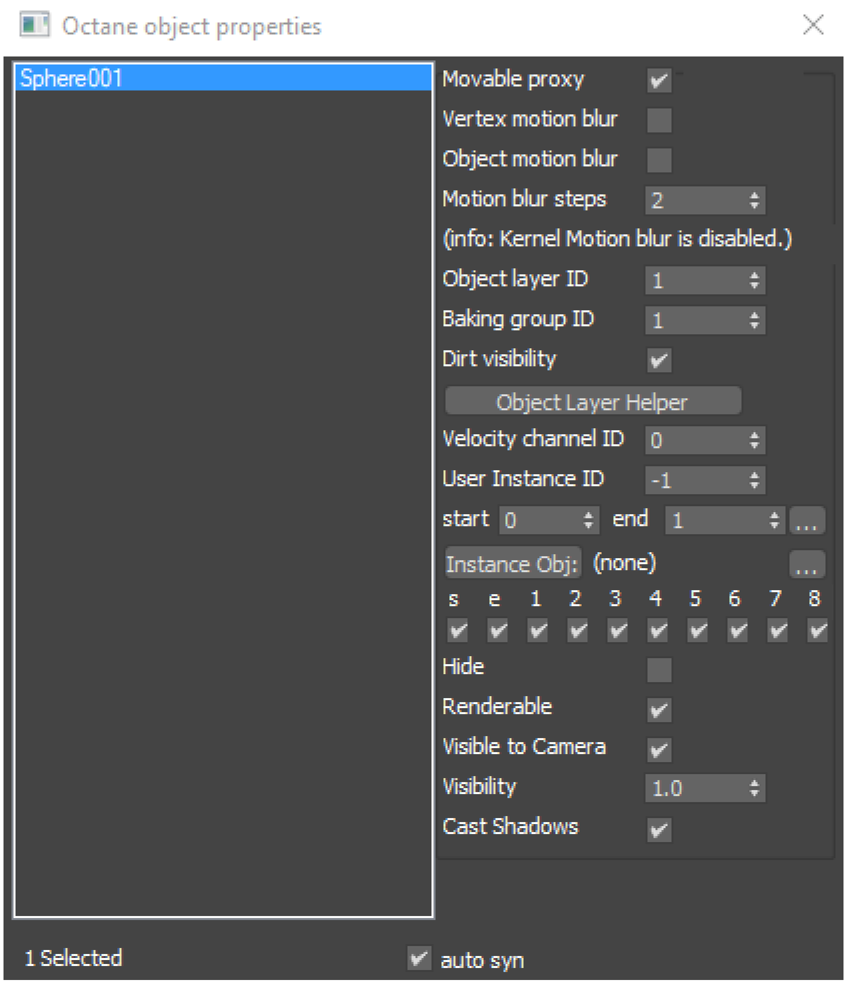


Figure 2: The Octane Object Properties window

Octane Object Parameters

Movable Proxy[1] - Allows real-time updates in the OctaneRender® **Viewport** for OctaneRender® proxies that change in a scene. Unchecked, it will ignore any updates to the object, which can noticeably affect the updating process on large scenes. This is required for instanced objects will can help manage memory more efficiently.

Vertex Motion Blur[2] - Enable this option if you need motion blur and the Object contains vertex-level deformations.

Object Motion Blur - Enable this option if an Object does not have any deformation animation, but the object is moving or rotating. If the Object contains deforming animation, then enable both Object Motion Blur and Vertex Motion Blur for the Object.

Motion Blur Steps - Determines how many steps before and after the current frame OctaneRender® applies motion blur. If you have fast curved objects moving, then increase this value.

Object Layer ID - Works with render layers. for more information, see the **Render Layers**[3] topic in this manual.

Baking Group ID - This ID type works with the texture baking process. For more information, see the **Texture Baking**[4] topic in this manual.

Dirt Visibility - Enabled by default. Set to false to prevent object from being evaluated by **Dirt Texture** node.

Object Layer Helper - Creates and edits an Object Layer Helper in the scene. When you use a **Baking** camera, this helper lets you edit the **Baking** transform matrix.

Velocity Channel ID - The ID for the channel with vertex velocity information.

---

[1]An object saved as a separate file with the purpose of being reused in larger scenes. This is used to minimize any addition to the total polygon count in the scene, especially if the scene requires the same object to appear several times. If used in conjunction with instancing, Proxies help keep very large scenes from reaching polygon limits and also keeps the relative file size of the main project file manageable.
[2]An optical phenomenon that occurs when a camera's shutter opens and closes too slowly to capture movement without recording a blurring of the subject.
[3]Render layers allow users to separate their scene geometry into parts, where one part is meant to be visible and the rest of the other parts "capture" the side effects of the visible geometry. The layers allow different objects to be rendered into separate images where, in turn, some normal render passes may be applied. The Render layers are meant for compositing and not to hide parts of the scene.
[4]A process in which scene lighting is "baked" into a texture map based on an object's UV texture coordinates. The resulting texture can then be mapped back onto the surface to create realistic lighting in a real-time rendering environment. This technique is frequently used in game engines and virtual reality for creating realistic environments with minimal rendering overhead.

User Instance ID - Set the instance IDs for all instances of the same Object. This is done by selecting an **Instance Object** as the first instance, then specifying the **Start** and **End** parameters according to the number of instances present. Clicking on the ellipses provides options as to the order that the IDs are applied to each corresponding instance. This parameter's value is **-1** by default. Used with **Instance Color** and **Instance Range** Nodes.

Instance Obj - This is an instance conversion tool.

1. Select and set an Object to be used as an instance source.
2. Once instance source is set, select object(s) to be converted to the instance source.
3. Clicking on the ellipses provide options to work with the instance.

S, E & 1 - 8 - These parameters specify light passes where **S** is for Sunlight, **E** is for Environment, and **1 - 8** are regular OctaneRender® light ID flags. By default, all of these flags are active. If, for example, there is an OctaneRender® light in the scene, its **Light Pass ID** is set to **2**, and the **2** flag is deactivated in an Object's Octane Object Properties, that light will not illuminate the object.

auto sy.. - Automatically sync your selection to the Max explorer window.

Export Octane Material

This provides options for exporting OctaneRender® **Material**[1] to use in other projects.

---

[1] The representation of the surface or volume properties of an object.

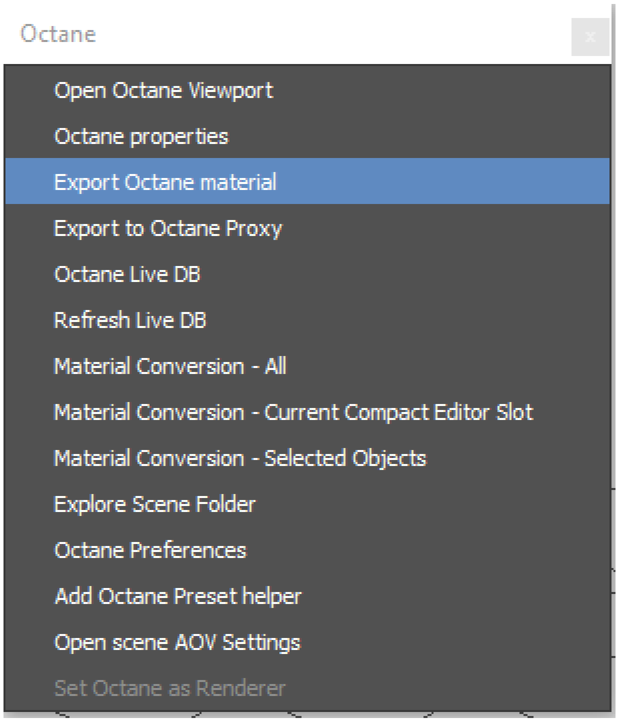


Figure 1: Accessing the Export Octane Material function from the Octane menu

You can store the Material in the Local DB, upload it to the Live DB, or export it as an OctaneRender® package in **ORBX**[1]® format.

[1]The ORBX file format is the best way to transfer scene files from 3D Authoring software programs that use the Octane Plug-in such as Octane for Maya, Octane for Cinema 4D, or OctaneRender Standalone. This format is more efficient than FBX when working with Octane specific data as it provides a flexible, application independent format. ORBX is a container format that includes all animation data, models, textures etc. that is needed to transfer an Octane scene from one application to another.

Figure 2: The Export Octane Material window

Export To Octane Proxy

This item lets you save a scene **Object** as an OctaneRender® proxy (Figure 1). A proxy is an Object saved as a separate file so you can use it in larger scenes. Mainly used to optimize viewport performance with a low resolution proxy object. Detailed information for working with OctaneRender® proxies is in the **Octane Proxy**[1] topic in this manual.

[1] An object saved as a separate file with the purpose of being reused in larger scenes. This is used to minimize any addition to the total polygon count in the scene, especially if the scene requires the same object to appear several times. If used in conjunction with instancing, Proxies help keep very large scenes from reaching polygon limits and also keeps the relative file size of the main project file manageable.

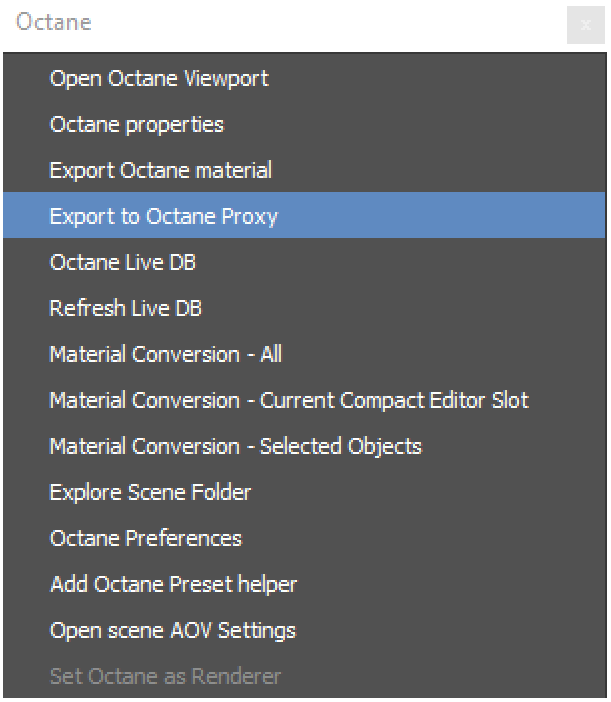


Figure 1: Accessing the Export to Octane Proxy option from the Octane menu

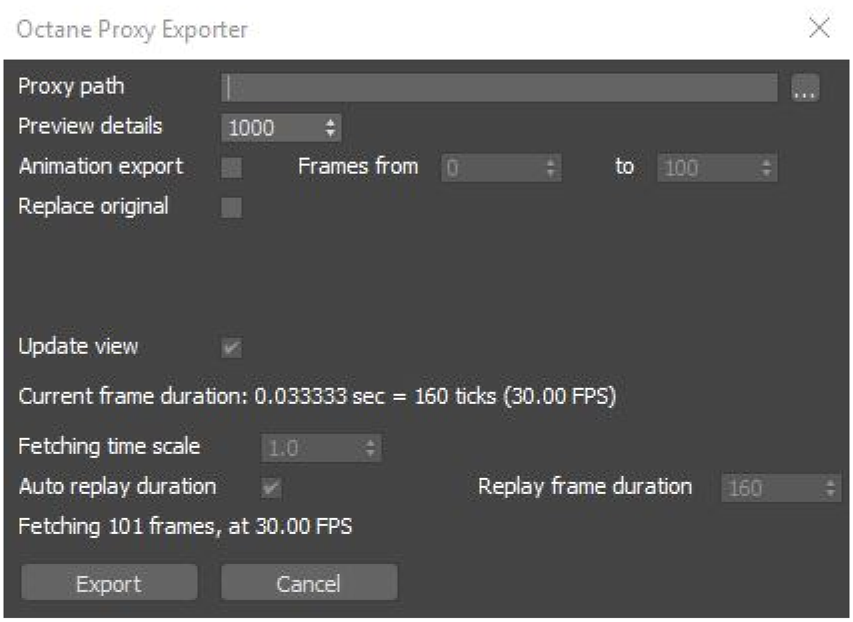


Figure 2: The Octane Proxy Exporter parameters

Octane LiveDB

The 3DS Max® plugin can access the OctaneRender® online LiveDB.

The LiveDB is OctaneRender's asset database. It stores not just **Materials**[1], but groups of **Nodes** and even whole scenes shared by the OctaneRender® community and the OctaneRender® team. The asset database makes it easier for moving groups of Nodes, scenes, and assets across a myriad of OctaneRender® plugins and Standalone edition.

You can access the LiveDB from the **Octane** menu (Figure 1), or by opening the **Render Setup** window, activating the **Tools** tab, and clicking on the **Open LiveDB** button (Figure 2).

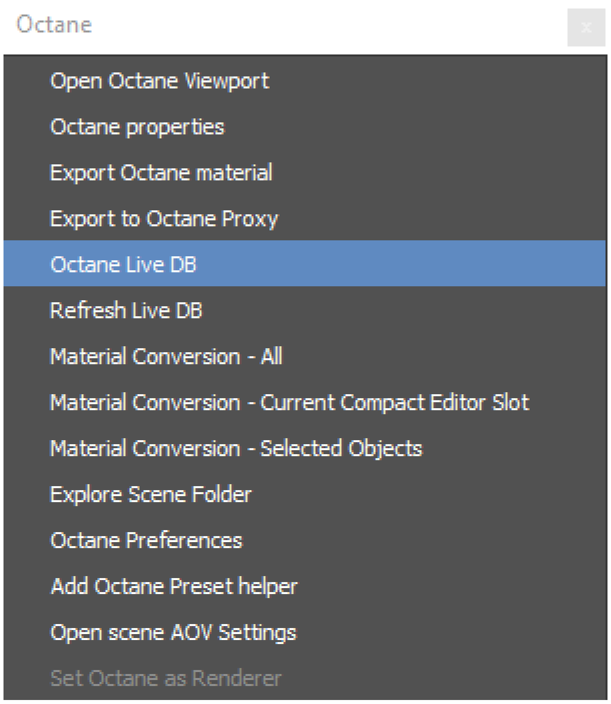


Figure 1: Octane Menu - Octane Live DB

[1] A set of attributes or parameters that describe surface characteristics.

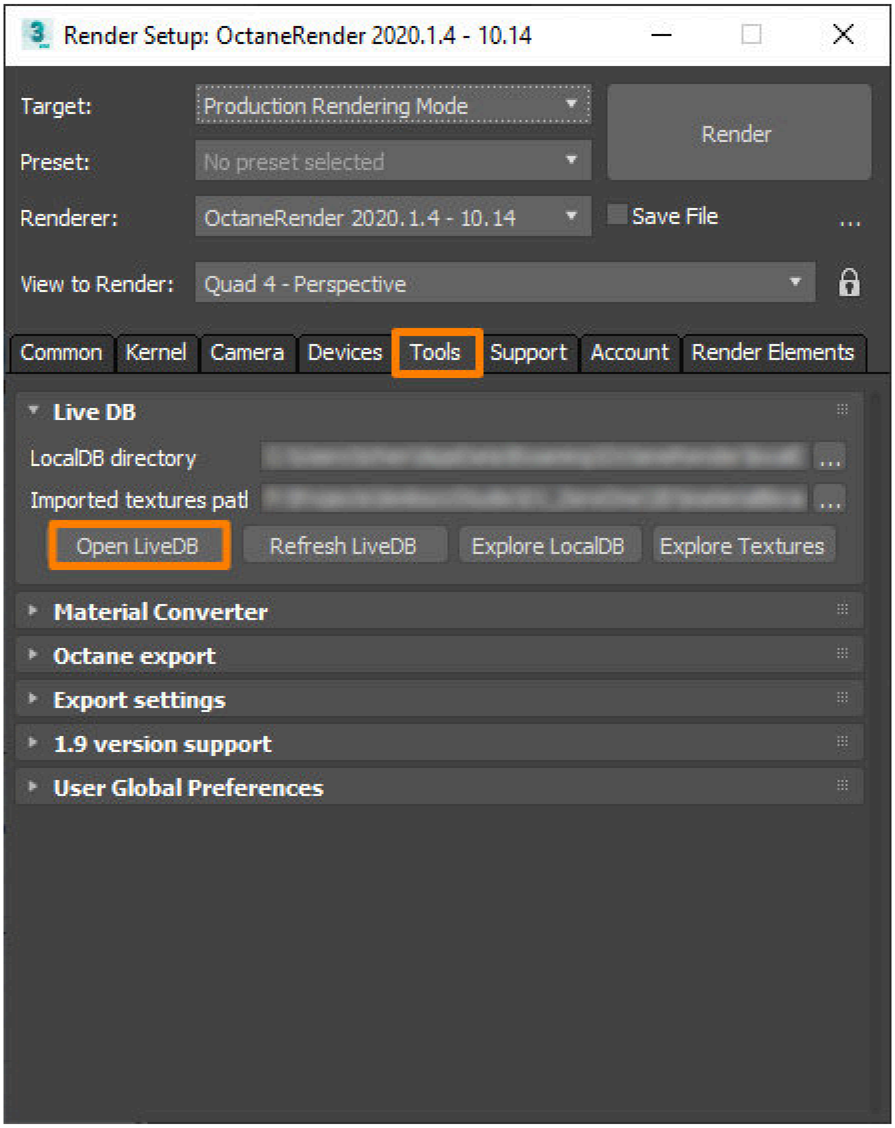


Figure 2: Accessing the LiveDB from the Render Setup window

Many of the Materials contain **Texture** maps. 3DS Max® cannot store these Texture maps within a scene, so it saves the Texture maps to disk under a specified folder and adds the **Material**[1] name to the path. You can customize the path as shown in Figure 3.

[1]The representation of the surface or volume properties of an object.

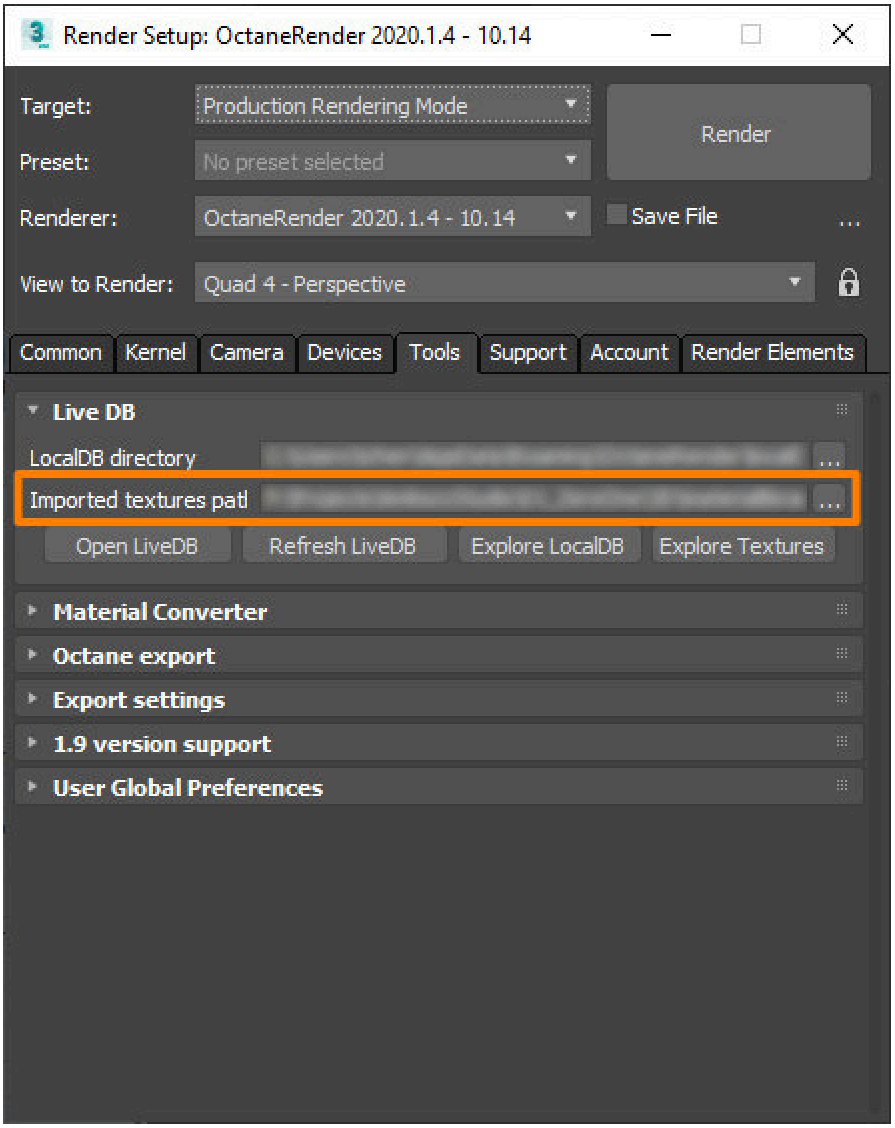


Figure 3: Setting the path where LiveDB stores Texture maps

To use a Material in the LiveDB, right-click on the material thumbnail and choose **Import**. You can access the imported material in the **Sample Slots** of the Material Editor.

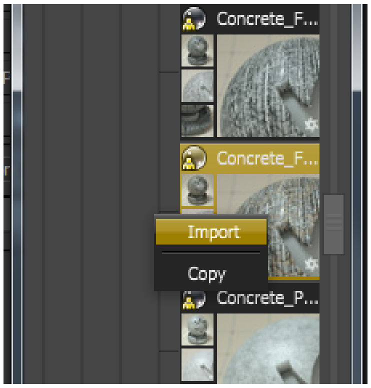


Figure 4: Import option

Refresh LiveDB

This button provides a quick refresh/update for LiveDB.

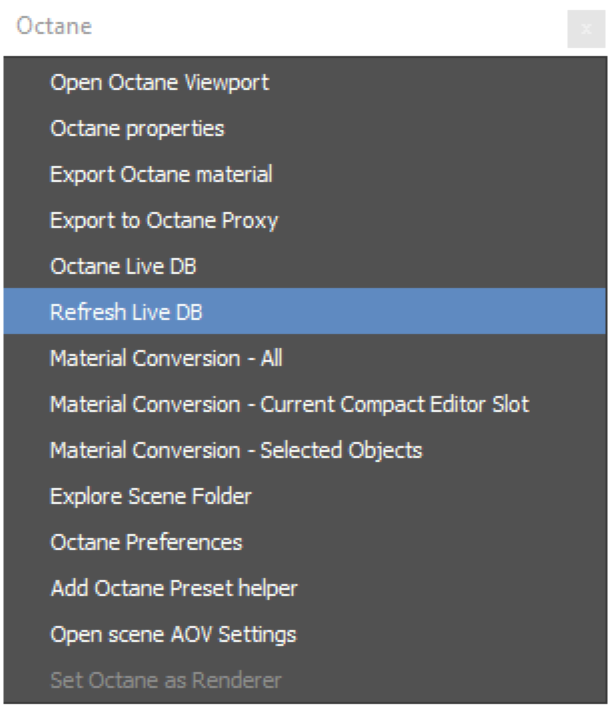


Figure 1: Octane Menu - Refresh Live DB

Material Conversion - All

There may be situations where you need to convert **Materials**[1] from other render engines to OctaneRender® Materials in order for the Materials to render in OctaneRender for 3DS Max®. To do this, click on the **Octane** menu, then click on the**Material**[2] **Conversion - All** menu item (Figure 1). You can also go to the **Render Setup** window, then click on the **Tools** tab, followed by clicking on the **Material Converter** rollout, then clicking the **Convert All** button (Figure 2).

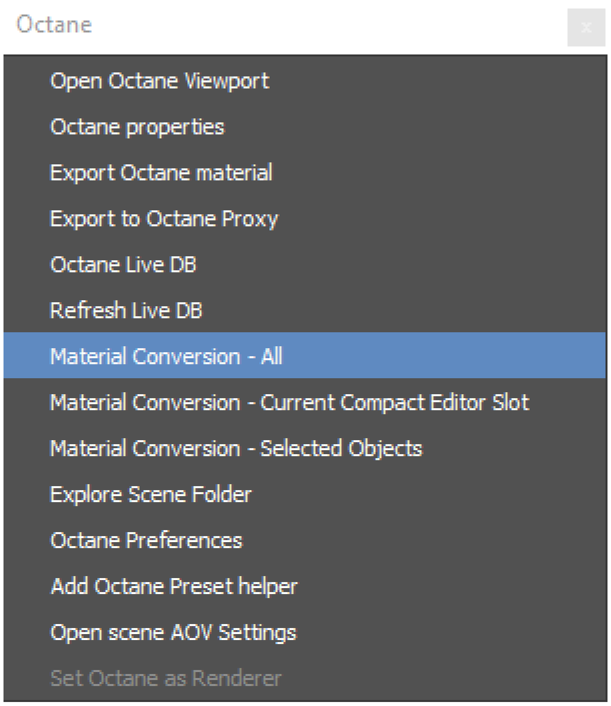


Figure 1: Using the Material Conversion - All to convert other Materials and light sources to OctaneRender®

---

[1] A set of attributes or parameters that describe surface characteristics.
[2] The representation of the surface or volume properties of an object.

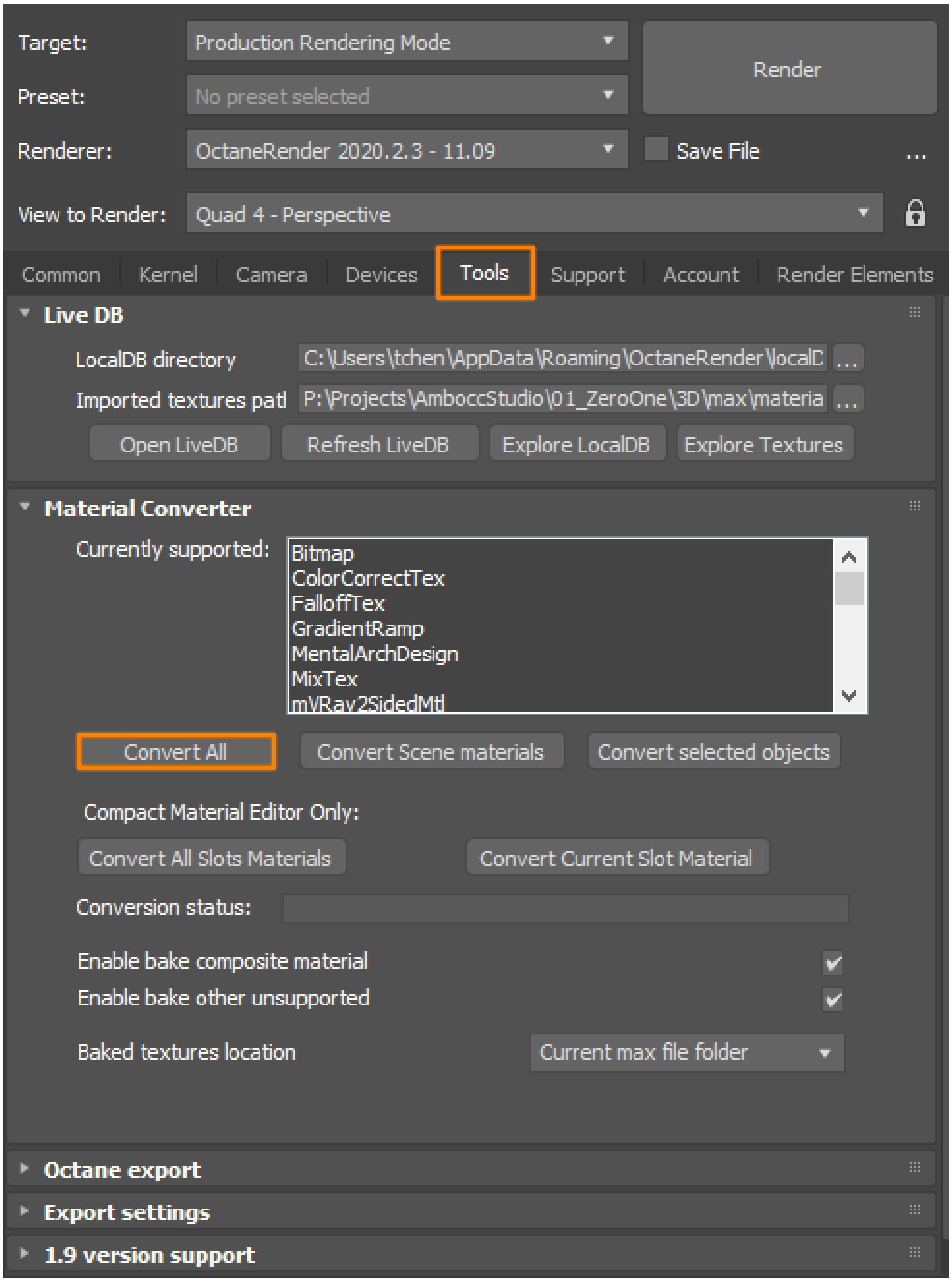


Figure 2: Using the Convert All button in the Render Setup window to convert Materials and lights to OctaneRender® format

Material Converter Parameters

Enable Bake Composite - Bakes 3DS Max® Material composites into a single Texture that is compatible with OctaneRender®.

Enable Bake Other Unsupported - Bakes Material composites designed with other render engines, such as Vray. However, the baked OctaneRender® result may require additional tweaking, depending on the render engine.

Baked Textures[1] Location - Determines the location to save the baked textures.

The following rendering was completed using VRay. The scene uses a VRay Light and a VRay Material with the VRay Dirt texture connected to the Vray Material's **Diffuse**[2] map slot.

Figure 3: A teapot render using VRay

[1] Textures are used to add details to a surface. Textures can be procedural or imported raster files.
[2] Amount of diffusion, or the reflection of light photons at different angles from an uneven or granular surface. Used for dull, non-reflecting materials or mesh emitters.

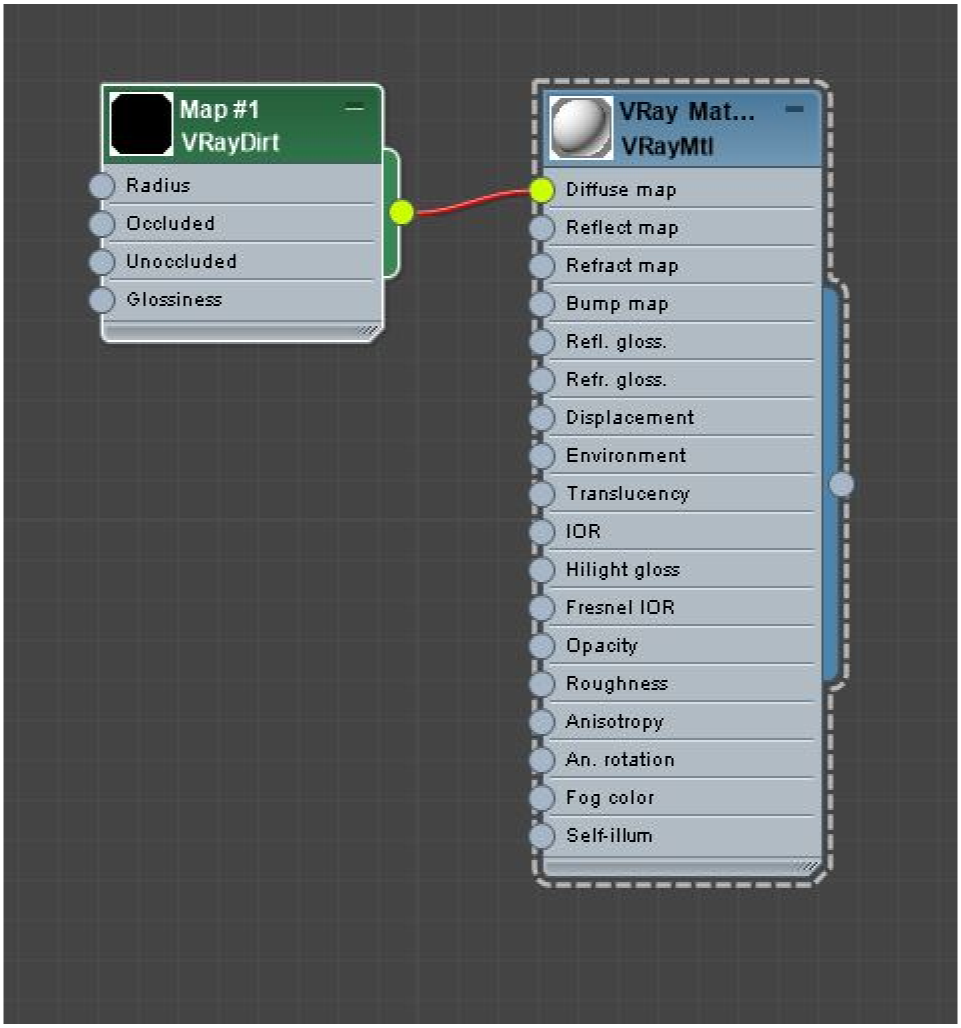


Figure 4: The VRay material network in the Slate Material editor

Once the scene is switched over to the OctaneRender® engine, the VRay material in the Slate Editor should not be affected. After clicking on the **Material Conversion - All** menu item, the VRay Material converts to an OctaneRender® **Diffuse material**[1] with an OctaneRender® Dirt texture replacing the VRay Dirt texture. The VRay light also converts to an OctaneRender® light source. Certain Objects, such as light sources, may need adjustments once the conversion process is complete.

---

[1]Used for dull, non-reflecting materials or mesh emitters.

Material Conversion - Material Editor Current

Similar to **Material**[1] **Conversion - All**, but only convert materials in the Material Editor.

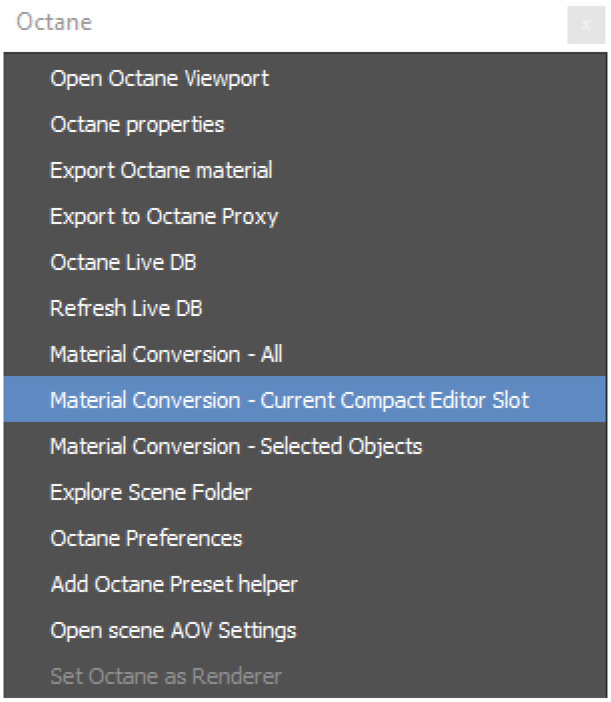


Figure 1: Using the Material Conversion - Material Editor Current

Material Conversion - Selected Objects

Similar to **Material**[2] **Conversion - All**, but only converts selected Objects.

---

[1] The representation of the surface or volume properties of an object.

[2] The representation of the surface or volume properties of an object.

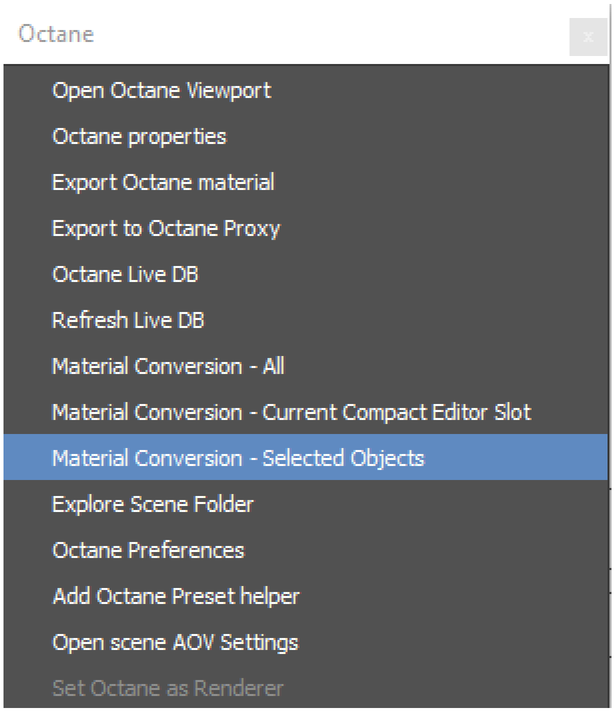


Figure 1: Using the Material Conversion - Selected Objects

Explore Scene Folder

This quickly provides access to the directory of the current scene.

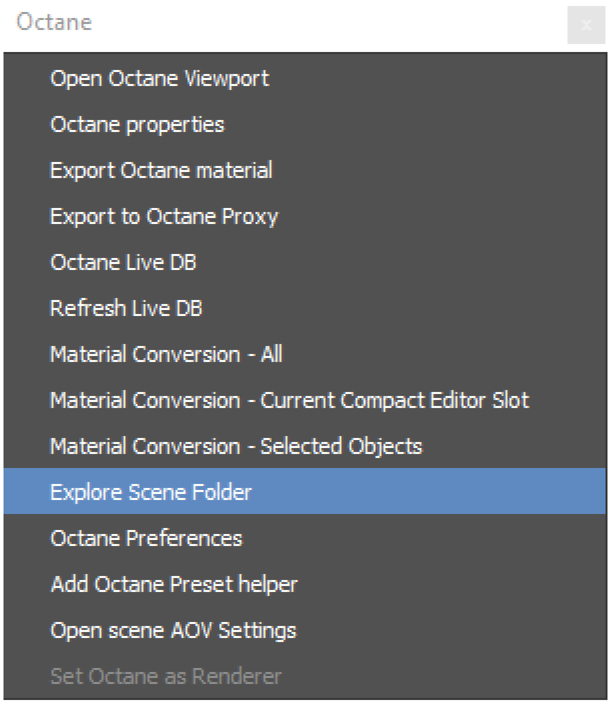


Figure 1: Explore Scene Folder

Octane Preferences Setting

The **Octane Preferences Setting** window contains a large list of parameters and settings related to OctaneRender® within 3DS Max®. The plugin lets you control and streamline how it handles **Object** data generated by third-party applications and other OctaneRender®-supported plugins. These settings include access to tools like OctaneRender's log files and related folders, access to settings used for conversions, and toggle features for troubleshooting. You can access the User Global Settings from the **Octane** menu (Figure 1), or the **Tools** tab in the **Render Setup** window (Figure 2).

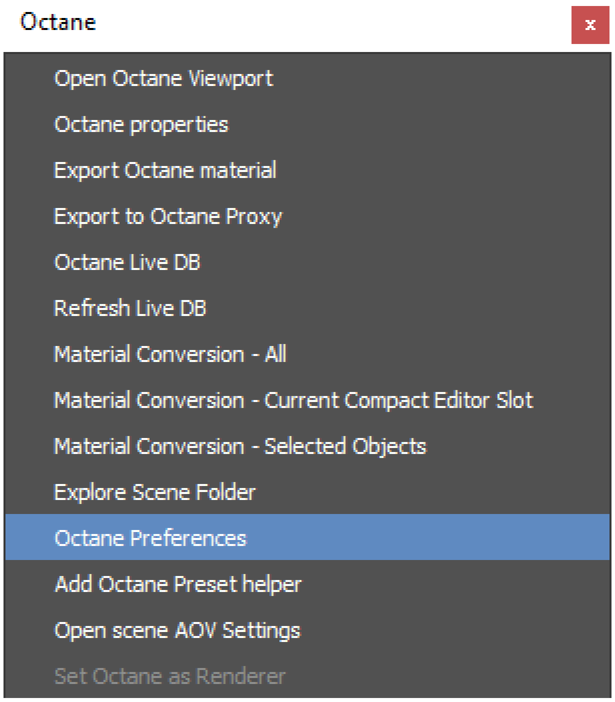


Figure 1: Accessing the User Global Settings from the Octane menu

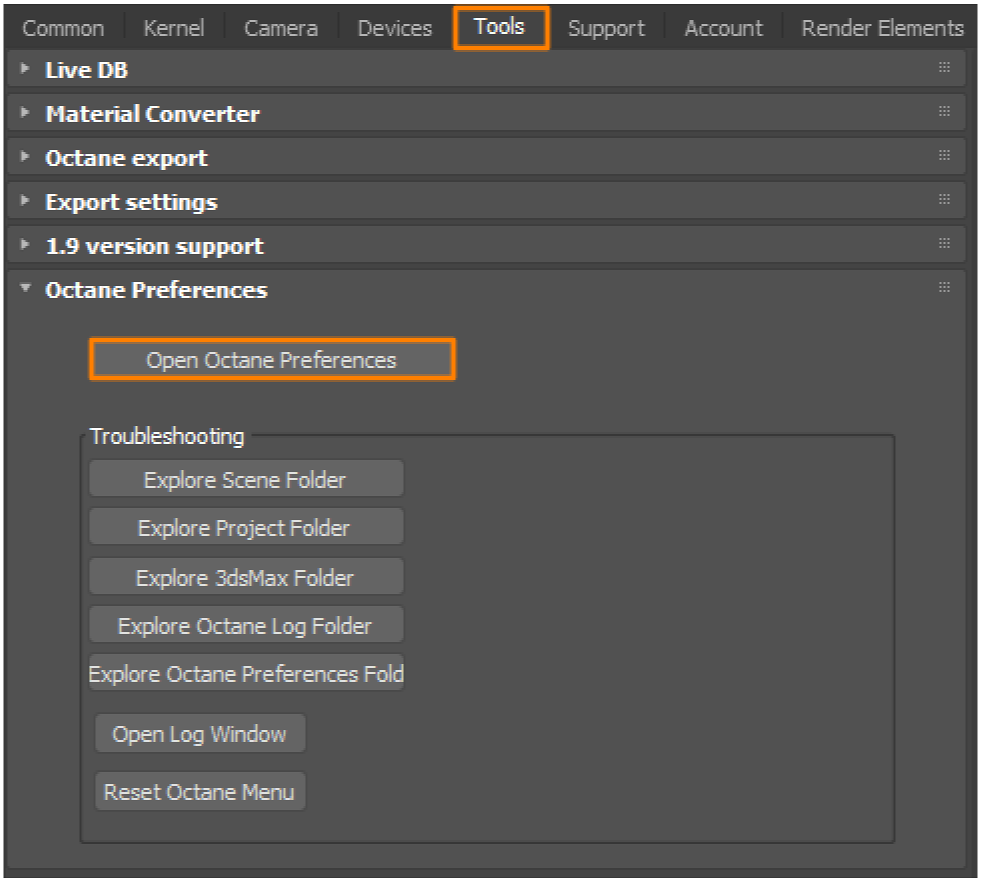


Figure 2: Accessing the Octane Preferences from the Tools tab

Set Octane As Renderer

The **Set Octane As Renderer** option activates the OctaneRender® engine in the **Render Setup** window as the production renderer.

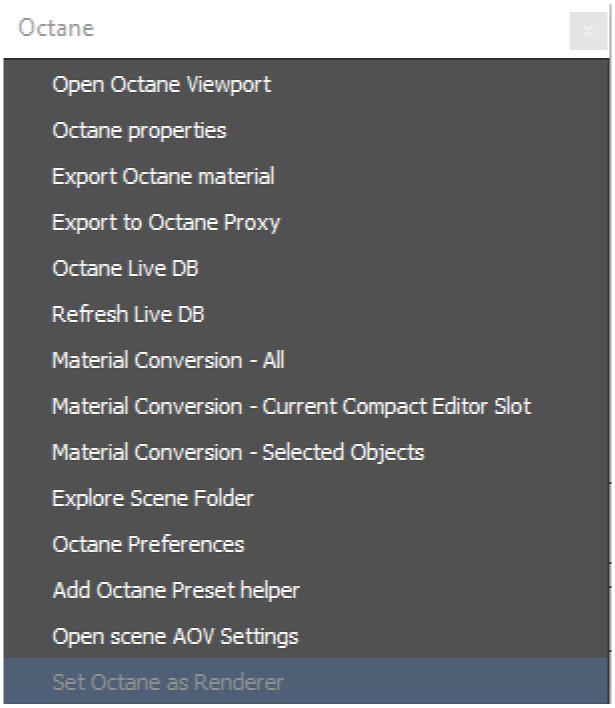


Figure 1: The Set Octane As Renderer option from the Octane menu

OctaneRender® Viewport

The **Viewport** is an updating progressive rendering window. The Viewport updates changes to material and light parameters without reloading the scene onto the **GPU**[1]. If transformations (move, scale, rotate) are made to the scene geometry, including lights, then the Viewport needs to refresh by reloading geometry onto the GPU in order to show these changes. You can access the OctaneRender® Viewport from the **Octane Menu** (Figure 1) or **Render Setup** window (Figure 2).

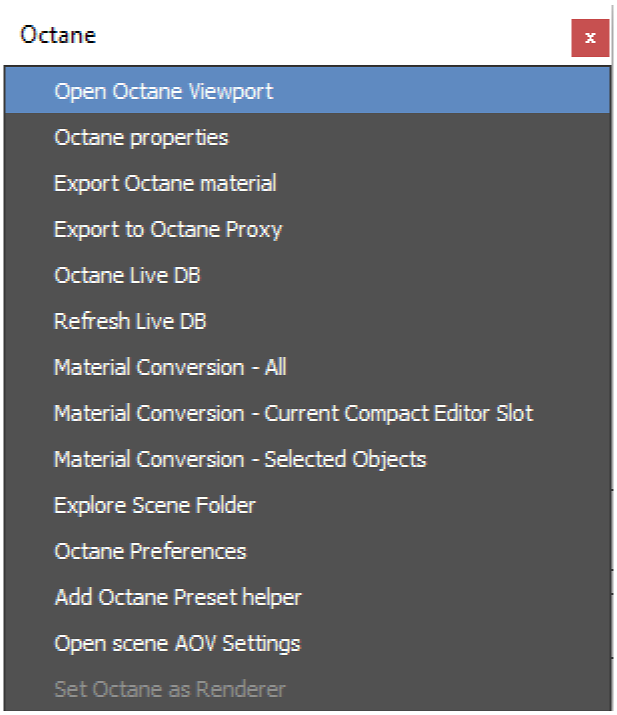


Figure 1: Accessing the OctaneRender® Viewport from the Octane Menu

---

[1]The GPU is responsible for displaying graphical elements on a computer display. The GPU plays a key role in the Octane rendering process as the CUDA cores are utilized during the rendering process.

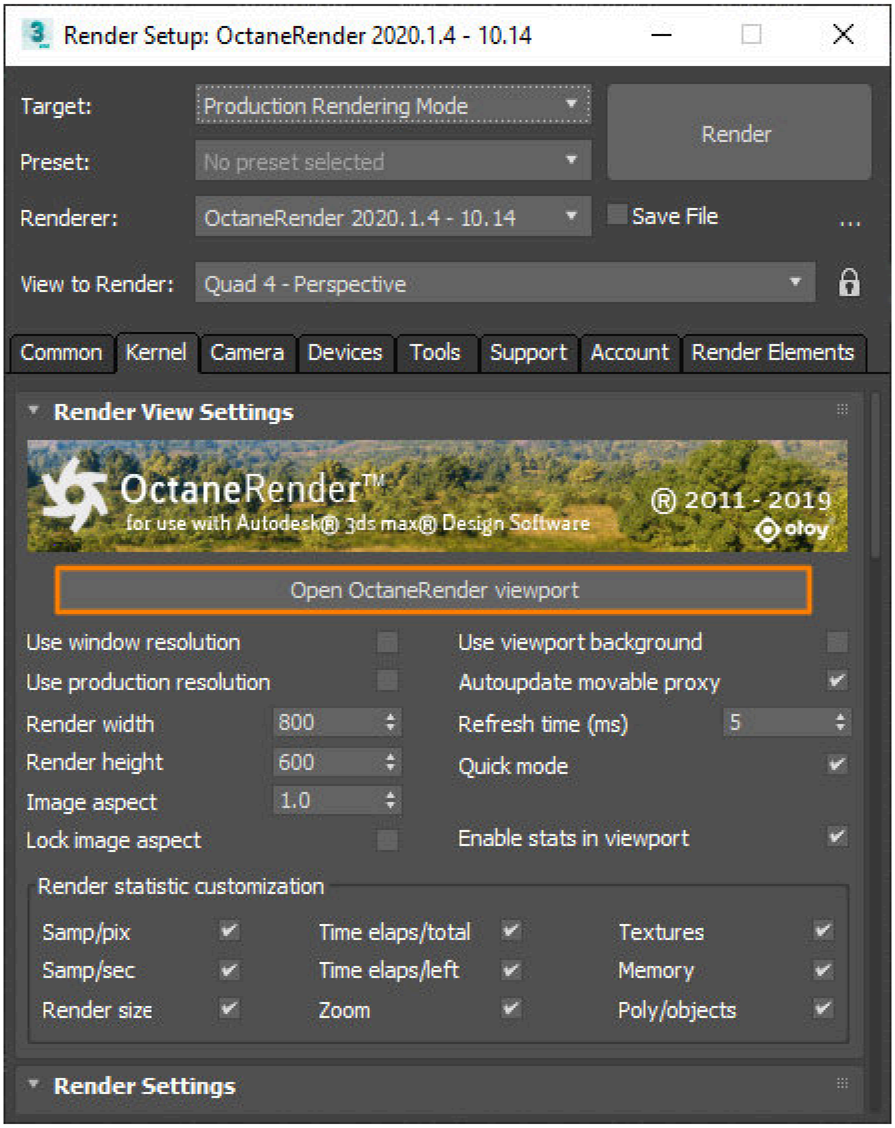


Figure 2: Accessing the OctaneRender® Viewport from Render Setup

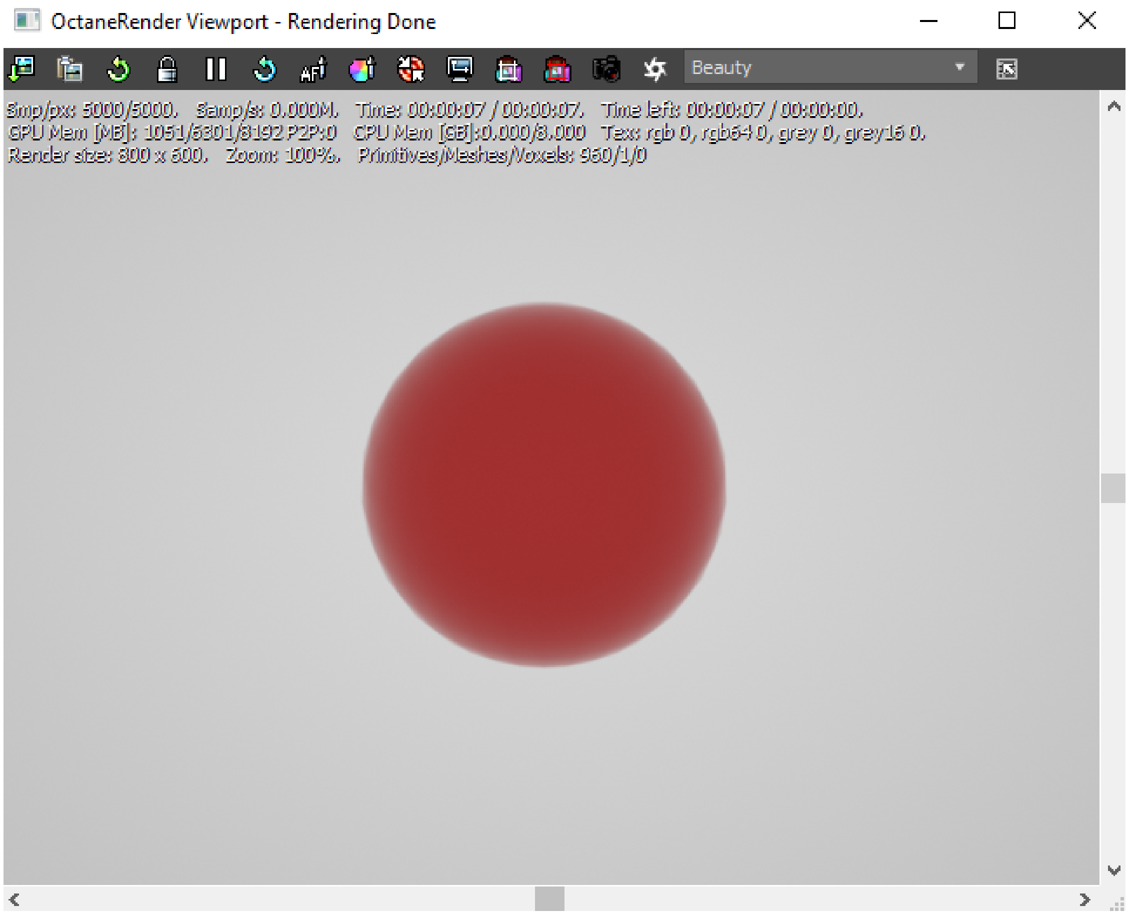


Figure 3: The OctaneRender® Viewport Window

OctaneRender® Viewport Buttons

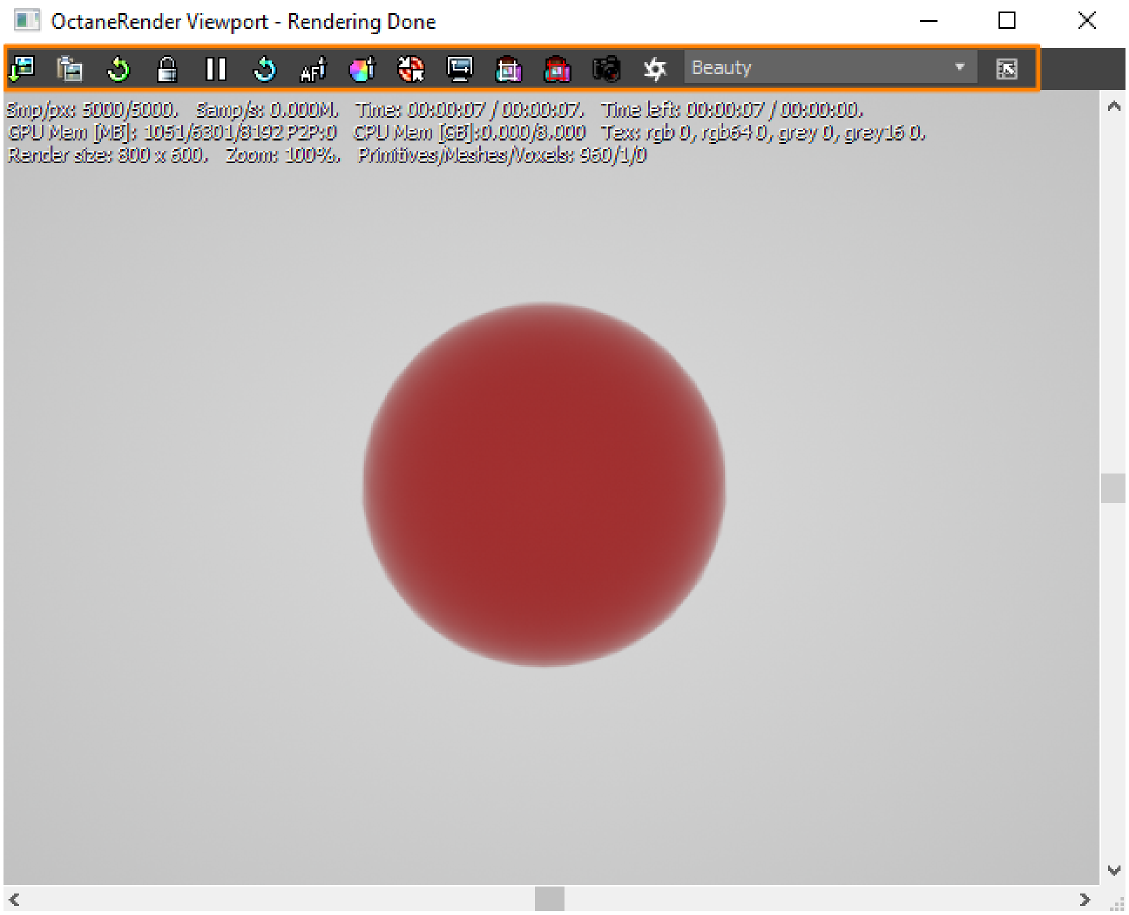
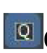


Figure 1: Viewport Buttons

**Save Image** - Saves the current render from the OctaneRender Viewport to disk.

**Port To Max Framebuffer** - Sends the current render to the 3DS Max® frame buffer (render window).

**Quick Update Mode** - When disabled, OctaneRender® tests all objects every 5 ms for changes. This can get very slow when there are many Objects. When you enable this mode, changes are detected automatically using 3DS Max® messages, making changes much more responsive.

Update Render Geometry - Reloads the scene geometry onto the **GPU**[1]. Click this button if you add or change scene elements while the OctaneRender® Viewport is open. Otherwise, scene geometry reloads onto the GPU whenever you close and reopen the Viewport.

Lock Update - Locks the Viewport so it does not update the render even if you edit scene elements, such as moving the camera or adjusting materials and lights. However, the Viewport continues to render the scene.

Pause Render - Similar to the Lock Update button except it also pauses the scene's progressive rendering.

Rebuild Scene - Rebuilds all the scene elements, including lighting and materials.

Focus Picker - Lets you sample an area of the Viewport as the render's focus point. Using this feature will automatically disable the camera's **Auto-focus** and set **Focus depth**to the proper value.

White Balance Picker - Samples an area in the Viewport to determine the render's white balance.

Align Image - Re-centers the rendering if you zoom in or reposition the render.

Lock Resolution - Locks the rendering resolution to the size and proportions of the render window.

Region Render - Limits the render window to render just the area included in the region selection box.

Film Region - Lets you move the red box corners to adjust the render boundaries. The renderer displays black outside of the rectangle.

Lock Camera - Locks the camera for the 3DS Max® Viewport that you want to render. If you disable this function, 3DS Max® restarts the render each time you select a different 3DS Max® Viewport.

Octane Node Viewer - Opens the OctaneRender® **Nodegraph Editor**. Primarily used to view and troubleshoot the node network translated to Octane. Changes cannot be saved.

Beauty **Passes Dropbox** - Lets you select and display any render passes enabled by **Select Multiple Passes**.

Select Multiple Passes - Lets you add Octane supported render passes. You can display the passes by using **Passes Dropbox**.

---

[1]The GPU is responsible for displaying graphical elements on a computer display. The GPU plays a key role in the Octane rendering process as the CUDA cores are utilized during the rendering process.

OctaneRender® Viewport Info

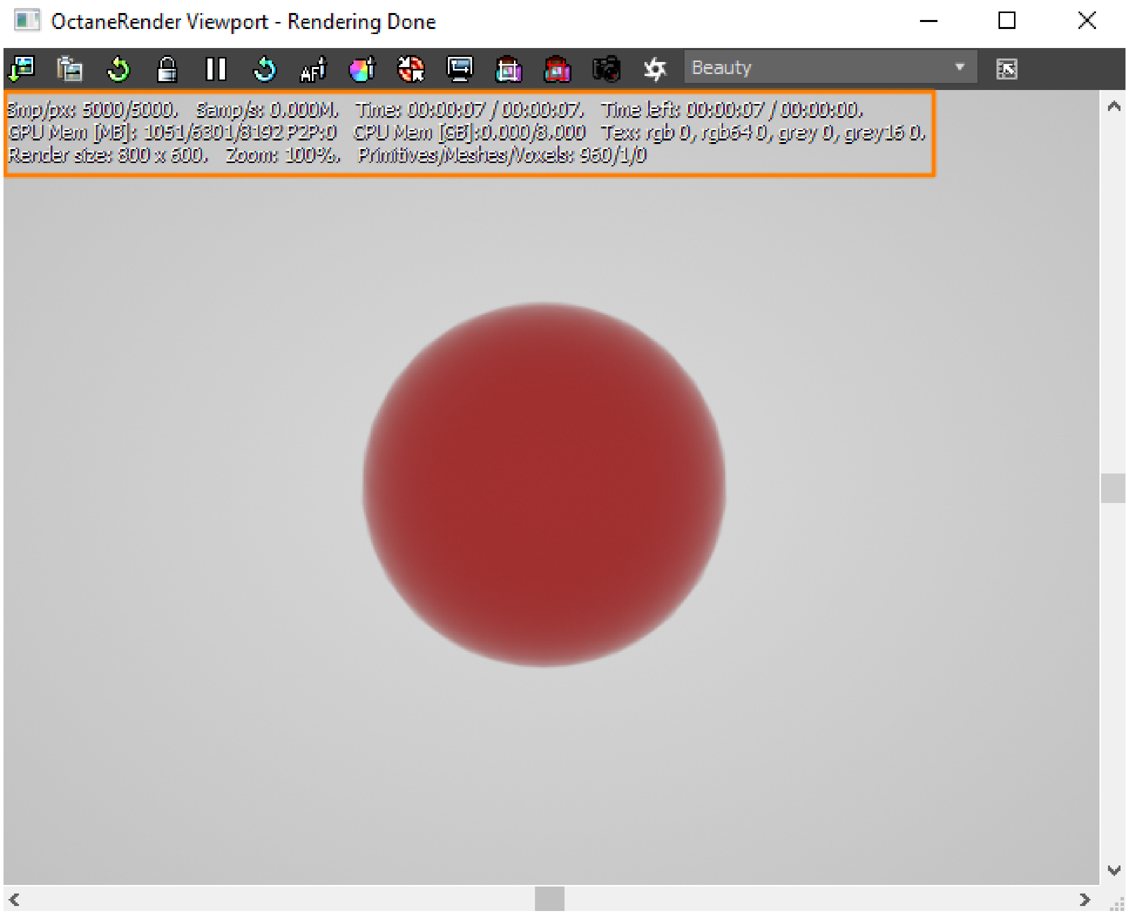


Figure 1: Render Scene Information

Information regarding the render includes:

Smp/px

- The number of tone mapped samples already visible
- The maximum samples to be rendered.

Samp/s

- Render speed

Time

- Time elapsed
- Estimated time for completion

Time Left

- Time elapsed
- Estimated time left for completion

GPU[1] Mem [MB]

- Shows the GPU memory used by the current process
- Total usable device memory, excluding memory used by other application running concurrently.
- Total Memory size available. If running multiple GPUs with different memory sizes, the system will limit the total memory access to the GPU with the least amount of VRAM.

PSP (Work in Progress)

- Display the amount of GPU **Peer to Peer** memory pool used.

CPU Mem [GB]

- Shows the Out of Core memory used.
- The maximum memory allocated to Out of Core. You can set this amount in **Render Setup.**

Tex

- rgb - Shows the number of **LDR**[2] color textures
- rgb64 - Shows the number of HDR color textures
- grey - Shows the number of LDR greyscale textures
- grey16 - Shows the number of HDR greyscale textures

Render size

- Current render resolution

Zoom

- Display the image zoom amount. Does not affect render or output resolution.

Primitives/Meshes/Voxels

- Shows the number of triangles.
- Shows the number of mesh objects.
- Shows the number of voxels.

---

[1] The GPU is responsible for displaying graphical elements on a computer display. The GPU plays a key role in the Octane rendering process as the CUDA cores are utilized during the rendering process.
[2] Image formats that have 8 bits per color channel such as the common image formats JPEG, PNG, GIF among others.

Material Editor

All **Materials**[1] and **Textures**[2] will be covered using **3DS Max® Slate Material[3] Editor**.

[1]A set of attributes or parameters that describe surface characteristics.
[2]Textures are used to add details to a surface. Textures can be procedural or imported raster files.
[3]The representation of the surface or volume properties of an object.

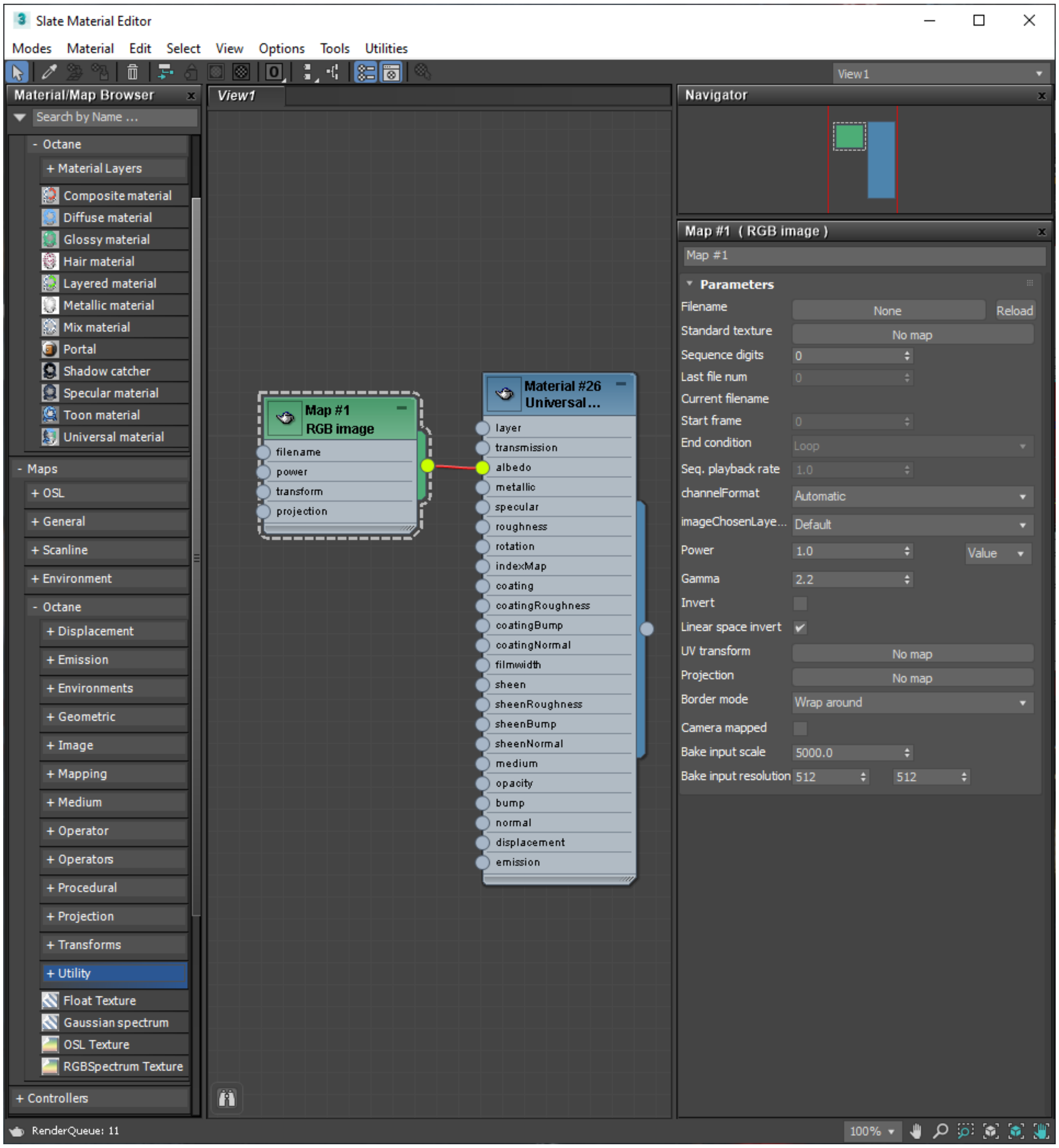


Figure 1: Slate Material Editor

Materials

There are nine types of **Materials**[1] available in OctaneRender®:

- **Diffuse**[2] - Used for dull, non-reflecting Materials or **Mesh** emitters.
- **Composite** - Mixes several Materials using masks.
- **Glossy**[3] - Used for shiny Materials such as plastics or metals.
- **Specular**[4] - Used for transparent Materials such as glass and water.
- **Metallic** - Similar to the **Glossy material**[5], except by default it exhibits more metal-like characteristics.
- **Mix** - Mixes any two **Material**[6] types.
- **Portal**[7] - Designates openings in scenes to allow the render kernel to better sample light from those areas. For more information, see the Lighting topic in this manual.
- **Toon** - Designs surfaces that look hand-drawn.
- **Universal** - Integrates more closely with **PBR**[8] workflows.

The Universal material with the PBR workflow is fast becoming the standard and can replicated a wide variety of materials.

---

[1] A set of attributes or parameters that describe surface characteristics.

[2] Amount of diffusion, or the reflection of light photons at different angles from an uneven or granular surface. Used for dull, non-reflecting materials or mesh emitters.

[3] The measure of how well light is reflected from a surface in the specular direction, the amount and way in which the light is spread around the specular direction, and the change in specular reflection as the specular angle changes. Used for shiny materials such as plastics or metals.

[4] Amount of specular reflection, or the mirror-like reflection of light photons at the same angle. Used for transparent materials such as glass and water.

[5] Used for shiny materials such as plastics or metals.

[6] The representation of the surface or volume properties of an object.

[7] A technique that assists the render kernel with exterior light sources that illuminate interiors. In interior renderings with windows, it is difficult for the path tracer to find light from the outside environment and optimally render the scene. Portals are planes that are added to the scene with the Portal material applied to them.

[8] A contemporary shading and rendering process that seeks to simplify shading characteristics while providing a more accurate representation of lighting in the real world.

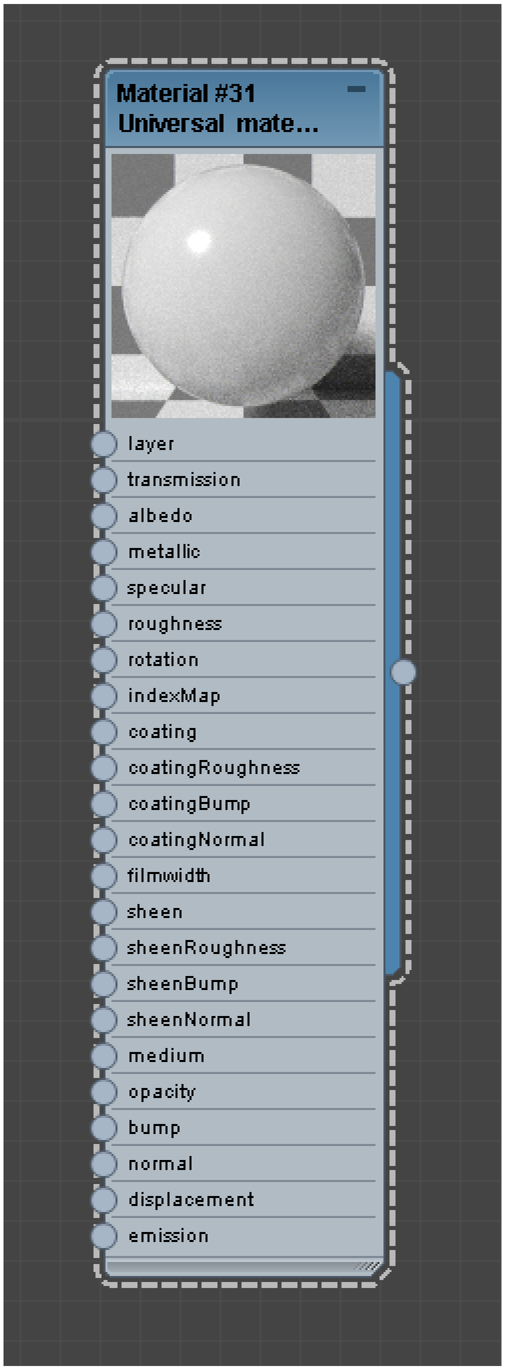


Figure 1: The four common OctaneRender® materials are Diffuse, Glossy, Specular, and Metallic

You can access the OctaneRender® Materials by expanding the **OctaneRender** rollout in the **Materials** section of the **Slate Material Editor**.

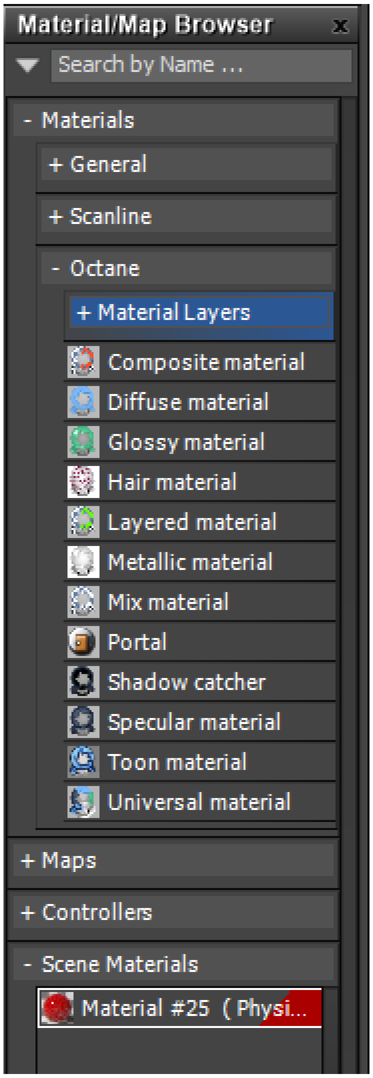


Figure 2: Editing Materials in the Slate Material Editor

Composite Material

The **Composite** material node mixes several materials using masks (Figure 1). This is much cleaner than using several chained **Mix** materials. If a mask is not connected, the material's **Opacity** is used. The first **Material**[1] pin becomes the base layer.

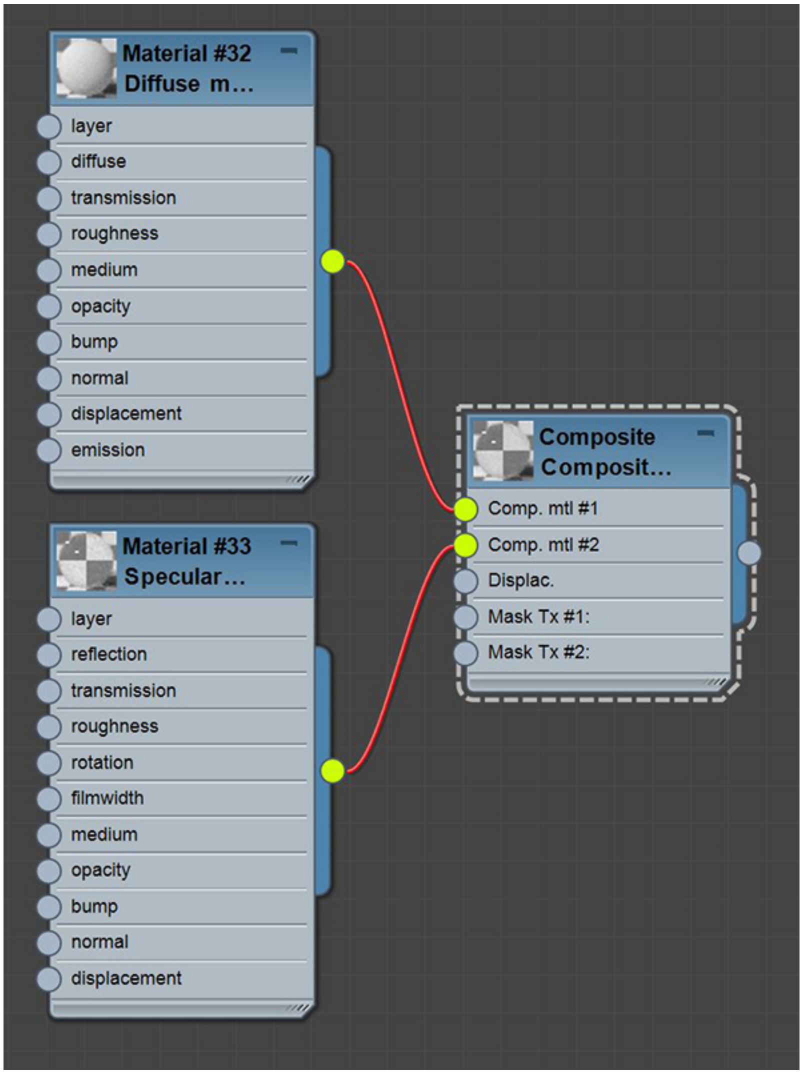


Figure 1: Diffuse[2] and Specular[3] materials connected to a Composite material node

---

[1]The representation of the surface or volume properties of an object.

[2]Amount of diffusion, or the reflection of light photons at different angles from an uneven or granular surface. Used for dull, non-reflecting materials or mesh emitters.

[3]Amount of specular reflection, or the mirror-like reflection of light photons at the same angle. Used for transparent materials such as glass and water.

Composite Material Parameters

Displacement[1] - Displacement for the Composite Material surface.

Add Input - Adds a new Material input to the end of the Node.

Remove Input - Removes the last Material input.

Comp Material - The Material input. When several **Materials**[2] are used, the first Material pin becomes the base layer.

Mask Tx - Controls the Material’s opacity using an Input map. If a mask is not connected, then OctaneRender® uses the Material's opacity.

Export Octane Material - Opens a dialog window that provides options for exporting the Material to **ORBX**[3], the Local DB, or the Live DB.

Diffuse Layer

The layered material system lets you construct complex materials that consist of a base layer and up to eight **Material**[4] Layers. The layers are based on components used in previous Octane materials. Using this set of unique layers, OctaneRender® now lets you recreate complex materials in a physically-based manner, as opposed to manually **mixing materials**[5] together.

---

[1] The process of utilizing a 2D texture map to generate 3D surface relief. As opposed to bump and normal mapping, Displacement mapping does not only provide the illusion of depth but it effectively displaces the actual geometric position of points over the textured surface.

[2] A set of attributes or parameters that describe surface characteristics.

[3] The ORBX file format is the best way to transfer scene files from 3D Authoring software programs that use the Octane Plug-in such as Octane for Maya, Octane for Cinema 4D, or OctaneRender Standalone. This format is more efficient than FBX when working with Octane specific data as it provides a flexible, application independent format. ORBX is a container format that includes all animation data, models, textures etc. that is needed to transfer an Octane scene from one application to another.

[4] The representation of the surface or volume properties of an object.

[5] Used to mix any two material types.

The **Diffuse**[1] layer is used for dull, non-reflective materials. See the **Diffuse Material** topic in this manual for more information. Material Layers can connect to the **Layered Material**, **Layer Group**, or **Material Layer** pins on standard materials (Figure 1).

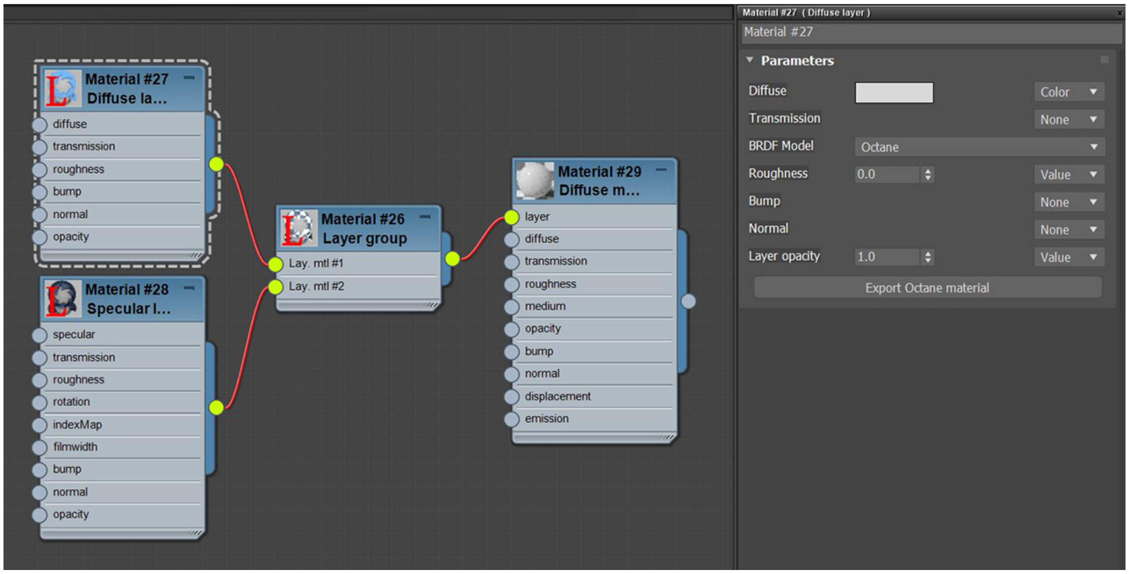


Figure 1: Diffuse and **Specular**[2] layers are mixed using the Layer Group node

Diffuse Layer Parameters

Diffuse - The layer's diffuse color.

Transmission[3] - The layer's transmission color. This can be used for thin translucent objects.

BRDF Model - Determines how light reflects or refracts. See the BRDF Models topic in this manual for more information.

Roughness - The Diffuse layer's roughness. High values simulate very rough surfaces like sandpaper or clay.

Bump - Simulates a relief by using a **Greyscale** texture interpreted as a **Height** map for the layer.

Normal - Distorts layer normals using an **RGB** image.

---

[1] Amount of diffusion, or the reflection of light photons at different angles from an uneven or granular surface. Used for dull, non-reflecting materials or mesh emitters.
[2] Amount of specular reflection, or the mirror-like reflection of light photons at the same angle. Used for transparent materials such as glass and water.
[3] A surface characteristic that determines if light may pass through a surface volume.

Layer Opacity - Controls the layer opacity with a Greyscale texture.

Export Octane Material - Opens a dialog window that provides options for exporting the Material to **ORBX**[1], the Local DB, or the Live DB.

Diffuse Material

The **Diffuse**[2] material is used for dull, non-reflecting materials or light-emitting surfaces. **Diffuse material**[3] simulates a rough surface that reflects light back into the environment in all directions. **Specular**[4] highlights and reflections do not appear on diffuse surfaces.

Figure 1: The Diffuse material

---

[1]The ORBX file format is the best way to transfer scene files from 3D Authoring software programs that use the Octane Plug-in such as Octane for Maya, Octane for Cinema 4D, or OctaneRender Standalone. This format is more efficient than FBX when working with Octane specific data as it provides a flexible, application independent format. ORBX is a container format that includes all animation data, models, textures etc. that is needed to transfer an Octane scene from one application to another.
[2]Amount of diffusion, or the reflection of light photons at different angles from an uneven or granular surface. Used for dull, non-reflecting materials or mesh emitters.
[3]Used for dull, non-reflecting materials or mesh emitters.
[4]Amount of specular reflection, or the mirror-like reflection of light photons at the same angle. Used for transparent materials such as glass and water.

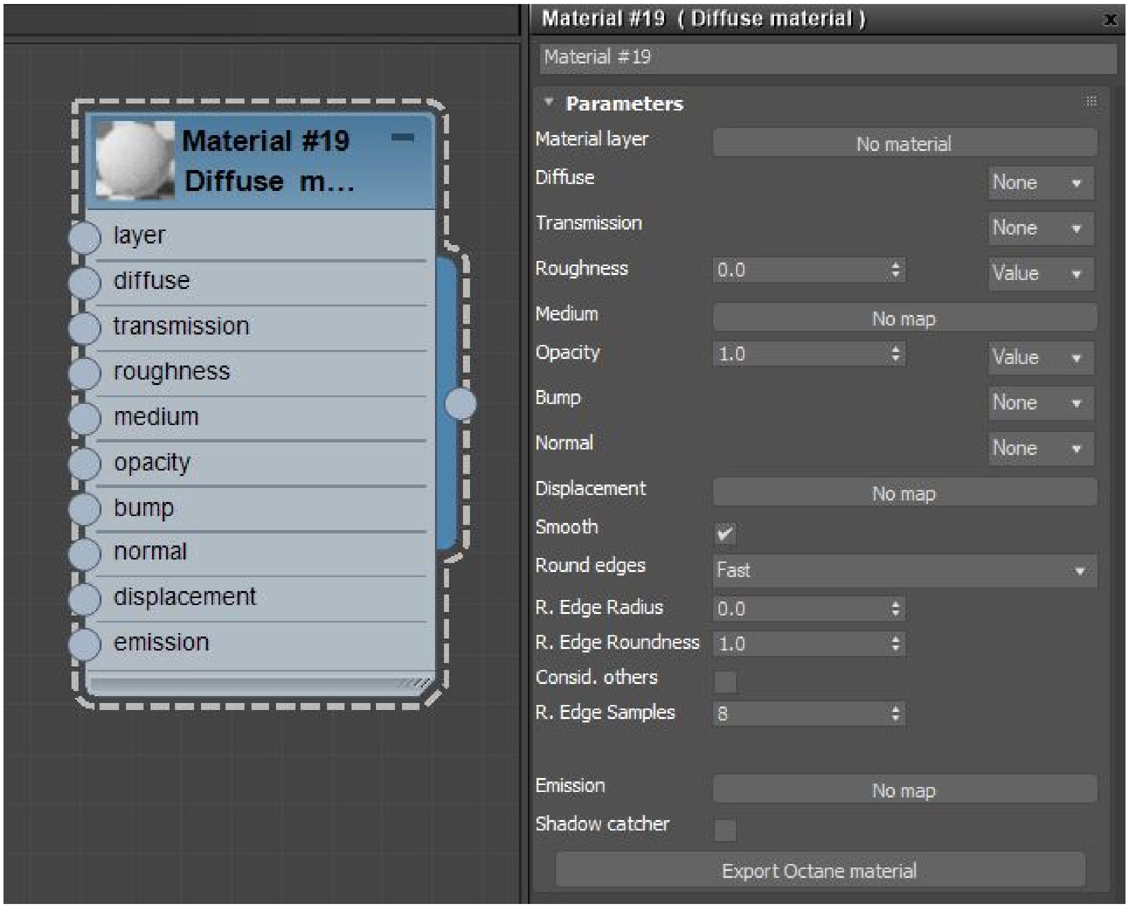


Figure 2: The Diffuse material parameters

Diffuse Material Parameters

Material[1] Layer - Adds a **Material Layer** above the base material. See the Material Layers topic in this manual for more details.

Diffuse- Provides color to the material. This is also known as base color or albedo. You can set Diffuse color by using a value, or connecting a **Procedural** or **Image** texture.

---

[1]The representation of the surface or volume properties of an object.

Transmission[1]- Uses a color or texture that is mixed with the material's Diffuse color, and is most noticeable in areas affected by indirect lighting.

Roughness - Determines the spread of highlights on the surface. A high Roughness value or light color can simulate very rough surfaces such as sand paper or clay. You can set Roughness using a value, or by connecting a Procedural or Image texture. A roughness value of **1** (white color) creates a diffuse sheen along the edges of the surface, simulating the look of crushed velvet.

Medium- OctaneRender® has three types of mediums to create translucent surfaces:

- **Absorption**[2] **Medium** - Produces the appearance of a material that absorbs light while passing through a surface. The resulting color depends on the distance that light travels through the material. For more information, see the Texture Overview topic in this manual.
- **Scattering**[3] **Medium** - Similar to the Absorption medium, but with an additional option for simulating subsurface scattering. Subsurface scattering is the phenomena that gives human skin and similar organic surfaces their characteristic glow under certain lighting conditions. It's a major component for creating the look of realistic skin. For more information, see the Texture Overview topic in this manual.
- **Volume Medium**[4] - Adds color and other qualities to a **VDB**[5] file. VDBs are a generic volume format for creating effects such as smoke, fog, vapor, and similar gaseous objects. VDBs can consist of a single frame, or an animated sequence. You can also download VDB files at http://www.openvdb.org/download/.
- **Random Walk -** Aa newer variant of subsurface scattering that utilizes a stochastic or random process for the scattering of light through an object. This provides the most realistic result when rendering scatter volumes.

Opacity - Determines what parts of the surface are visible in the render. Dark values indicate transparent areas, and light values indicate opaque areas. Values in-between light and dark indicate semi-transparent areas. You can lower the Opacity value to fade the object's overall visibility, or you can use a Texture map to vary the opacity across the surface. For example, if you want to make a simple polygon plane look like a leaf, you would connect a black-and-white image of the leaf's silhouette to the **Diffuse** shader's **Opacity** channel.

---

[1]A surface characteristic that determines if light may pass through a surface volume.

[2]Defines how fast light is absorbed while passing through a medium.

[3]Defines how fast light gets scattered when traveling through the medium.

[4]A shading system designed to render volumes such as smoke and fog.

[5]Dreamworks' open-source C++ library housing the data structures and tools implementation for storing and manipulating volume data, like smoke and other amorphous materials. The purpose of OpenVDB is mostly to have an efficient way to store volumetric data in memory and on disk. It has evolved into a more general toolkit that also lets you accomplish other things, such as fracturing volumes, converting meshes to volumes and vice versa. However, it does not include a computational fluid dynamics solver, and therefore it cannot procedurally generate smoke or fire. OpenVDB is fully integrated as a library in OctaneRender. For more information about OpenVDB, please see http://www.openvdb.org/.

When using an Image texture map, use a **Grayscale** image node for black-and-white images, to load an image for setting the transparency. To invert the transparency regions, select the image's **Invert** checkbox.

Bump - Creates fine details on the Material's surface using a Procedural or Image texture. When you connect a **Greyscale** texture to this parameter, light areas of the Texture look like protruding bumps, and dark areas look like indentations. You can adjust the Bump map's strength by adjusting the **Power** or **Gamma**[1] values on the Image texture. These attributes are covered in more detail under the Texture Overview topic in this manual.

Normal - Creates fine details on the surface. A Normal map is a special type of Image texture that uses red, green, and blue color values to perturb the surface's normals at render time, thus giving the appearance of added detail. They can be more accurate than Bump maps, but require specific software such as ZBrush®, Mudbox®, Substance Designer, Xnormal™, or others to generate. Note that Normal maps take precedence over Bump maps, so you cannot use a Normal map and a Bump map at the same time.

Displacement[2] - Adjusts the surface vertices' height at render time using an Image texture map. Displacement maps differs from Bump or Normal maps in that the geometry is altered by the Texture, as opposed to creating details. Displacement mapping is more complex than using a Bump or Normal map, but the results are more realistic, in particular along the surface's silhouette. Displacement mapping is covered in more detail under the Texture Overview topic in this manual.

Smooth - Smooths the transition between surface normals. If this option is disabled, the edges between the polygons of the surface are sharp, giving the surface a faceted look.

Round Edges - Rounds off geometry edges by using a shading effect instead of creating additional geometry. It's best used for rounded edges that will appear small in the final render. The **Fast** mode uses the rounding method introduced in OctaneRender® v3. The **Accurate** mode produces better-looking results, but may be slower. Accurate mode can select the affected edges by using the **Concave Only** or **Convex Only** options.

Rounded Edges Radius - Define the size of the rounded edge by radius.

Rounded Edges Roundness - Controls the rounded edge's shape. A value of 1 is completely round, while 0 is a chamfer.

Consider Other Objects - Controls how rounded edges are applied to different objects. When enabled, intersections between different objects are rounded. When disabled, only the current object is considered.

Rounded Edge Samples - Set samples for rendering edges.

---

[1] The function or attribute used to code or decode luminance for common displays. The computer graphics industry has set a standard gamma setting of 2.2 making it the most common default for 3D modelling and rendering applications.

[2] The process of utilizing a 2D texture map to generate 3D surface relief. As opposed to bump and normal mapping, Displacement mapping does not only provide the illusion of depth but it effectively displaces the actual geometric position of points over the textured surface.

Emission - Creates a surface that emits light (also known as a **Mesh** emitter). To use this option, connect a Diffuse material's Emission input to either a **Blackbody** or **Texture** emission node. These nodes are covered in more detail in the Texture Overview topic, and in the **Mesh Emitters**[1] topic under the Lighting Overview category in this manual.

Shadow Catcher[2] - Converts the Material into a shadow catcher. When it is active, the surface is visible in the areas that are in shadow, and all other areas are transparent in the render.

Export Octane Material - Opens a dialog window that provides options for exporting the Material to **ORBX**[3], the Local DB, or the Live DB.

Glossy Material

The **Glossy**[4] material is used for shiny materials such as plastics or metals.

---

[1] The ability for a surface to emit illumination usually described by a Black Body or Texture emission type.

[2] The Shadow Catcher can be used to create shadows cast by objects onto the surrounding background imagery. The shadows cast are not limited to simply a ground plane but can be cast onto other surfaces of varying shapes.

[3] The ORBX file format is the best way to transfer scene files from 3D Authoring software programs that use the Octane Plug-in such as Octane for Maya, Octane for Cinema 4D, or OctaneRender Standalone. This format is more efficient than FBX when working with Octane specific data as it provides a flexible, application independent format. ORBX is a container format that includes all animation data, models, textures etc. that is needed to transfer an Octane scene from one application to another.

[4] The measure of how well light is reflected from a surface in the specular direction, the amount and way in which the light is spread around the specular direction, and the change in specular reflection as the specular angle changes. Used for shiny materials such as plastics or metals.

Figure 1: The Glossy material[1]

[1]Used for shiny materials such as plastics or metals.

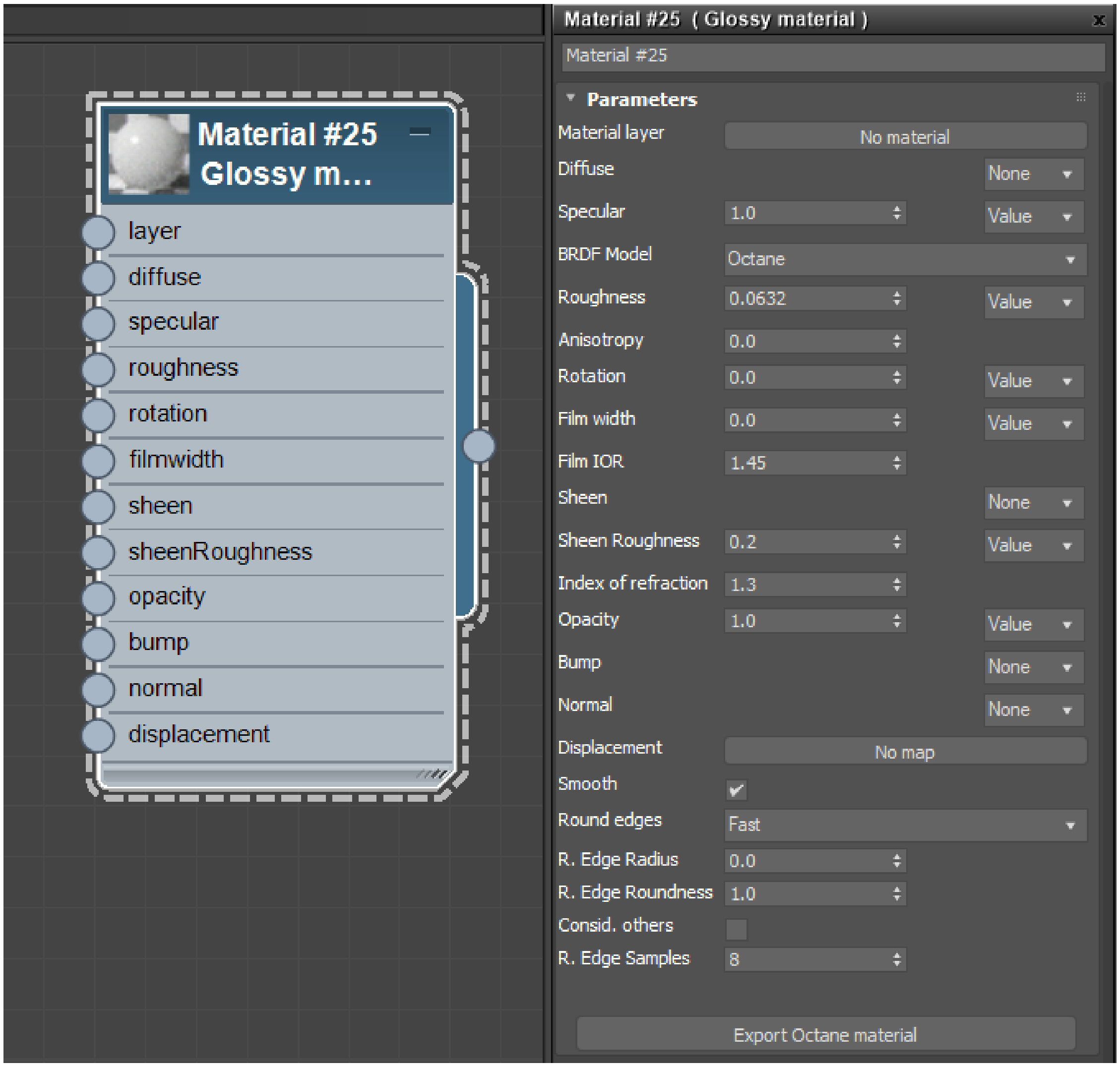


Figure 2: Glossy material parameters

Glossy Material Parameters

Material[1] **Layer** - Adds a **Material Layer** above the base material. See the Material Layers topic in this manual for more details.

---

[1]The representation of the surface or volume properties of an object.

Diffuse[1] - Gives color to the **Material**. In computer graphics, this is referred to as base color or albedo. You can set Diffuse color by using the color picker, or by connecting a **Procedural** or **Image** texture.

Specular[2] - Determines the intensity for Specular reflections on the surface. This parameter accepts color, values, or **Textures**[3]. In most cases, specular highlights are white or colorless. However, to simulate metallic surfaces, you should tint the Specular color using a color similar to the Diffuse parameter, like the bright yellow-orange highlights seen on a polished copper kettle.

BRDF Model - This attribute provides four options for determining the overall bi-directional reflectance distribution function (BRDF). The **Octane** option produces a more brushed-metal effect. The **Beckmann**, **GGX**, and **Ward** options produce more polished chrome-like effects.

Roughness - Determines how much the specular reflection spreads across the surface. In CG terminology, this is also known as reflection blur. A value of **0** simulates a perfect smooth reflective surface such as a mirror. Increasing the value simulates microfacets in the surface, which causes the reflective highlights to spread. For example, to create the look of worn plastic, increase the Roughness value. This parameter accepts a value or **Texture** map (**Procedural** or **Image**).

Anisotropy - Controls the material's reflectance uniformity. Reflectance changes based on surface orientation or rotation is anisotropic. If the reflectance is uniform in all directions and doesn't change based on the surface's orientation or rotation, then it is isotropic. This parameter's default value is **0**, which sets the metallic material as isotropic. Non-zero values mean the material exhibits anisotropic reflectance, where **-1** is horizontal and **1** is vertical.

Rotation - Controls the Anisotropy effect's orientation.

Film Width - Simulates the look of a thin film of material on the surface. This is useful when you want to create an effect like the rainbow colors that appear on an oil slick surface. Larger values increase the effect's strength.

Film IOR - Controls the film's Index of Refraction. Use this option to adjust the colors visible in the film.

Sheen - Applies a soft luster to a surface.

Sheen Roughness - Determines how the sheen spreads across the surface. Lower values create a sharp and narrow effect, and higher values spread the effect across a larger surface area.

Index Of Refraction - Determines the strength of reflections on the surface based on the Fresnel law. The Fresnel law describes the physical properties of light as it is reflected off of a surface at grazing angles. If Index of Refraction is set to a value higher than **1**, the reflection is strongest on the part of the surface that turns away from the viewer's angle (grazing angles), while the reflection appears weaker or less apparent on the parts of

---

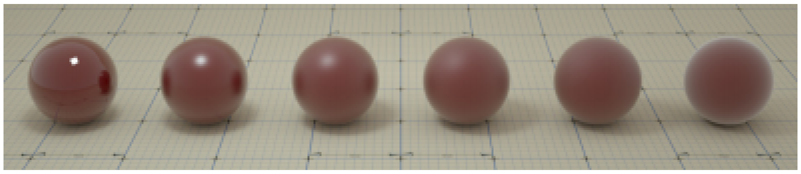
[1] Amount of diffusion, or the reflection of light photons at different angles from an uneven or granular surface. Used for dull, non-reflecting materials or mesh emitters.

[2] Amount of specular reflection, or the mirror-like reflection of light photons at the same angle. Used for transparent materials such as glass and water.

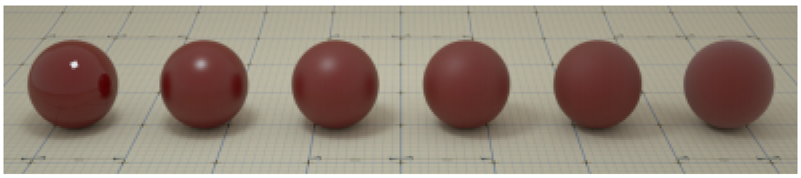
[3] Textures are used to add details to a surface. Textures can be procedural or imported raster files.

the surface perpendicular to the viewing angle. Since this is a physical phenomena, the result is a more realistic-looking surface. If Index of Refraction is set to a value lower than **1**, then the Fresnel effect is disabled and the reflection color simply appears as a uniform color across the highlight. The color of the reflective highlight itself is determined by the color connected to the Specular channel.

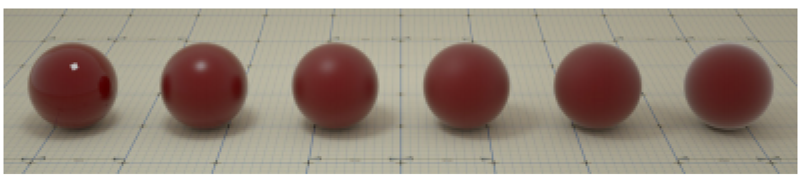
In the following examples, the six balls have a Roughness of **0**, **0.2**, **0.4**, **0.6**, **0.8**, and **1.0** (left to right), and only the Specular value and Index of Refraction are modified for each rendered image:



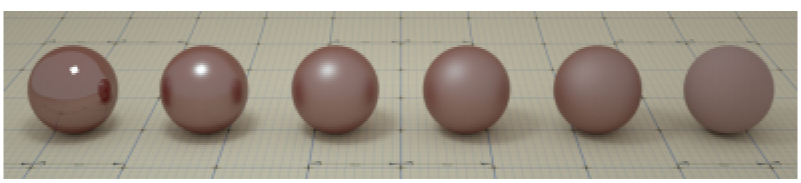
specular = 1.0
index = 2.0 (strong Fresnel reflection)



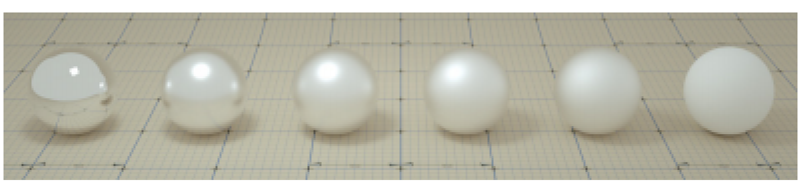
specular = 0.3
index = 3.0 (very strong Fresnel reflection)



specular = 1.0
index = 1.5 (normal Fresnel reflection)



specular = 0.3
index = 0.0 (no Fresnel reflection)



specular = 1.0
index = 0.0 (no Fresnel reflection)

Figure 3: Spheres rendered using different settings for Specular and Index

Opacity - Determines what parts of the surface are visible in the render. Dark values indicate transparent areas, and light values determine opaque areas. Values in-between light and dark create the look of semi-transparent areas. Lowering the Opacity value lowers the Object's overall visibility, and using a Texture map varies the opacity across the surface. For example, if you want to make a simple polygon plane look like a leaf, connect a black-and-white image of the leaf's silhouette to the Diffuse shader's Opacity channel.

Bump - Creates fine details on the Material's surface using a Procedural or Image texture. When you connect a **Grayscale** texture to this parameter, light areas of the Texture look like protruding bumps, and dark areas look like indentations. You can adjust the Bump map's strength by setting the **Power** or **Gamma**[1] values on the Image texture. These attributes are covered in more detail in the Texture Overview topic in this manual.

Normal - Creates fine details on the surface. A Normal map is a special type of Image texture that uses red, green, and blue color values to perturb the normals of the surface at render time, giving the appearance of added detail. They can be more accurate than Bump maps, but requires specific software to generate.

Displacement[2] - Adjusts the surface vertices' height at render time using an Image texture map. Displacement maps differs from Bump or Normal maps in that the geometry is altered by the Texture, as opposed to creating details. Displacement mapping is more complex than using a Bump or Normal map, but the results are more realistic, in particular along the surface's silhouette. Displacement mapping is covered in more detail under the Texture Overview topic in this manual.

Smooth - Smooths the transition between surface normals. If this option is disabled, the edges between the polygons of the surface are sharp, giving the surface a faceted look.

Round Edges - Rounds off geometry edges by using a shading effect instead of creating additional geometry. It's best used for rounded edges that will appear small in the final render. The **Fast** mode uses the rounding method introduced in OctaneRender® v3. The **Accurate** mode produces better-looking results, but may be slower. Accurate mode can select the affected edges by using the **Concave Only** or **Convex Only** options.

Rounded Edges Radius - Define the size of the rounded edge by radius.Bevels the surface edges at render time without altering or subdividing the geometry. Using this option enhances object realism by eliminating sharp edges. The value refers to the rounded edge's radius. Higher values produce rounder edges.

Rounded Edges Roundness - Controls the rounded edge's shape. A value of 1 is completely round, while 0 is a chamfer.

---

[1] The function or attribute used to code or decode luminance for common displays. The computer graphics industry has set a standard gamma setting of 2.2 making it the most common default for 3D modelling and rendering applications.

[2] The process of utilizing a 2D texture map to generate 3D surface relief. As opposed to bump and normal mapping, Displacement mapping does not only provide the illusion of depth but it effectively displaces the actual geometric position of points over the textured surface.

Consider Other Objects - Controls how rounded edges are applied to different objects. When enabled, intersections between different objects are rounded. When disabled, only the current object is considered.

Rounded Edge Samples - Set samples for rendering edges.

Export Octane Material - Opens a dialog window that provides options for exporting the Material to **ORBX**[1], the Local DB, or the Live DB.

Hair Material

The **Hair Material**[2] parameters are focused on characteristics common with hair and fur strands (Figure 1).

[1]The ORBX file format is the best way to transfer scene files from 3D Authoring software programs that use the Octane Plug-in such as Octane for Maya, Octane for Cinema 4D, or OctaneRender Standalone. This format is more efficient than FBX when working with Octane specific data as it provides a flexible, application independent format. ORBX is a container format that includes all animation data, models, textures etc. that is needed to transfer an Octane scene from one application to another.
[2]The representation of the surface or volume properties of an object.

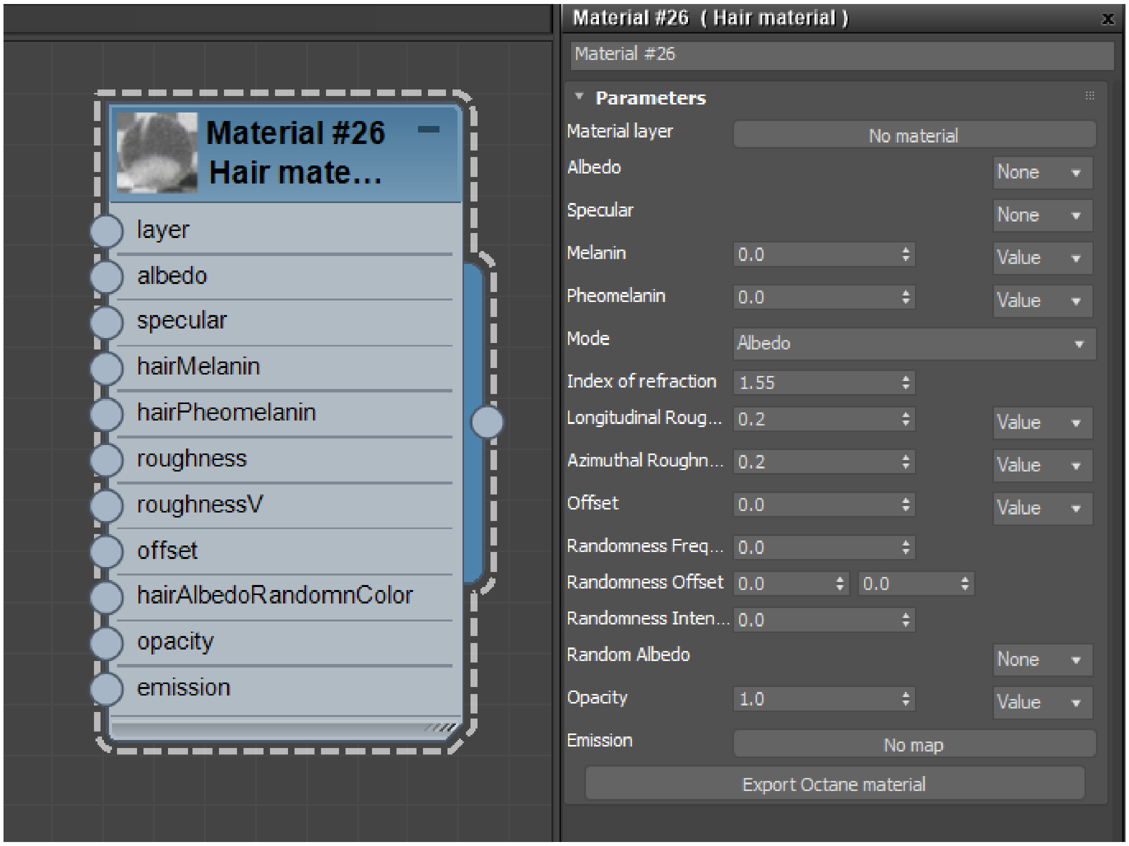


Figure 2: Hair Material parameters

Hair Material Parameters

Albedo - The hair base color.

Specular[1] - The hair specular or shininess color.

Melanin - The quantity of pigment for the hair base color.

Pheomelanin - The amount of redness in the hair strand.

Mode - Determines whether to use the Albedo or the Melanin/Pheomelanin parameters to determine the hair color.

Index of Refraction - This parameter controls the level of the Fresnel effect on the specular reflection.

---

[1]Amount of specular reflection, or the mirror-like reflection of light photons at the same angle. Used for transparent materials such as glass and water.

Longitudinal Roughness - Controls the roughness along the hair strand.

Azimuthal Roughness - Controls the roughness along a hair strand's cross section.

Offset - Scale offset on the surface of the hair. A value of 0 demotes perfectly smooth cylindrical hair. Increasing this value shifts the specular highlight away from a perfectly reflective direction.

Randomness Frequency - Controls the frequency of randomness on the hair for a more believable effect.

Randomness Offset - Works much like a seed value and offsets the randomness effect.

Randomness Intensity - Controls the intensity of the randomness on each hair strand.

Random Albedo - Specifies the target random albedo on the hair. This parameter will only work with the Albedo mode enabled.

Opacity - Controls the transparency value of the hair using greyscale values.

Emission -Determines whether the hair material will function as an emission surface.

Material Layer - Adds a Material Layer above the base material. See the Material Layers topic in this manual for more details.

Export Octane Material - Opens a dialog window that provides options for exporting the Material to **ORBX**[1], the Local DB, or the Live DB.

Layer Group

The **Layer Group** node adds more than one **Material**[2] **Layer** type to an existing **Material** through the Material's **Layer** input pin. The layered material system lets you construct complex **Materials**[3] that consist of a base layer and up to eight Material Layers. The Layers are based on components used in previous OctaneRender® Materials.

---

[1]The ORBX file format is the best way to transfer scene files from 3D Authoring software programs that use the Octane Plug-in such as Octane for Maya, Octane for Cinema 4D, or OctaneRender Standalone. This format is more efficient than FBX when working with Octane specific data as it provides a flexible, application independent format. ORBX is a container format that includes all animation data, models, textures etc. that is needed to transfer an Octane scene from one application to another.

[2]The representation of the surface or volume properties of an object.

[3]A set of attributes or parameters that describe surface characteristics.

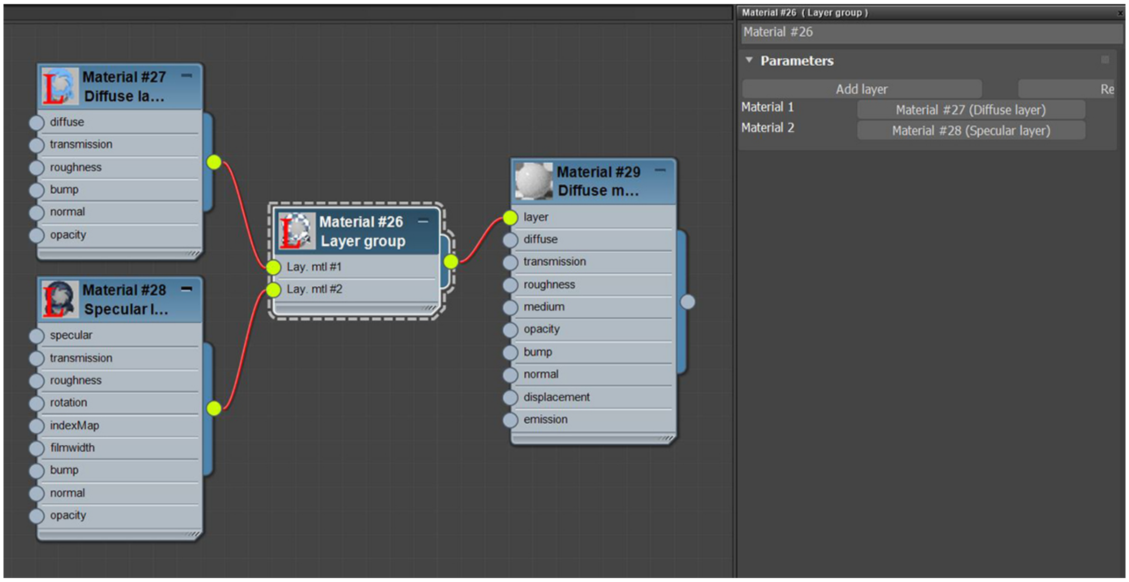


Figure 1: A Diffuse[1] and a Specular[2] layer are mixed using the Layer Group node

Using this set of unique Layers, OctaneRender® lets you recreate complex Materials in a physically-based manner, as opposed to manually **mixing materials**[3] together.

The following Material Layer nodes are available:

- **Diffuse Layer** - Used for dull, non-reflective materials.
- **Layer Group** - Adds multiple Material layers to existing Materials.
- **Metallic Layer** - Used for highly reflective Materials.
- **Sheen Layer** - Simulates the grazing coloration in fabrics.
- **Specular Layer** - Used for shiny Materials like plastic, or clear Materials like glass.

Material Layers can connect to the **Layered Material**, **Layer Group**, or **Material Layer** pins on standard Materials.

Layered Material

[1]Amount of diffusion, or the reflection of light photons at different angles from an uneven or granular surface. Used for dull, non-reflecting materials or mesh emitters.
[2]Amount of specular reflection, or the mirror-like reflection of light photons at the same angle. Used for transparent materials such as glass and water.
[3]Used to mix any two material types.

The **Layered Material**[1] node constructs complex **Materials**[2] that consist of a base layer and up to eight **Material Layers**. You can create complex Materials in a physically-based manner, as opposed to manually **mixing Materials**[3] together.

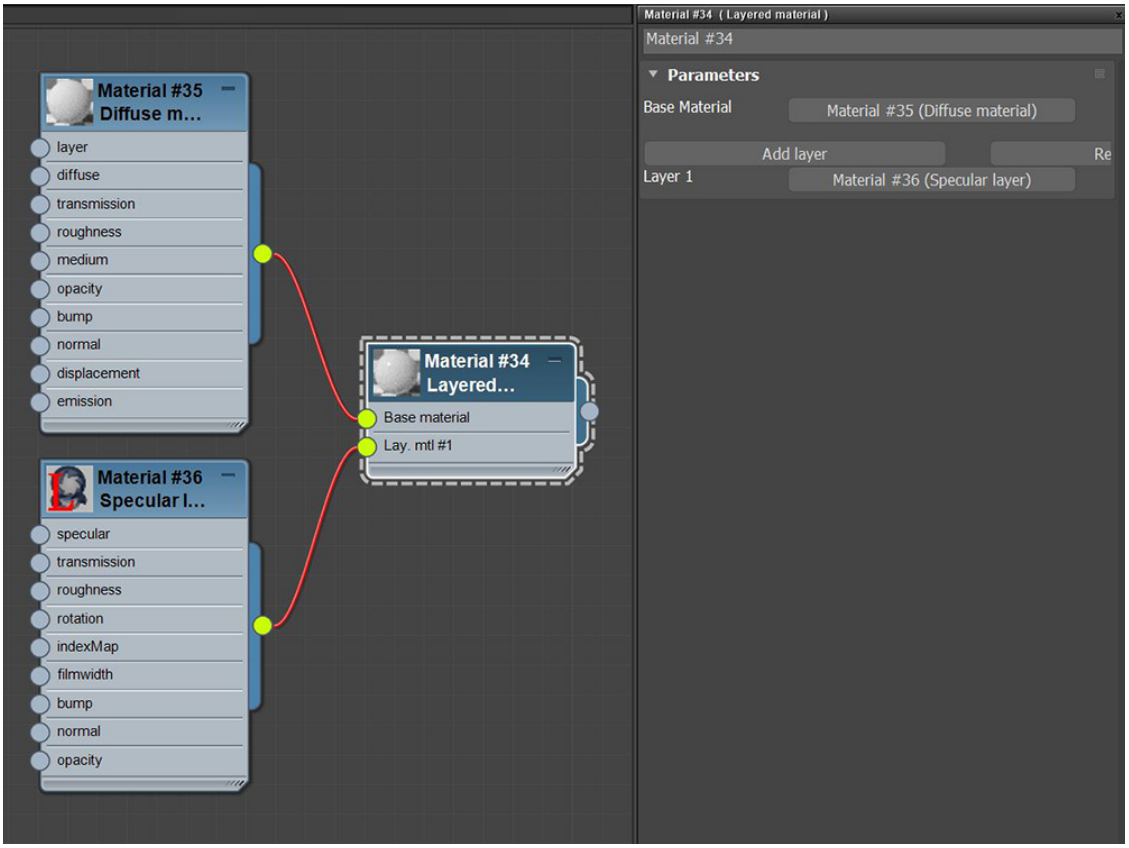


Figure 1: The Layered Material with a Diffuse material[4] as the base Material, and a Specular[5] layer as the Layer 1 input

---

[1] The representation of the surface or volume properties of an object.

[2] A set of attributes or parameters that describe surface characteristics.

[3] Used to mix any two material types.

[4] Used for dull, non-reflecting materials or mesh emitters.

[5] Amount of specular reflection, or the mirror-like reflection of light photons at the same angle. Used for transparent materials such as glass and water.

Layered Material Parameters

Add Layer - Adds a new **Layer** input to the end of the **Node**. You can add up to eight Layer inputs.

Remove Layer - Removes the last Layer input on the Node.

Base Material - The Material that sits below any additional Material Layers.

Layer 1 - 8 - The Material Layer inputs.

With the Layered material, you are given all Material Layers used in OctaneRender®, letting you reconstruct pre-existing OctaneRender® Materials or your own uber-Material.

Metallic Layer

The layered material system lets you construct complex **Materials**[1] that consist of a base layer and up to eight **Material**[2] Layers. The Layers are based on components used in previous OctaneRender® Materials. Using this set of unique layers, OctaneRender® now lets you recreate complex Materials in a physically-based manner, as opposed to manually **mixing Materials**[3] together.

The **Metallic** layer is used for highly reflective Materials that have colored reflections. For more information, see the Metallic Material topic in this manual. Material Layers can connect to the **Layered Material**, **Layer Group**, or **Material Layer** pins on standard Materials.

---

[1] A set of attributes or parameters that describe surface characteristics.

[2] The representation of the surface or volume properties of an object.

[3] Used to mix any two material types.

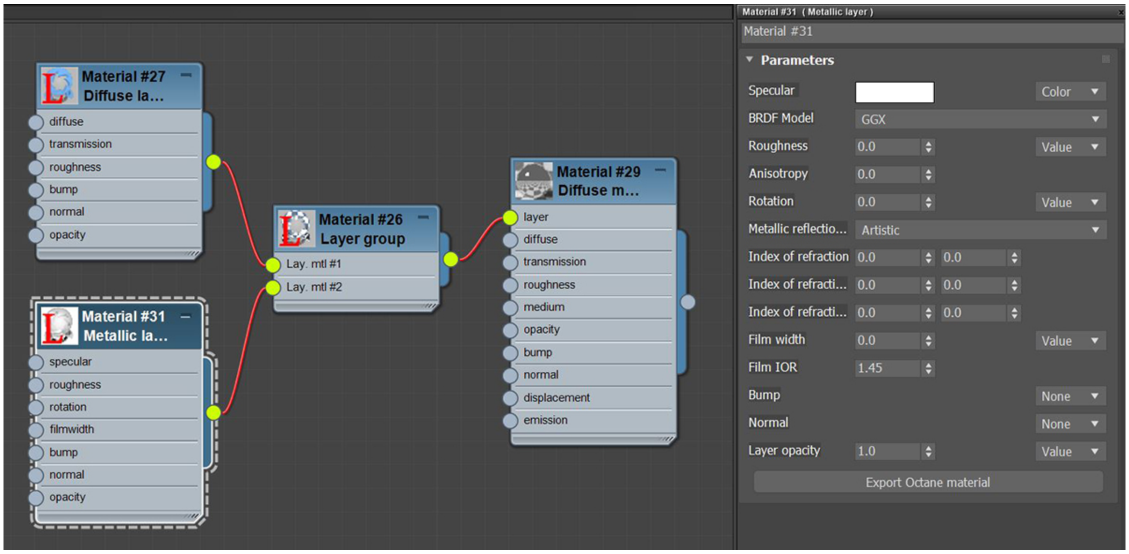


Figure 1: **Diffuse**[1] and Metallic layers are mixed by using the Layer Group node

Metallic Layer Parameters

Specular[2] - The Layer's coating color.

BRDF Model - Determines how light reflects or refracts. See the BRDF Models topic in this manual for more information.

Roughness - Adjusts how rough the Metallic layer looks.

Anisotropy - The Metallic layer's anisotropy. A value of **-1** is horizontal, while **1** is vertical. A value of **0** is isotropic.

Rotation - The Metallic anisotropic reflection's rotation.

Metallic Reflection Mode - This changes how OctaneRender® calculates reflectivity.

- **Artistic** - Uses the Metallic color.
- **IOR + Color** - Uses the Metallic color and adjusts the brightness using the IOR.
- **RGB IOR** - Uses the three IOR values (for 650 nm, 550 nm, and 450 nm) and ignores the Metallic color.

---

[1] Amount of diffusion, or the reflection of light photons at different angles from an uneven or granular surface. Used for dull, non-reflecting materials or mesh emitters.
[2] Amount of specular reflection, or the mirror-like reflection of light photons at the same angle. Used for transparent materials such as glass and water.

Index Of Refraction - Complex-valued IOR (n-k*i) controlling the specular reflection's Fresnel effect, where n = the refractive index and k = the attenuation or extinction coefficient. For RGB mode, the IOR for red light (650nm).

Index Of Refraction (Green) - For RGB mode, the IOR for red light (550nm).

Index Of Refraction (Blue) - For RGB mode, the IOR for red light (450nm).

Film Width - Sets the film coating's thickness.

Film IOR - Sets the film coating's Index Of Refraction.

Bump - Simulates a relief using a **Greyscale** texture interpreted as a **Height** map for the Layer.

Normal - Distorts the Layer normals using an **RGB** image.

Layer Opacity - Controls the Layer's opacity with a Greyscale texture.

Export Octane Material - Opens a dialog window that provides options for exporting the Material to **ORBX**[1], the Local DB, or the Live DB.

Metallic Material

The **Metallic** material has similar attributes and surface characteristics as the **Glossy**[2] material. However, its default settings produce a more accurate metallic surface without any adjustments.

---

[1]The ORBX file format is the best way to transfer scene files from 3D Authoring software programs that use the Octane Plug-in such as Octane for Maya, Octane for Cinema 4D, or OctaneRender Standalone. This format is more efficient than FBX when working with Octane specific data as it provides a flexible, application independent format. ORBX is a container format that includes all animation data, models, textures etc. that is needed to transfer an Octane scene from one application to another.

[2]The measure of how well light is reflected from a surface in the specular direction, the amount and way in which the light is spread around the specular direction, and the change in specular reflection as the specular angle changes. Used for shiny materials such as plastics or metals.

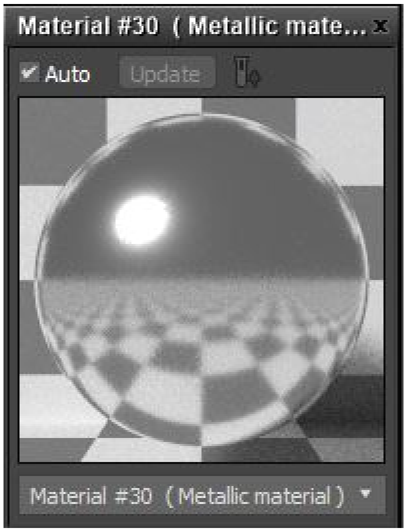


Figure 1: The Metallic material

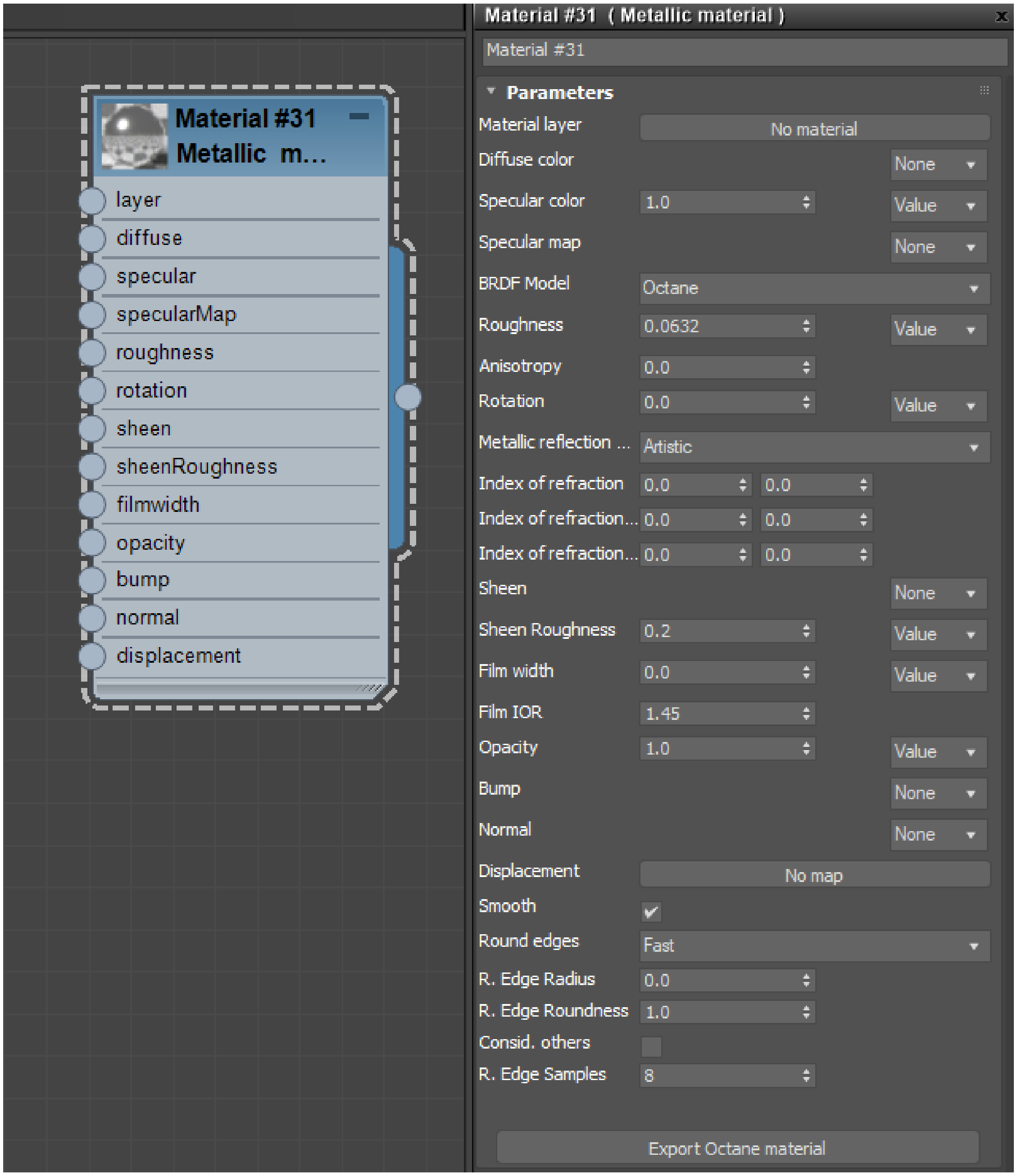


Figure 2: Metallic material parameters

Metallic Material Parameters

Material[1] Layer - Adds a Material layer above the base material. See the Material Layers topic in this manual for more details.

Diffuse[2] Color - The diffuse texture for the **Reflection** channel.

Specular[3] Color - The specular reflection channel, which determines the metallic color. If the **IOR** is set to a value less than **0**, OctaneRender® adjusts the color brightness to match the Fresnel equations.

Specular Map - Controls the blend between the **Diffuse** and **Specular** channels.

BRDF Model - Provides four options for determining the overall bi-directional reflectance distribution function (BRDF). The **Octane** option produces a more brushed-metal effect. The **Beckmann**, **GGX**, and **Ward** options produce more polished chrome-like effects.

Roughness - Determines how much the specular reflection spreads across the surface. In CG terminology, this is also known as reflection blur. A value of **0** simulates a perfect smooth reflective surface such as a mirror. Increasing the value simulates microfacets in the surface, which causes the reflective highlights to spread. For example, to create the look of worn plastic, increase the Roughness value. This parameter accepts a value or **Texture** map (**Procedural** or **Image**).

Anisotropy- Adjusts the amount of change that a surface's reflectance has, depending on viewing direction.

Rotation - Controls the Anisotropy effect's orientation.

Metallic Refl. Mode - This attribute, along with the IOR attributes, provide options to control the IOR across a surface.

> **Artistic** - Uses the albedo color.
>
> **IOR + Color** - Uses the albedo color and further adjusts the surface brightness using the IOR.
>
> **RBG IOR** - This is the most-used mode. It uses the three IOR values and ignores the albedo color.

Index Of Refraction - Complex-valued IOR (n-k*i) controlling the specular reflection's Fresnel effect, where n = the refractive index and k = the attenuation or extinction coefficient. For RGB mode, the IOR for red light (650nm).

Index Of Refraction (Green) - For RGB mode, the IOR for red light (550nm).

Index Of Refraction (Blue) - For RGB mode, the IOR for red light (450nm).

Sheen - The subtle lustre's color on the material's surface.

---

[1] The representation of the surface or volume properties of an object.

[2] Amount of diffusion, or the reflection of light photons at different angles from an uneven or granular surface. Used for dull, non-reflecting materials or mesh emitters.

[3] Amount of specular reflection, or the mirror-like reflection of light photons at the same angle. Used for transparent materials such as glass and water.

Sheen Roughness - The Roughness channel for the sheen that is present on Metallic and Glossy materials.

Film Width - Simulates the look of a thin film of material on the surface. This is useful when you want to create an effect such as the rainbow colors that appear on an oil slick surface. Larger values increase the effect's strength.

Film IOR - Controls the thin film's IOR by adjusting its visible colors.

Opacity - Controls the **Toon** material's transparency with a **Grayscale** texture.

Bump - Simulates a relief using a **Grayscale** texture interpreted as a **Height** map.

Normal - Distorts normals based on an **RGB** image.

Displacement[1] - Creates very detailed geometry with a low memory footprint.

Smooth - Enables or disables normal interpolation. If normal interpolation is disabled, triangle meshes appear faceted.

Round Edges - Rounds off geometry edges by using a shading effect instead of creating additional geometry. It's best used for rounded edges that will appear small in the final render. The **Fast** mode uses the rounding method introduced in OctaneRender® v3. The **Accurate** mode produces better-looking results, but may be slower. Accurate mode can select the affected edges by using the **Concave Only** or **Convex Only** options.

Rounded Edges Radius - Define the size of the rounded edge by radius. Bevels the surface edges at render time without altering or subdividing the geometry. Using this option enhances object realism by eliminating sharp edges. The value refers to the rounded edge's radius. Higher values produce rounder edges.

Rounded Edges Roundness - Controls the rounded edge's shape. A value of 1 is completely round, while 0 is a chamfer.

Consider Other Objects - Controls how rounded edges are applied to different objects. When enabled, intersections between different objects are rounded. When disabled, only the current object is considered.

Rounded Edge Samples - Set samples for rendering edges.

Export Octane Material - Opens a dialog window that provides options for exporting the Material to **ORBX**[2], the Local DB, or the Live DB.

---

[1] The process of utilizing a 2D texture map to generate 3D surface relief. As opposed to bump and normal mapping, Displacement mapping does not only provide the illusion of depth but it effectively displaces the actual geometric position of points over the textured surface.

[2] The ORBX file format is the best way to transfer scene files from 3D Authoring software programs that use the Octane Plug-in such as Octane for Maya, Octane for Cinema 4D, or OctaneRender Standalone. This format is more efficient than FBX when working with Octane specific data as it provides a flexible, application independent format. ORBX is a container format that includes all animation data, models, textures etc. that is needed to transfer an Octane scene from one application to another.

Mix Material

The **Mix** material mixes any two **Material**[1] types. It accepts any two Material nodes, and you control the mix with a value, color, or **Texture**.

In the following example (Figure 2), the **Mix material**[2] mixes two **Diffuse**[3] materials together using a **Checks** texture as the **Mix Amount**.

---

[1] The representation of the surface or volume properties of an object.

[2] Used to mix any two material types.

[3] Amount of diffusion, or the reflection of light photons at different angles from an uneven or granular surface. Used for dull, non-reflecting materials or mesh emitters.

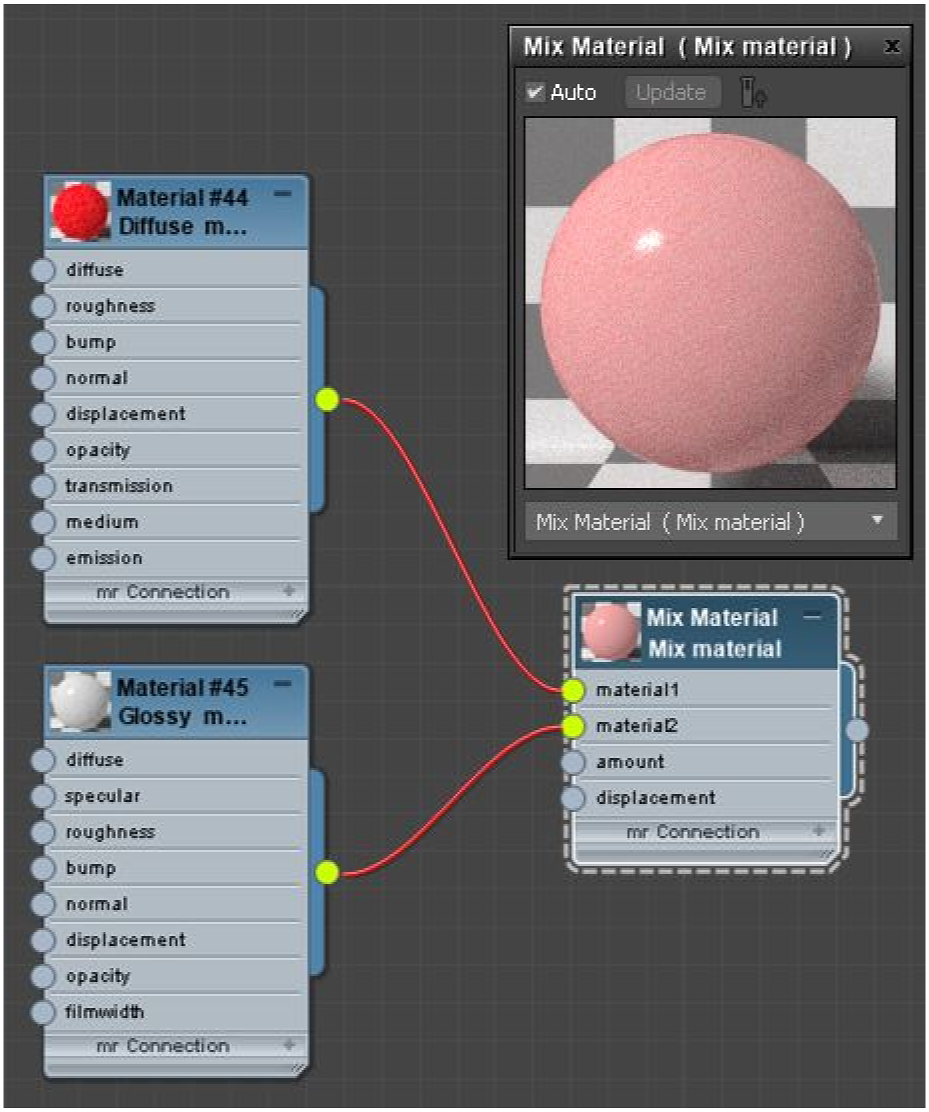


Figure 1: The Mix material

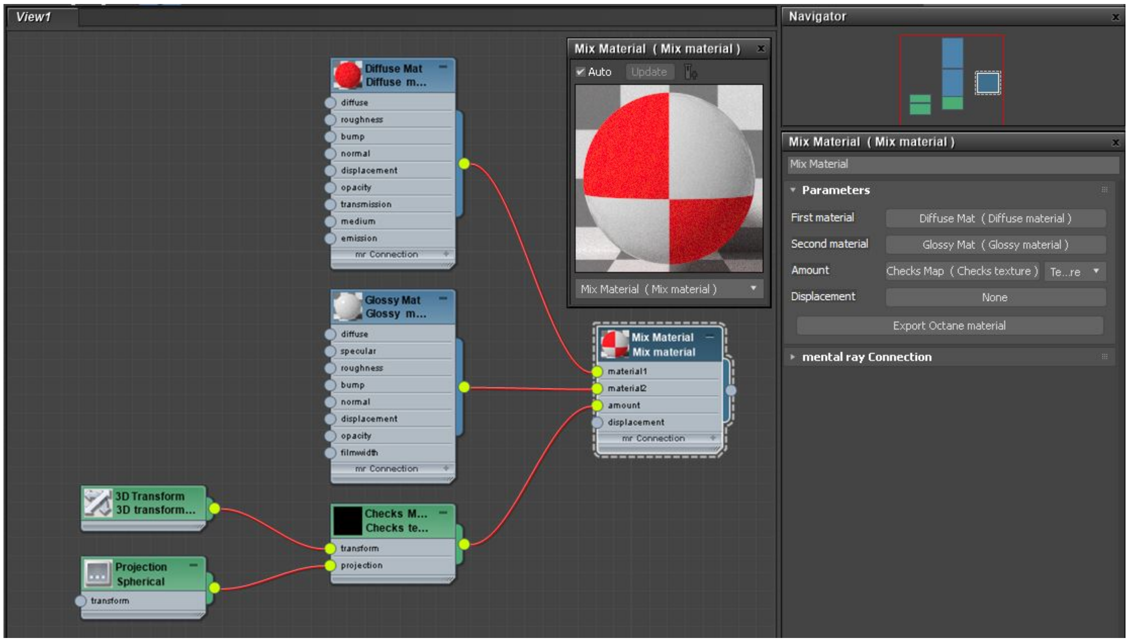


Figure 2: The Checks texture controls mixing two Materials[1]

Mix Material Parameters

First Material and **Second Material** - These slots connect OctaneRender® Materials to the Mix material. You can connect any OctaneRender® Material to a slot, including other Mix materials. You can create a complex Material by connecting multiple Mix materials together. Common uses for the Mix material include combining a **Diffuse material**[2] with an **Emissive** output together with a **Glossy**[3] material that has specular reflections, or combining a shiny **Glossy material**[4] with a **Specular material**[5] that uses a Medium to create subsurface scattering effects.

Amount - The slider goes from **0** - **1** and controls the amount of influence each Material has on the surface. A value of **1** means **First Material** has the highest amount of influence. A value of **0** means **Second Material** has the highest amount of influence. Values in-between blend the materials together. Click on the checker

---

[1] A set of attributes or parameters that describe surface characteristics.

[2] Used for dull, non-reflecting materials or mesh emitters.

[3] The measure of how well light is reflected from a surface in the specular direction, the amount and way in which the light is spread around the specular direction, and the change in specular reflection as the specular angle changes. Used for shiny materials such as plastics or metals.

[4] Used for shiny materials such as plastics or metals.

[5] Used for transparent materials such as glass and water.

swatch to the right of the slider to connect a Texture node to control the mix amount. You can use **Procedural** or **Image** textures to control the amount. Dark and light Texture values correspond to an amount setting of **0** and **1**.

Displacement[1] - Connect a **Displacement** node to the Mix material's **Displacement** slot. This is preferable to mixing Materials that have their own Displacement nodes and settings.

Export Octane Material - Opens a dialog window that provides options for exporting the Material to **ORBX**[2], the Local DB, or the Live DB.

Portal Material

Portal[3] materials optimize rendering light sources. They accomplish this by helping the render kernel find important light sources in the scene. For example, for interior scenes illuminated by an outside light source that comes in through windows, it can be difficult for the path tracer to optimize the light as it enters the interior environment. To help the path tracer find these light sources, you can place a polygon plane outside the window and then apply a Portal material to the plane, which creates a portal plane. This setup improves the light quality and increases the render efficiency.

---

[1] The process of utilizing a 2D texture map to generate 3D surface relief. As opposed to bump and normal mapping, Displacement mapping does not only provide the illusion of depth but it effectively displaces the actual geometric position of points over the textured surface.

[2] The ORBX file format is the best way to transfer scene files from 3D Authoring software programs that use the Octane Plug-in such as Octane for Maya, Octane for Cinema 4D, or OctaneRender Standalone. This format is more efficient than FBX when working with Octane specific data as it provides a flexible, application independent format. ORBX is a container format that includes all animation data, models, textures etc. that is needed to transfer an Octane scene from one application to another.

[3] A technique that assists the render kernel with exterior light sources that illuminate interiors. In interior renderings with windows, it is difficult for the path tracer to find light from the outside environment and optimally render the scene. Portals are planes that are added to the scene with the Portal material applied to them.

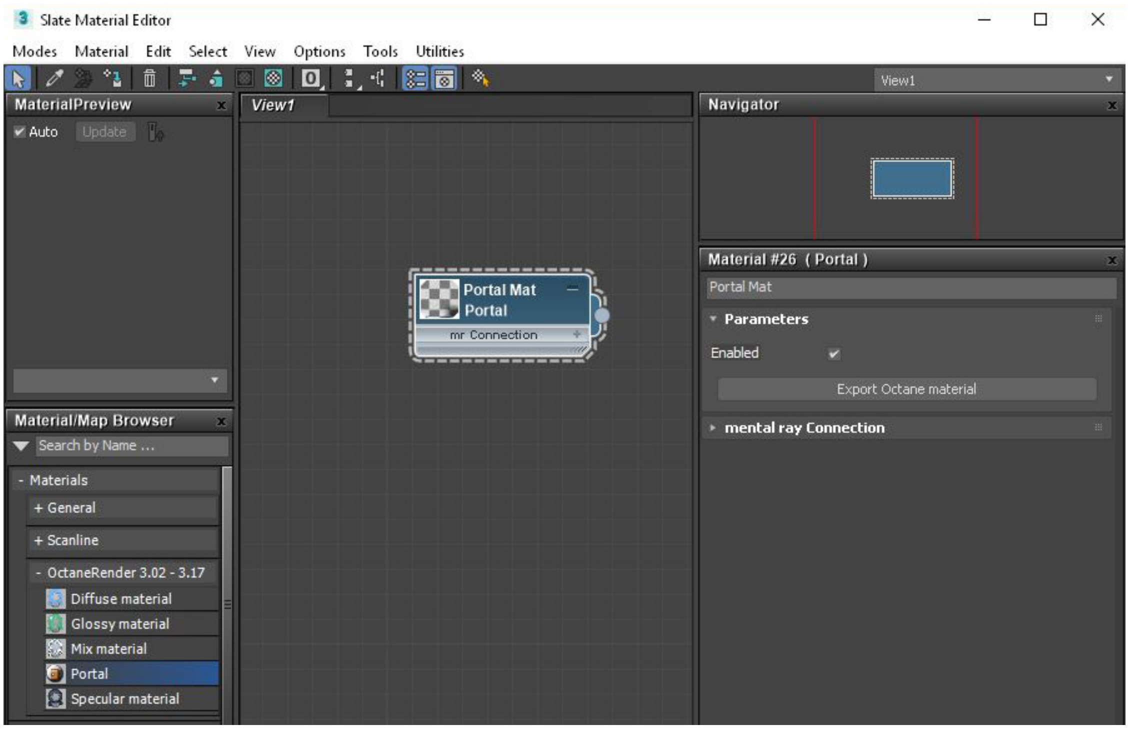


Figure 1: A Portal material in the Slate Material[1] Editor

To set up a scene using Portal materials, make sure that every window or opening in the environment is covered by a portal plane. It will not work if only one window has a portal over it when all other windows do not have a portal over them. And the normal direction of the portal plane should be facing inwards towards the interior, or the scene will not render properly. Don't block portal planes with other geometry like glass surfaces. Objects with the Portal material applied are invisible as geometry in the rendering.

We recommend using the least amount of geometry for Portals. A few simple rectangular planes are best, as dense geometry used for portal planes can slow down rendering. It is possible to use a single piece of portal geometry to cover several openings such as multiple windows on a single wall. However, if the geometry is too large, that can reduce rendering efficiency. It's important to strike a balance between an opening's coverage and the size of the geometry that uses the Portal material.

Use Portal materials with the **Pathtracing** and **PMC** kernels, as it will not work when rendering with the **Direct Light** kernel.

The two images in Figure 3 show the rendering results with and without a Portal material. The scene shows a glass sphere rendered in a room lit by light coming through a window. The scene is rendered using 500

---

[1] The representation of the surface or volume properties of an object.

samples. The first image does not have a portal plane placed over the opening, and it is noisier than the second image, which does have a portal plane.

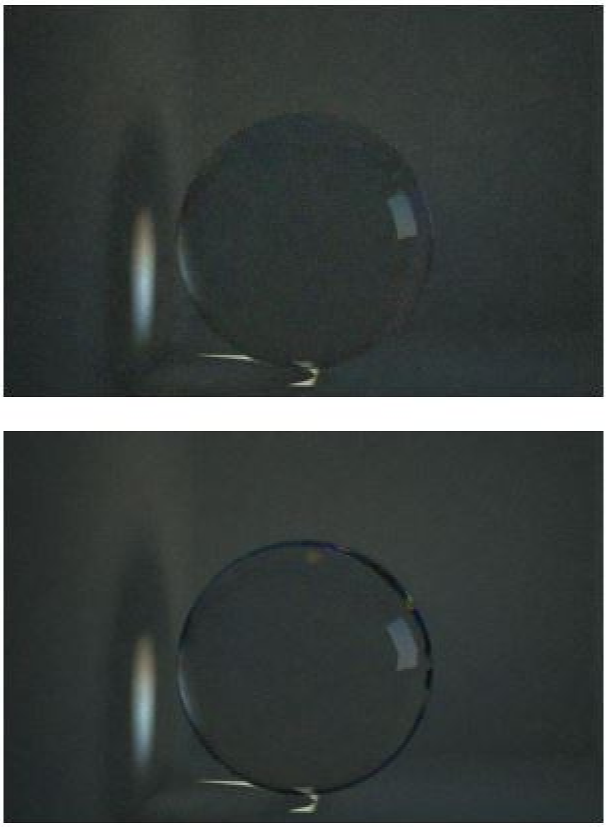


Figure 2: The first image is rendered without a Portal material, the second is rendered with a Portal material

Portal Material Parameters

Enabled - Enabled or disable checkbox

Export Octane Material - Opens a dialog window that provides options for exporting the Material to **ORBX**[1], the Local DB, or the Live DB.

---

[1]The ORBX file format is the best way to transfer scene files from 3D Authoring software programs that use the Octane Plug-in such as Octane for Maya, Octane for Cinema 4D, or OctaneRender Standalone. This format is more efficient than FBX when working with Octane specific data as it provides a flexible, application independent format. ORBX is a container format that includes all animation data, models, textures etc. that is needed to transfer an Octane scene from one application to another.

Shadow Catcher

The **Shadow Catcher**[1] option creates shadows cast by objects onto the surrounding geometry. The shadows cast are not limited to a ground plane, but can be cast onto other surfaces of varying shapes.

You can enable Shadow Catcher by activating the **Matte** option on the **Diffuse**[2] material. The objects receiving the shadow should have a **Diffuse material**[3] applied to it.

In the **Render Settings** window, activate **Alpha Channel**[4] and disable **Keep Environment**. When the image is rendered, the shadows appear over the transparent parts of the surface. You can use this image in a compositing package to merge the object and the shadows into the composition.

---

[1] The Shadow Catcher can be used to create shadows cast by objects onto the surrounding background imagery. The shadows cast are not limited to simply a ground plane but can be cast onto other surfaces of varying shapes.

[2] Amount of diffusion, or the reflection of light photons at different angles from an uneven or granular surface. Used for dull, non-reflecting materials or mesh emitters.

[3] Used for dull, non-reflecting materials or mesh emitters.

[4] A greyscale image used to determine which areas of a texture map are opaque and which areas are transparent.

Figure 1 : Shadows appear in the transparent parts of the image where the Shadow Catcher objects were placed

(By Icelaglace)

Shadow Catcher Parameters

Enabled - The material is transparent unless there is some direct shadow cast onto the material, which makes it less transparent depending on the shadow strength.

Opacity - Control visibility strength.

Export Octane Material[1] - Opens a dialog window that provides options for exporting the Material to **ORBX**[2], the Local DB, or the Live DB.

Sheen Layer

The layered material system lets you construct complex **Materials**[3] that consist of a base layer and up to eight **Material**[4] **Layers**. The Layers are based on components used in previous OctaneRender® Materials. Using this set of unique Layers, OctaneRender® recreates complex Materials in a physically-based manner, as opposed to manually **mixing Materials**[5] together.

The **Sheen** layer simulates the grazing coloration or rim lighting in fabrics like velvet. It can also simulate layers of dust. See the Universal Material topic in this manual for more information. Material Layers can connect to the **Layered Material**, **Layer Group**, or **Material Layer** pins on standard Materials.

---

[1]The representation of the surface or volume properties of an object.

[2]The ORBX file format is the best way to transfer scene files from 3D Authoring software programs that use the Octane Plug-in such as Octane for Maya, Octane for Cinema 4D, or OctaneRender Standalone. This format is more efficient than FBX when working with Octane specific data as it provides a flexible, application independent format. ORBX is a container format that includes all animation data, models, textures etc. that is needed to transfer an Octane scene from one application to another.

[3]A set of attributes or parameters that describe surface characteristics.

[4]The representation of the surface or volume properties of an object.

[5]Used to mix any two material types.

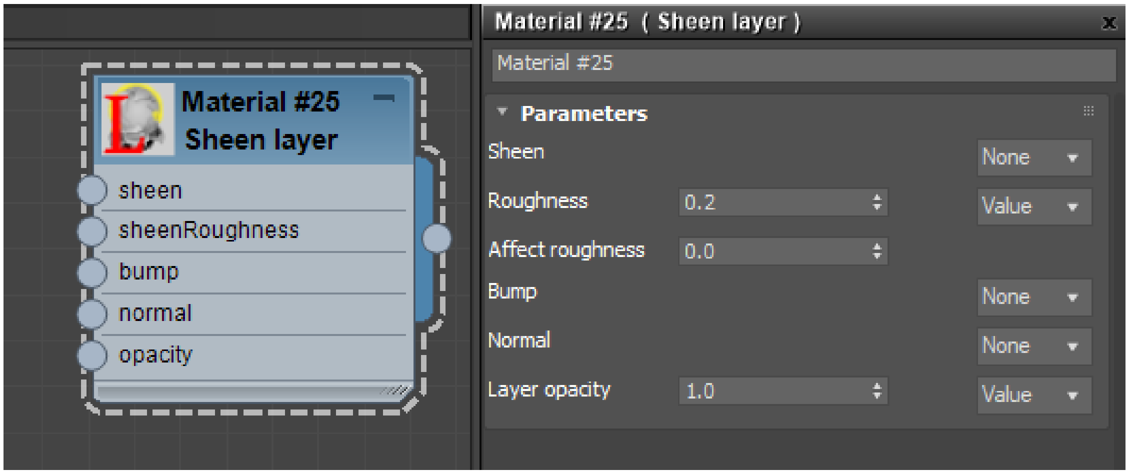


Figure 1: Sheen layer parameters

Sheen Layer Parameters

Sheen - The Sheen layer's color.

Roughness - The Sheen layer's roughness. Higher values spread the sheen color to larger parts of the surface.

Affect roughness - Override the roughness value of the base material with the Sheen roughness value.

Bump - Simulates a relief by using a **Greyscale** texture interpreted as a **Height** map for the layer.

Normal - Distorts Layer normals by using an **RGB** image.

Layer Opacity - Controls the Layer opacity with a Greyscale texture.

Specular Layer

The **Specular**[1] layer is intended for shiny **Materials**[2] like plastics, or clear Materials like glass. Refer to the **Glossy**[3], Specular, and Universal **Material**[4] topics in this manual for more information. Material Layers can connect to the **Layered Material**, **Layer Group**, or **Material Layer** pins on standard Materials.

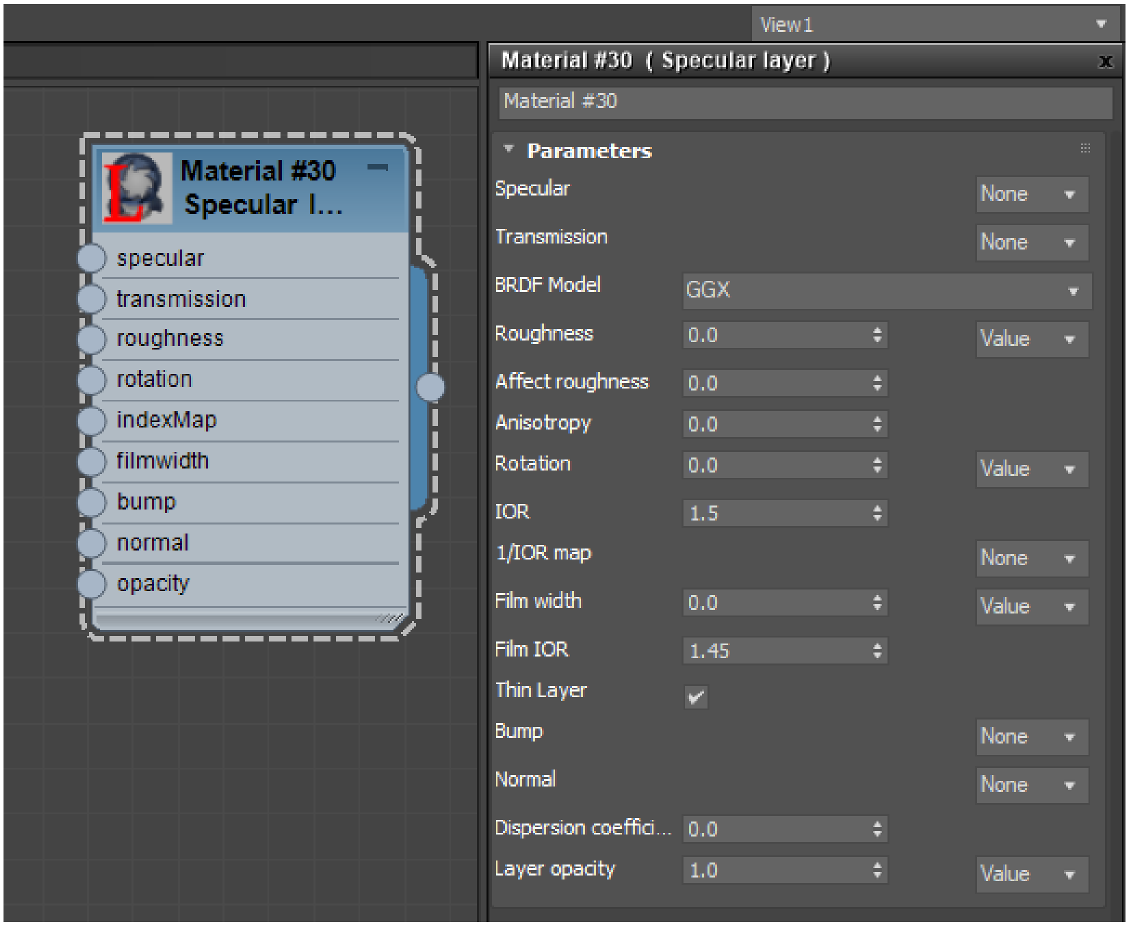


Figure 1: Specular layer parameters

[1] Amount of specular reflection, or the mirror-like reflection of light photons at the same angle. Used for transparent materials such as glass and water.

[2] A set of attributes or parameters that describe surface characteristics.

[3] The measure of how well light is reflected from a surface in the specular direction, the amount and way in which the light is spread around the specular direction, and the change in specular reflection as the specular angle changes. Used for shiny materials such as plastics or metals.

[4] The representation of the surface or volume properties of an object.

Specular Layer Parameters

Specular - The Layer's coating color.

Transmission[1] - The Layer's transmission color.

BRDF Model - Determines how light reflects or refracts. For more information, see the BRDF Models topic in this manual.

Roughness - The layer's roughness.

Affect roughness - Override the roughness value of the base material with the Sheen roughness value.

Anisotropy - The layer's anisotropy. A value of **-1** is horizontal, while **1** is vertical. A value of **0** is isotropic.

Rotation - The specular anisotropic reflection's rotation.

IOR - The specular Reflection's or Transmission's Index Of Refraction.

1/IOR Map - The Index of Refraction map. Each texel represents 1/IOR. When this is empty, OctaneRender® uses the IOR value. If this is not empty, then this parameter overrides the Index Of Refraction set by the IOR value.

Film Width - Sets the film coating's thickness.

Film IOR - Sets the film coating's Index Of Refraction.

Thin Layer - Makes the layer very thin so light reflects or goes straight though the layer.

Bump - Simulates a relief by using a **Greyscale** texture interpreted as a **Height** map for the Layer.

Normal - Distorts Layer normals using an **RGB** image.

Dispersion Coefficient - The B parameter of the Cauchy dispersion model. Increasing this value increases the coloration amount and dispersion in the Layer’s transmission and caustics.

Layer Opacity - Controls the Layer's opacity with a Greyscale texture.

Specular Material

The **Specular**[2] material creates transparent materials like glass and water.

---

[1] A surface characteristic that determines if light may pass through a surface volume.

[2] Amount of specular reflection, or the mirror-like reflection of light photons at the same angle. Used for transparent materials such as glass and water.

Figure 1: The Specular material[1]

[1]Used for transparent materials such as glass and water.

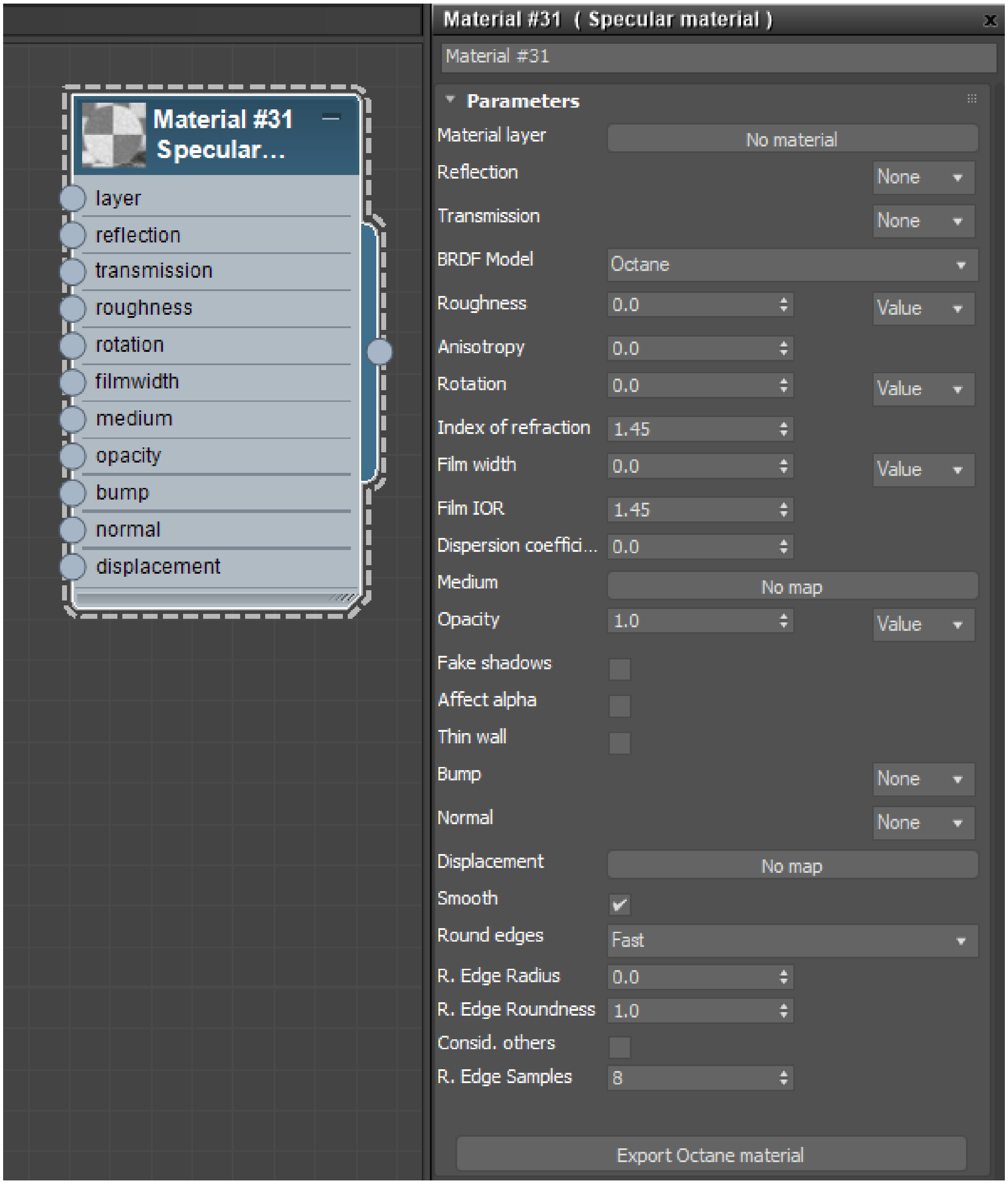


Figure 2: Specular material parameters

Specular Material Parameters

Material[1] Layer - Adds a **Material Layer** above the base material. See the Material Layers topic in this manual for more details.

Reflection - Determines the strength of reflections visible on the surface. Lower values increase its ability to transmit light through the **Object** volume. Reflection is closely tied with the **Index of Refraction** (IOR), and the two parameters work together to tune the Specular material's reflectivity.

Transmission[2] - Controls how light passes through a transparent surface. It works with the Index of Refraction to control the surface's transparency, and it accepts color or **Texture** inputs. A value of **1** means light passes through the surface. To create a mirror surface, set Transmission to **0** and Index of Refraction to **0**. To create colored glass, change the color input to something other than white. Transmission is not the same as **Opacity**. Opacity controls the surface's visibility, while Transmission controls the transparency. Use Transmission to create a reflective glass surface, and use Opacity to create a hole in the surface.

BRDF Model - This attribute provides four options for determining the overall bi-directional reflectance distribution function (BRDF). The **Octane** option produces a more brushed-metal effect. The **Beckmann**, **GGX**, and **Ward** options produce more polished chrome-like effects.

Roughness - Creates microfacets in the surface, which blurs both the surface's reflections and the transparency. One way to create translucent plastic is to make a surface that has a high Transmission value and a Roughness value above **0**. Roughness accepts a color value, or a **Procedural** or **Image** texture (we recommend using a **Grayscale** image). Hue information doesn't affect the roughness.

Anisotropy - Adjusts the amount of change in a surface's reflection, depending on viewing direction.

Rotation - Controls the Anisotropy effect's orientation.

Index Of Refraction - As light photons move through surfaces like water, they slow down and change direction. This shift is visible as object distortion on the other side of the water's surface. The vacuum's index of refraction (IOR) is **1**, and the water's IOR is **1.33**, meaning that light travels 1.33 times faster through a vacuum than it does through water. You can find the IOR of most transparent surfaces by searching the internet. Knowing the correct IOR of a surface is key to replicating the look of the surface when rendering with OctaneRender®.

Film Width - Film Width simulates the look of a thin film of material on the surface. This is useful when you want to create an effect like the rainbow colors that appear on an oil slick's surface. Larger values increase the effect's strength.

Film IOR - Controls the thin film's Index of Refraction. Use this option to adjust the film's visible colors.

Dispersion Coefficient - Increasing the Dispersion value increases the amount of coloration and dispersion in the Object's transmission and caustics.

---

[1] The representation of the surface or volume properties of an object.

[2] A surface characteristic that determines if light may pass through a surface volume.

Medium - OctaneRender® for 3DS Max® has three types of **Mediums**[1]:

- **Absorption**[2] **Medium** - Produces a Material that absorbs light while passing through a surface. The color resulting from this absorption depends on the distance light travels through the Material. The Absorption map type is covered in more detail in the Texture Overview topic in this manual.
- **Scattering**[3] **Medium** - Similar to the Absorption medium, but with an additional option to simulate subsurface scattering. Subsurface scattering is the phenomena that gives human skin, and similar organic surfaces, their characteristic glow under certain lighting conditions. It is a major component in creating the look of realistic skin. The Scattering map type is covered in more detail in the Texture Overview topic in this manual.
- **Volume Medium**[4] - Creates the effect of volumetric surfaces when applied to **VDB**[5] files imported into 3DS Max® using openVDB.
- **Random Walk -** A newer variant of subsurface scattering that utilizes a stochastic or random process for the scattering of light through an object. This provides the most realistic result when rendering scatter volumes.

Opacity - Determines what parts of the surface are visible in the render. Dark values indicate transparent areas, and light values indicate opaque areas. Values in-between light and dark indicate semi-transparent areas. You can lower the Opacity value to fade the object's overall visibility, or you can use a Texture map to vary the opacity across the surface. For example, if you want to make a simple polygon plane look like a leaf, you would connect a black-and-white image of the leaf's silhouette to the **Diffuse**[6] shader's **Opacity** channel.

Fake Shadows - Activates the **Architectural** glass option for all meshes sharing that material. When enabled, Specular materials exhibit Architectural glass characteristics with its transparent feature, allowing light to illuminate enclosed spaces or frame an exterior view.

---

[1]The behavior of light inside a surface volume described by scatter, absorption, and transmission characteristics.

[2]Defines how fast light is absorbed while passing through a medium.

[3]Defines how fast light gets scattered when traveling through the medium.

[4]A shading system designed to render volumes such as smoke and fog.

[5]Dreamworks' open-source C++ library housing the data structures and tools implementation for storing and manipulating volume data, like smoke and other amorphous materials. The purpose of OpenVDB is mostly to have an efficient way to store volumetric data in memory and on disk. It has evolved into a more general toolkit that also lets you accomplish other things, such as fracturing volumes, converting meshes to volumes and vice versa. However, it does not include a computational fluid dynamics solver, and therefore it cannot procedurally generate smoke or fire. OpenVDB is fully integrated as a library in OctaneRender. For more information about OpenVDB, please see http://www.openvdb.org/.

[6]Amount of diffusion, or the reflection of light photons at different angles from an uneven or granular surface. Used for dull, non-reflecting materials or mesh emitters.

Affect Alpha - This option lets refractions affect the **Alpha Channel**[1], as long as you enable the **Alpha Channel** in the **Kernel** settings.

Thin wall - When enabled, the geometry becomes very thin, so the ray bounce exits the material immediately, rather than entering the medium.

Bump - Creates fine details on the Material's surface using a Procedural or Image texture. When you connect a Grayscale texture to this parameter, the Texture's light areas look like protruding bumps, and the dark areas look like indentations. You can adjust the Bump map's strength by adjusting the Image texture's **Power** or **Gamma**[2] values. These attributes are covered in more detail in the Texture Overview topic in this manual.

Normal - Creates fine details on the surface. A Normal map is a special type of Image texture that uses red, green, and blue color values to perturb the surface's normals at render time, thus giving the appearance of added detail. They can be more accurate than Bump maps, but require specific software such as ZBrush®, Mudbox®, Substance Designer, Xnormal™, or others to generate.

Displacement[3] - Adjusts the surface vertices' height at render time using an Image texture map. Displacement maps differs from Bump or Normal maps in that the geometry is altered by the Texture, as opposed to creating details. Displacement mapping is more complex than using a Bump or Normal map, but the results are more realistic, in particular along the surface's silhouette. Displacement mapping is covered in more detail under the Texture Overview topic in this manual.

Smooth - Smooths the transition between surface normals. If this option is disabled, the edges between the polygons of the surface are sharp, giving the surface a faceted look.

Round Edges - Rounds off geometry edges by using a shading effect instead of creating additional geometry. It's best used for rounded edges that will appear small in the final render. The **Fast** mode uses the rounding method introduced in OctaneRender® v3. The **Accurate** mode produces better-looking results, but may be slower. Accurate mode can select the affected edges by using the **Concave Only** or **Convex Only** options.

Rounded Edges Radius - Define the size of the rounded edge by radius. Bevels the surface edges at render time without altering or subdividing the geometry. Using this option enhances object realism by eliminating sharp edges. The value refers to the rounded edge's radius. Higher values produce rounder edges.

Rounded Edges Roundness - Controls the rounded edge's shape. A value of 1 is completely round, while 0 is a chamfer.

---

[1] A greyscale image used to determine which areas of a texture map are opaque and which areas are transparent.

[2] The function or attribute used to code or decode luminance for common displays. The computer graphics industry has set a standard gamma setting of 2.2 making it the most common default for 3D modelling and rendering applications.

[3] The process of utilizing a 2D texture map to generate 3D surface relief. As opposed to bump and normal mapping, Displacement mapping does not only provide the illusion of depth but it effectively displaces the actual geometric position of points over the textured surface.

Consider Other Objects - Controls how rounded edges are applied to different objects. When enabled, intersections between different objects are rounded. When disabled, only the current object is considered.

Rounded Edge Samples - Set samples for rendering edges.

Export Octane Material - Opens a dialog window that provides options for exporting the Material to **ORBX**[1], the Local DB, or the Live DB.

Toon Material

The **Toon** material can design non-photorealistic renderings that have hand-drawn characteristics. You can use it in conjunction with the **Toon Ramp** texture connected to any of the ramp attributes to design more detailed, toon-like **Material**[2] effects.

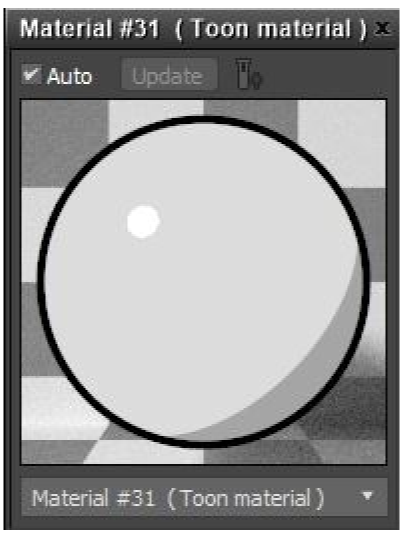


Figure 1: The Toon material

[1]The ORBX file format is the best way to transfer scene files from 3D Authoring software programs that use the Octane Plug-in such as Octane for Maya, Octane for Cinema 4D, or OctaneRender Standalone. This format is more efficient than FBX when working with Octane specific data as it provides a flexible, application independent format. ORBX is a container format that includes all animation data, models, textures etc. that is needed to transfer an Octane scene from one application to another.
[2]The representation of the surface or volume properties of an object.

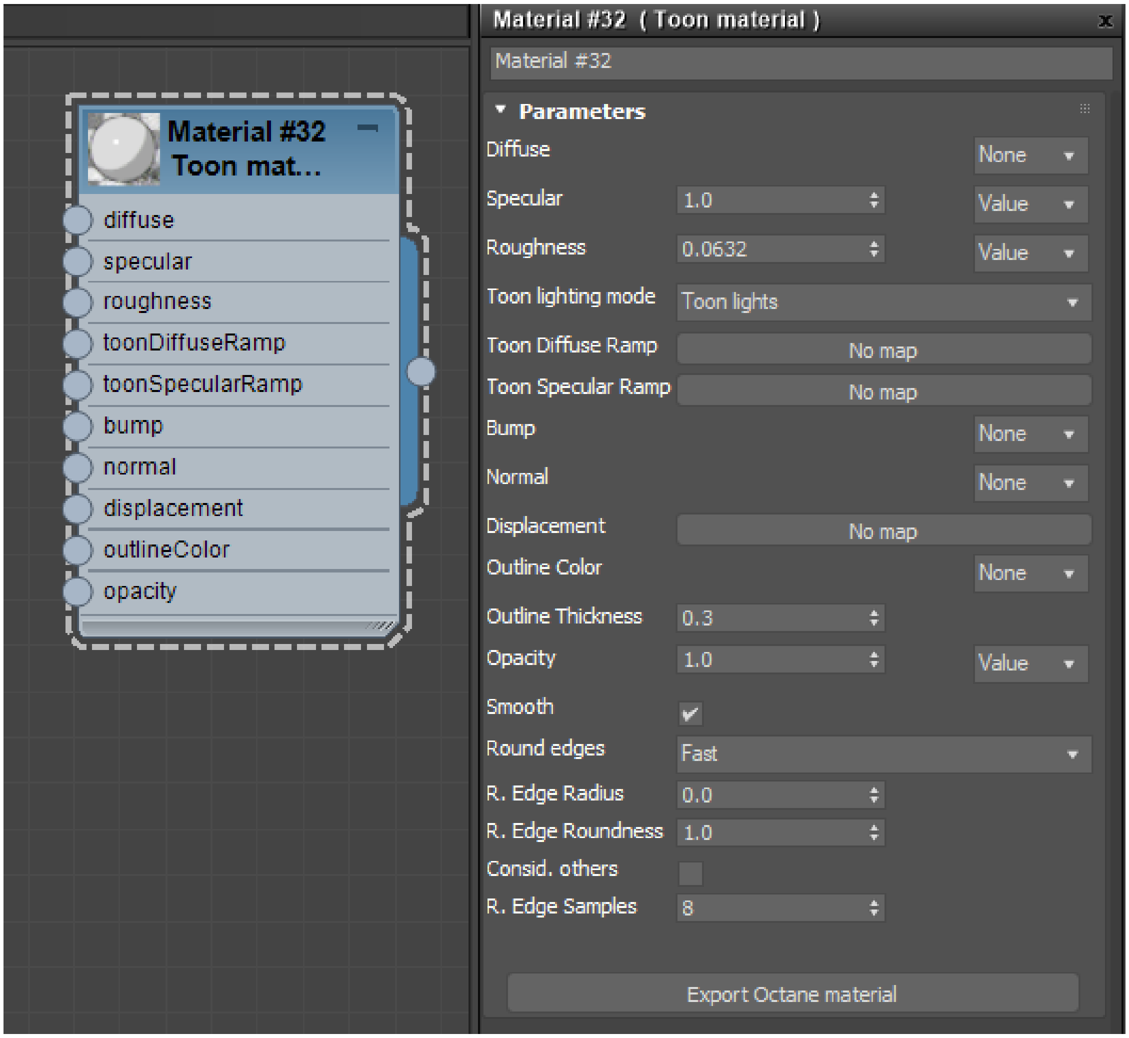


Figure 2: Toon Material parameters

Toon Material Parameters

Diffuse[1] - The Diffuse reflection channel, or the albedo value of the Toon shader.

---

[1]Amount of diffusion, or the reflection of light photons at different angles from an uneven or granular surface. Used for dull, non-reflecting materials or mesh emitters.

Specular[1] - The Specular reflection channel, which behaves like a coating on top of the Diffuse layer and creates a highlight on the surface depending on the incident light angle and the camera's viewpoint. A value of **0** means there is no highlight at all.

Roughness - The Specular reflection channel's roughness. The appearance of the Toon shading's Specular reflection becomes more prevalent as the roughness of the Specular reflection channels decreases.

Toon Lighting Mode - Since Toon Lighting is required for Toon materials to work, this attribute defines where the Toon lighting is drawn from. This can be from the camera direction, or from OctaneRender® **Toon Lights**. If Toon Lights is the selected mode, Toon materials will need either a **Toon** point light or a **Toon** directional light included in the scene in order to work.

Toon Diffuse Ramp - The color/float range that defines how the Toon shading's albedo value (or diffuse color) varies over a surface.

Toon Specular Ramp - The color/float range that defines how the Toon shading's Specular value varies over a surface.

Bump - Simulates a relief using a **Grayscale** texture interpreted as a height map.

Normal - Distorts normals based on an **RGB** image.

Displacement[2] - Creates very detailed geometry with a low memory footprint.

Outline Color - The color used for the surface's outline and contour edges.

Outline Thickness - Defines and propagates the outline and contour edges used in the Toon shading. A thickness of **0.0** means there is no outline for that surface.

Opacity - Controls the Toon material transparency with a Grayscale texture.

Smooth - Enables normal interpolation. If disabled, triangle meshes will appear faceted.

Round Edges - Rounds off geometry edges by using a shading effect instead of creating additional geometry. It's best used for rounded edges that will appear small in the final render. The **Fast** mode uses the rounding method introduced in OctaneRender® v3. The **Accurate** mode produces better-looking results, but may be slower. Accurate mode can select the affected edges by using the **Concave Only** or **Convex Only** options.

Rounded Edges Radius - Define the size of the rounded edge by radius. Bevels the surface edges at render time without altering or subdividing the geometry. Using this option enhances object realism by eliminating sharp edges. The value refers to the rounded edge's radius. Higher values produce rounder edges.

Rounded Edges Roundness - Controls the rounded edge's shape. A value of 1 is completely round, while 0 is a chamfer.

---

[1] Amount of specular reflection, or the mirror-like reflection of light photons at the same angle. Used for transparent materials such as glass and water.

[2] The process of utilizing a 2D texture map to generate 3D surface relief. As opposed to bump and normal mapping, Displacement mapping does not only provide the illusion of depth but it effectively displaces the actual geometric position of points over the textured surface.

Consider Other Objects - Controls how rounded edges are applied to different objects. When enabled, intersections between different objects are rounded. When disabled, only the current object is considered.

Rounded Edge Samples - Set samples for rendering edges.

Export Octane Material - Opens a dialog window that provides options for exporting the Material to **ORBX**[1], the Local DB, or the Live DB.

Universal Material

The **Universal** material puts **Substance** maps and **PBR**[2] outputs into OctaneRender®. Substance Painter and other engines map well to this material.

Universal materials blend between dielectric and metallic with a **Metallic** parameter value from **0** - **1**. Compared to other materials, the Universal material is equivalent to the **Metallic** material when its Metallic parameter is set to **1.0**, and it is similar to the **Glossy**[3] material when its Metallic parameter is set to **0.0**.

The Universal material is designed to follow after the workflow in the PBR model, since the Metallic material falls short of the **Metallic** and **Roughness** maps that are often derived from Substance Painter and other tools. It handles dielectric material (**Diffuse**[4] and Glossy BRDF) and also Metallic material (Glossy BRDF) with assumed **IOR** or custom IOR for both dielectric and metallic surfaces.

Material[5] IOR in the base layer of Universal materials is also not limited to scalar values, and this can be controlled procedurally with texture-type nodes and OSL shaders connected to a new **IOR** texture input pin.

---

[1] The ORBX file format is the best way to transfer scene files from 3D Authoring software programs that use the Octane Plug-in such as Octane for Maya, Octane for Cinema 4D, or OctaneRender Standalone. This format is more efficient than FBX when working with Octane specific data as it provides a flexible, application independent format. ORBX is a container format that includes all animation data, models, textures etc. that is needed to transfer an Octane scene from one application to another.

[2] A contemporary shading and rendering process that seeks to simplify shading characteristics while providing a more accurate representation of lighting in the real world.

[3] The measure of how well light is reflected from a surface in the specular direction, the amount and way in which the light is spread around the specular direction, and the change in specular reflection as the specular angle changes. Used for shiny materials such as plastics or metals.

[4] Amount of diffusion, or the reflection of light photons at different angles from an uneven or granular surface. Used for dull, non-reflecting materials or mesh emitters.

[5] The representation of the surface or volume properties of an object.

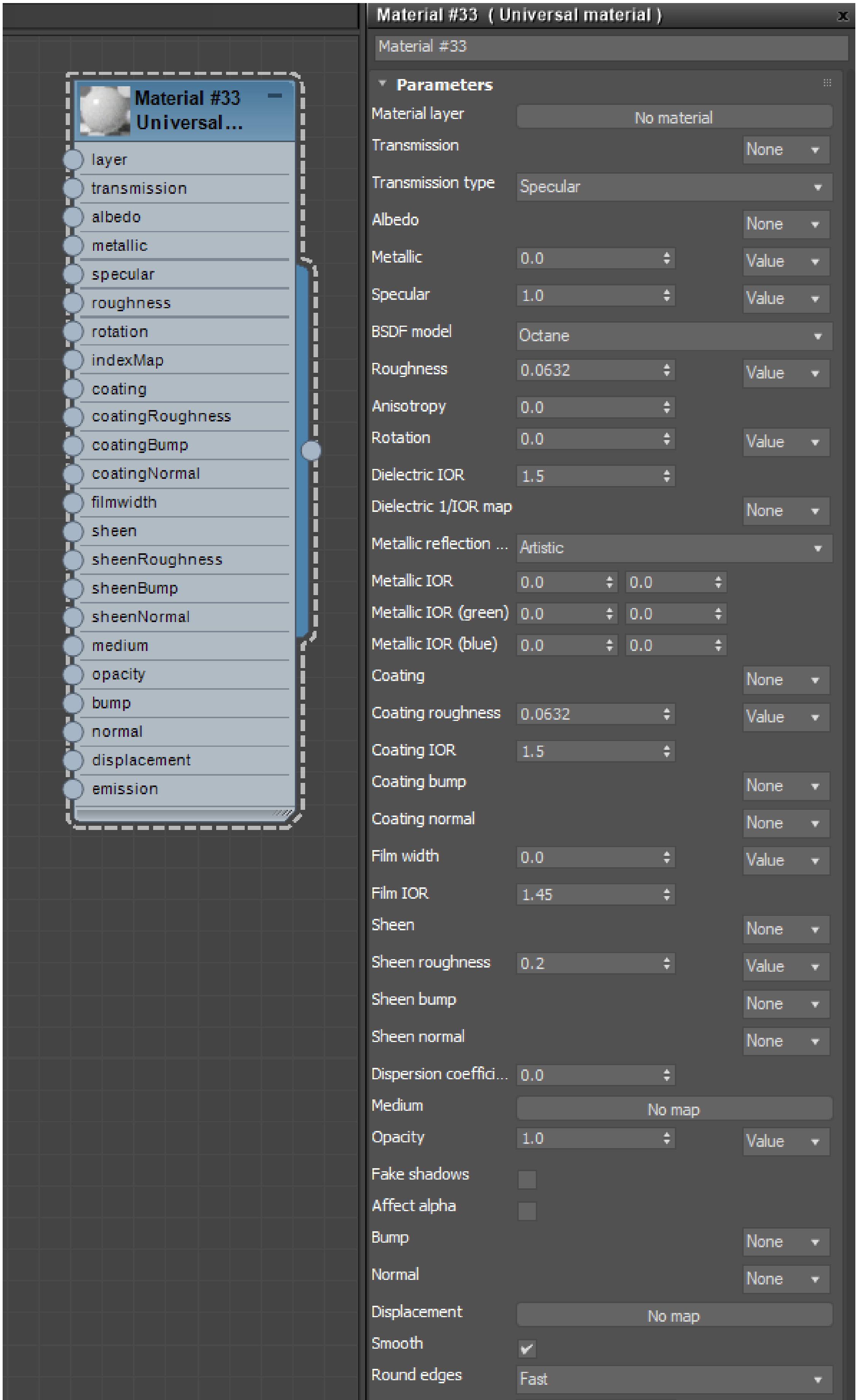

Material #33 (Universal material)
Material #33
Universal...
layer
transmission
albedo
metallic
specular
roughness
rotation
indexMap
coating
coatingRoughness
coatingBump
coatingNormal
filmwidth
sheen
sheenRoughness
sheenBump
sheenNormal
medium
opacity
bump
normal
displacement
emission
Parameters
Material layer No material
Transmission None
Transmission type Specular
Albedo None
Metallic 0.0 Value
Specular 1.0 Value
BSDF model Octane
Roughness 0.0632 Value
Anisotropy 0.0
Rotation 0.0 Value
Dielectric IOR 1.5
Dielectric 1/IOR map None
Metallic reflection ... Artistic
Metallic IOR 0.0 0.0
Metallic IOR (green) 0.0 0.0
Metallic IOR (blue) 0.0 0.0
Coating None
Coating roughness 0.0632 Value
Coating IOR 1.5
Coating bump None
Coating normal None
Film width 0.0 Value
Film IOR 1.45
Sheen None
Sheen roughness 0.2 Value
Sheen bump None
Sheen normal None
Dispersion coeffici... 0.0
Medium No map
Opacity 1.0 Value
Fake shadows
Affect alpha
Bump None
Normal None
Displacement No map
Smooth
Round edges Fast

Figure 1: The Universal material parameters

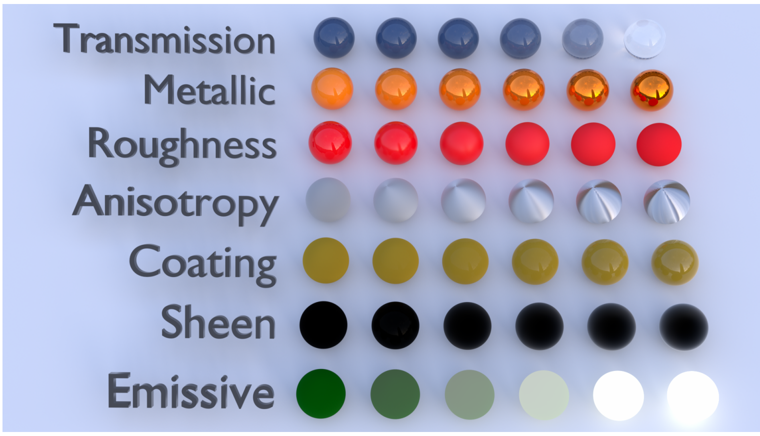


Figure 2: Creating basic and complex materials with Universal materials

You can also import the BaseColor maps, Height maps, Normal maps, Occlusion maps, and other texture maps for a scene derived from major 3D painting software into OctaneRender®, and then re-link these texture maps to the corresponding Universal material node pins. The Universal material node blends the Glossy and Metallic materials, depending on the Metallic input settings. You can then adjust each texture's settings in greater detail. For example, you can place real-world IOR values of Metallic objects as part of the Universal material's Red, Green, and Blue IOR metallic input channels.

Universal Material Parameters

Material Layer - Adds a material layer above the base layer. See the Material Layers topic in this manual for more details.

Transmission Layer

Transmission[1] - Controls how light passes through a transparent surface. It is tied with the **Dielectric IOR** parameter to control surface transparency. It accepts a **Color**, **Texture**, or value as an input, but Color input

---

[1] A surface characteristic that determines if light may pass through a surface volume.

provides the most control. To create colored glass, change the Color input to something other than black and set the **Albedo** to black. Transmission is not the same as **Opacity**. Opacity controls the visibility of the surface, while Transmission controls the transparency. Use Transmission to create a reflective glass surface, and use Opacity to create a hole in the surface.

Transmission type - Determines how light refracts.

Base Layer

Albedo - The material's base color.

Metallic - The material's metallic appearance. Blends between dielectric and metallic material.

Specular Layer

Specular[1] - Determines the color of glossy reflections for Dielectric materials (when the Metallic parameter is set to **0**). Set the **Dielectric IOR** parameter higher than **1.0** for the Specular parameter to contribute to the surface characteristics.

BSDF Model - There are three options for determining the overall bi-directional scattering distribution function (BSDF). The **Octane** option produces a more brushed-metal effect. The **Beckmann** and **GGX** options produce more polished, chrome-like effects.

Roughness

Roughness - Determines how much the Specular and Transmission characteristics spread across the surface.

Anisotropy - Determines the shape of the Specular and Transmission highlights. A value of **-1** creates a horizontal shape, and a value of **1** creates a vertical shape.

Rotation - Controls the Anisotropy shape's rotation.

---

[1] Amount of specular reflection, or the mirror-like reflection of light photons at the same angle. Used for transparent materials such as glass and water.

IOR

Dielectric IOR - This is the IOR that controls the Fresnel effect of the specular reflection or transmission. By default, if the **Dielectric 1/IOR** parameter is empty, then the dielectric specular uses this IOR parameter instead.

Dielectric 1/IOR - Overrides the Dielectric IOR when a map or value is applied. This parameter is an index of refraction map, where each texel represents 1/IOR.

Metallic Refl. Mode - This attribute, along with the IOR attributes, provide options for controlling the Index of Refraction across a surface.

> **Artistic** - Uses the Albedo color.
>
> **IOR + Color** - Uses the Albedo color and further adjusts the surface brightness by using the IOR.
>
> **RBG IOR** - This is the most used mode. It uses the just the three IOR values and ignores the Albedo color altogether.

Metallic IOR - Complex-valued IOR (n-k*i) that controls the Fresnel effect of the specular reflection for Metallic materials. For RGB IOR mode, this serves as the IOR for the red light (650nm).

Metallic IOR (Green) - For RGB IOR mode, this is the IOR for the green light (550nm).

Metallic IOR (Blue) - For RGB IOR mode, this is the IOR for the blue light (450nm).

Coating Layer

Coating - Adds a second layer of reflection to the surface.

Coating Roughness - Determines how much the Coating characteristic spreads across the surface.

Coating IOR - Controls the Fresnel effect for the Coating characteristics of the surface.

Coating Bump - Much like a regular **Bump** map, this creates fine details on the Material's Coating attribute using a **Procedural** or **Image** texture.

Coating Normal - Creates fine details on the surface's coating. However, a **Normal** map is a special type of **Image** texture that uses red, green, and blue color values to perturb the surface's normals at render time, thus giving added detail. They can be more accurate than **Bump** maps, but require specific 3D software to generate.

Thin Film Layer

Film Width - Simulates the look of a thin film of material on the surface. This is useful when you want to create an effect like the rainbow colors that appear on an oil slick's surface. Larger values increase the effect's strength.

Film IOR - Controls the thin film's IOR and its visible colors.

Sheen Layer

Sheen - Adds a second layer of glossiness to the surface.

Sheen Roughness - Determines how much the Sheen characteristic spreads across the surface.

Sheen Bump - Much like a regular Bump map, this creates fine details in the Material’s Sheen attribute using a **Procedural** or **Image** texture.

Sheen Normal - This attribute also creates fine details in the surface's sheen. However, a Normal map is a special type of Image texture that uses red, green, and blue color values to perturb the surface normals at render time, thus giving the appearance of added detail. They can be more accurate than Bump maps, but require specific 3D software to generate.

Transmission Properties

Dispersion Coefficient - Increasing the **Dispersion** value increases the amount of coloration and dispersion in the Object’s transmission and in caustics.

Medium - OctaneRender® for 3DS Max® has three types of **Mediums**[1] to create translucent surfaces: **Absorption**[2], **Scattering**[3], and **Volume**. To use these options, connect the **Medium** input of the **Diffuse** material to either the **Absorption** or **Scattering** medium nodes.

> **Absorption Medium** - Produces Material that absorbs light while passing through a surface. The color resulting from this Absorption depends on the distance light travels through the Material. The Absorption map type is covered in more detail in the Texture Overview topic of this manual.
>
> **Scattering Medium** - Similar to the Absorption medium, but with an additional option to simulate subsurface scattering. Subsurface scattering is the phenomena that gives human skin and similar organic surfaces their characteristic glow under certain lighting conditions. It is a major component in creating the look of realistic skin. The Scattering map type is covered in more detail in the Texture Overview topic of this manual.

---

[1]The behavior of light inside a surface volume described by scatter, absorption, and transmission characteristics.

[2]Defines how fast light is absorbed while passing through a medium.

[3]Defines how fast light gets scattered when traveling through the medium.

Volume Medium[1] - Creates volumetric surfaces when applied to **VDB**[2] files imported into 3DS Max® using openVDB.

Random Walk - Aa newer variant of subsurface scattering that utilizes a stochastic or random process for the scattering of light through an object. This provides the most realistic result when rendering scatter volumes.

Opacity - Determines what parts of the surface are visible in the render. Dark values indicate transparent areas, and light values determine opaque areas. Values in-between light and dark indicate semi-transparent areas. You can lower the Opacity value to fade the Object's overall visibility, or you can use a Texture map to vary the opacity across the surface. For example, to make a simple polygon plane look like a leaf, connect a black-and-white image of the leaf's silhouette to the **Diffuse** shader's **Opacity** channel.

Fake Shadows - If enabled, light traces through the Material during the shadow calculation, ignoring refraction.

Affect Alpha - Enabling this option causes refraction to affect the **Alpha Channel**[3]. This parameter has an effect if the Alpha Channel is enabled in the Render settings' **Kernel** parameters.

Geometric Properties

Bump - Creates fine details on the Material's surface using a Procedural or Image texture. When you connect a Grayscale texture to this parameter, light areas of the texture indicate protruding bumps, and dark areas indicate indentation. You can adjust the Bump map's strength by setting the **Power** or **Gamma**[4] values on the **Image** texture node. These attributes are covered in more detail in the Texture Overview topic of this manual.

---

[1] A shading system designed to render volumes such as smoke and fog.

[2] Dreamworks' open-source C++ library housing the data structures and tools implementation for storing and manipulating volume data, like smoke and other amorphous materials. The purpose of OpenVDB is mostly to have an efficient way to store volumetric data in memory and on disk. It has evolved into a more general toolkit that also lets you accomplish other things, such as fracturing volumes, converting meshes to volumes and vice versa. However, it does not include a computational fluid dynamics solver, and therefore it cannot procedurally generate smoke or fire. OpenVDB is fully integrated as a library in OctaneRender. For more information about OpenVDB, please see http://www.openvdb.org/.

[3] A greyscale image used to determine which areas of a texture map are opaque and which areas are transparent.

[4] The function or attribute used to code or decode luminance for common displays. The computer graphics industry has set a standard gamma setting of 2.2 making it the most common default for 3D modelling and rendering applications.

Normal - Also creates fine detail on the surface. A Normal map is a special type of Image texture that uses red, green, and blue color values to perturb the surface normals at render time, giving the appearance of added detail. They can be more accurate than Bump maps, but require specific 3D software to generate.

Displacement[1] - Adjusts the height of the vertices of a surface at render time using a Texture map. Displacement maps differs from Bump or Normal maps in that the geometry is altered by the Texture, as opposed to creating the appearance of detail. Displacement mapping is more complex than using a Bump or Normal map, but the results can be more realistic, in particular along the surface's silhouette. Displacement mapping is covered in more detail under the Texture Overview category.

Smooth - Smooths out the transition between surface normals. If this option is disabled, the edges between the polygons of the surface are sharp, giving the surface a faceted look.

Round Edges - Rounds off geometry edges by using a shading effect instead of creating additional geometry. It’s best used for rounded edges that will appear small in the final render. The **Fast** mode uses the rounding method introduced in OctaneRender® v3. The **Accurate** mode produces better-looking results, but may be slower. Accurate mode can select the affected edges by using the **Concave Only** or **Convex Only** options.

Rounded Edges Radius - Define the size of the rounded edge by radius. Bevels the surface edges at render time without altering or subdividing the geometry. Using this option enhances object realism by eliminating sharp edges. The value refers to the rounded edge's radius. Higher values produce rounder edges.

Rounded Edges Roundness - Controls the rounded edge's shape. A value of 1 is completely round, while 0 is a chamfer.

Consider Other Objects - Controls how rounded edges are applied to different objects. When enabled, intersections between different objects are rounded. When disabled, only the current object is considered.

Rounded Edge Samples - Set samples for rendering edges.

Miscellaneous Properties

Emission - Creates a surface that emits light (also known as a **Mesh** emitter). To use this option, connect the **Emission** input of the **Diffuse** material to either a **Blackbody** or **Texture** emission node. These nodes are covered in more detail in the Texture Overview topic, and in **Mesh Emitters**[2] under the Lighting Overview category in this manual.

---

[1] The process of utilizing a 2D texture map to generate 3D surface relief. As opposed to bump and normal mapping, Displacement mapping does not only provide the illusion of depth but it effectively displaces the actual geometric position of points over the textured surface.

[2] The ability for a surface to emit illumination usually described by a Black Body or Texture emission type.

Shadow Catcher[1] - This converts the material into a shadow catcher. When it is active, the surface is visible in the areas that are in shadow, and all other areas are transparent in the render.

Export Octane Material - Opens a dialog window with options for exporting the material to **ORBX**[2], the Local DB, or the Live DB.

Textures

Textures[3] create flexible **Materials**[4]. This section details all the Texture nodes available in the **Maps** rollout in the **Slate Material**[5] **Editor**.

Texture types are classified into the following:

- **Texture Generators** - These generate colors, import **Texture** maps, and create **Procedural** textures similar to specific patterns used in the real world.
- **Texture Modifiers** - These nodes modify existing **Texture** nodes in some way, such as color correction or inverting.
- **Transforms** - These nodes control Texture placement on an Object's surface.
- **Projections**[6] - These specify how the **Procedural** textures or **Image** textures are mapped across a surface.
- **Emissions**[7] - This set of nodes provides illumination functionality for an Object's surface.

Material Texture Interface

Material parameters with Texture map capabilities have a dropbox with the following options.

---

[1]The Shadow Catcher can be used to create shadows cast by objects onto the surrounding background imagery. The shadows cast are not limited to simply a ground plane but can be cast onto other surfaces of varying shapes.

[2]The ORBX file format is the best way to transfer scene files from 3D Authoring software programs that use the Octane Plug-in such as Octane for Maya, Octane for Cinema 4D, or OctaneRender Standalone. This format is more efficient than FBX when working with Octane specific data as it provides a flexible, application independent format. ORBX is a container format that includes all animation data, models, textures etc. that is needed to transfer an Octane scene from one application to another.

[3]Textures are used to add details to a surface. Textures can be procedural or imported raster files.

[4]A set of attributes or parameters that describe surface characteristics.

[5]The representation of the surface or volume properties of an object.

[6]Methods for orienting 2D texture maps onto 3D surfaces.

[7]The process by which a Black body or Texture is used to emit light from a surface.

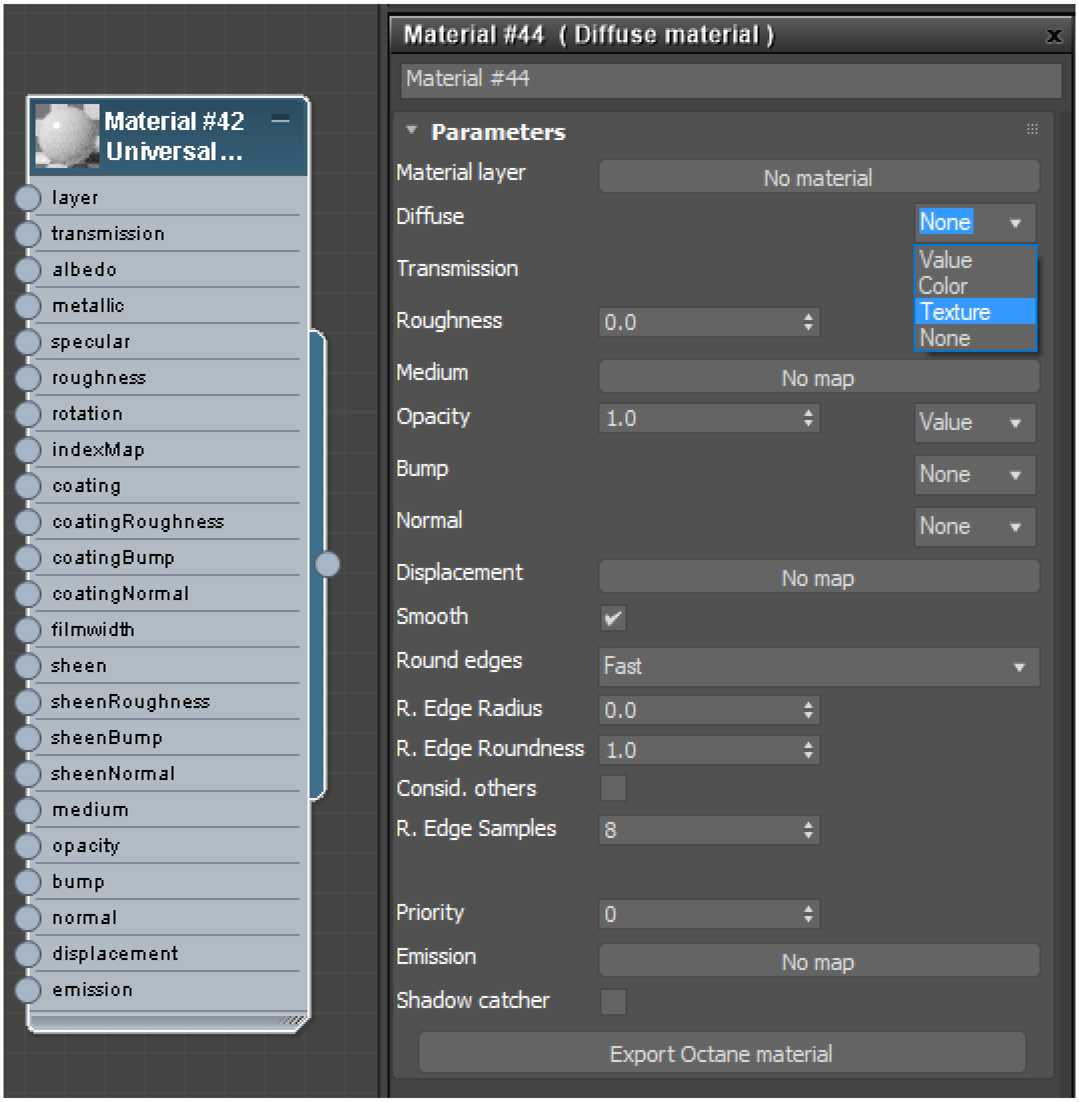


Figure 1: Textures connect by using the dropdown menu in the Material parameters

Displacement Vertex Mixer

The **Vertex Displacement**[1] **Mixer** mixes multiple **Vertex Displacement** nodes.

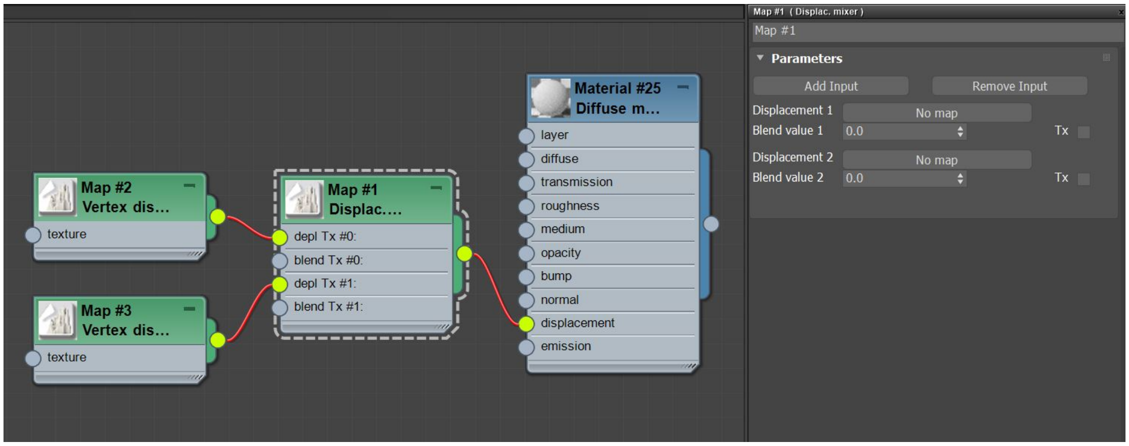


Figure 1: Two Vertex Displacement nodes blended together using the Displacement Vertex Mixer node

Vertex Displacement Mixer Parameters

Add Input - Add a new **Displacement** input to the end of the **Node**.

Remove Input - Removes the last Displacement input.

Displacement 1 - Connects a Vertex Displacement node.

Blend Weight 1 - Controls the mix amount for the connected Vertex Displacement node.

Vertex Displacement

Vertex Displacement[2] is a more robust displacement system that does not suffer from the same limitations as **Texture Displacement**. It works with all **Textures**[3] and **Projections**[4], including **Procedurals**, **OSL**

[1]The process of utilizing a 2D texture map to generate 3D surface relief. As opposed to bump and normal mapping, Displacement mapping does not only provide the illusion of depth but it effectively displaces the actual geometric position of points over the textured surface.
[2]The process of utilizing a 2D texture map to generate 3D surface relief. As opposed to bump and normal mapping, Displacement mapping does not only provide the illusion of depth but it effectively displaces the actual geometric position of points over the textured surface.
[3]Textures are used to add details to a surface. Textures can be procedural or imported raster files.
[4]Methods for orienting 2D texture maps onto 3D surfaces.

textures, and **Images**. **Height** maps and **Vector Displacement** maps are also supported, and you can mix them by using the Vertex Displacement mixer node.

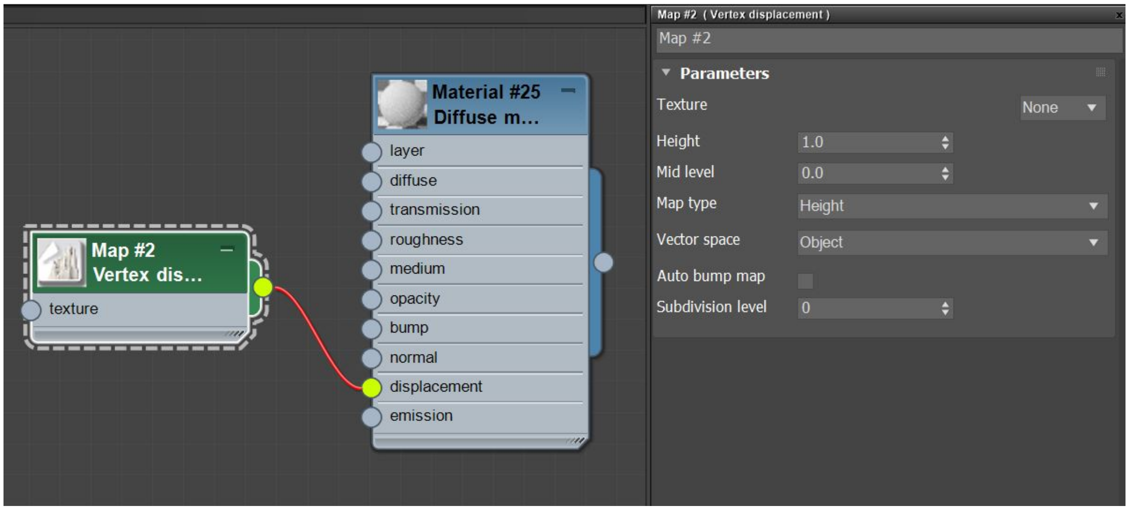


***Figure 1: A Vertex Displacement node connected to the Displacement input on a Diffuse material*[1]**

Vertex Displacement Parameters

Texture - All **Texture** types are supported, including Images, Procedurals, and OSL textures.

Height - The displacement height in meters.

Mid Level - The image value that corresponds to no displacement. The range is always normalized to [0,1]. Set this value to **0.5** for Image textures that use 50% to represent no displacement.

Map Type - Choose between **Height** maps and **Vector Displacement** maps.

Vector Space - Valid when Map Type is set to **Vector Displacement**.

Auto Bump Map - Generates an **Automatic** bump map to achieve fine details without requiring high subdivision levels. Only supports Height displacement maps.

Subdivision Level - This refers to the subdivision level applied to **Polygons** using this **Material**[2]. It overrides the subdivision level set in **Geometry** preferences. Higher values achieve greater displacement detail, but can also increase rendering and pre-processing times.

---

[1] Used for dull, non-reflecting materials or mesh emitters.

[2] The representation of the surface or volume properties of an object.

Texture Generators

Texture Generators provide a set of **Nodes** in the **Slate Material**[1] **Editor** that create solid colors, gradients, **Procedural** textures, and import external **Texture** maps. You can access these Nodes from the **Maps** rollout in the Slate Material Editor.

[1] The representation of the surface or volume properties of an object.

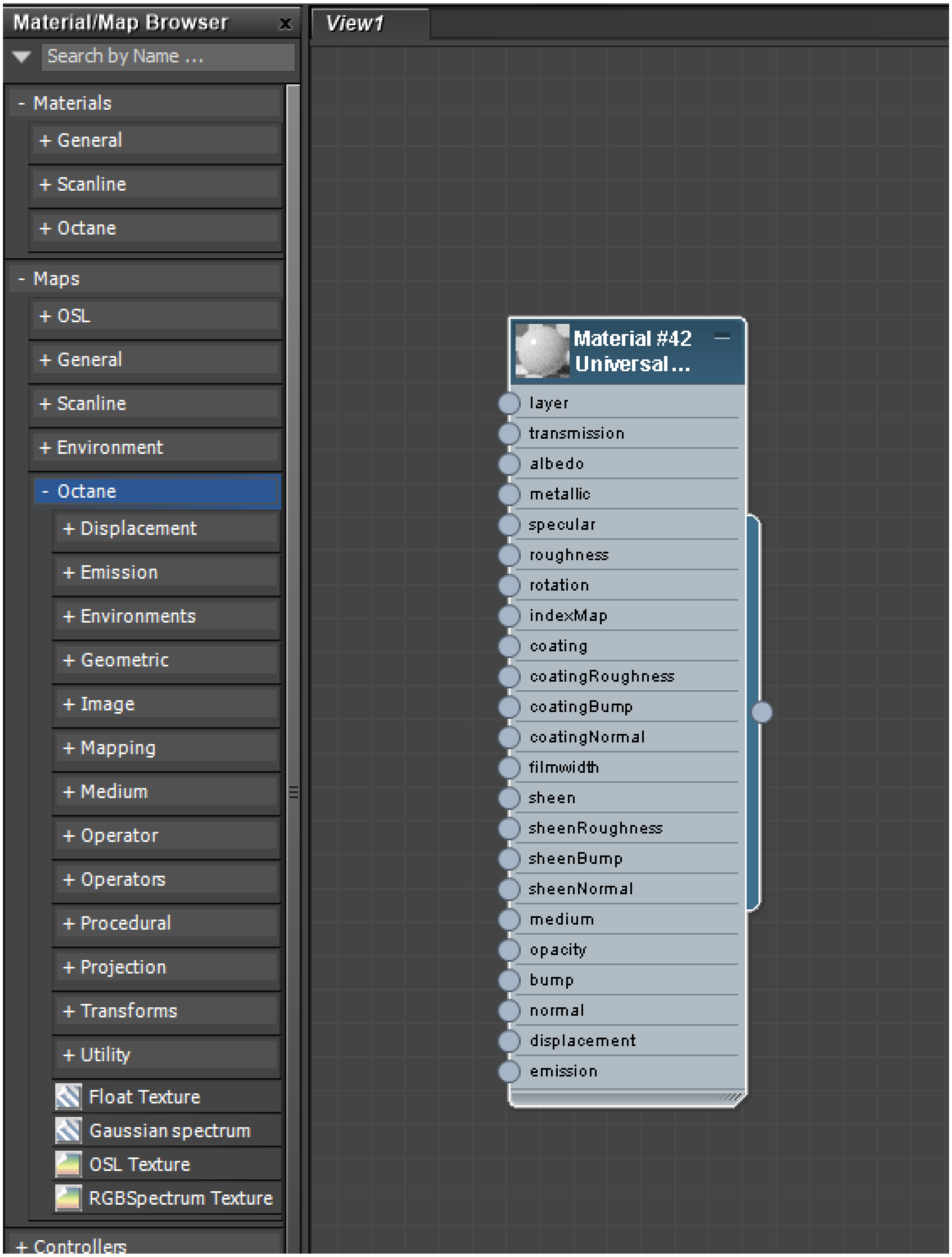

Material/Map Browser
Search by Name ...
- Materials
+ General
+ Scanline
+ Octane
- Maps
+ OSL
+ General
+ Scanline
+ Environment
- Octane
+ Displacement
+ Emission
+ Environments
+ Geometric
+ Image
+ Mapping
+ Medium
+ Operator
+ Operators
+ Procedural
+ Projection
+ Transforms
+ Utility
Float Texture
Gaussian spectrum
OSL Texture
RGBSpectrum Texture
+ Controllers
View1
Material #42
Universal ...
layer
transmission
albedo
metallic
specular
roughness
rotation
indexMap
coating
coatingRoughness
coatingBump
coatingNormal
filmwidth
sheen
sheenRoughness
sheenBump
sheenNormal
medium
opacity
bump
normal
displacement
emission

Figure 1: Accessing OctaneRender® Map nodes in the Slate Material Editor

Alpha Image

The **Alpha Image** utilizes an Image's native Alpha channel to provide transparency. This type accepts PNG, TIF, and **EXR**[1] image types.

---

[1] Also known as OpenEXR. This image file format was developed by Industrial Light & Magic and provides a High Dynamic Range image capable of storing deep image data on a frame-by-frame basis.

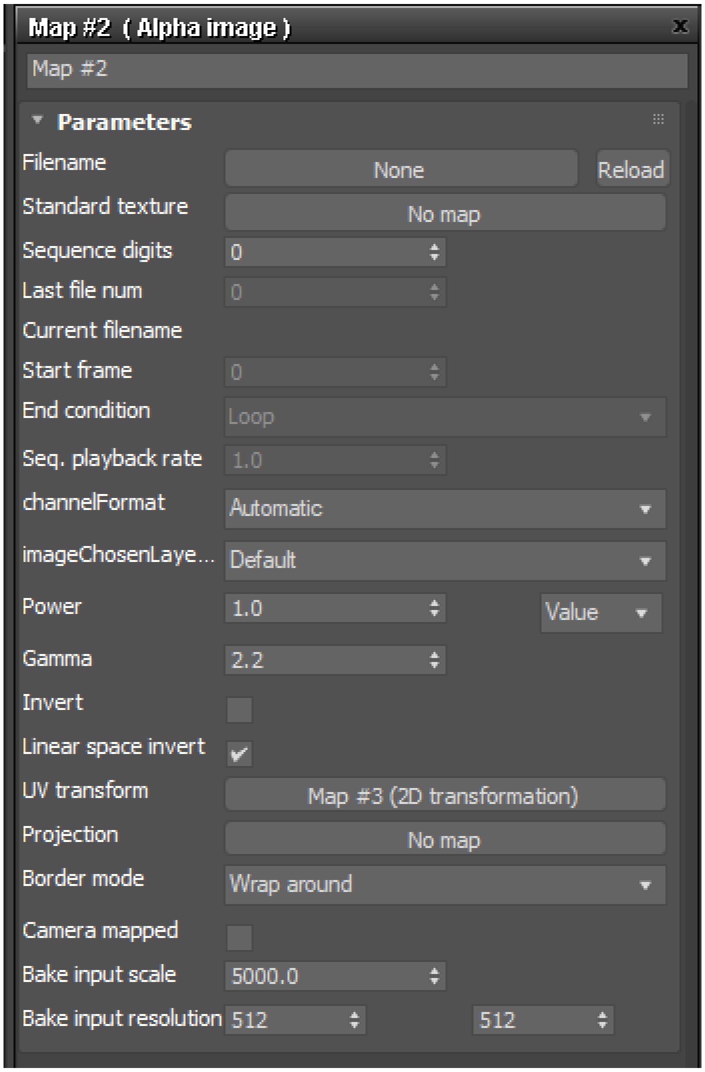


Figure 1: Alpha Image parameters

Alpha Image Parameters

Power - Controls image brightness. Lowering the value makes the image look darker.

Gamma[1] - Controls input image luminance, and tunes or color-corrects images if needed.

Invert - Inverts the texture values.

UV Transform - Positions, rotates, and scales the surface texture.

Projection - Accepts OctaneRender® **Projection** nodes. If nothing is connected to this input, the Image texture uses the surface's **UV** texture coordinates by default. This also changes the UV set if the original surface contains more than one UV set. For more details, see the Octane **Projections**[2] topic in this manual.

Border Mode - Sets the behavior of the space around the image if it doesn't cover the entire geometry. **Wrap Around** is the default behavior, which repeats the image in the areas outside the image's coverage. If you set this parameter to **White Color** or **Black Color**, the area outside the image turns to white or black, respectively.

Baking Texture

The **Baking Texture** node converts any procedural texture to image texture (Figure 1). For example, if you want to use **Displacement**[3] in Octane, this can only be done with an image texture. With this node, however, it is possible to convert the procedural texture to the image texture and use it in the displacement channel.

The baking process uses the texture preview system, which looks like an **Image** texture to the rest of the system. The baking is done whenever you change an input, and it is calculated on-the-fly. The internal image is not stored in the project, so it needs recalculating whenever you load the project.

The Baking texture takes an input from any Procedural texture map and then connects it to a Displacement node, which connects to the **Displacement** pin on an OctaneRender® **Material**[4].

---

[1] The function or attribute used to code or decode luminance for common displays. The computer graphics industry has set a standard gamma setting of 2.2 making it the most common default for 3D modelling and rendering applications.

[2] Methods for orienting 2D texture maps onto 3D surfaces.

[3] The process of utilizing a 2D texture map to generate 3D surface relief. As opposed to bump and normal mapping, Displacement mapping does not only provide the illusion of depth but it effectively displaces the actual geometric position of points over the textured surface.

[4] The representation of the surface or volume properties of an object.

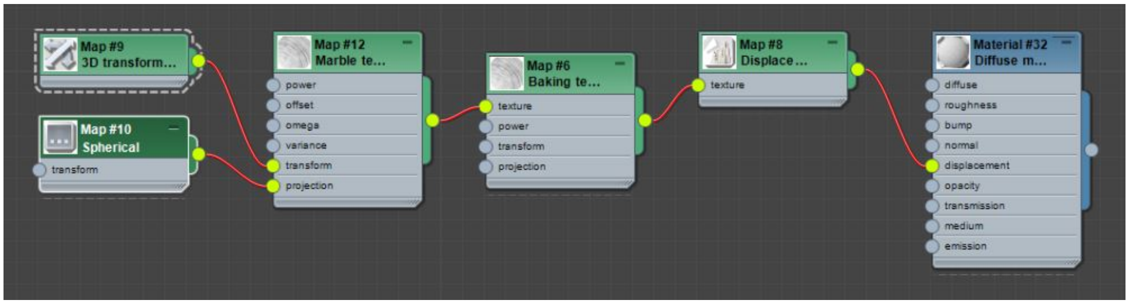


Figure 1: The typical Slate material network for using the Baking texture node with Procedural texture maps

Baking Texture Parameters

Texture - Accepts any procedural texture.

Enable Baking - Enables the baking process.

Rsolution - Determines the resolution of the texture to be converted. Higher resolutions require more RAM and **GPU**[1] power.

Samples per Pixel - Determines how many samples will be used per pixel.

Type - Determines the bit-depth of the bake texture. The options include **LDR**[2] or low dynamic range and HDR Linear Space.

RGB Baking - Converts RGB values according to the type of procedural texture. If the procedural texture has RGB values, enable this option. If the procedural texture uses greyscale values, leave this option off.

Power - Adjusts the intensity value of the baked texture.

Gamma[3] - Adjusts the gamma value of the baked texture.

Invert - Inverts the baked texture.

---

[1]The GPU is responsible for displaying graphical elements on a computer display. The GPU plays a key role in the Octane rendering process as the CUDA cores are utilized during the rendering process.
[2]Image formats that have 8 bits per color channel such as the common image formats JPEG, PNG, GIF among others.
[3]The function or attribute used to code or decode luminance for common displays. The computer graphics industry has set a standard gamma setting of 2.2 making it the most common default for 3D modelling and rendering applications.

Figure 2: The rendered results from Figure 1

Checks Texture

The **Checks** procedural texture is useful for making stripes, checkerboard, and grid patterns. It requires a **3D Transform** map or **UV** coordinates for a correct projection across a surface.

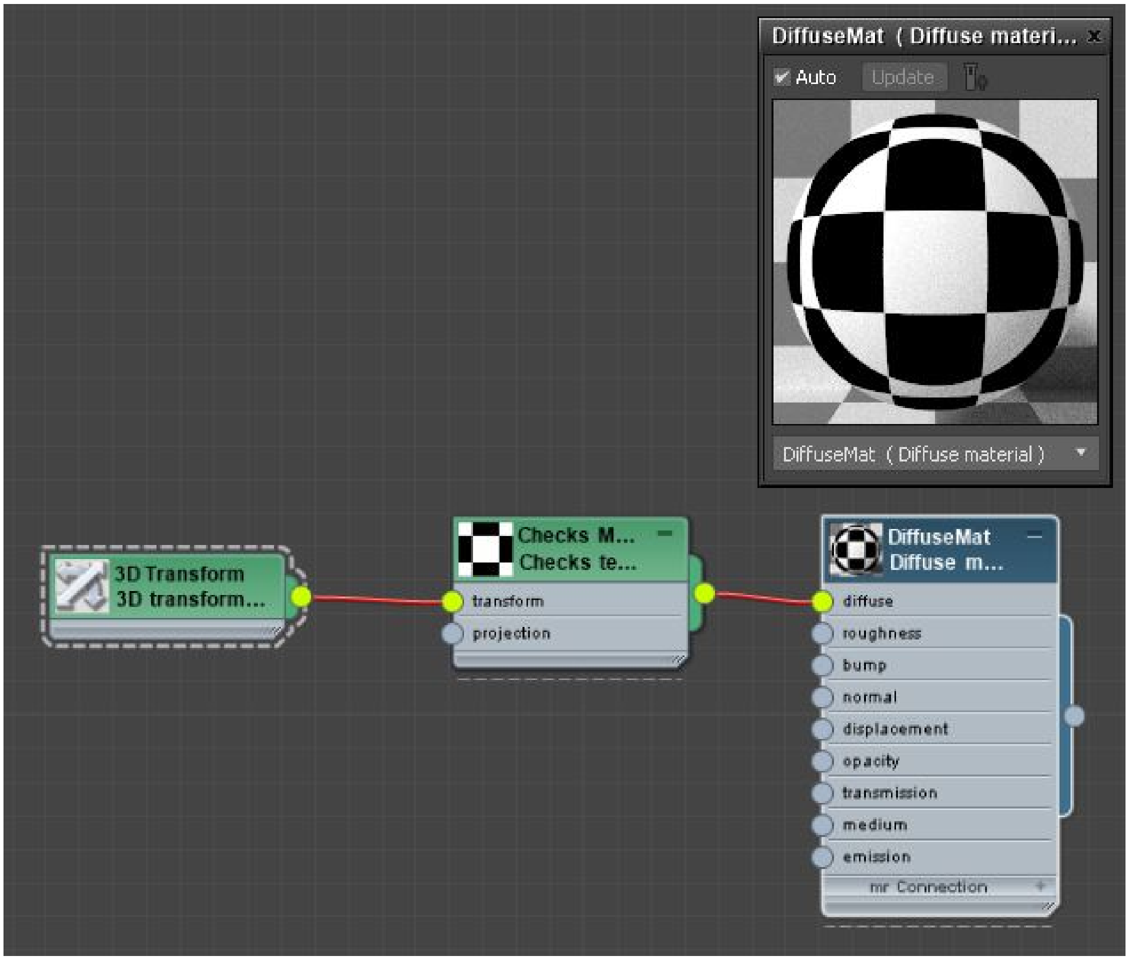


Figure 1: Connecting a 3D Transform map to a Checks texture

Float Texture

The **Float** texture provides a map where the **Texture** data is a floating point value from **0.00** - **1.00**. This signifies a greyscale value, where **0.00** is darkest and **1.00** is lightest.

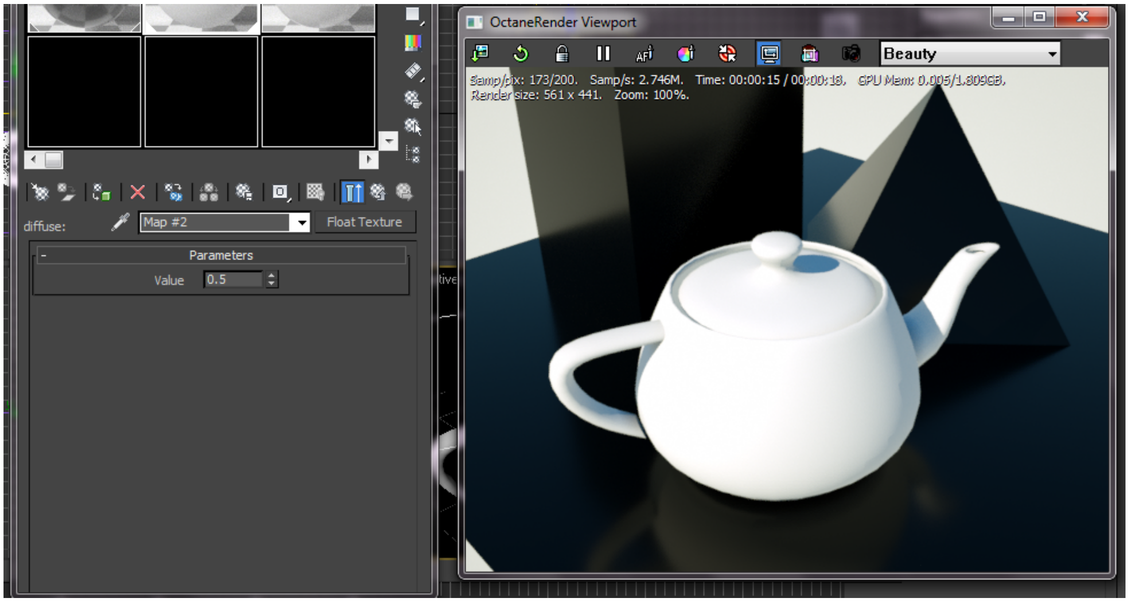


Figure 1: Float texture

Gaussian Spectrum

The **Gaussian Spectrum** texture is based on a Gaussian distribution spectrum. **Wavelength** sets the spectrum's center, and **Width** sets the curve's width. Narrower widths create purer and more saturated colors.

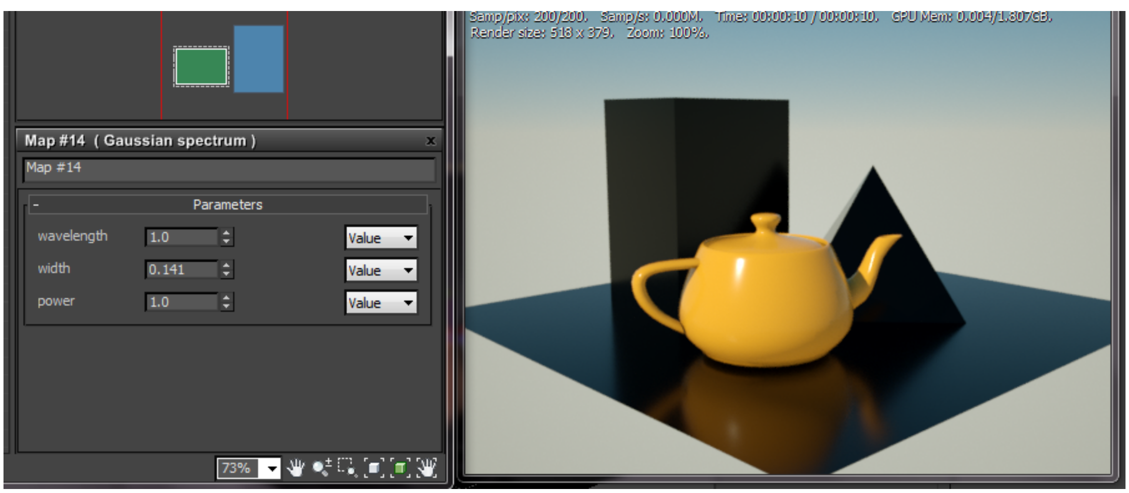


Figure 1: Gaussian Spectrum texture

Gaussian Spectrum Parameters

Wavelength - This represents the mean wavelength approximation between **380nm** - **720nm**. Lower wavelength values appear bluish, while higher wavelengths (around **700nm**) appear reddish.

Width - Almost no color is visible when using a width of **0.000**. On the other hand, a width of **1.000** means the color is spread thin over a large space, and the Texture appears faint.

Power - The Texture's brightness.

Gradient Texture

The **Gradient** texture affects graded linear changes to represent slopes, depth, distance, or color progressions of **Procedural** texture maps.

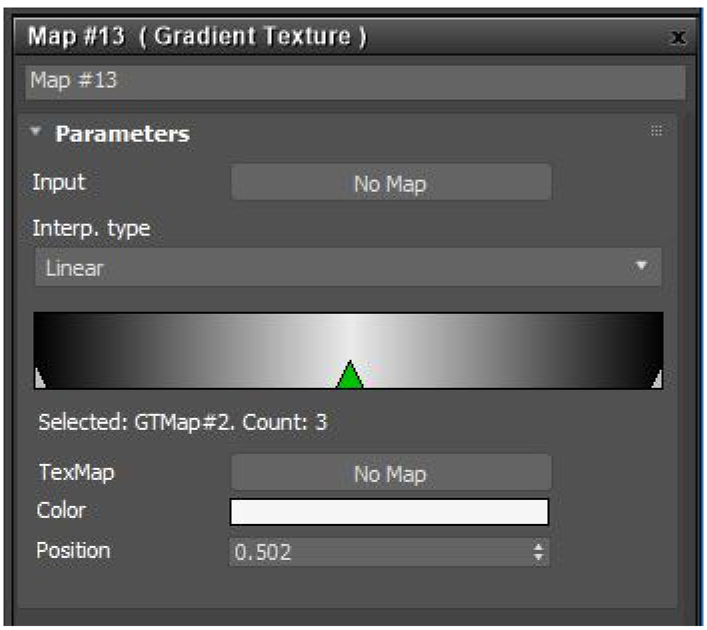


Figure 1: The Gradient texture parameters

You can define the colors through a series of control points on the curve in the same way that 3DS Max® creates Gradients.

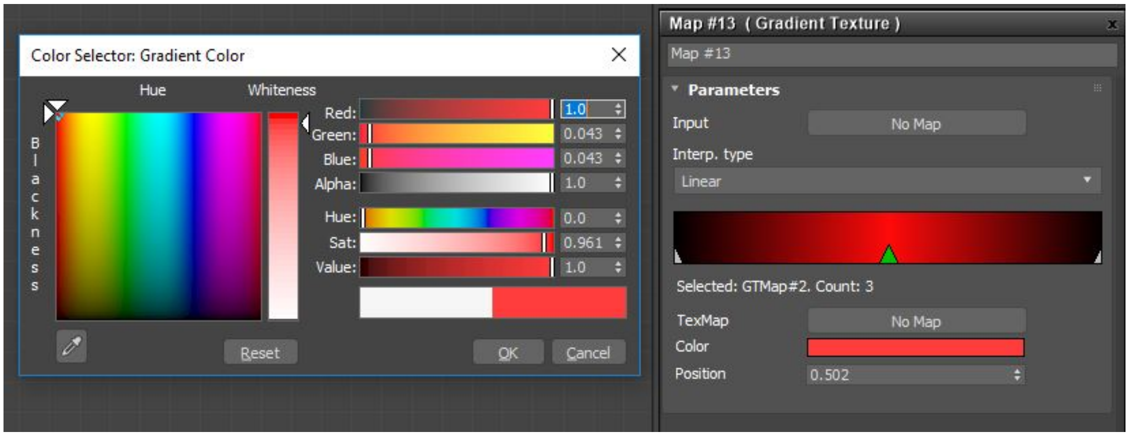


Figure 2: Defining colors for the Gradient texture node

You can also use the Gradient texture to control the color distribution for Procedural textures.

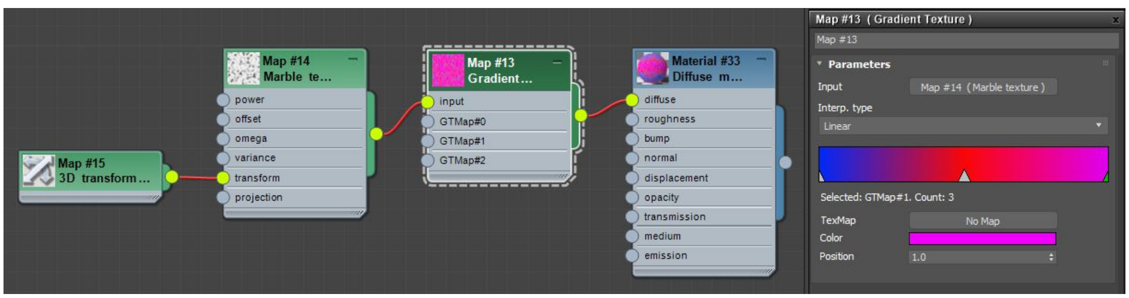


Figure 3: The Gradient texture colors a Marble texture

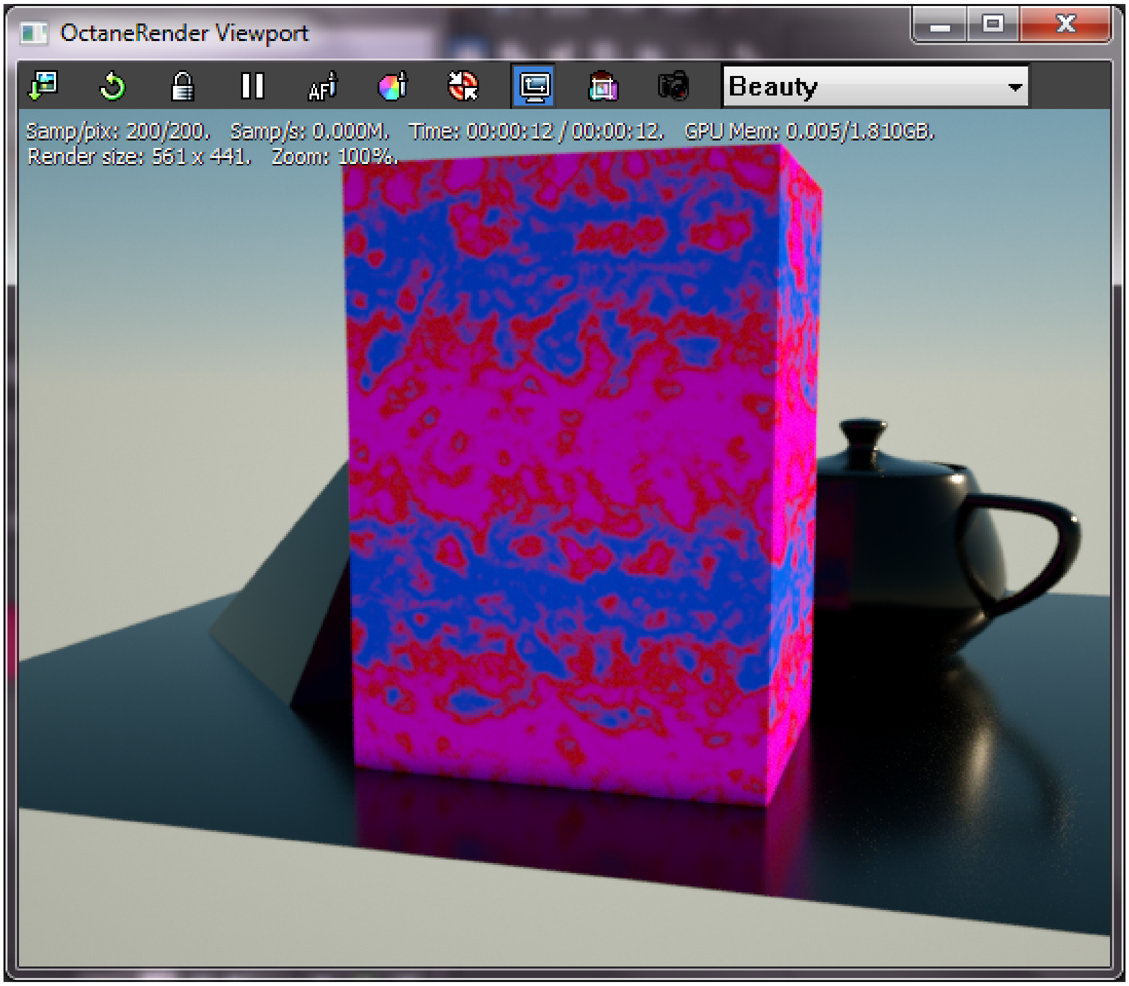


Figure 4: The results of the Slate Material[1] network from Figure 3

Gradient Parameters

Input - Determines how the color maps to the surface.

Interpolation - Select **Constant**, **Linear**, or **Cubic** to determine the color-blending rate from one marker to the next.

---

[1]The representation of the surface or volume properties of an object.

Gradient - Determines the gradient's colors. Click in the gradient to add marker and drag to either ends to remove marker. Each new marker creates an arrow and a new color input option. You can place the color on different parts of the gradient by dragging the marker around.

Gradient Start Value - Use the color swatches or RGB values to set the gradient's starting color.

Gradient End Value - Use the color swatches or RGB values to set the gradient's ending color.

TexMap - **Textures**[1] used to tile after a marker position.

Color - Define a RGB value for the gradient at the marker position.

Position - Define the marker placement on the gradient.

Grayscale Image

The **Grayscale** image map interprets any **Texture** map as grayscale, even if it is a full-color image, and saves VRAM. You can use the **Invert** checkbox to invert the image, which is useful for **Bump** and **Opacity** maps.

[1]Textures are used to add details to a surface. Textures can be procedural or imported raster files.

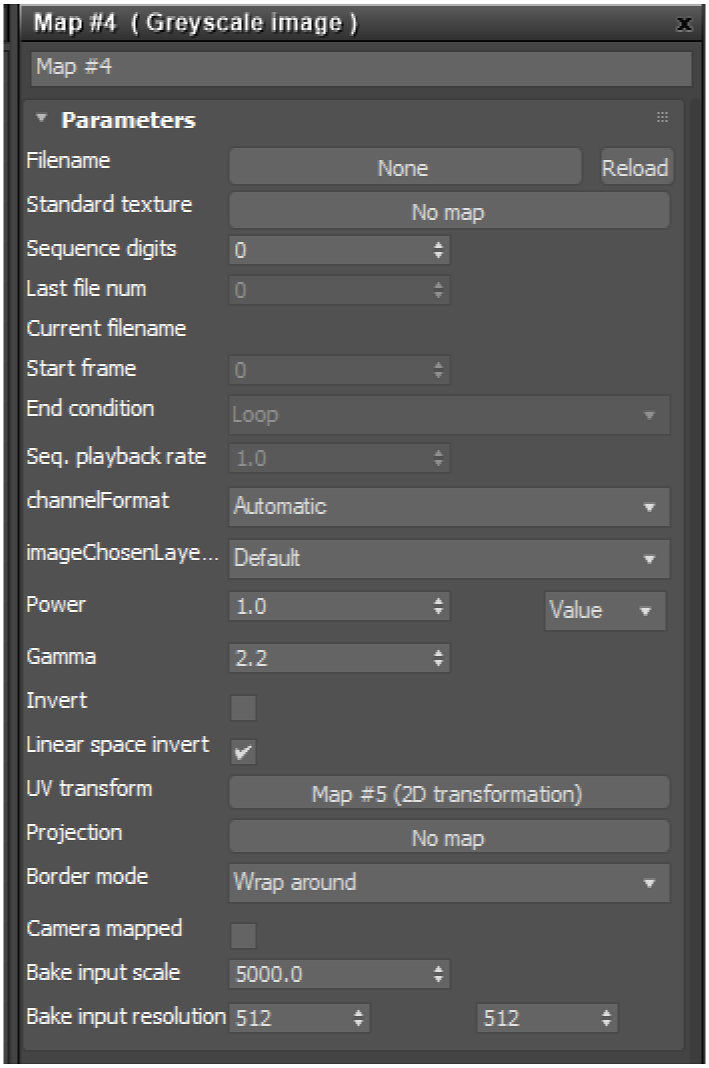


Figure 1: The Grayscale image node parameters

Greyscale Image Parameters

Standard Texture - Connects non-OctaneRender® Texture nodes to an OctaneRender material network.

Num. Sequence Digits/Start Frame/End Frame/End Condition/Sequence Playback Rate - These refer to the advanced file sequence settings for **Image** textures and **Volume** objects. The **Num.**

Sequence Digits setting is applicable when a direct file sequence is used (not an **IFL**[1] file). The **End Condition** is applicable when an IFL file is used. The **Start Frame** and **Playback Rate** are used in both modes.

Channel Format - Indicates the preferred channel format for loading the image. This is ignored for 8-bit images.

Power- Controls image brightness. Lower values cause the image to appear darker. When used as a Bump map, this setting alters the bump height on the surface.

Gamma[2] - Controls the input image luminance, and it also tunes or color-corrects the image.

Invert - Inverts the texture values.

Transform - Positions, rotates, and scales the surface texture.

Projection - Accepts OctaneRender® **Projection** nodes. If nothing is connected to this input, the Image texture uses the surface's **UV** texture coordinates by default. This also changes the UV set if the original surface contains more than one UV set.

Border Mode - Sets the behavior of the space around the image if it doesn't cover the entire geometry. **Wrap Around** is the default behavior, which repeats the image in the areas outside the image's coverage. If you set this parameter to **White Color** or **Black Color**, the area outside the image turns to white or black, respectively.

Camera Mapped - Sets the current rendering camera to project the image onto the surface.

Bake input scale -Affects only 3D texmaps. For example, 'Marble' is a 3D texmap, but 'Checker' is not. 5000 is fine with a default scene. With other scene units, objects size, etc, it might be necessary to adjust this scale.

Bake Input Resolution - These parameters provide an on-the-fly texture baking approach. OctaneRender® will render by using standard 3DS Max® CPU-calculated nodes and bake all **Textures**[3] every time it requests a scene translation. If there are many Textures with high-baking resolutions, this increases the scene translation time by a lot.

IES Texture

---

[1](Image File List) file is an ASCII file that constructs an animation by listing single-frame bitmap files to be used for each rendered frame. When you assign an IFL file as a bitmap, rendering steps through each specified frame, resulting in an animated map. (reference: https://knowledge.autodesk.com/support/3DS-max/learn-explore/caas/CloudHelp/cloudhelp/2017/ENU/3DSMax/files/GUID-CA63616D-9E87-42FC-8E84-D67E1990EE71-htm.html)

[2]The function or attribute used to code or decode luminance for common displays. The computer graphics industry has set a standard gamma setting of 2.2 making it the most common default for 3D modelling and rendering applications.

[3]Textures are used to add details to a surface. Textures can be procedural or imported raster files.

The **IES**[1] texture uses an IES profile with geometry as an illumination source. To create an IES light source using geometry:

1. Apply a **Diffuse**[2] material to the geometry.

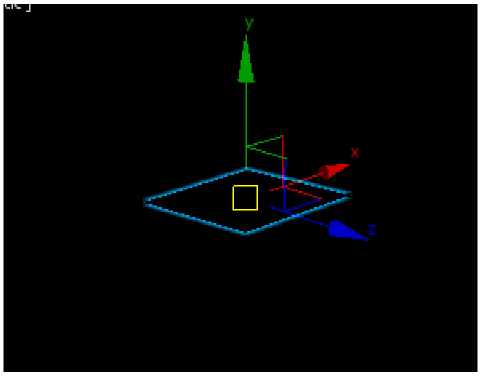
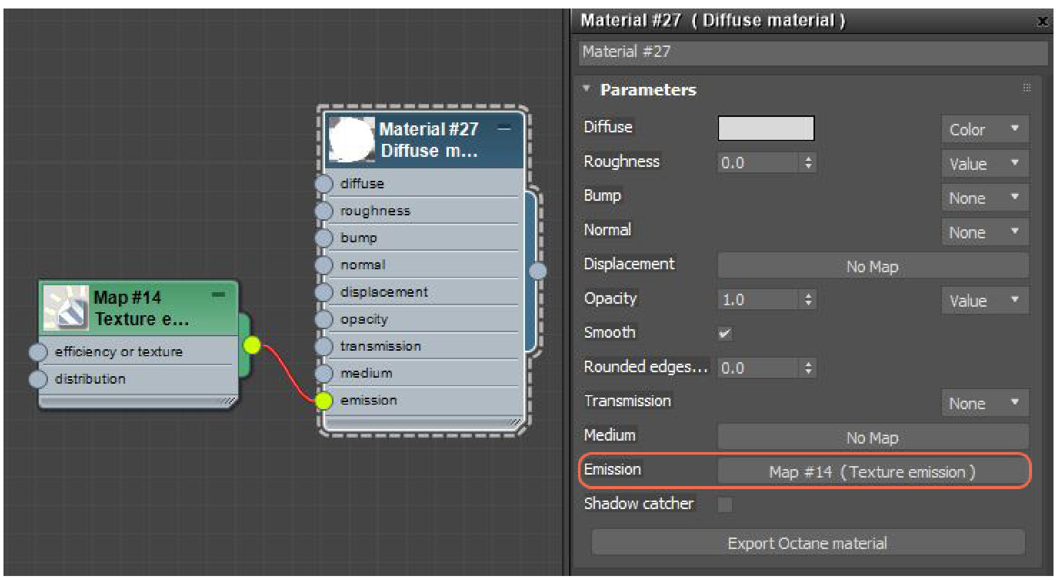


Figure 1: A Diffuse material[3] applied to a plane

2. Attach a **Texture** emission map to the Diffuse material's **Emission** parameter.



---

[1]An IES light is the lighting information representing the real-world lighting values for specific light fixtures. For more information, visit http://www.ies.org/lighting/.
[2]Amount of diffusion, or the reflection of light photons at different angles from an uneven or granular surface. Used for dull, non-reflecting materials or mesh emitters.
[3]Used for dull, non-reflecting materials or mesh emitters.

Figure 2: Connecting the Texture emission node

3. Attach an IES texture to the Texture emission's **Distribution** parameter and import an appropriate IES file in the IES texture node.

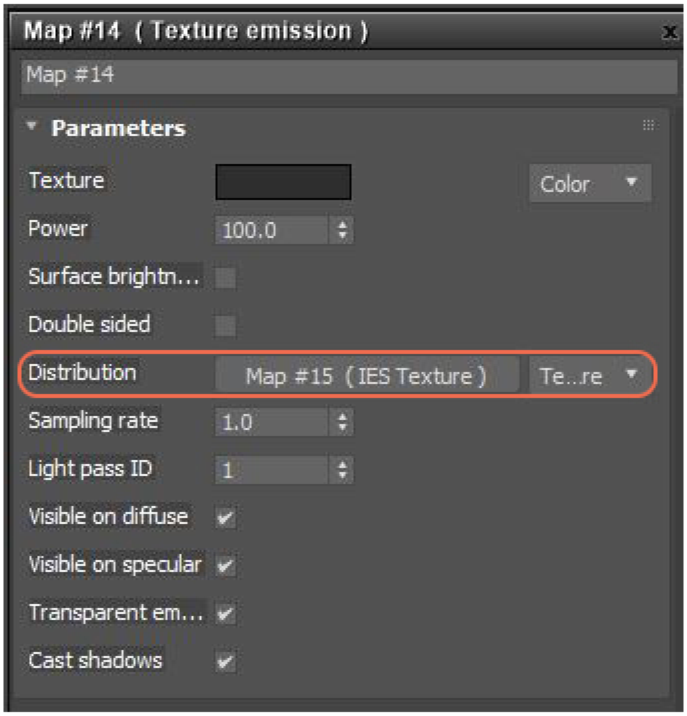


Figure 3: Adding an IES texture map to an Emission's Distribution slot

4. Select **Normal Space** as the coordinate space for the IES light projection.

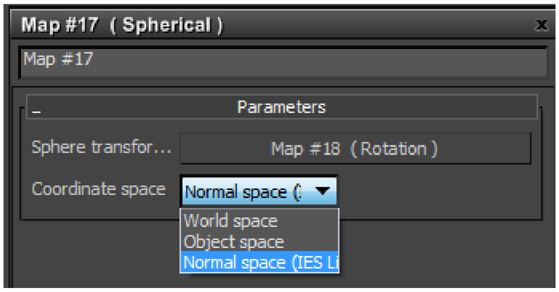


Figure 4: Selecting Normal Space in the Spherical projection node

Pictured below are the typical **Slate Editor** nodes and connections for a IES lighting setup.

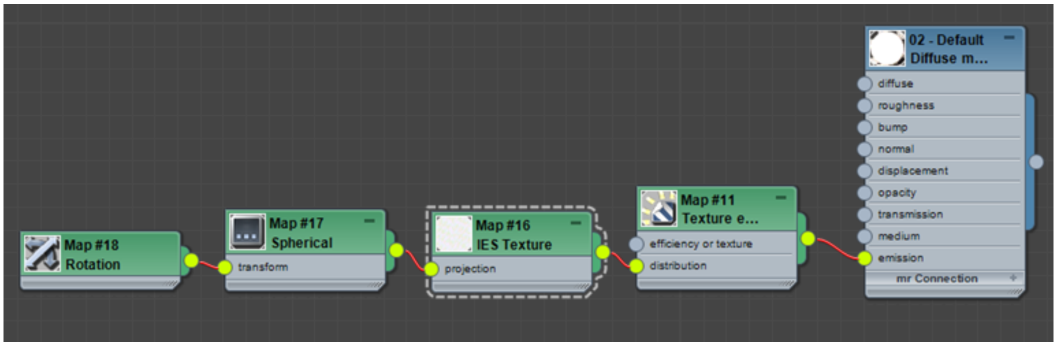


Figure 5: IES lighting node connections

Instance Color Texture

The **Instance Color** texture holds an image, and it prepares to map each pixel to geometric instance IDs.

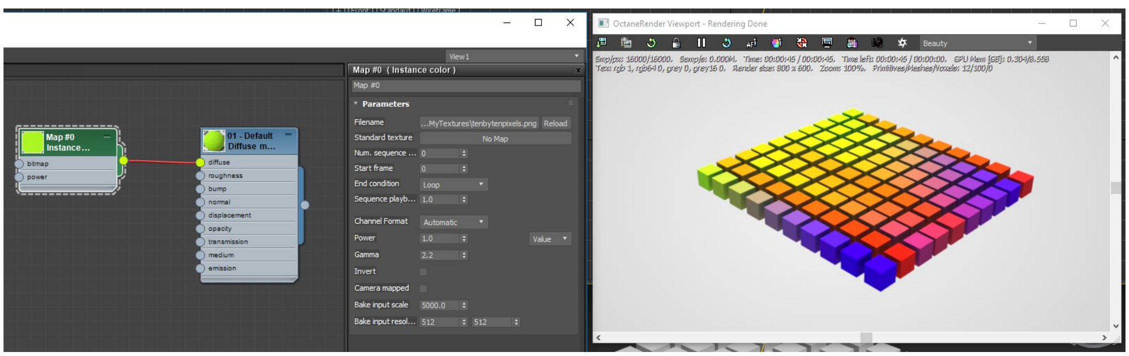


Figure 1: A color texture map applied to instances with the Instance Color texture node

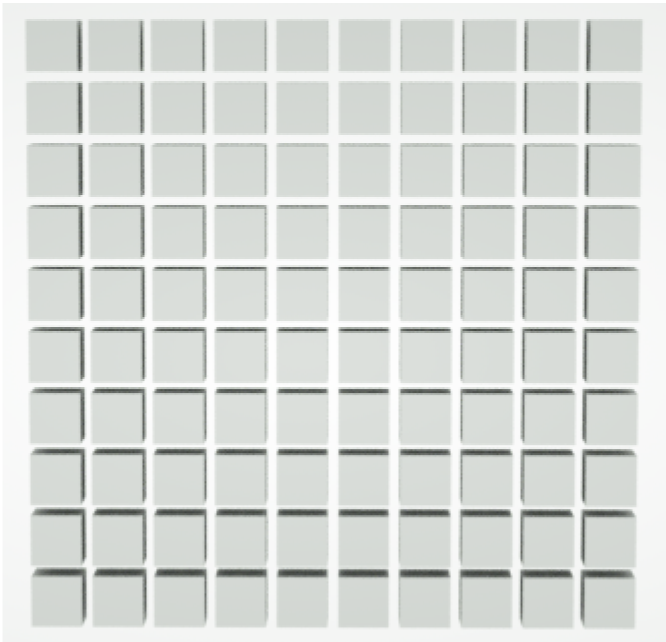


Figure 2: A cube and 99 instances of the same cube are shown here, forming a 10x10 grid of cubes

You can plug an image with 10x10 pixels into the Instance Color texture to match these dimensions. OctaneRender® maps each pixel and assigns them to the instance IDs.

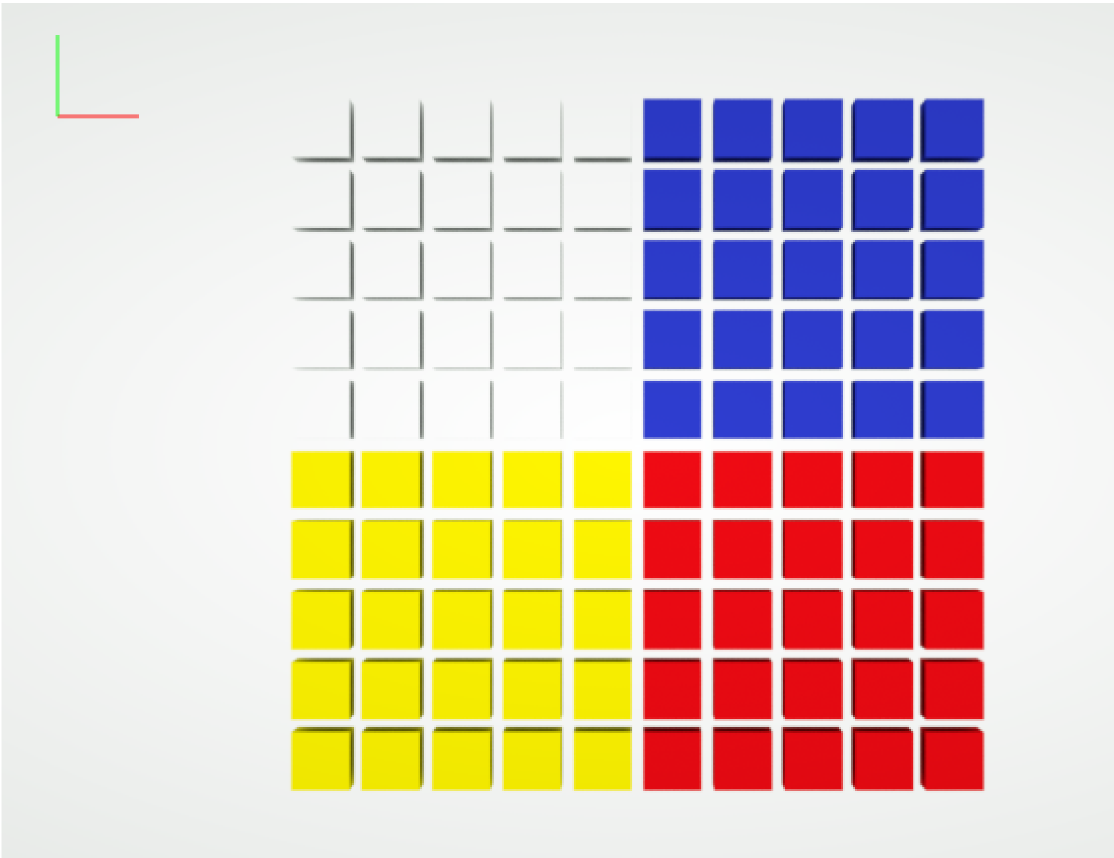


Figure 3: 10x10 grid with four colors

Instance Range Texture

Can be used with the **Gradient Texture** Node. The **Instance Range** texture holds a gradient color with the range of **0** - **Maximum ID**, and prepares this range to map to geometric instance IDs. This node has one attribute, **Maximum ID**, which corresponds to the total number of instanced geometry in the scene for which the **Material**[1] is applied.

---

[1]The representation of the surface or volume properties of an object.

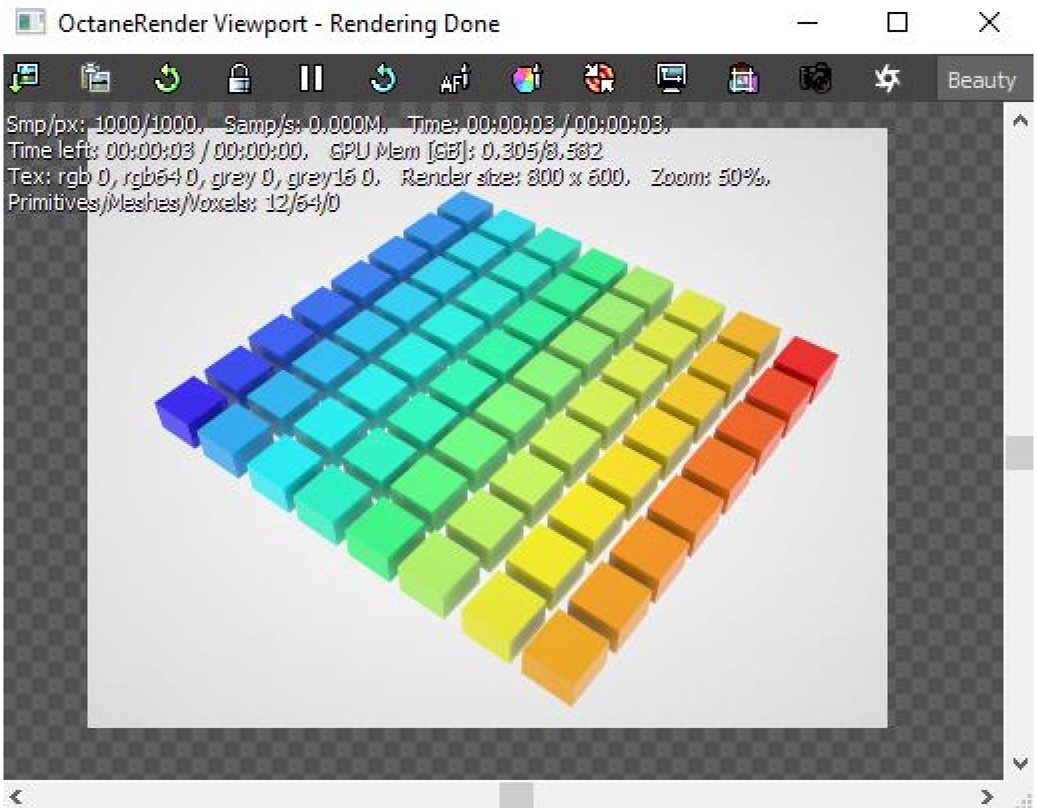


Figure 1: The Instance Range texture maps a Gradient onto a series of instanced cubes

You should set the Instance Range texture's **Maximum ID** parameter to the corresponding number of instances its Material is applied to in a scene.

Figure 2: The Maximum ID value set to a corresponding number of instances in a scene

Object instance IDs are set in an Object's **Octane Properties** window, which you can find in the **Quad** menu when right-clicking on an Object.

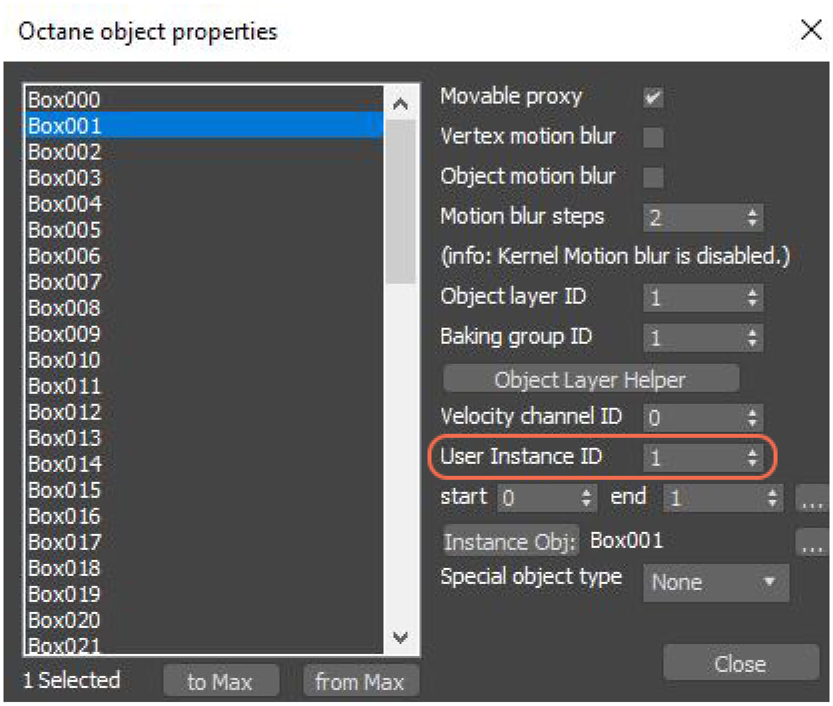


Figure 3: Specifying Instance IDs from the Octane Object Properties window

Marble Texture

The **Marble** texture is a **Procedural** texture that creates marble-like noise. It is similar to a **Turbulence** texture, but more fine-tuned to create marble-like patterns.

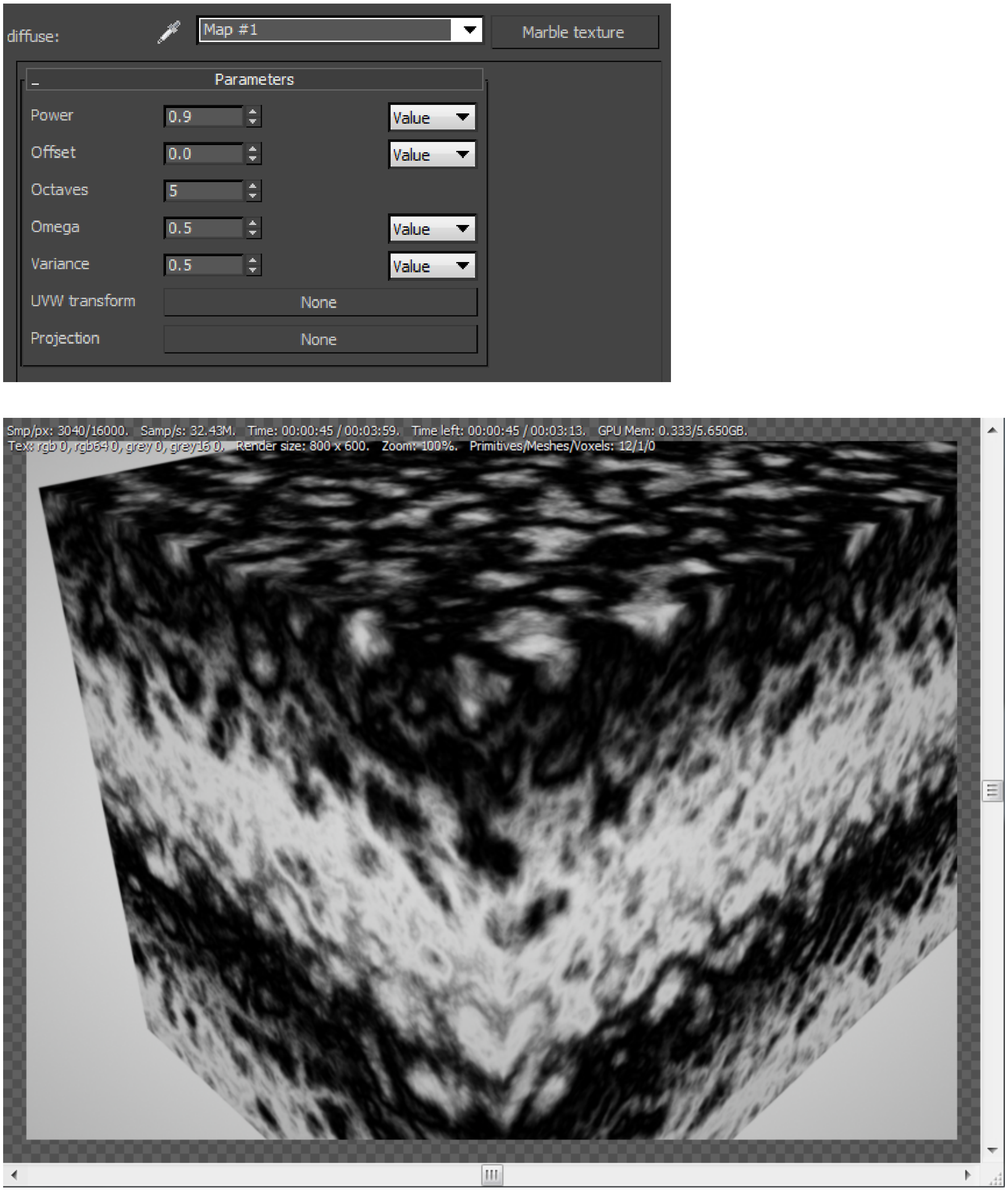


Figure 1: Marble texture

Marble Texture Parameters

Power - Controls the texture's overall brightness.

Offset - Sets the texture position in 3D space.

Octaves - Number of intervals for finer texture scale.

Omega - Controls detail in the underlying fractal pattern.

Variance - Randomizes the Marble pattern.

UVW Transform - Positions, scales, and rotates the surface texture.

Projection - Sets how the texture projects onto the surface.

Noise Texture

The **Noise** texture map is useful for creating and controlling intentional noise. There are four noise types:

- **Perlin** - Similar to the **Turbulence** node with **Use Turbulence** disabled.
- **Turbulence** - Similar to the Turbulence node with **Use Turbulence** enabled.
- **Circular** - Produces a Worley noise.
- **Chips** - Produces a Voronoi noise.

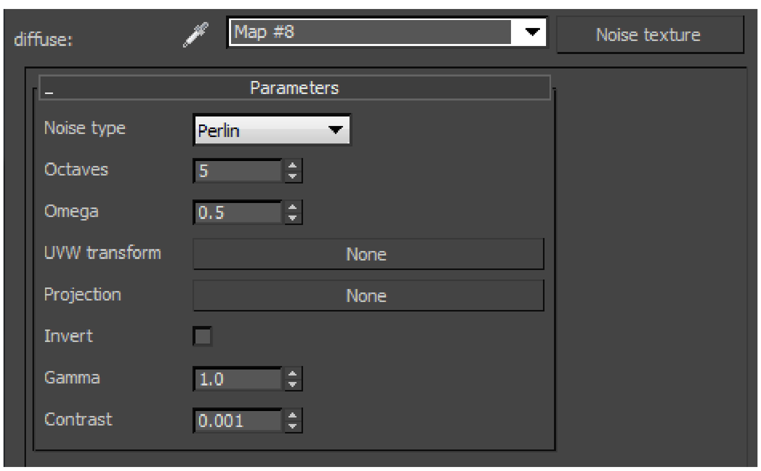


Figure 1: Noise texture parameters

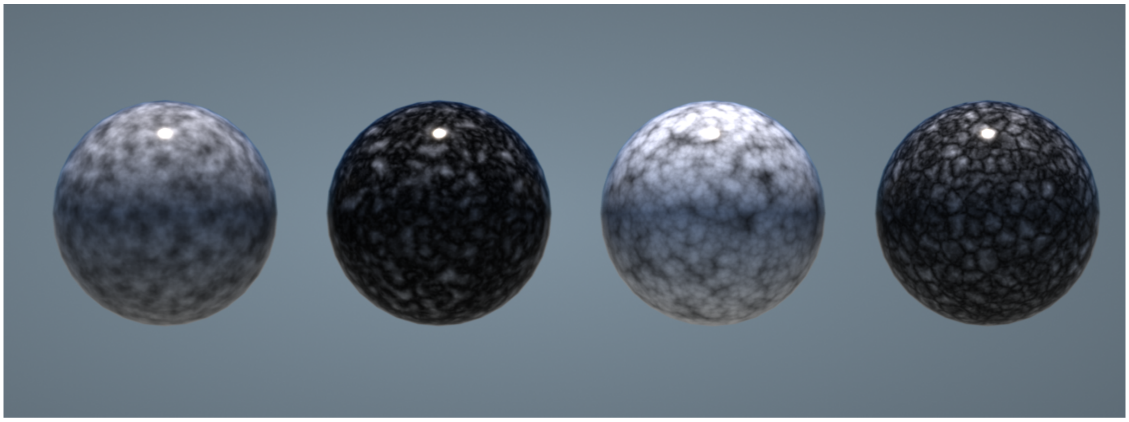


Figure 2: Noise texture samples

Noise Texture Parameters

Noise Type - Select from four different noise generators.

Octaves - Sets the noise detail's scale.

Omega - Controls the fractal pattern detail.

UVW Transform - Positions, scales, and rotates the surface texture.

Projection - Sets how the texture projects onto the surface.

Invert - Inverts the Noise texture values.

Gamma[1] - Adjust the Noise texture's luminance values.

Contrast - Adjusts the Noise detail sharpness.

OSL Texture

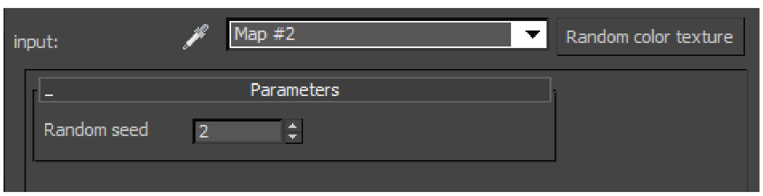
[1]The function or attribute used to code or decode luminance for common displays. The computer graphics industry has set a standard gamma setting of 2.2 making it the most common default for 3D modelling and rendering applications.

The **OSL** texture node is a scriptable node. You can write scripts using the OSL (**Open Shader Language**[1]) to define arbitrary **Texture** types to create customized OctaneRender® **Materials**[2] and shaders. To learn about the generic OSL standard, read the OSL Readme and PDF documentation.

Figure 1: The OSL texture parameters

Random Color Texture

The **Random Color** texture is a **Grayscale** texture map that you can plug into a **Gradient** node to achieve random colors.



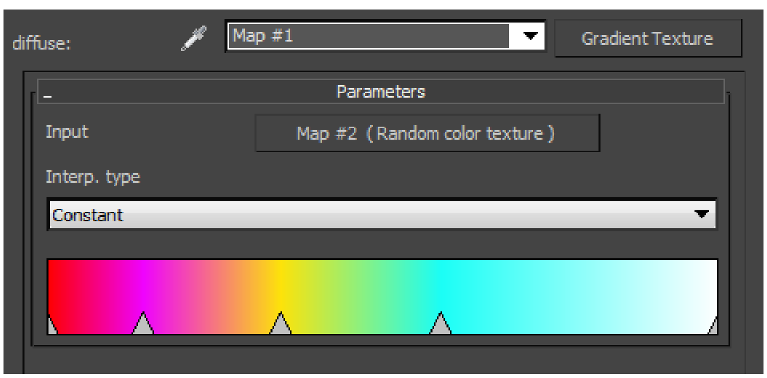


Figure 1: Random Color parameters

---

[1]A shading language developed by Sony Pictures Imageworks. There are multiple render engines that utilize OSL as it is particularly suited for physically-based renderers.
[2]A set of attributes or parameters that describe surface characteristics.

It creates quasi-random float values between **0** and **1** that are constant, but different for instances of the same **Mesh**. You can set random seeds in the **Object Layer** node to control the beginning of an instance chain. This is useful in modifying instance colors that have the same **Material**[1], such as simulating different shades of green of instanced trees in a common scene.

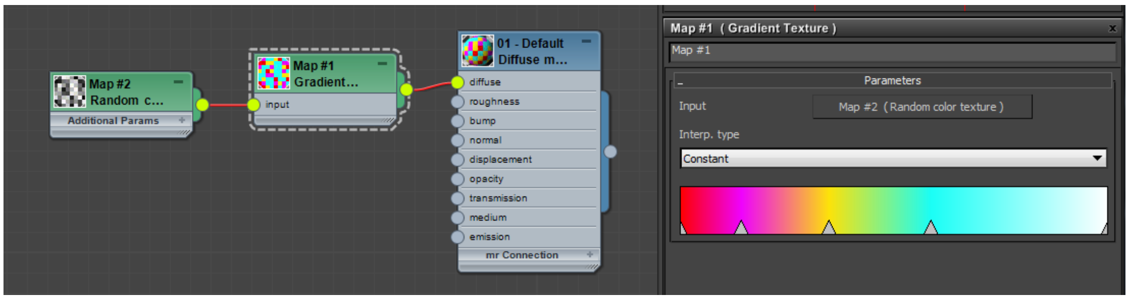


Figure 2: Random Color texture attached to a Gradient

[1] The representation of the surface or volume properties of an object.

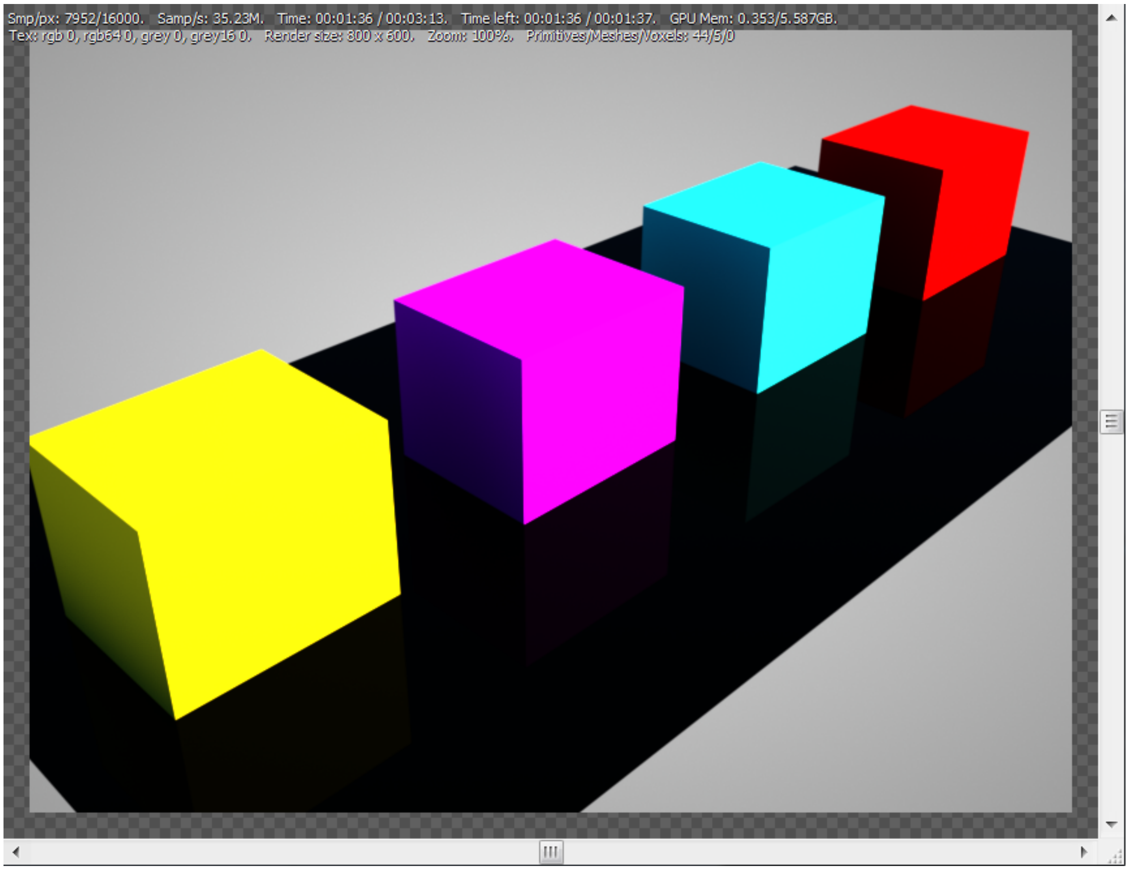


Figure 3: Random Color results

RGB Image

The **RGB Image** map imports external **Texture** maps to any **Material**[1] parameters that accept a Texture map.

---

[1] The representation of the surface or volume properties of an object.

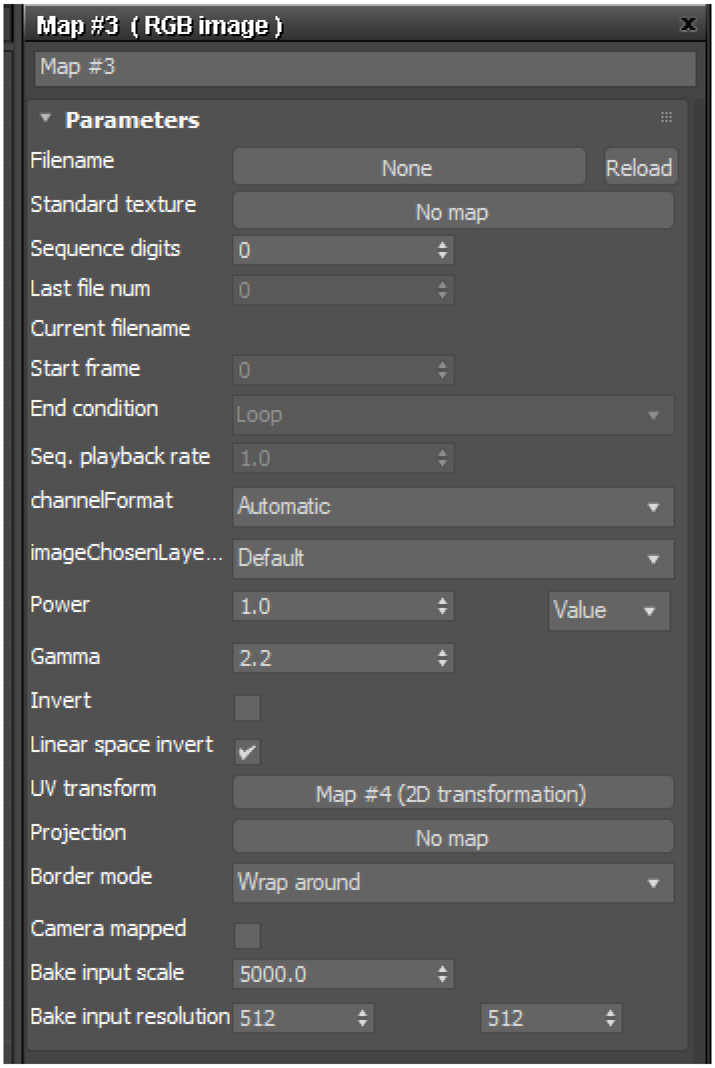


Figure 1: The RGB Image node parameters

RGB Image Parameters

Standard Texture - Connects non-OctaneRender® Texture nodes to an OctaneRender material network.

Sequence Digits/Start Frame/End Frame/End Condition/Sequence Playback Rate - These refer to the advanced file sequence settings for **Image** textures and **Volume** objects. The **Sequence Digits**

setting is applicable when a direct file sequence is used (not an **IFL**[1] file). The **End Condition** is applicable when an IFL file is used. The **Start Frame** and **Playback Rate** are used in both modes.

Channel Format - Indicates the preferred channel format for loading the image. This is ignored for 8-bit images.

Power- Controls image brightness. Lower values cause the image to appear darker. When used as a Bump map, this setting alters the bump height on the surface.

Gamma[2] - Controls the input image luminance, and it also tunes or color-corrects the image.

Invert - Inverts the texture values.

Transform - Positions, rotates, and scales the surface texture.

Projection - Accepts OctaneRender® **Projection** nodes. If nothing is connected to this input, the Image texture uses the surface's **UV** texture coordinates by default. This also changes the UV set if the original surface contains more than one UV set.

Border Mode - Sets the behavior of the space around the image if it doesn't cover the entire geometry. **Wrap Around** is the default behavior, which repeats the image in the areas outside the image's coverage. If you set this parameter to **White Color** or **Black Color**, the area outside the image turns to white or black, respectively.

Camera Mapped - Sets the current rendering camera to project the image onto the surface.

Bake input scale -Affects only 3D texmaps. For example, 'Marble' is a 3D texmap, but 'Checker' is not. 5000 is fine with a default scene. With other scene units, objects size, etc, it might be necessary to adjust this scale.

Bake Input Resolution - These parameters provide an on-the-fly texture baking approach. OctaneRender® will render by using standard 3DS Max® CPU-calculated nodes and bake all **Textures**[3] every time it requests a scene translation. If there are many Textures with high-baking resolutions, this increases the scene translation time by a lot.

---

[1](Image File List) file is an ASCII file that constructs an animation by listing single-frame bitmap files to be used for each rendered frame. When you assign an IFL file as a bitmap, rendering steps through each specified frame, resulting in an animated map. (reference: https://knowledge.autodesk.com/support/3DS-max/learn-explore/caas/CloudHelp/cloudhelp/2017/ENU/3DSMax/files/GUID-CA63616D-9E87-42FC-8E84-D67E1990EE71-htm.html)
[2]The function or attribute used to code or decode luminance for common displays. The computer graphics industry has set a standard gamma setting of 2.2 making it the most common default for 3D modelling and rendering applications.
[3]Textures are used to add details to a surface. Textures can be procedural or imported raster files.

RGB Spectrum Texture

The **RGBSpectrum** texture map outputs the color specified in the **RGB Color** parameter.

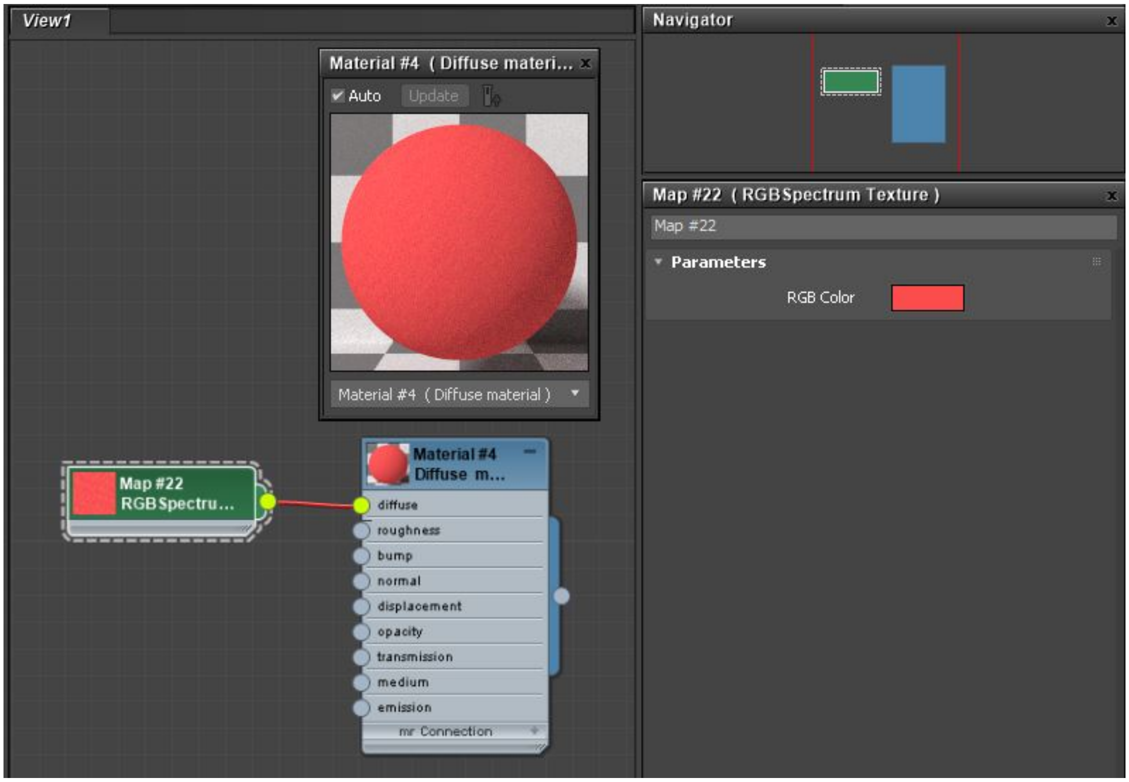


Figure 1: RGB Spectrum texture

Ridged Fractal Texture

The **Ridged Fractal** texture produces a fractal pattern in grayscale format. This generates a memory-efficient Texture with ridged waves and fractal patterns.

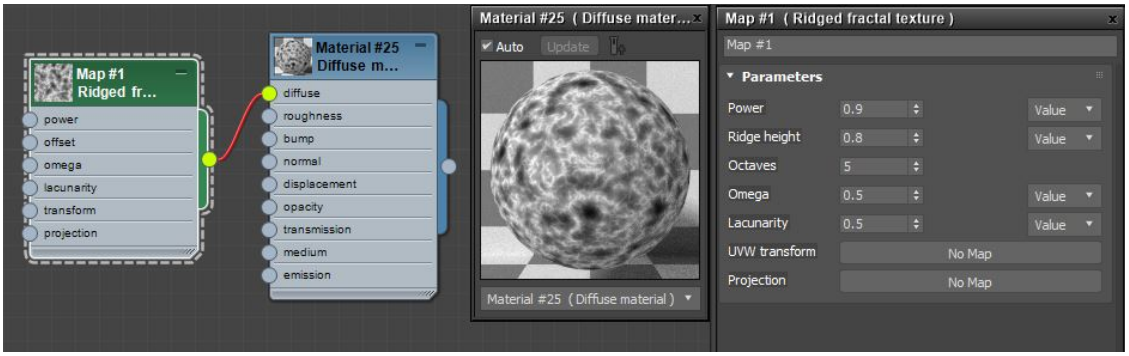


Figure 1 : A Ridged Fractal texture node connected to a Diffuse material[1]'s Diffuse[2] channel

Ridged Fractal Parameters

Power - Controls the texture's overall brightness.

Ridge Height - This specifies the height of the elevated parts of the fractal pattern.

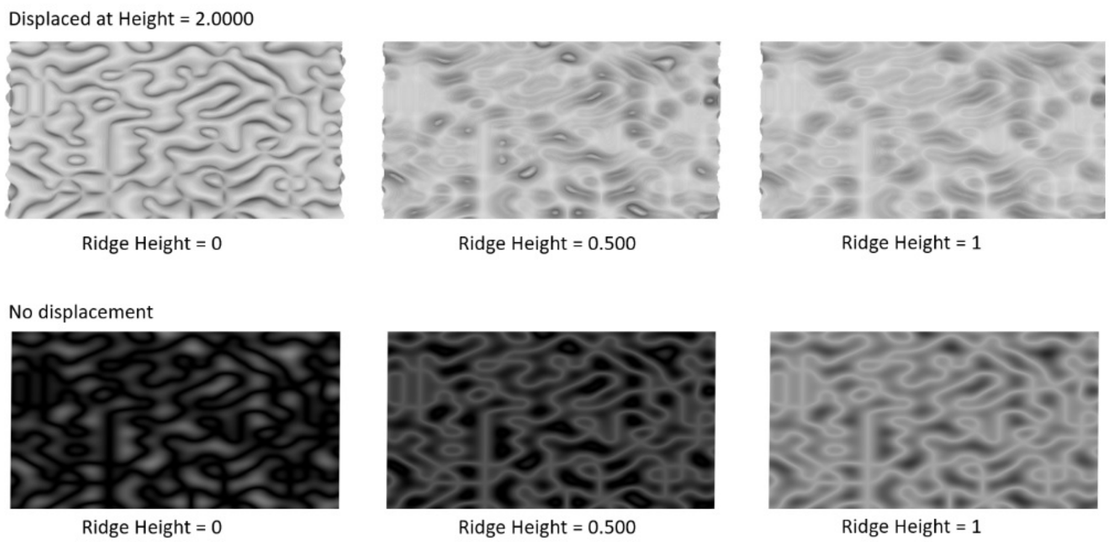


Figure 2: Ridge Height examples

---

[1]Used for dull, non-reflecting materials or mesh emitters.

[2]Amount of diffusion, or the reflection of light photons at different angles from an uneven or granular surface. Used for dull, non-reflecting materials or mesh emitters.

Octaves - Controls the amount of detail in the texture.

Omega - This specifies the difference per interval.

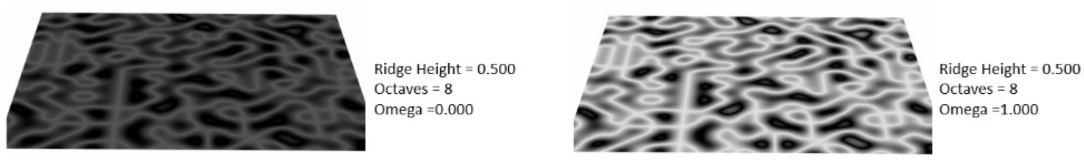


Figure 3: Omega setting examples

Lacunarity - Controls the size of the gaps in the fractal pattern.

UVW Transform - Controls the texture's position, scale, and rotation on the surface.

Projection - Determines how the texture projects onto the surface.

Saw Wave Texture

The **Saw Wave** texture generates a memory-efficient wave texture with sharp jagged-edged patterns that can look like wood.

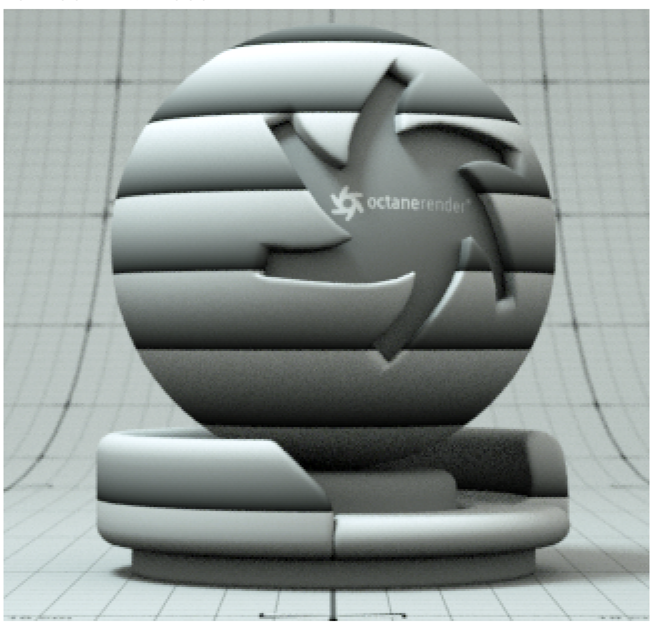


Figure 1: Saw Wave texture

Sine Wave Texture

The **Sine Wave** texture generates a memory-efficient **Texture** with simple, smooth wave patterns for marble or wood.

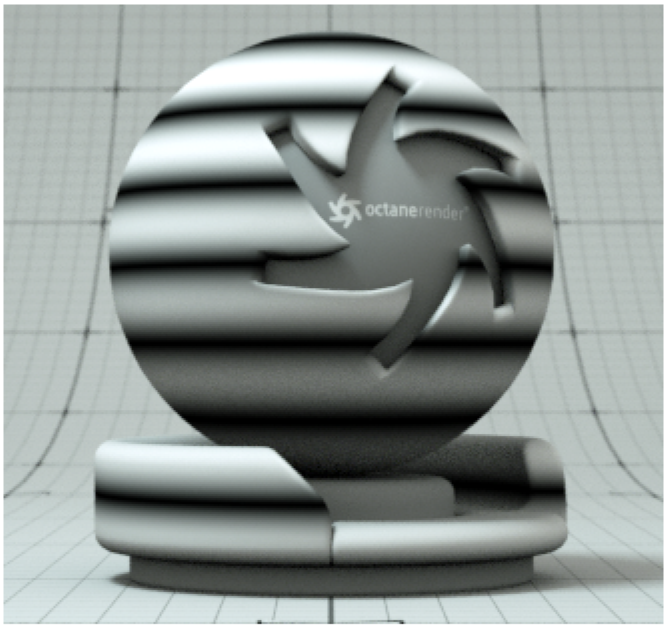


Figure 1: Sine Wave texture

Triangle Wave Texture

The **Triangle Wave** texture generates a memory-efficient **Texture** with sharp triangular wave patterns.

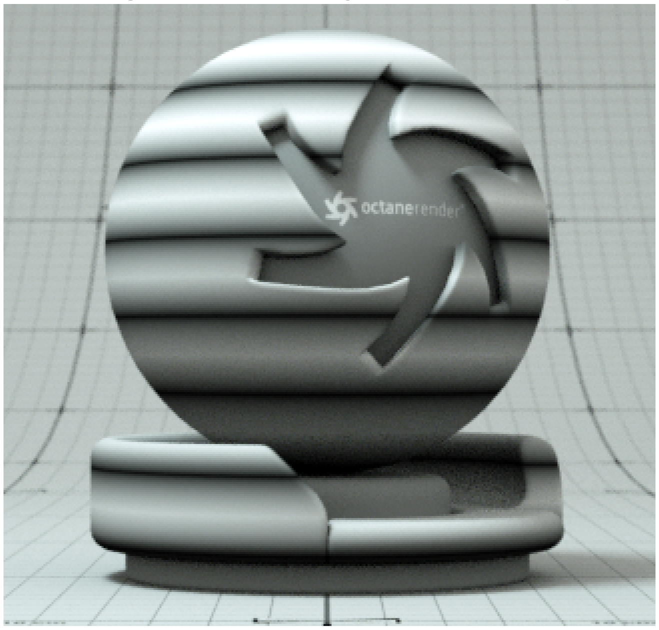


Figure 1: Triangle Wave texture

Turbulence Texture

The **Turbulence** texture creates many different effects based on banded noise. This flexible **Texture** can create wood, marble, flesh, and many other useful **Textures**[1].

[1]Textures are used to add details to a surface. Textures can be procedural or imported raster files.

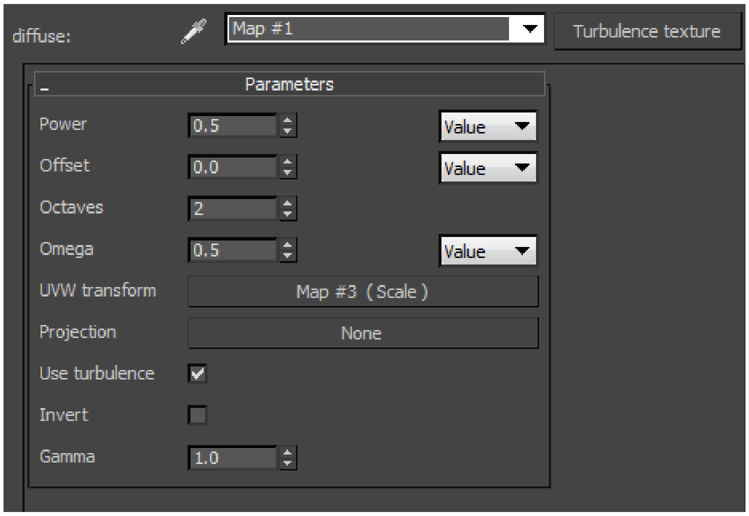
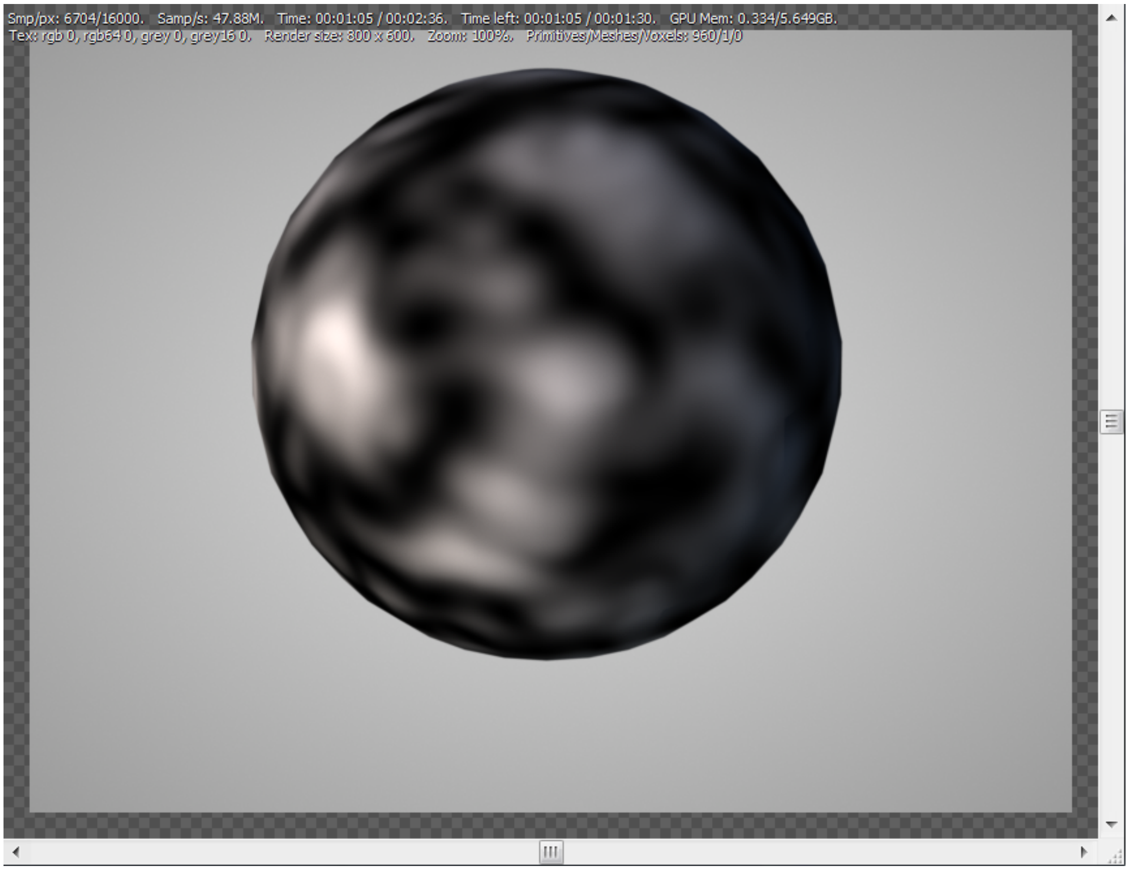


Figure 1: Turbulance texture parameters


Smp/px: 6704/16000. Samp/s: 47.88M. Time: 00:01:05 / 00:02:36. Time left: 00:01:05 / 00:01:30. GPU Mem: 0.334/5.649GB.
Tex: rgb 0, rgb64 0, grey 0, grey16 0. Render size: 800 x 600. Zoom: 100%. Primitives/Meshes/Voxels: 960/1/0

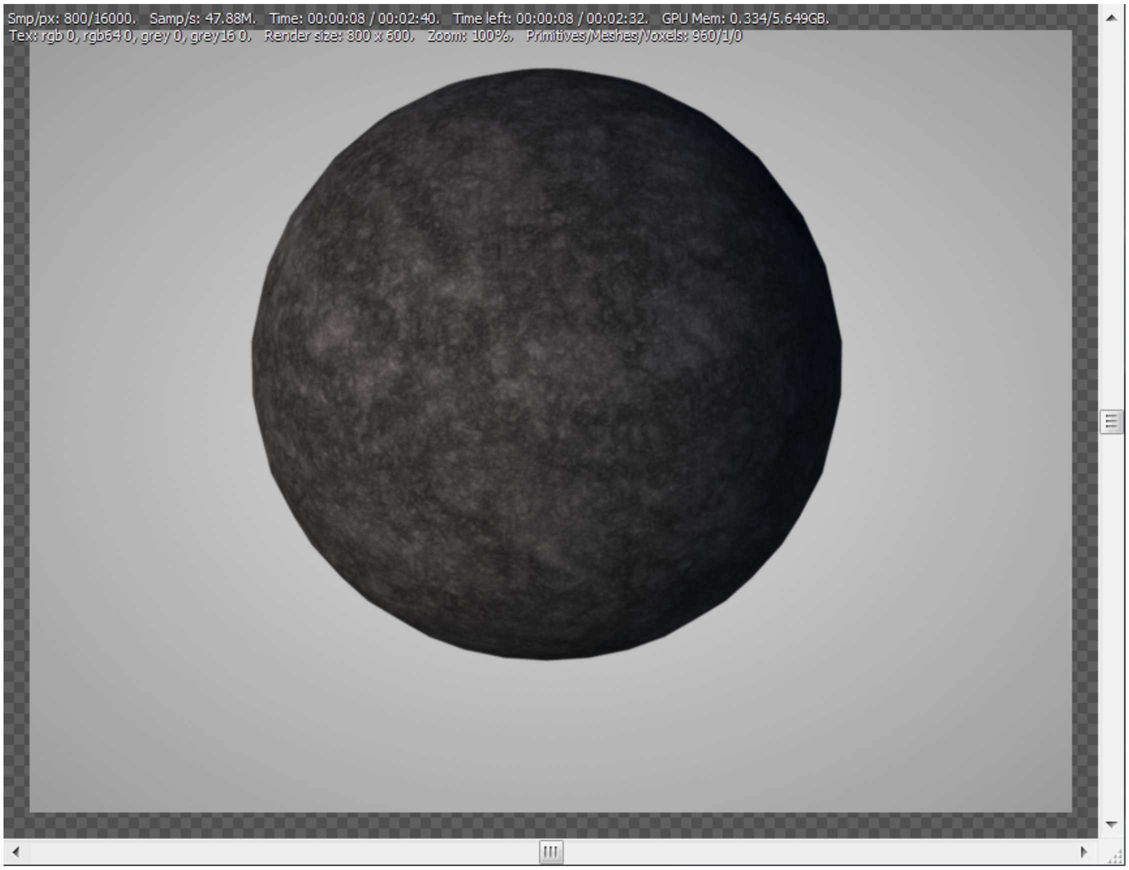


Figure 2: Turbulance texture

Turbulence Texture Parameters

Power - Controls overall texture brightness.

Offset - Shifts the Turbulence pattern in 3D space.

Octaves - Sets the noise detail's scale.

Omega - Controls fractal pattern detail.

UVW Transform - Positions, scales, and rotates the surface texture.

Projection - Determines how the texture projects onto the surface.

Use Turbulence - Toggles the turbulent noise calculation, which multiplies against procedural noise.

Invert - Inverts the Noise texture values.

Gamma[1] - Adjusts the Noise texture's luminance values.

Color Vertex Attribute

The **Color Vertex Attribute** texture works in conjunction with the 3DS Max® Vertex Paint modifier. This Texture node displays vertex paint data on native OctaneRender® **Materials**[2].

Your mesh object will need to apply the **Octane Geometry Properties** modifier to successfully read the MapChannel properties correctly.

[1] The function or attribute used to code or decode luminance for common displays. The computer graphics industry has set a standard gamma setting of 2.2 making it the most common default for 3D modelling and rendering applications.
[2] A set of attributes or parameters that describe surface characteristics.

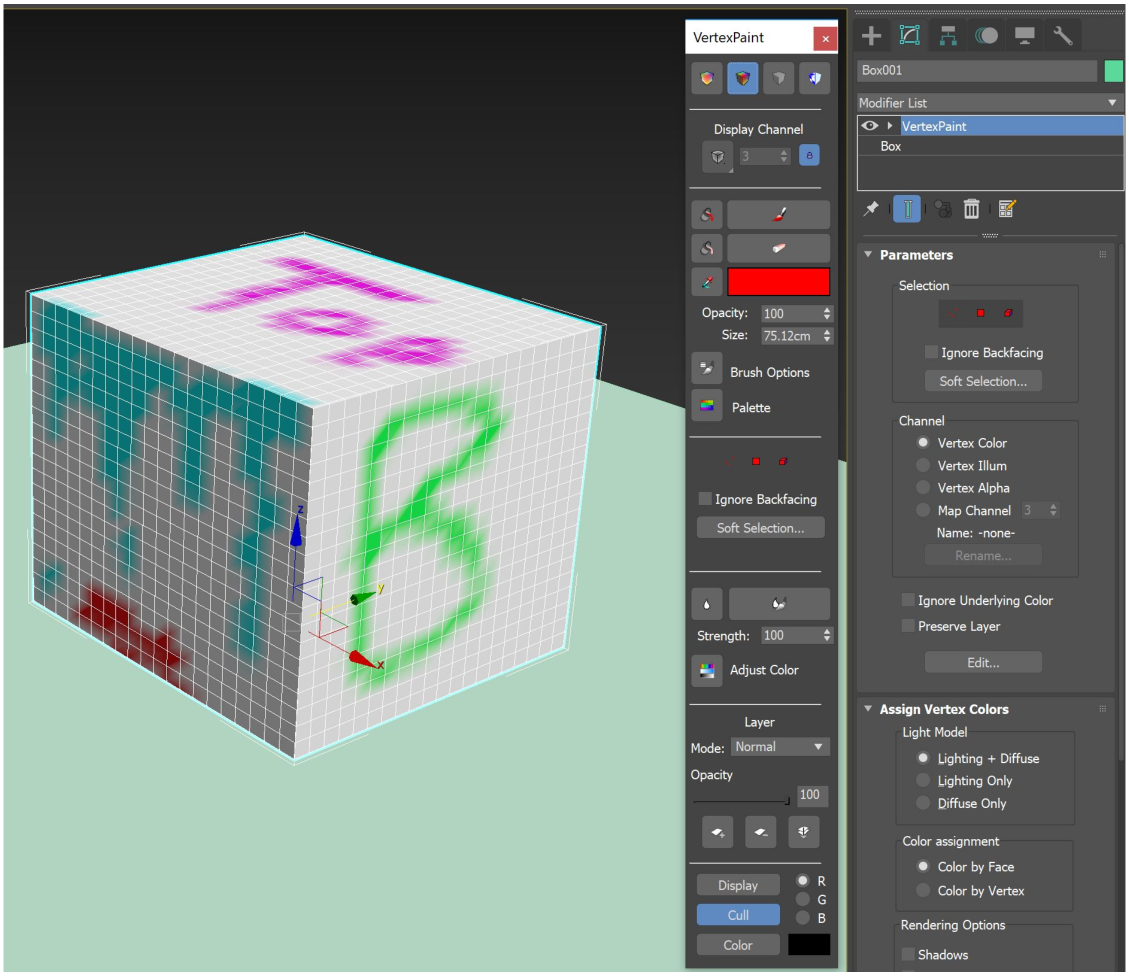


Figure 1: Using the Vertex Paint Modifier to paint color onto a cube surface

Scene **Objects** display the Color Vertex Attribute node if you set them to **Movable Proxy**[1] in the **Octane Object Properties** window.

---

[1] An object saved as a separate file with the purpose of being reused in larger scenes. This is used to minimize any addition to the total polygon count in the scene, especially if the scene requires the same object to appear several times. If used in conjunction with instancing, Proxies help keep very large scenes from reaching polygon limits and also keeps the relative file size of the main project file manageable.

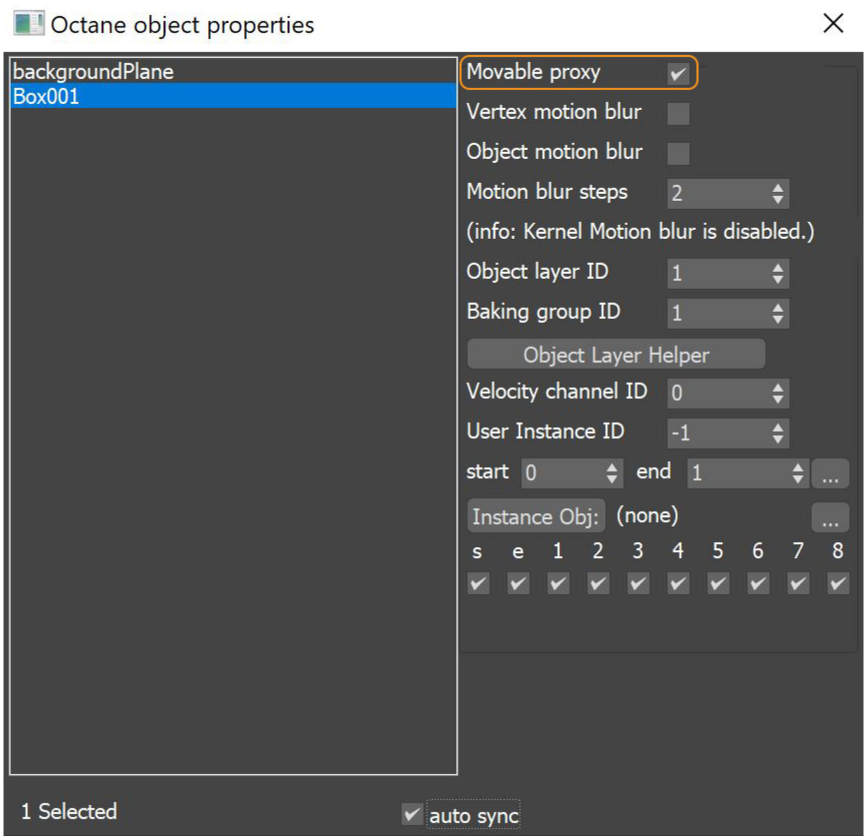


Figure 2: Setting an object to Movable Proxy

After painting the Object surface with the Vertex Paint modifier, connect the Color Vertex Attribute node where appropriate in the Object's shader node network.

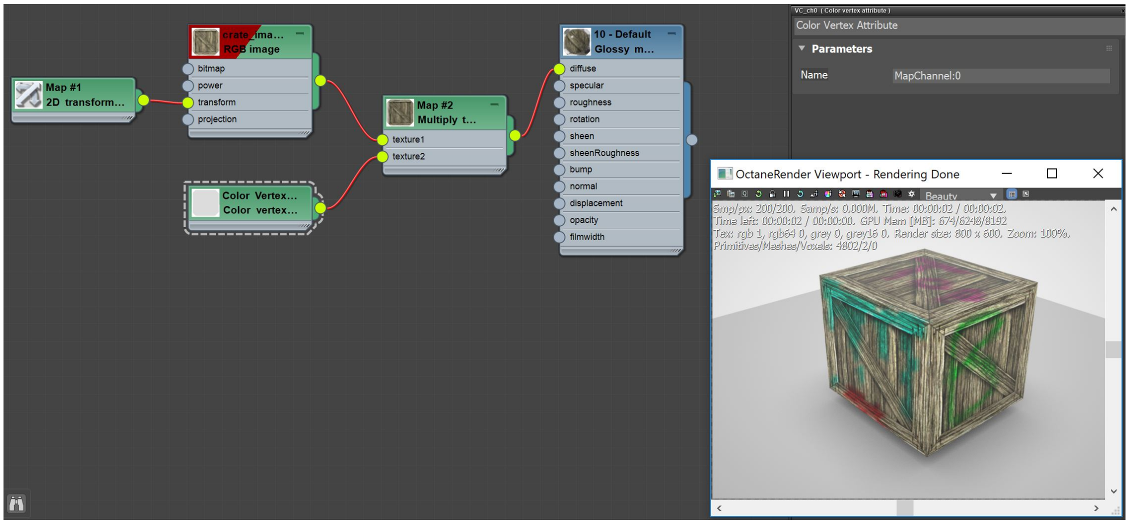


Figure 3: The Color Vertex Attribute node is mixed with a RGB Texture node

The Vertex Paint modifier's default **MapChannel** for storing the color data is **0**. Specify this in the Color Vertex Attribute node's **Name** parameter.

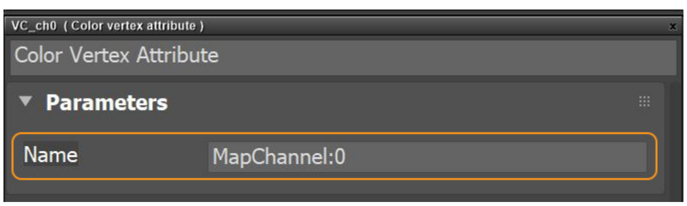


Figure 4: Specifying the default MapChannel in the Color Vertex Attribute node

Grayscale Vertex Attribute

The **Grayscale Vertex Attribute** node operates similar to the **Color Vertex Attribute** node. The difference is that this **Node** produces a float value from the vertex paint data to use for **Material**[1] attributes such as **Opacity**. Set the corresponding **Name** parameter to **MapChannel:-2** as the default channel for Opacity.

[1] The representation of the surface or volume properties of an object.

Your mesh object will need to apply the **Octane Geometry Properties** modifier to successfully read the MapChannel properties correctly.

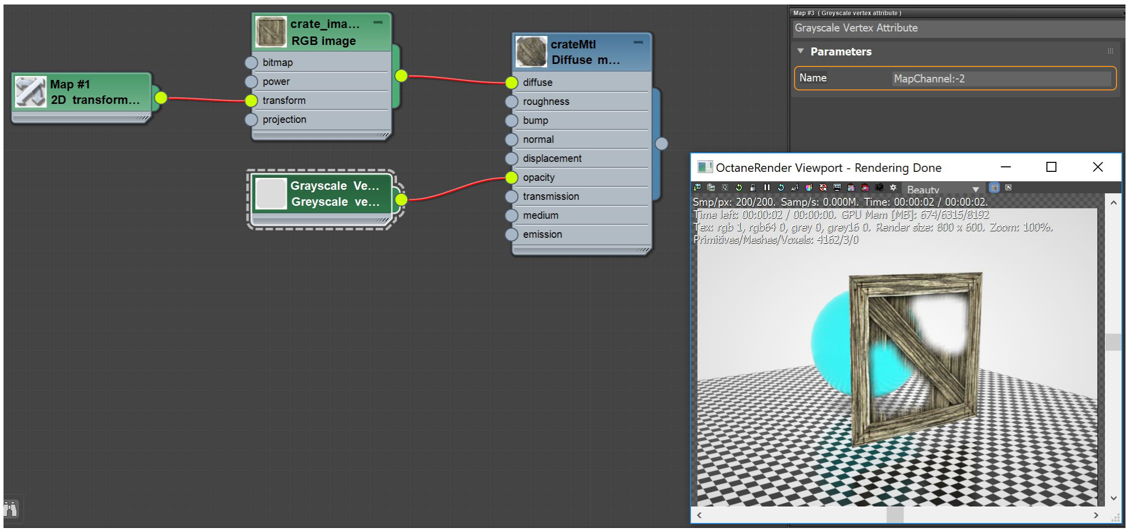


Figure 1: Setting the Grayscale Vertex Attribute node's Name parameter to -2

Texture Modifiers

Texture Modifiers provide a set of **Nodes** in the **Slate Material**[1] **Editor** that modify existing **Texture** maps. You can access these Nodes from the **Maps** rollout in the Slate Material Editor.

[1]The representation of the surface or volume properties of an object.

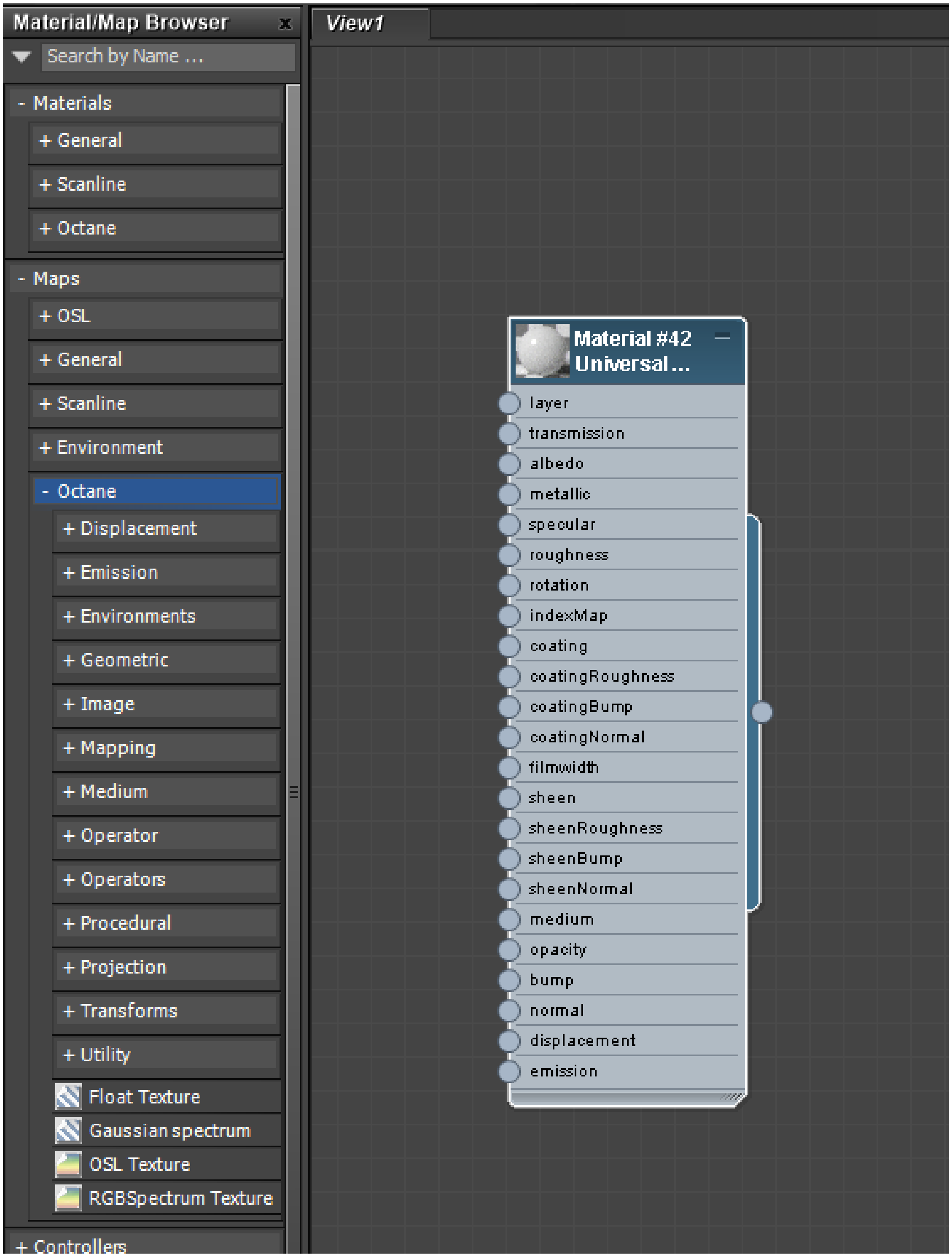

Material/Map Browser
Search by Name ...
- Materials
+ General
+ Scanline
+ Octane
- Maps
+ OSL
+ General
+ Scanline
+ Environment
- Octane
+ Displacement
+ Emission
+ Environments
+ Geometric
+ Image
+ Mapping
+ Medium
+ Operator
+ Operators
+ Procedural
+ Projection
+ Transforms
+ Utility
Float Texture
Gaussian spectrum
OSL Texture
RGBSpectrum Texture
+ Controllers
View1
Material #42
Universal...
layer
transmission
albedo
metallic
specular
roughness
rotation
indexMap
coating
coatingRoughness
coatingBump
coatingNormal
filmwidth
sheen
sheenRoughness
sheenBump
sheenNormal
medium
opacity
bump
normal
displacement
emission

Figure 1: Accessing OctaneRender Map nodes in the Slate Material Editor

Add Texture

The **Add** texture node adds two **Textures**[1] together. The calculation is similar to the Add Layer mode used in Photoshop® to add the color values of two layers. Figure 1 shows how the Add texture adds a red color to a **Ridged Fractal** texture. The result is that the dark parts of the Ridged Fractal pattern are tinted red.

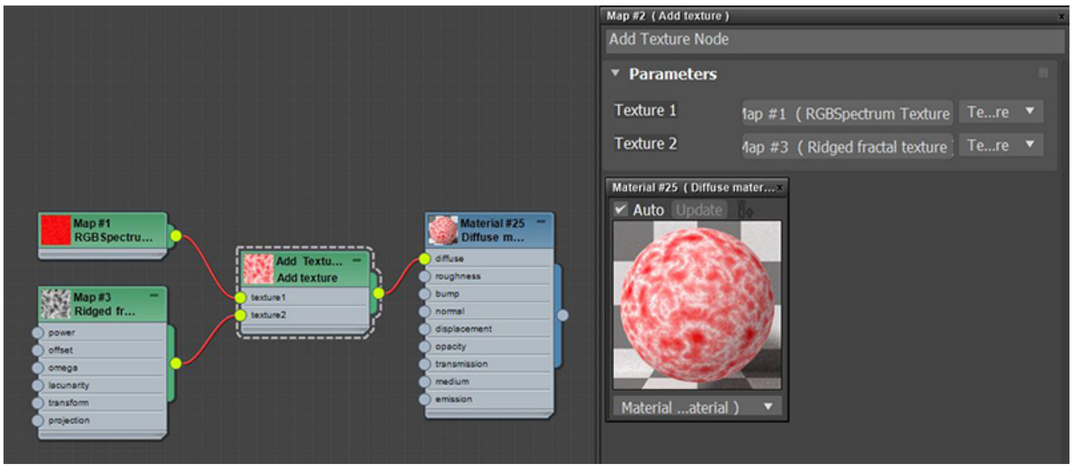


Figure 1 : An Add texture tints a Ridged Fractal texture with a red color

Clamp Texture

The **Clamp** texture requires a **Texture** input to have the Texture clamp with the **Minimum** and **Maximum** sliders.

---

[1]Textures are used to add details to a surface. Textures can be procedural or imported raster files.

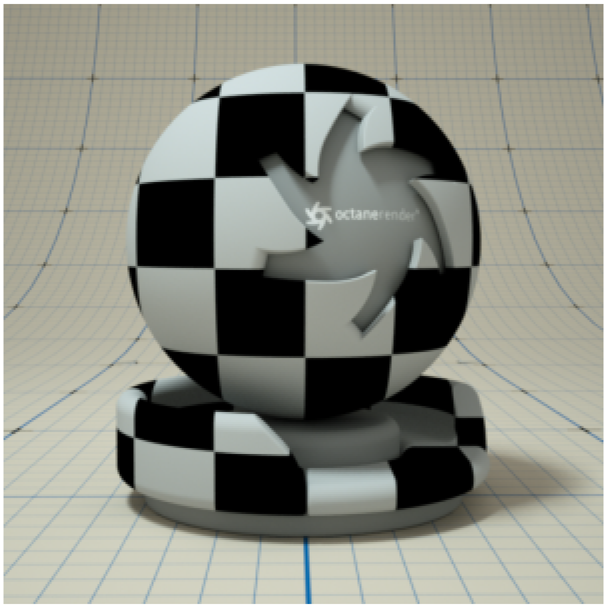


Figure 1: Clamp texture with settings Minimum = 0, Maximum = 0.5

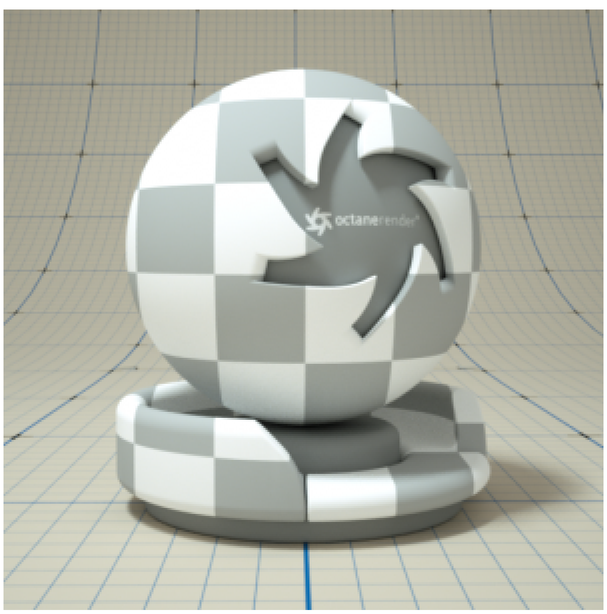


Figure 2: Clamp texture with settings Minimum = 0.5, Maximum = 1.0

Color Correction

The **Color Correction** map adjusts typical image attributes for a **Texture** map with **Brightness**, **Hue**, **Saturation**, **Gamma**, and **Contrast** parameters.

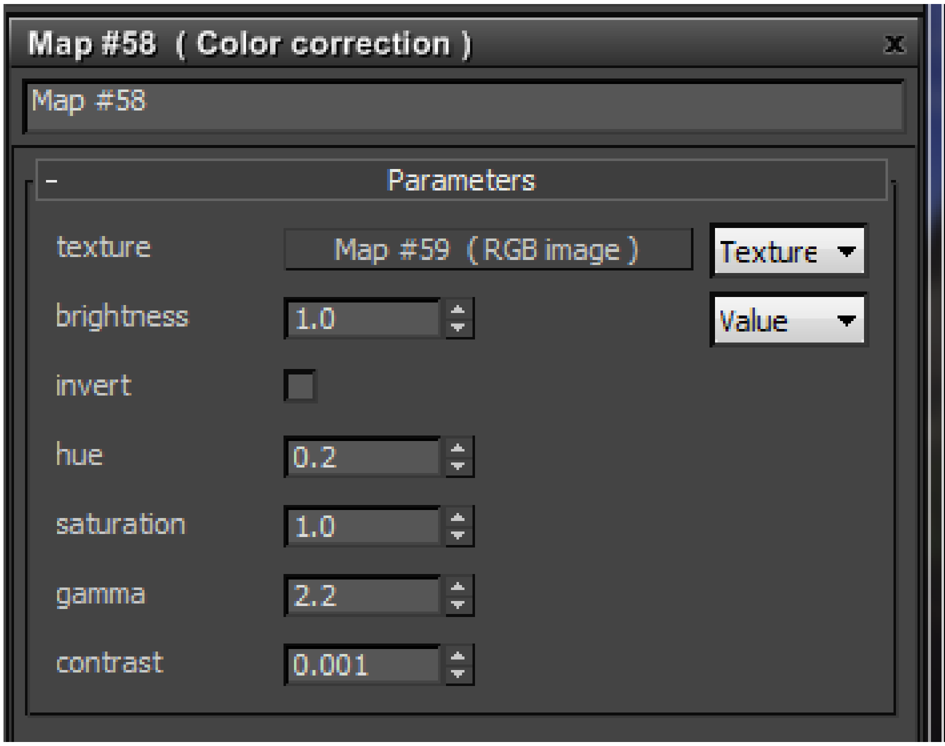


Figure 1: Color Correction parameters

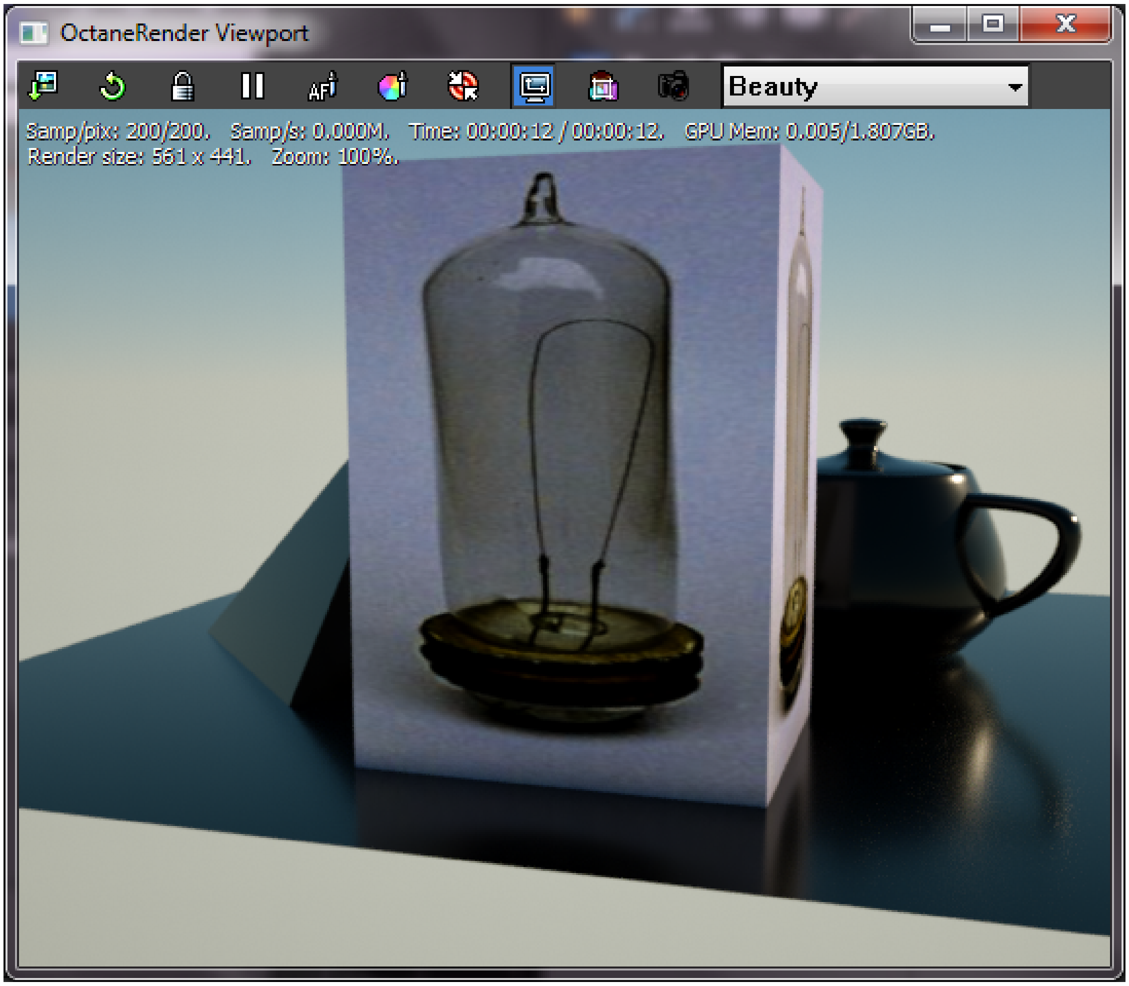


Figure 2: One example of gamma correction

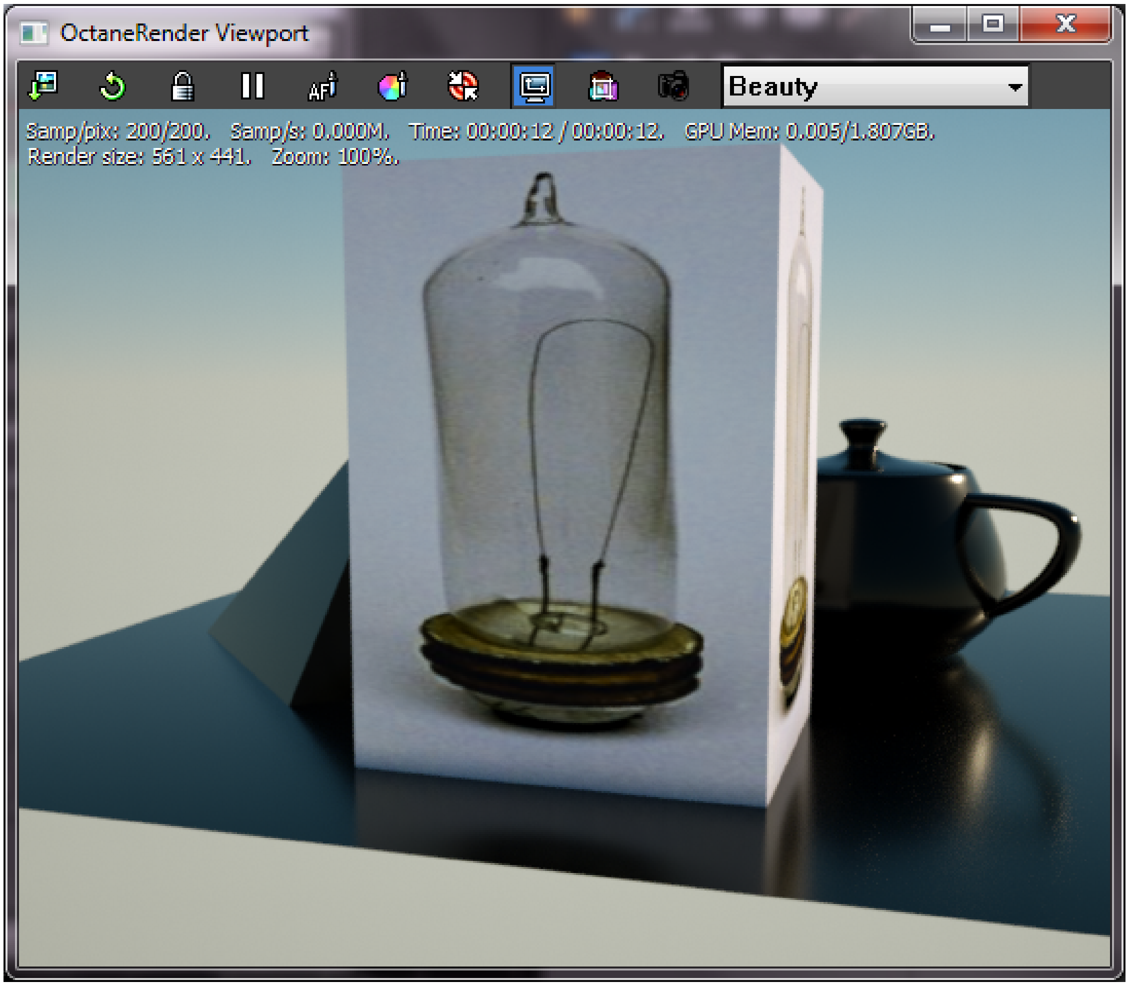


Figure 3: Another example of gamma correction

When using the Color Correction node, the **Node** structure looks like this:

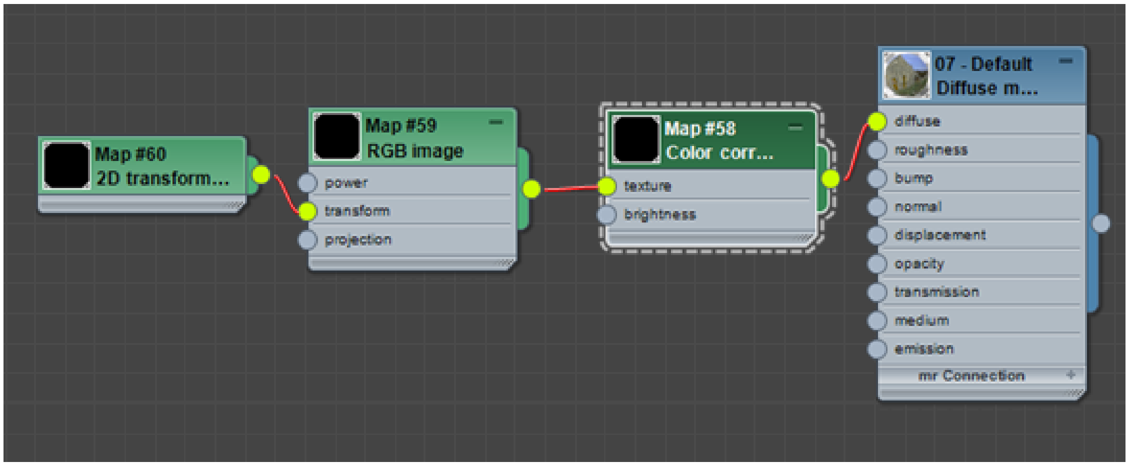


Figure 4: Color Correction node structure

The Color Correction map is between the **Image** map and the **Material**[1] node.

Comparison

The **Comparison** texture uses a logical comparison operator to combine **Textures**[2]. The **Node** takes four inputs. The first two inputs are the Textures to compare. The second two inputs are the comparison results. In Figure 1, **Input A** is a **Noise** texture, and **Input B** is a **Gradient** texture that is mapped from a **Falloff** texture. The **Compare** texture looks at the color values of Inputs A and B. Wherever the color values of A are less than B, a green **RGB** texture maps to the surface. Wherever the color values of Input A are equal or greater than the values of Input B, a red RGB texture maps to the surface. This example is simple, but you can create very complex Textures using the Compare texture, in particular when combined with other Compare textures.

[1] The representation of the surface or volume properties of an object.

[2] Textures are used to add details to a surface. Textures can be procedural or imported raster files.

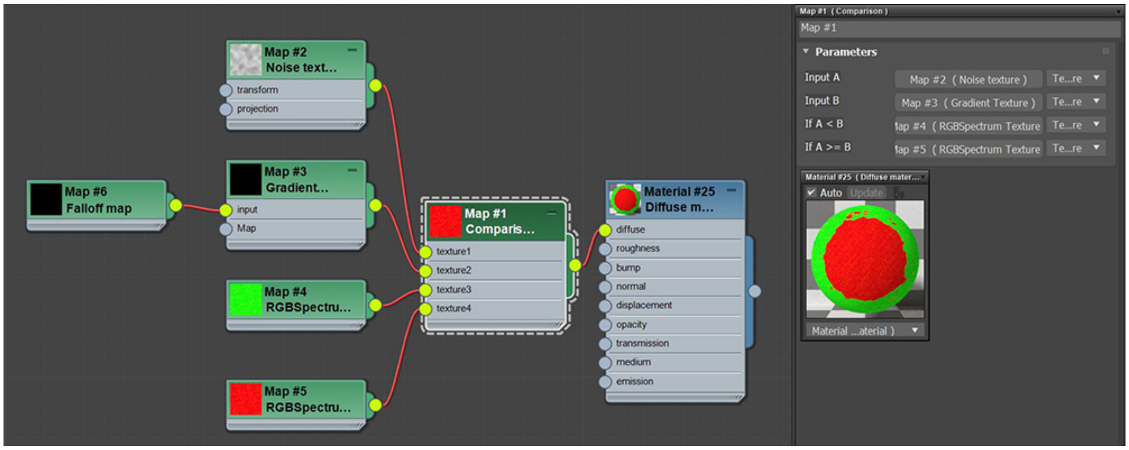


Figure 1: The Comparison texture uses a logical operator to compare two Textures and uses the result to compare two other Textures

Cosine Mix Texture

The **Mix** textures mix two **Textures**[1] together, either linearly or according to a cosine wave. In the example below, a **Checks** mix combines with a **Gaussian Spectrum** using a **Cosine Mix** texture, and connects to a **Diffuse**[2] material's **Diffuse** channel.

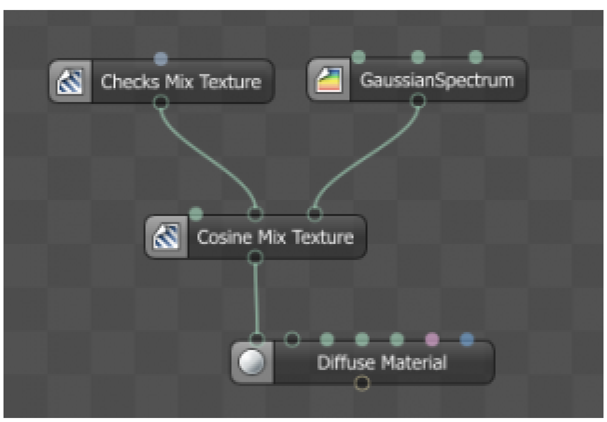


Figure 1: Cosine Mix texture example

---

[1]Textures are used to add details to a surface. Textures can be procedural or imported raster files.
[2]Amount of diffusion, or the reflection of light photons at different angles from an uneven or granular surface. Used for dull, non-reflecting materials or mesh emitters.

Combining a Checks texture with a Gaussian Spectrum by using a Cosine Mix texture, which is set to a **Diffuse material**[1]'s Diffuse channel.

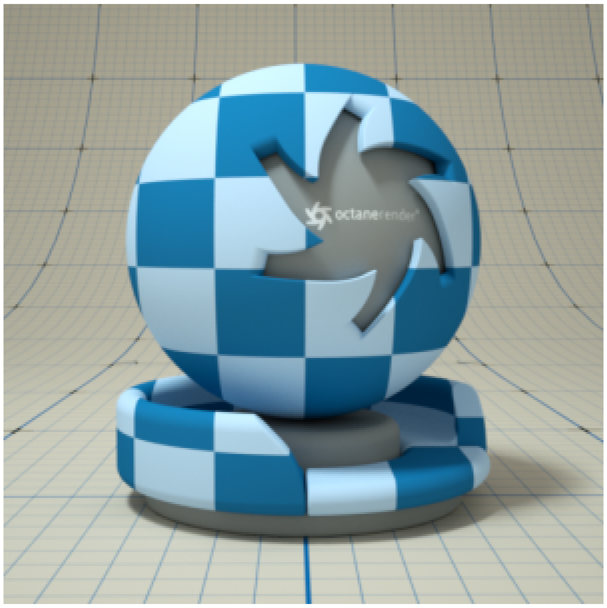


Figure 2: Using a Cosine Mix texture to combine Checks and Gaussian Spectrum textures

Dirt Texture

The **Dirt** texture creates different effects based on ambient occlusion calculations. This **Texture** always returns a random value and can simulate dirt, dust, or wear and tear.

This Texture is often plugged into the **Diffuse**[2], **Bump**, or **Transmission**[3] parameters.

[1] Used for dull, non-reflecting materials or mesh emitters.

[2] Amount of diffusion, or the reflection of light photons at different angles from an uneven or granular surface. Used for dull, non-reflecting materials or mesh emitters.

[3] A surface characteristic that determines if light may pass through a surface volume.

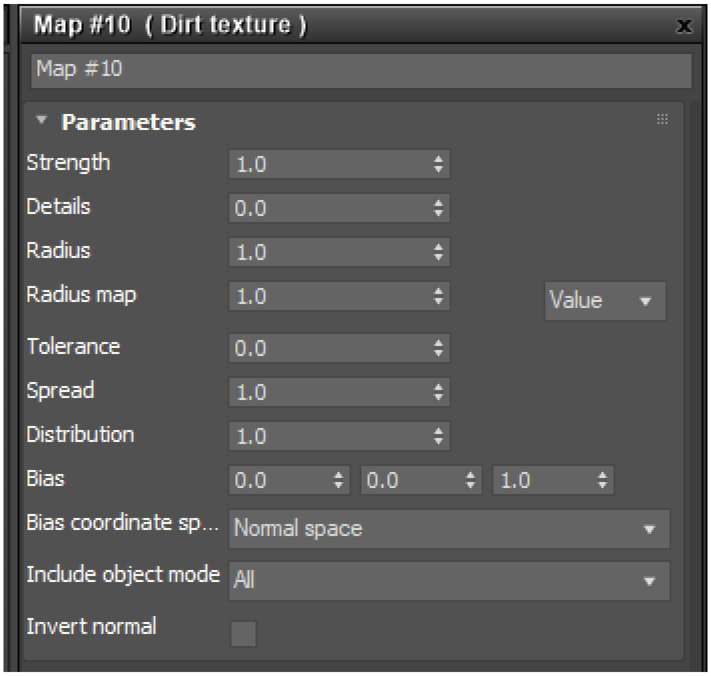
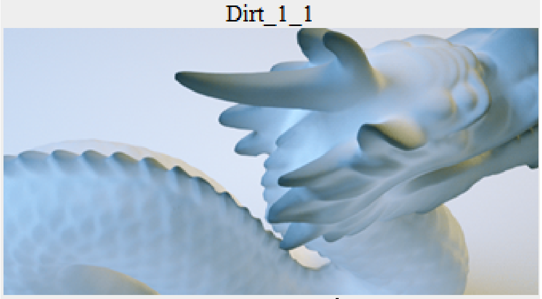


Figure 1: Dirt parameters


No Dirt


Dirt_1_1


Dirt_1_1_Inv


Dirt_1_20

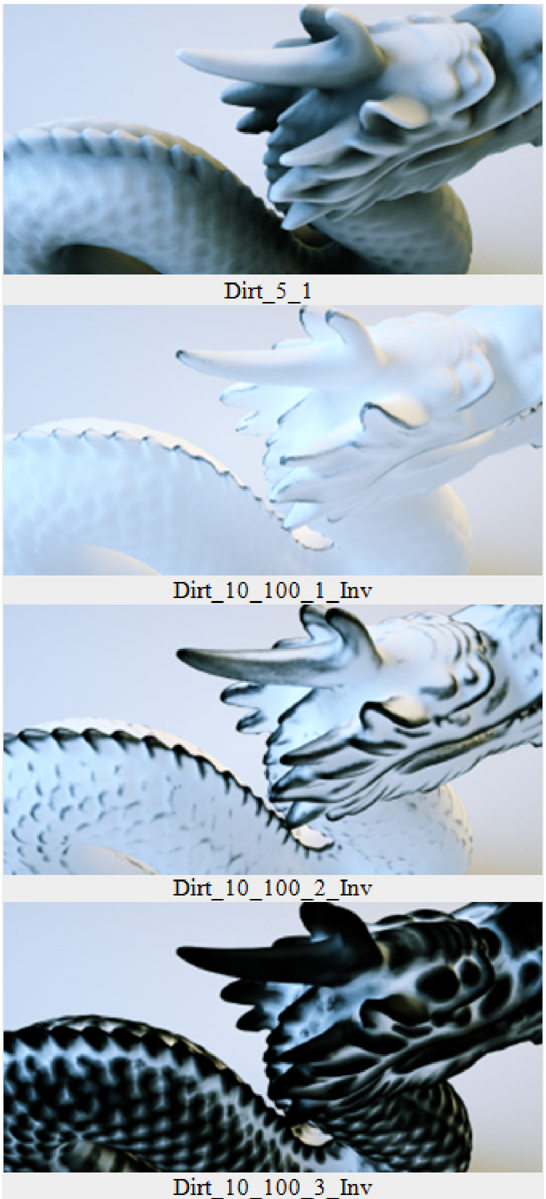


Figure 2: Dirt texture with differing Strength, Details, Radius, and Invert settings

Dirt Texture Parameters

Strength - Controls the Dirt intensity across the geometry surface.

Details - Controls the Details intensity.

Radius - Controls the dirt spread across the model's surface from the recessed parts towards the exposed parts.

Tolerance - Reduces black edges on rough tessellated meshes.

Invert Normal - Reverses the Dirt texture effect based on the normal surface direction.

Falloff Map

The **Falloff** map blends two **Materials**[1], depending on the viewing angle of the material's geometry. This helps simulate, for example, coating effects visible in car shaders, or layered effects visible in velvet cloth or frosted glass.

This also mimics architectural glass for **Glossy**[2] and **Specular**[3] materials by plugging the Texture into the **Material**[4] node's **Opacity** channel, letting light pass through based on the Falloff texture.

---

[1] A set of attributes or parameters that describe surface characteristics.

[2] The measure of how well light is reflected from a surface in the specular direction, the amount and way in which the light is spread around the specular direction, and the change in specular reflection as the specular angle changes. Used for shiny materials such as plastics or metals.

[3] Amount of specular reflection, or the mirror-like reflection of light photons at the same angle. Used for transparent materials such as glass and water.

[4] The representation of the surface or volume properties of an object.

Figure 1: The Falloff map varies the surface's color, depending on viewing angle

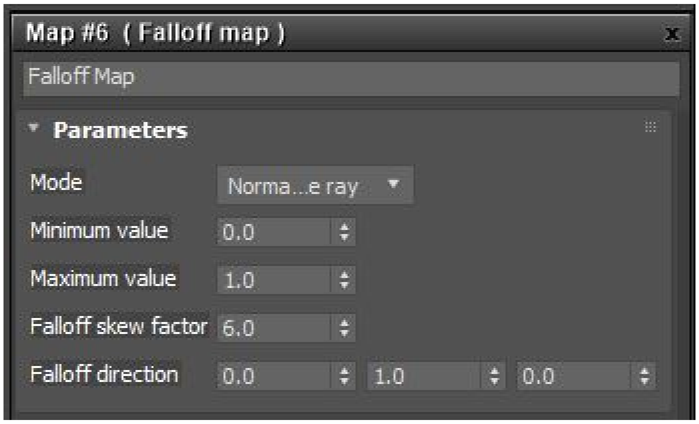


Figure 2: The Falloff map parameters

Falloff Map Parameters

Mode - Provides directional options for the falloff: **Normal vs. Eye Ray**, **Normal vs. Vector 90 Degrees**, and **Normal vs. Vector 180 Degrees**.

Minimum Value - Controls the blending distance from the falloff edges.

Maximum Value - Controls the blending distance from the camera-facing surface to the edge.

Falloff Skew Factor - Controls the area where the **Min** and **Max** values converge.

Falloff Direction - Controls the falloff's orientation if the **Mode** is set to **Normal vs. Vector 90 Degrees** or **Normal vs. Vector 180 Degrees**.

Invert Texture

The **Invert** texture map reverses the colors or values in a **Texture** or **Procedural** map. In the example below, the Invert texture map flips the black-and-white areas of a **Checks** texture map.

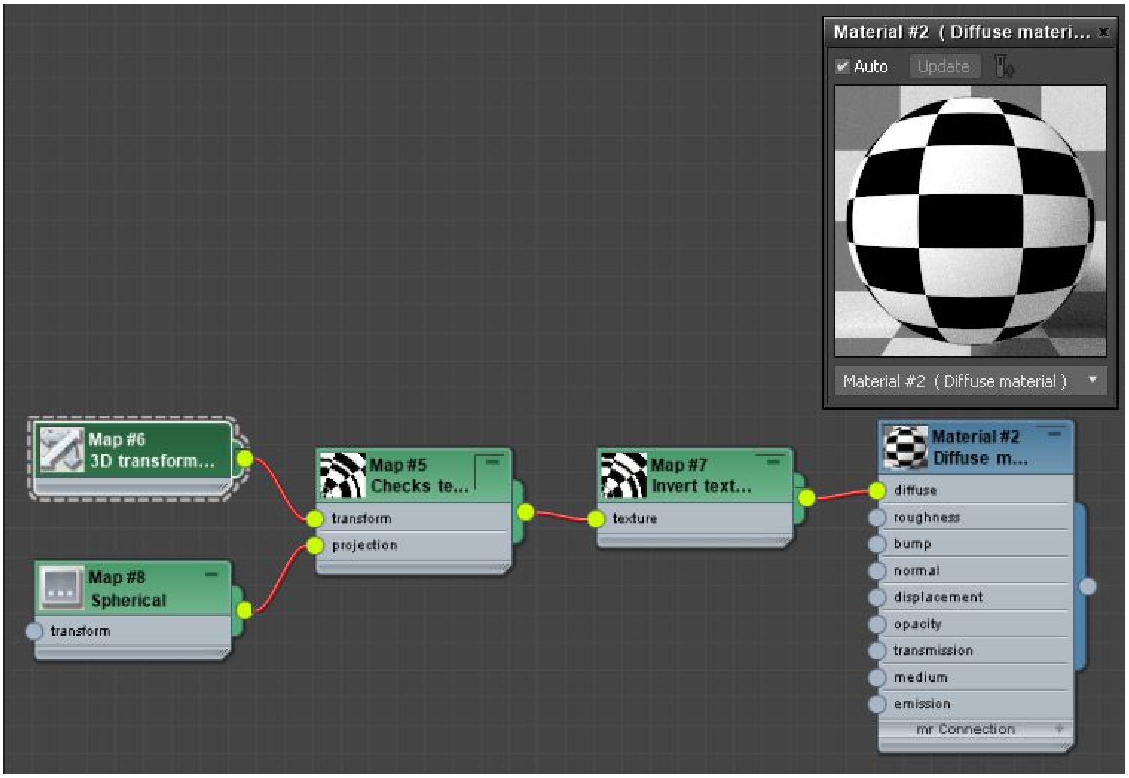
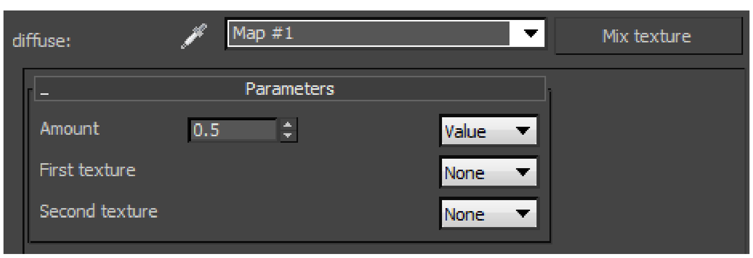


Figure 1: Invert texture used on a Checks texture

Mix Texture

The **Mix** texture mixes two **Textures**[1] together with a value, **Color**, or **Texture** map. Values in-between blend the two textures together in a linear fashion.



[1] Textures are used to add details to a surface. Textures can be procedural or imported raster files.

Figure 1: Mix texture parameters

In the example below, a Mix texture blends the **Marble** texture map with a **Checks** texture map . The **Blend** value is set to **0.05**.

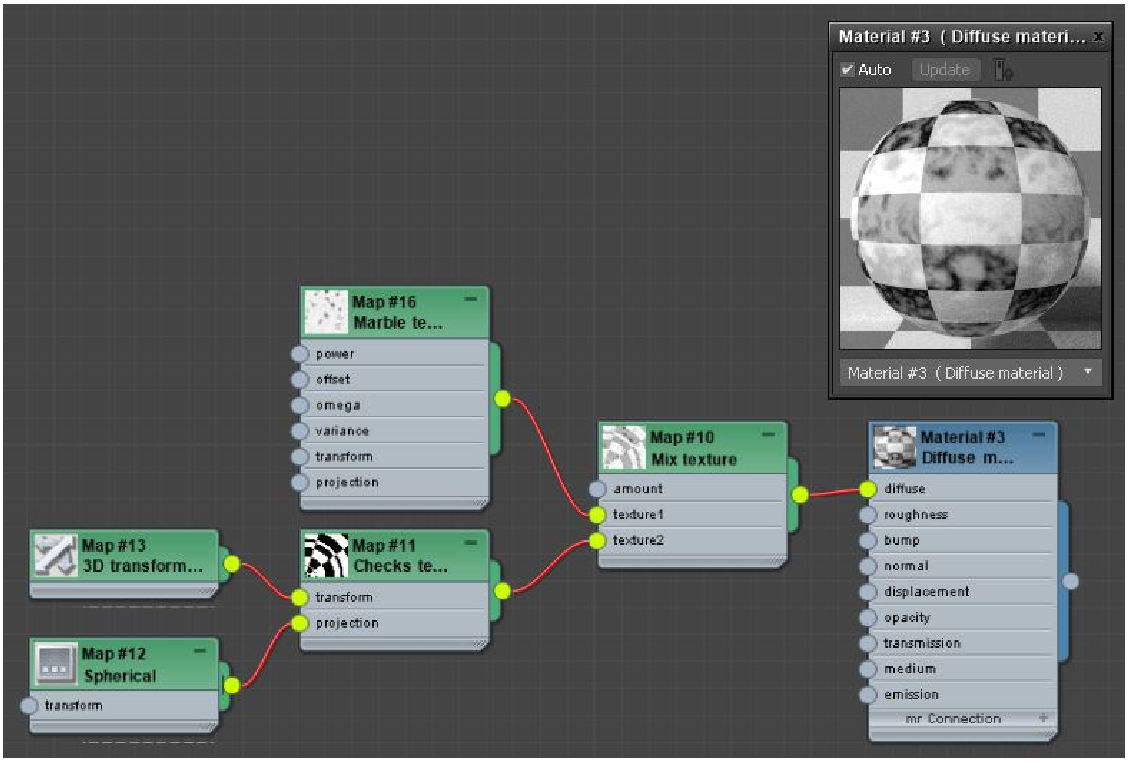


Figure 1: A Mix texture blending Marble and Checks textures together

Multiply Texture

The **Multiply** texture mixes two **Textures**[1] or colors together by overlaying them. Similar to placing two slides over each other, all the values in the image are multiplied. In the example below, the chestnut **RGB** image is multiplied by a **Marble** texture.

---

[1]Textures are used to add details to a surface. Textures can be procedural or imported raster files.

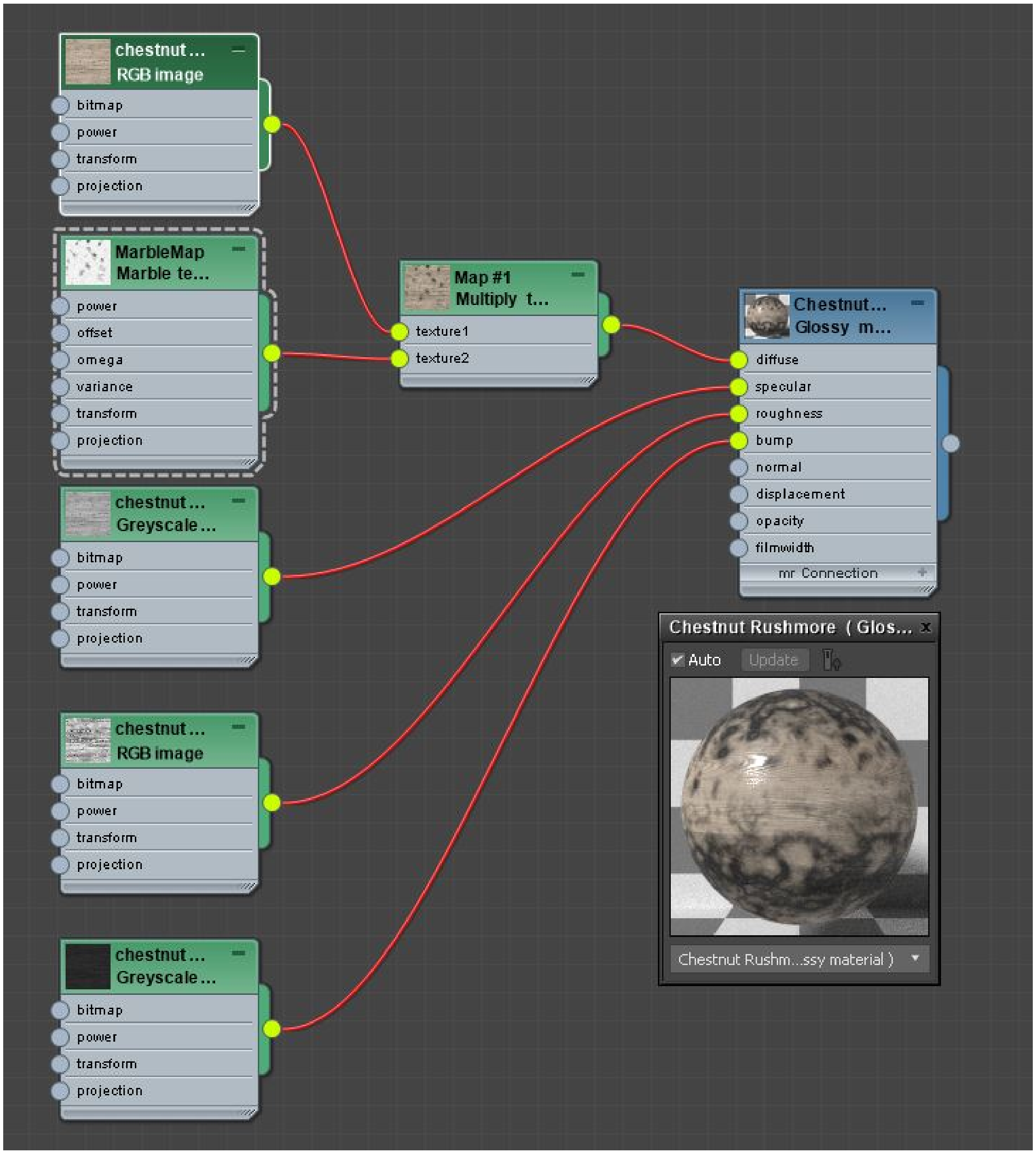


Figure 1: A Marble texture multiplying a chestnut RGB image texture

Polygon Side

The **Polygon Side** texture renders white on the front face and black on the backface of a polygon. You can use this for backface culling by putting this **Texture** into the **Opacity** channel. You can create double-sided **Materials**[1] by placing it into a **Mix** material, or create double-sided **Textures**[2] by placing it into a **Mix** texture.

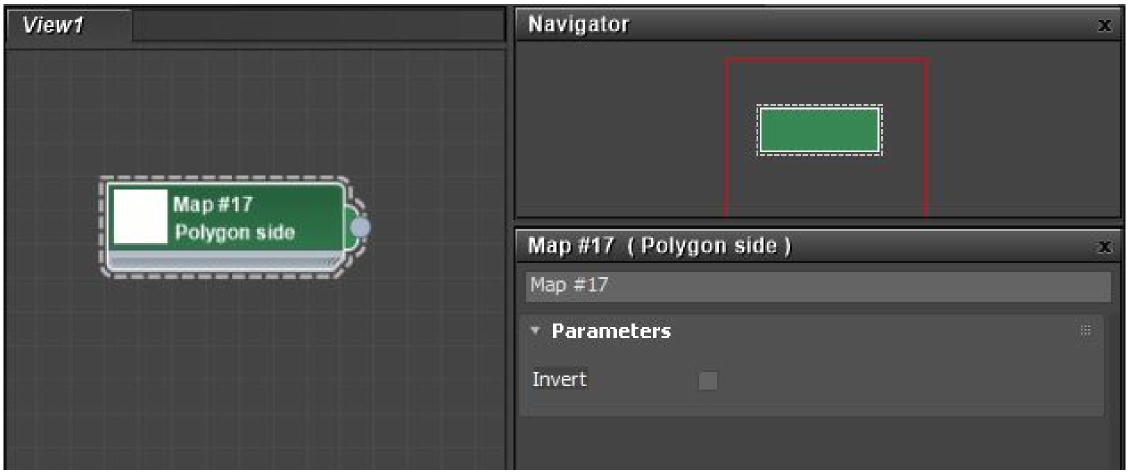


Figure 1: Polygon Side parameters

Subtract Texture

The **Subtract** texture subtracts the value of one **Texture** from another, similar to the Subtract Layer mode in Photoshop®.

[1]A set of attributes or parameters that describe surface characteristics.

[2]Textures are used to add details to a surface. Textures can be procedural or imported raster files.

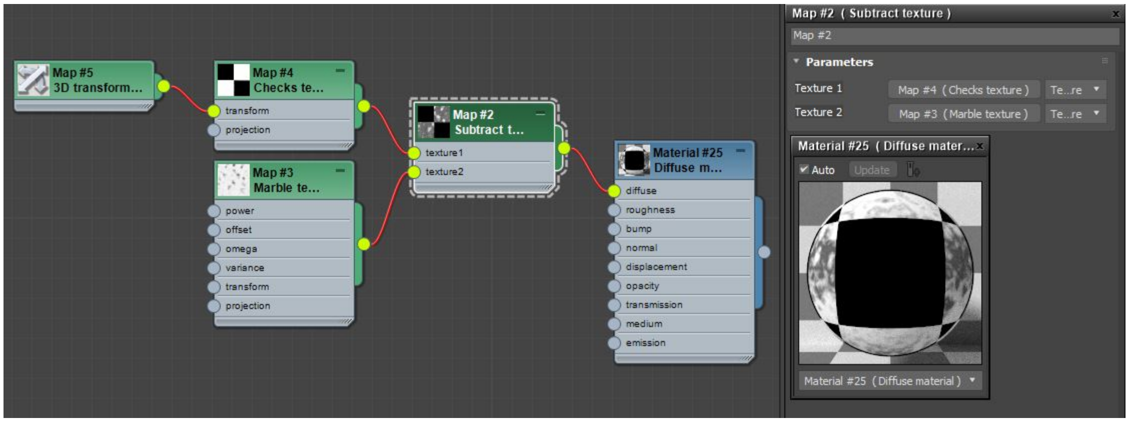


Figure 1: The Subtract texture subtracting parts of a Marble texture with a Checks texture

W Coordinate

The **W Coordinate** texture accesses the OctaneRender® W coordinate system, which places **Gradients** on hair geometry. This is where you store inherent hair gradient interpolation, along with hair data exported from other 3D modelling applications. It takes into account the basis of the interpolation set in the **Octane Global Preferences** tab.

You must generate the W with a supported third-party plugin for this to work. When using the W Coordinate texture, make sure the **Enable Custom Hair Ws** option is enabled in the **Octane Global Preferences** tab. When you set **W Coordinate Map** as **Gradient Texture** input for hair material, the Texture uses third-party-generated Ws instead of automatic Ws from the host application.

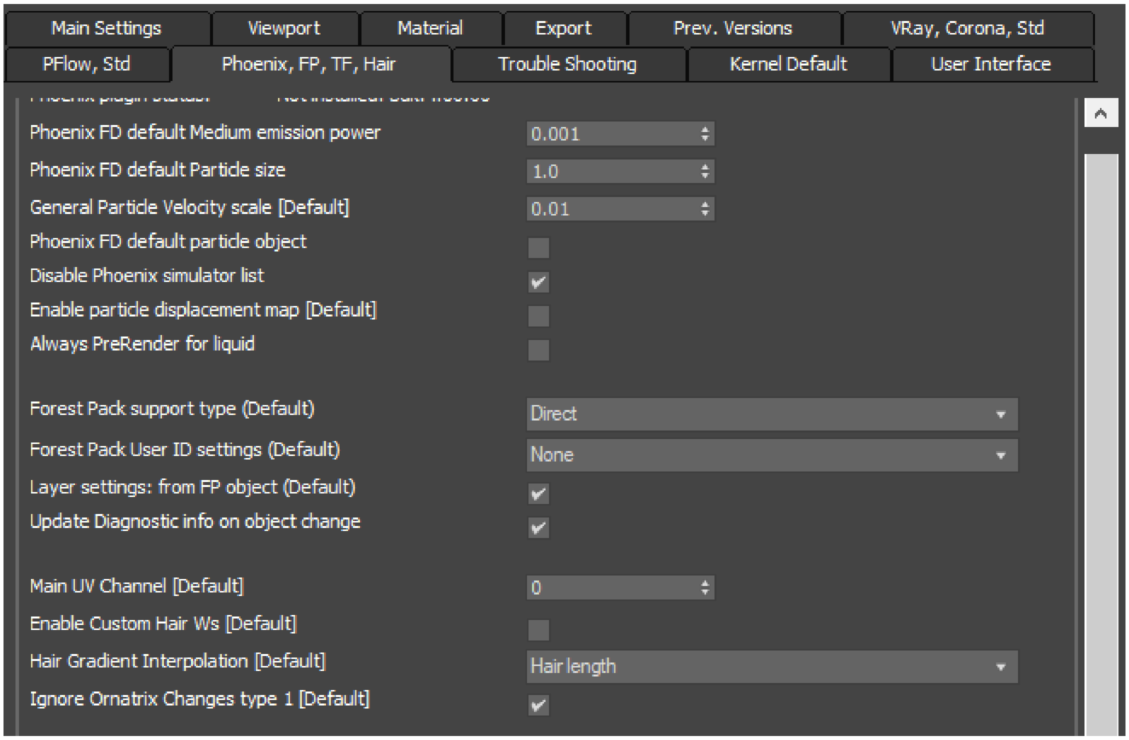


Figure 1: Setting the Hair Gradient Interpolation in the Octane Global Preferences

When you designate the W Coordinate texture as the **Input** texture of a **Gradient** texture, it tells OctaneRender® to render the inputs as a gradient mapping based on either the hair length or the segment count per strand.

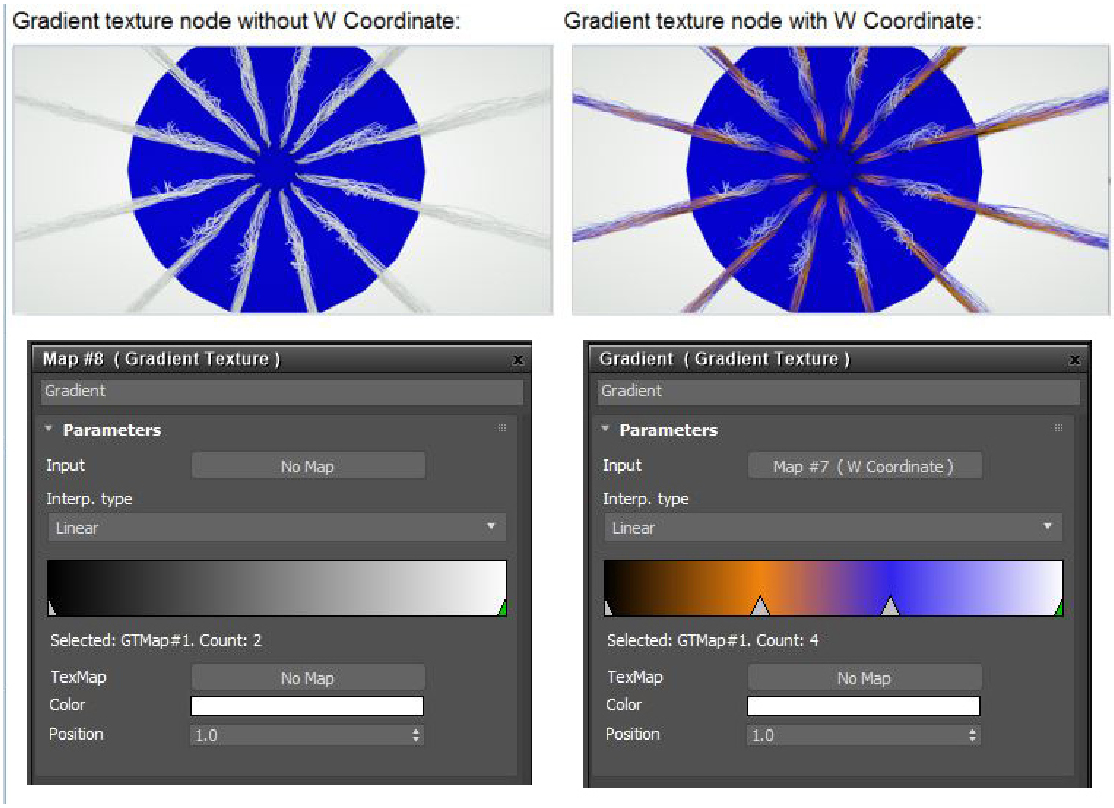


Figure 2: Hair segment comparison between a Gradient texture node without the W Coordinate, and one with the W Coordinate

Transforms

Transforms provide a set of **Nodes** in the **Slate Material**[1] **Editor** that move, scale, and rotate **Texture** maps on an Object's surface. You can access these Nodes from the **Maps** rollout in the Slate Material Editor.

[1] The representation of the surface or volume properties of an object.

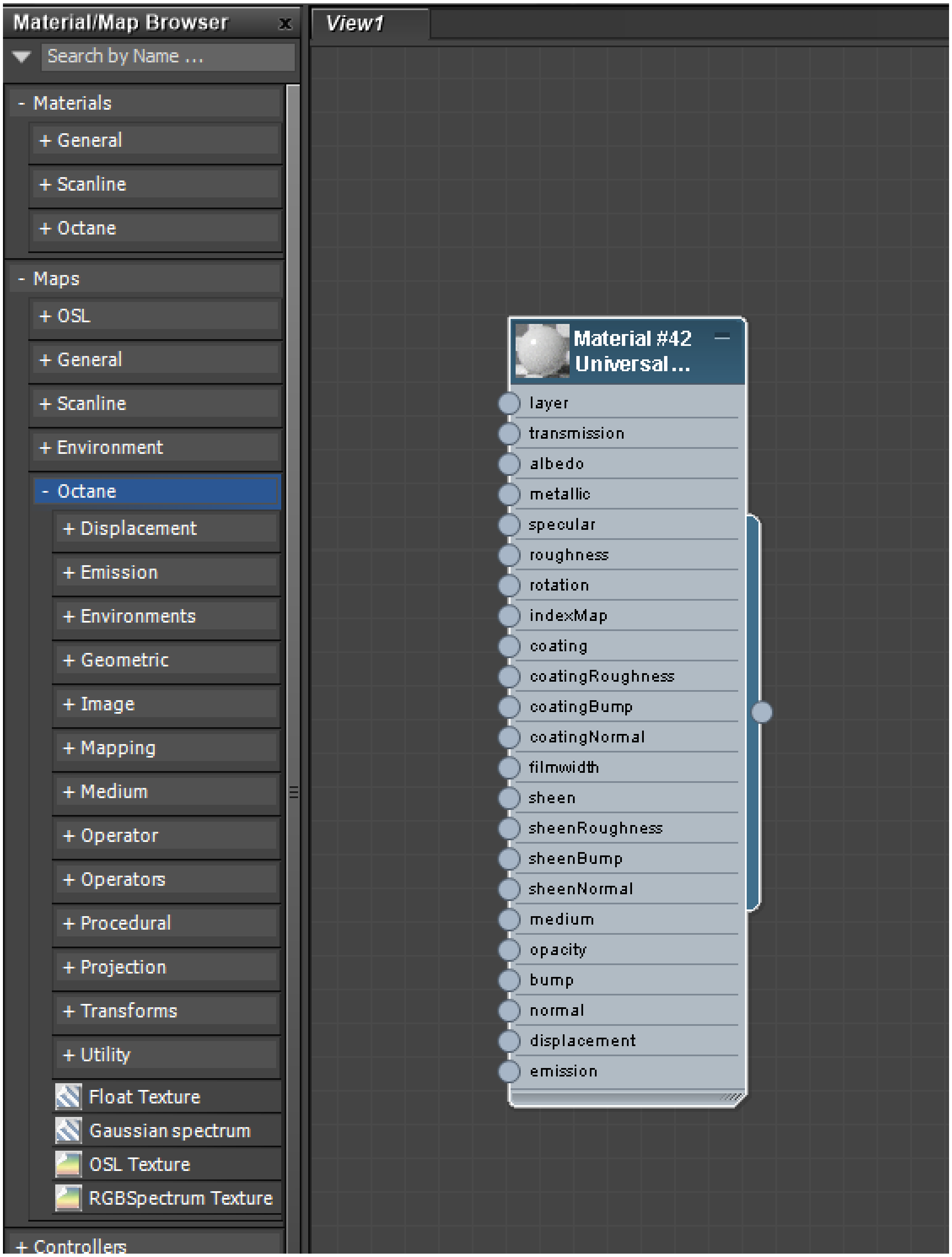

Material/Map Browser
Search by Name ...
- Materials
+ General
+ Scanline
+ Octane
- Maps
+ OSL
+ General
+ Scanline
+ Environment
- Octane
+ Displacement
+ Emission
+ Environments
+ Geometric
+ Image
+ Mapping
+ Medium
+ Operator
+ Operators
+ Procedural
+ Projection
+ Transforms
+ Utility
Float Texture
Gaussian spectrum
OSL Texture
RGBSpectrum Texture
+ Controllers
View1
Material #42
Universal ...
layer
transmission
albedo
metallic
specular
roughness
rotation
indexMap
coating
coatingRoughness
coatingBump
coatingNormal
filmwidth
sheen
sheenRoughness
sheenBump
sheenNormal
medium
opacity
bump
normal
displacement
emission

Figure 1: Accessing Map nodes in the Slate Material Editor

2D Transformation

2D Transformation sets the orientation of other **Textures**[1]. It connects to the **Transform** input pin on any given **Texture** map. It provides planar positioning with Texture maps - unlike the **3D Transformation** node, which provides positioning on all three axes.

Figure 1: The 2D Transformation node

3D Transformation

3D Transformation sets the orientation of other **Textures**[2]. It connects to the **Transform** input pin on any given Texture map. It provides true 3D positioning of Texture maps - unlike the **2D Transformation** node, which only provides planar positioning for Texture maps.

---

[1] Textures are used to add details to a surface. Textures can be procedural or imported raster files.
[2] Textures are used to add details to a surface. Textures can be procedural or imported raster files.

Figure 1: The 3D Transformation node

Rotation

The **Rotation** map modifies rotational data for a **Projection** map (**Box**, **Cylindrical**, etc.). In the following example, the Rotation map connects to the **Spherical** map, which determines the Projection of a **Checks** map.

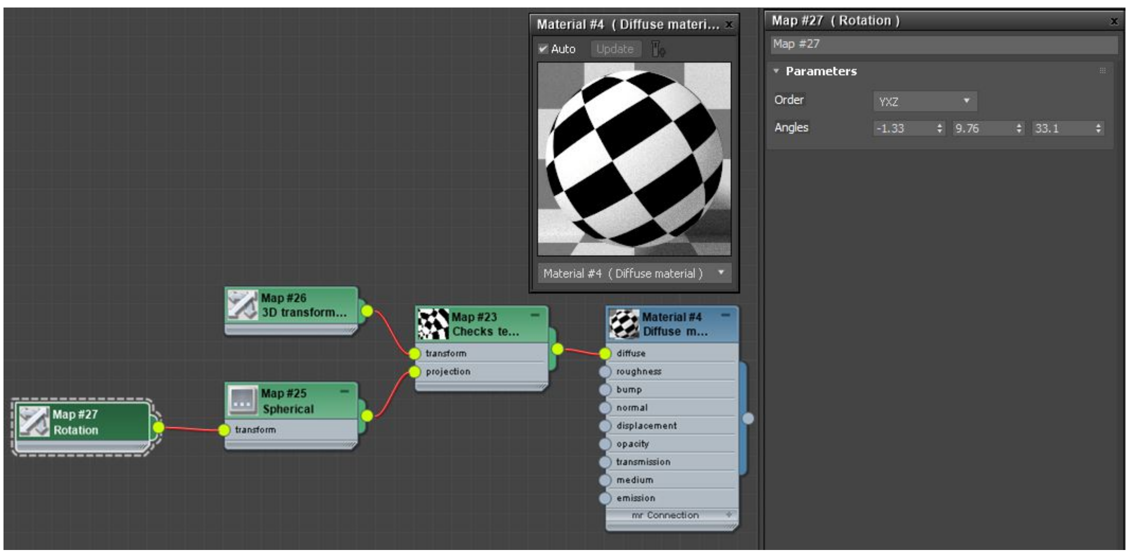


Figure 1: A Rotation map connected to a Spherical map

Scale

The **Scale** map controls a **Projection** map's transform data. In the following example, the Scale map connects to the **Spherical** map, altering the **Scale** parameter's **X** data by **5** units.

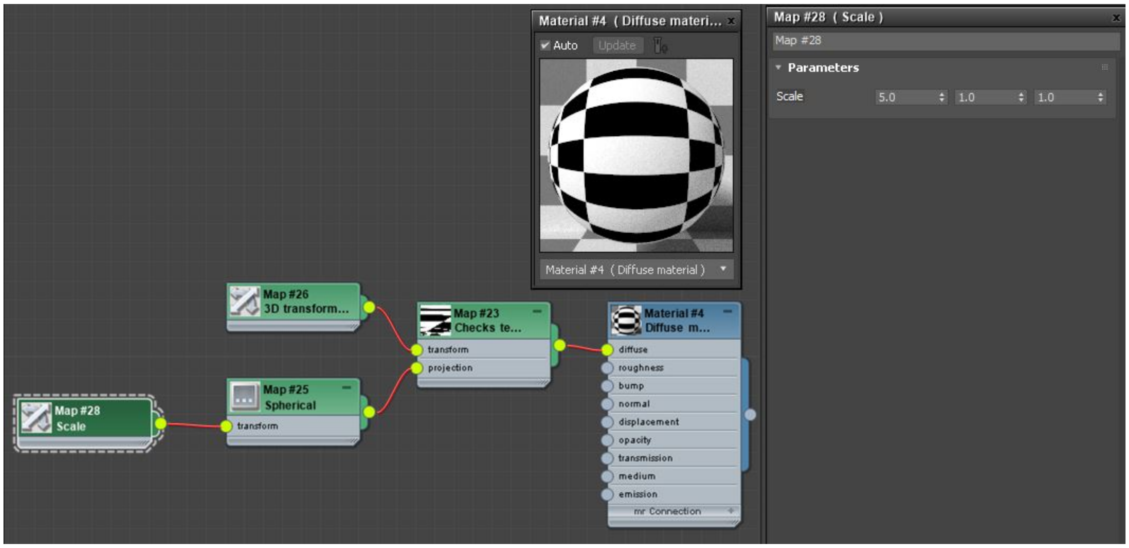


Figure 1: A Scale map connected to a Spherical map

UVW Transform

The **UVW Transform** texture takes an **Input** texture and applies a map to transform the Input texture’s UV layout on top of the Input texture’s own UV coordinate transformation.

Figure 1: The UVW Transform texture applied to a cube

The UVW Transform texture works with other mapping textures like the **Triplanar** map texture, **Mix** texture, **Cosine Mix** texture, the logical Texture maps (**Comparison**), or arithmetic Texture maps (**Add**, **Subtract**, **Multiply**).

Transform Value

Transform Value sets the other **Textures**[1]' orientation. It connects to the **Transform** input pin on any given **Texture** map and provides true 3D positioning for Texture maps, unlike the **2D Transformation** node, which provides planar positioning for Texture maps.

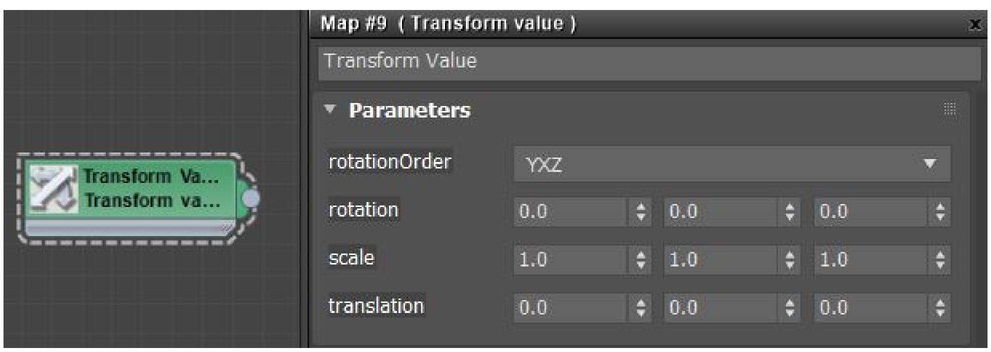


Figure 1: Transform Value parameters

---

[1] Textures are used to add details to a surface. Textures can be procedural or imported raster files.

Projections

Projections[1] are a set of **Nodes** in the **Slate Material**[2] **Editor** that orient **Texture** maps on an Object's surface. You can access these Nodes from the **Maps** rollout in the Slate Material Editor.

[1] Methods for orienting 2D texture maps onto 3D surfaces.

[2] The representation of the surface or volume properties of an object.

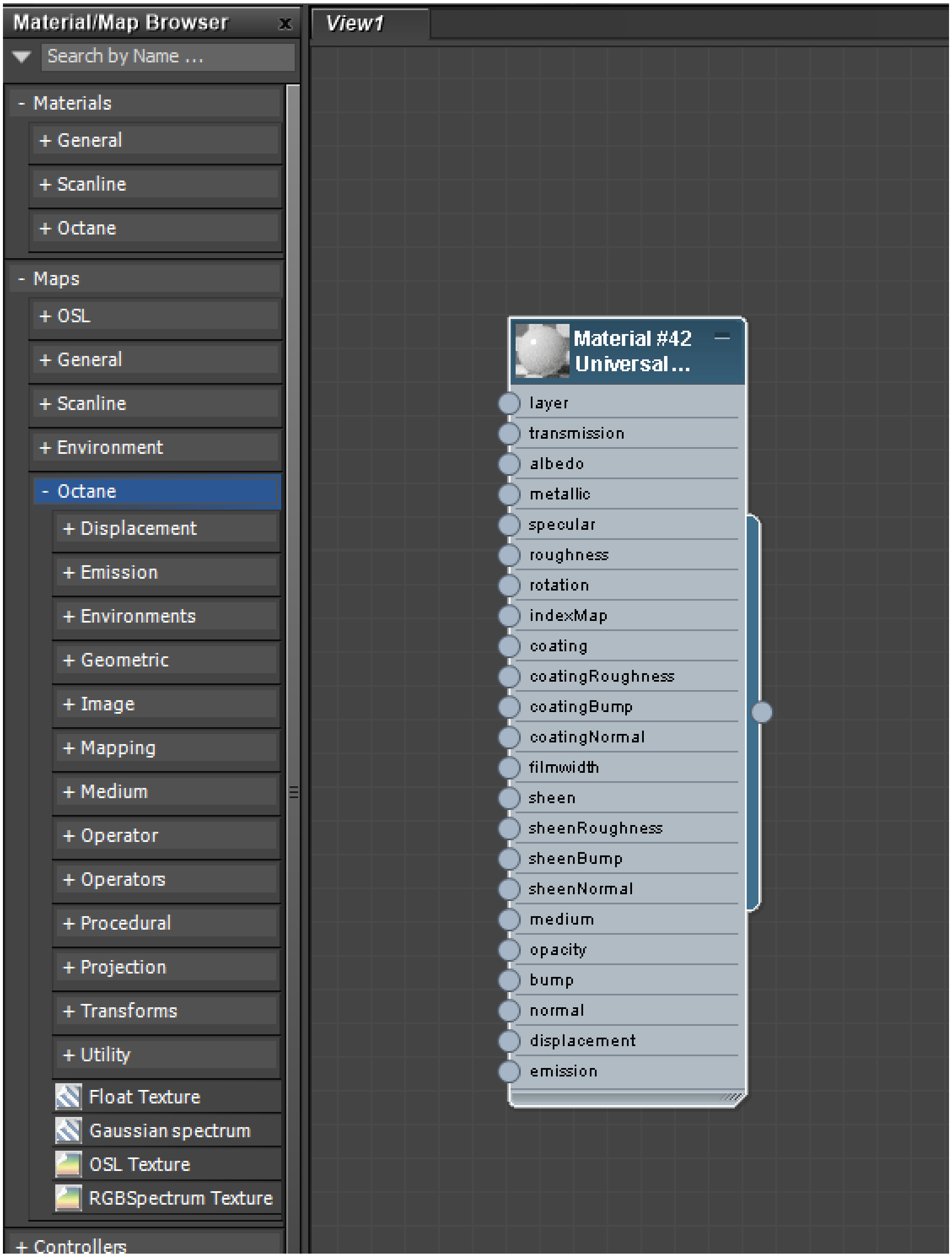

Material/Map Browser
Search by Name ...
- Materials
+ General
+ Scanline
+ Octane
- Maps
+ OSL
+ General
+ Scanline
+ Environment
- Octane
+ Displacement
+ Emission
+ Environments
+ Geometric
+ Image
+ Mapping
+ Medium
+ Operator
+ Operators
+ Procedural
+ Projection
+ Transforms
+ Utility
Float Texture
Gaussian spectrum
OSL Texture
RGBSpectrum Texture
+ Controllers
View1
Material #42
Universal ...
layer
transmission
albedo
metallic
specular
roughness
rotation
indexMap
coating
coatingRoughness
coatingBump
coatingNormal
filmwidth
sheen
sheenRoughness
sheenBump
sheenNormal
medium
opacity
bump
normal
displacement
emission

Figure 1: Accessing Map nodes in the Slate Material Editor

Box

The **Box** map is also known as cube mapping. This **Projection** is an extension of **XYZ To UVW** mapping. It picks a different Projection axis, depending on the normal. This gives a quick way to map a **Texture** on any **Object** without too much distortion, but with a lot of possible seams.

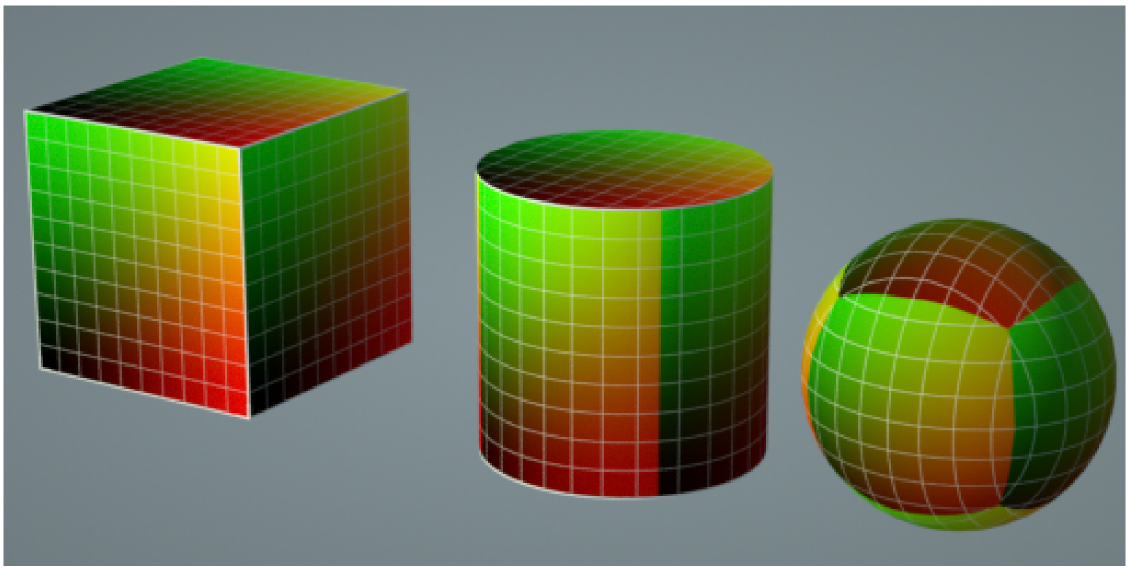


Figure 1: Box maps

Cylindrical

Cylindrical mapping wraps **Texture** maps on a surface with a cylindrical shape.

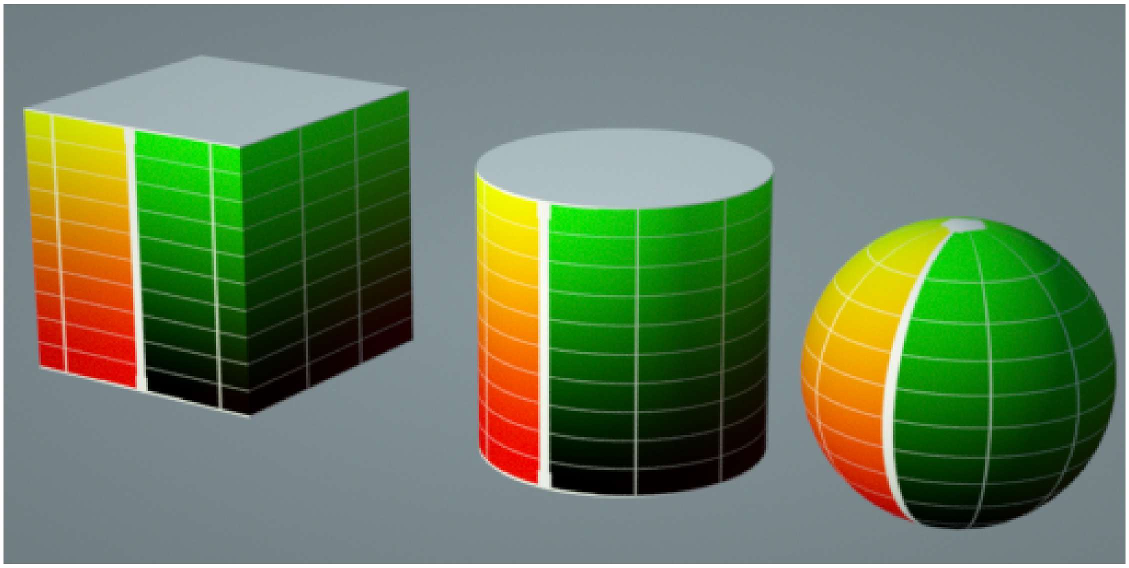


Figure 1: Cylindrical maps

Mesh UV Projection

The **Mesh UV** projection has a single parameter to extract a specified **UV Set** from the applicable **Mesh**.

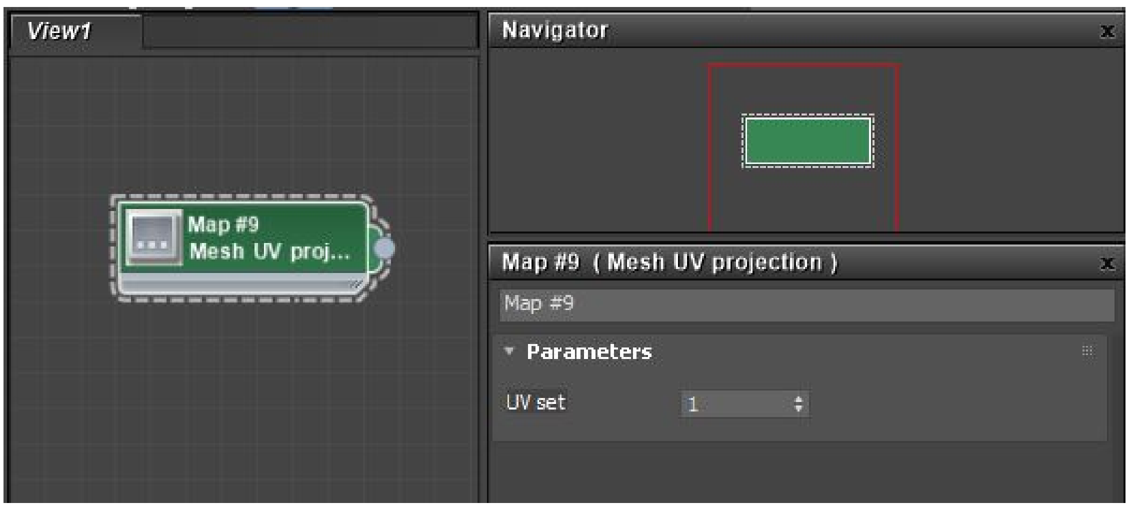


Figure 1: Mesh UV Set parameter

OSL Projection

The **OSL** projection node is a scriptable **Node** where you write OSL (**Open Shader Language**[1]) scripts using the defined arbitrary **Projection** types. It works similar to an **OSL** texture node, but connects to a **Projection** input. To learn about the generic OSL standard, information is provided from the OSL Readme and PDF documentation.

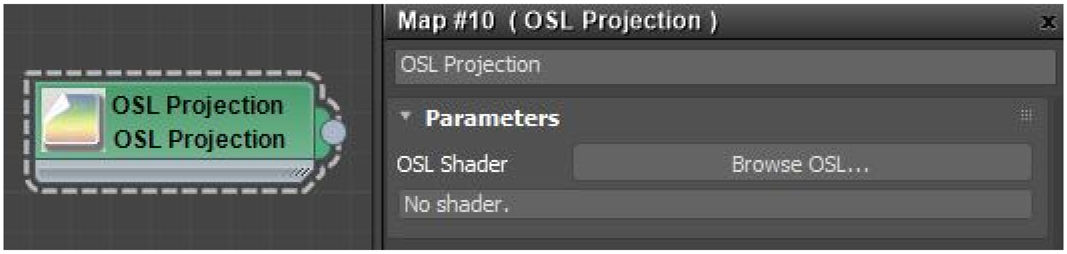


Figure 1: The OSL Projection parameters

OSL Delayed UV

The **OSL Delayed UV** projection node is a scriptable **Node** where you write OSL scripts using the defined **UV** projection type.

Figure 1: The OSL UV projection does not have any specific parameters

Perspective

[1]A shading language developed by Sony Pictures Imageworks. There are multiple render engines that utilize OSL as it is particularly suited for physically-based renderers.

Perspective mapping takes the world space coordinates and divides the X and Y coordinates by the Z coordinate (Figure 1). One way this is useful is by using a texture with this projection as the distribution, with black border mode. You can also use it for camera mapping.

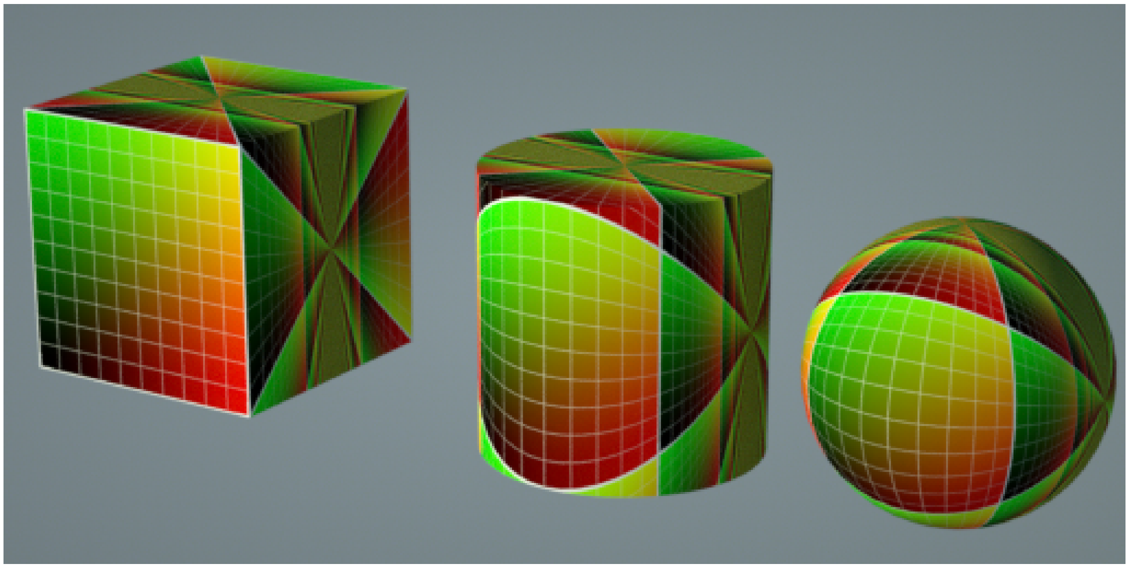


Figure 1: Perspective maps

Spherical

The **Spherical** map is used for **Environment** textures and **IES**[1] Light distributions. It performs latitude-longitude mapping for the U and V coordinates, and for **Procedural** textures, the W coordinate is the distance from the origin.

---

[1] An IES light is the lighting information representing the real-world lighting values for specific light fixtures. For more information, visit http://www.ies.org/lighting/.

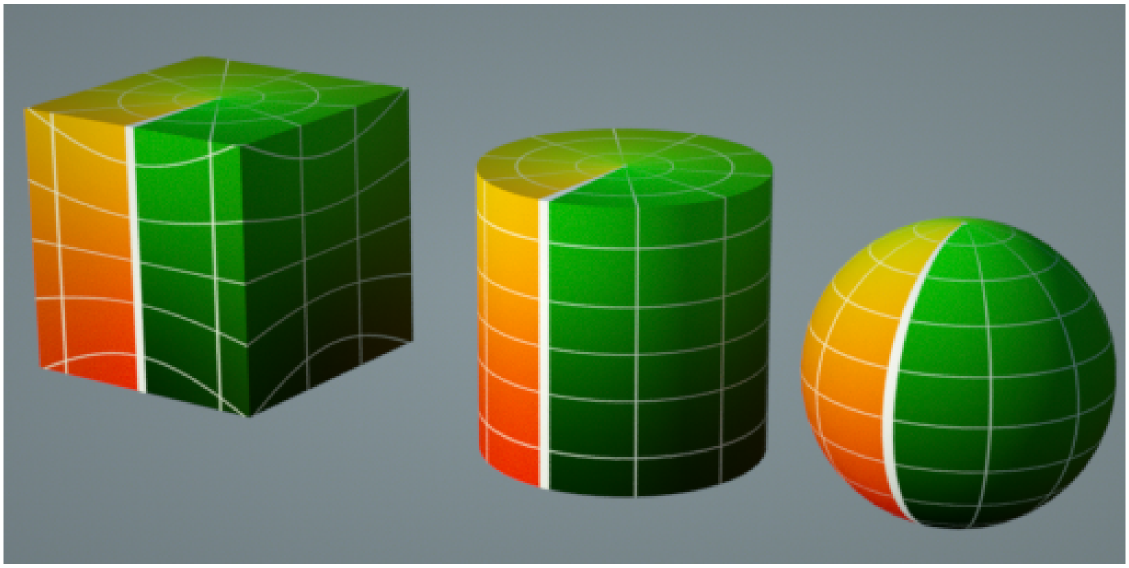


Figure 1: Spherical maps

Triplanar Map

The **Triplanar** map assigns 2D **Textures**[1] on **Objects** that do not have UV coordinates. This **Node** works by projecting one or more Textures along either Object or world space X,Y, and Z axes. In the following image (Figure 1), different colored **RGB Spectrum** maps connect to all six texture position inputs (**TexturePosX**, **TextureNegX**, **TexturePosY**, **TextureNegY**, **TexturePosZ**, **TextureNegZ**) of the Triplanar map node. The result is illustrated in Figure 2.

---

[1]Textures are used to add details to a surface. Textures can be procedural or imported raster files.

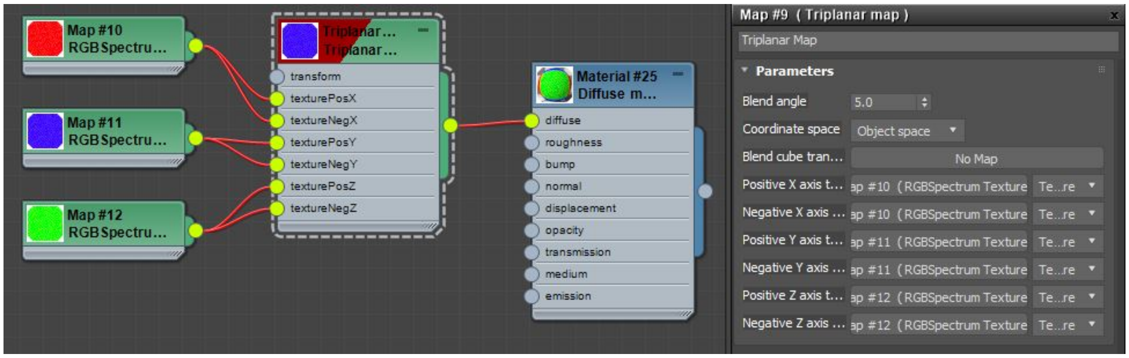


Figure 1: RGB Spectrum maps connects to a Triplanar map's six input pins

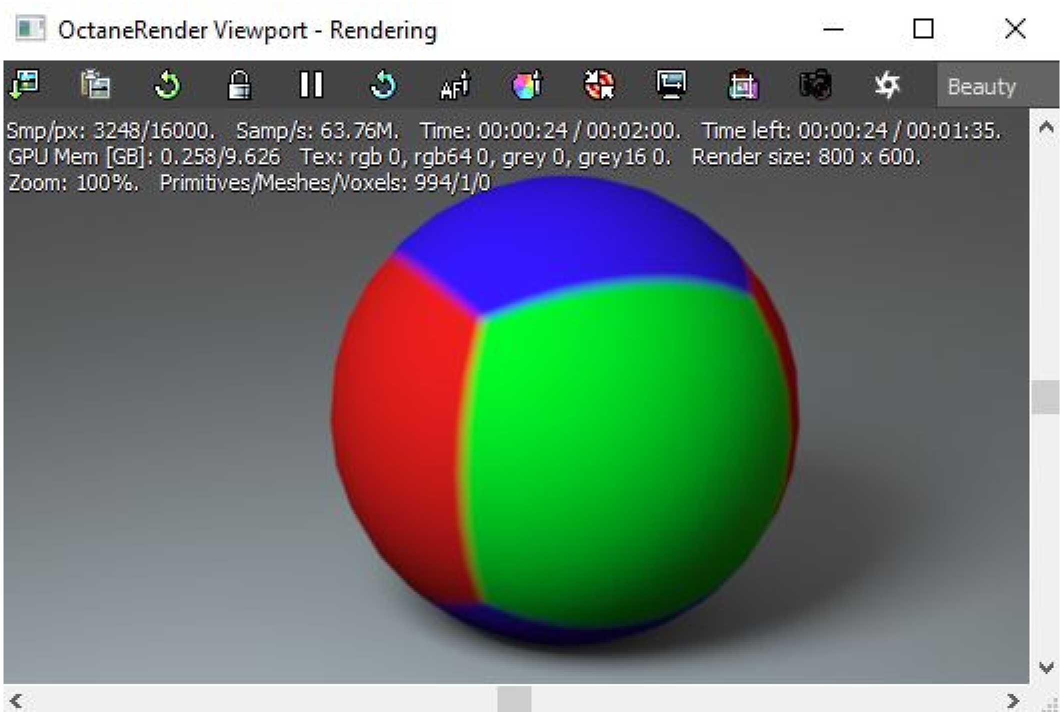


Figure 2: The results of a Triplanar map applied to an Object's surface

XYZ To UVW

The **XYZ To UVW** map is also known as planar projection or flat mapping. This **Projection** map takes the coordinates in world or object space and use them as UVW coordinates. For images, only the X and Y coordinates are relevant, which are mapped to U and V. In other words, the images project flat mapping along the Z axis.

Rotating the mapping around the Z axis rotates the image around the center, as the UVW rotation would do.

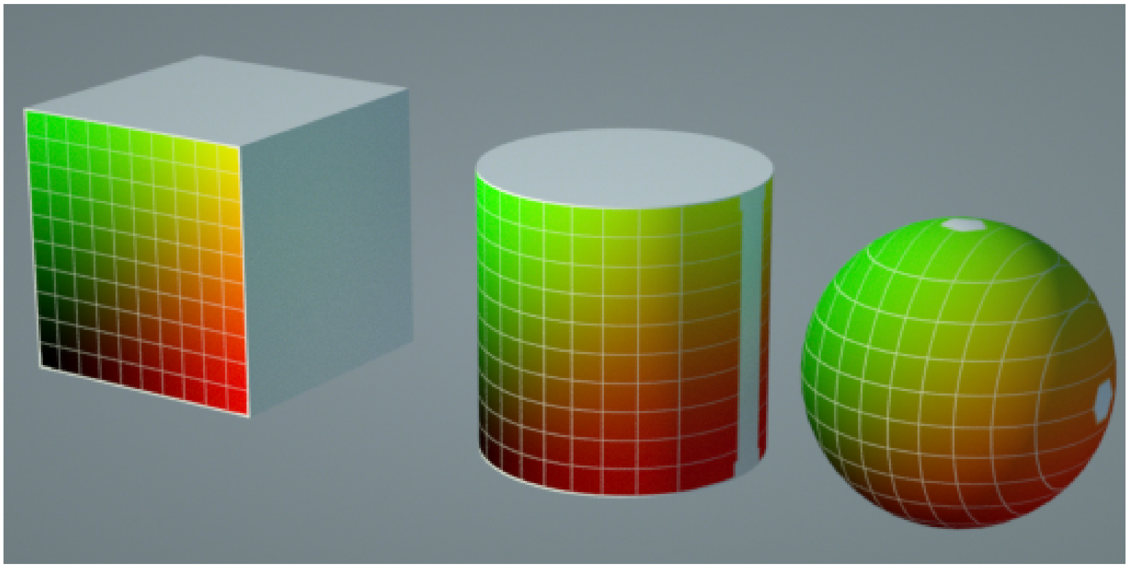


Figure 1: XYZ To UVW map

Emissions

Emission nodes provide a set of **Nodes** in the **Slate Material**[1] **Editor** that generate illumination from an Object's surface. You can access these Nodes from the **Maps** rollout in the Slate Material Editor (Figure 1).

[1]The representation of the surface or volume properties of an object.

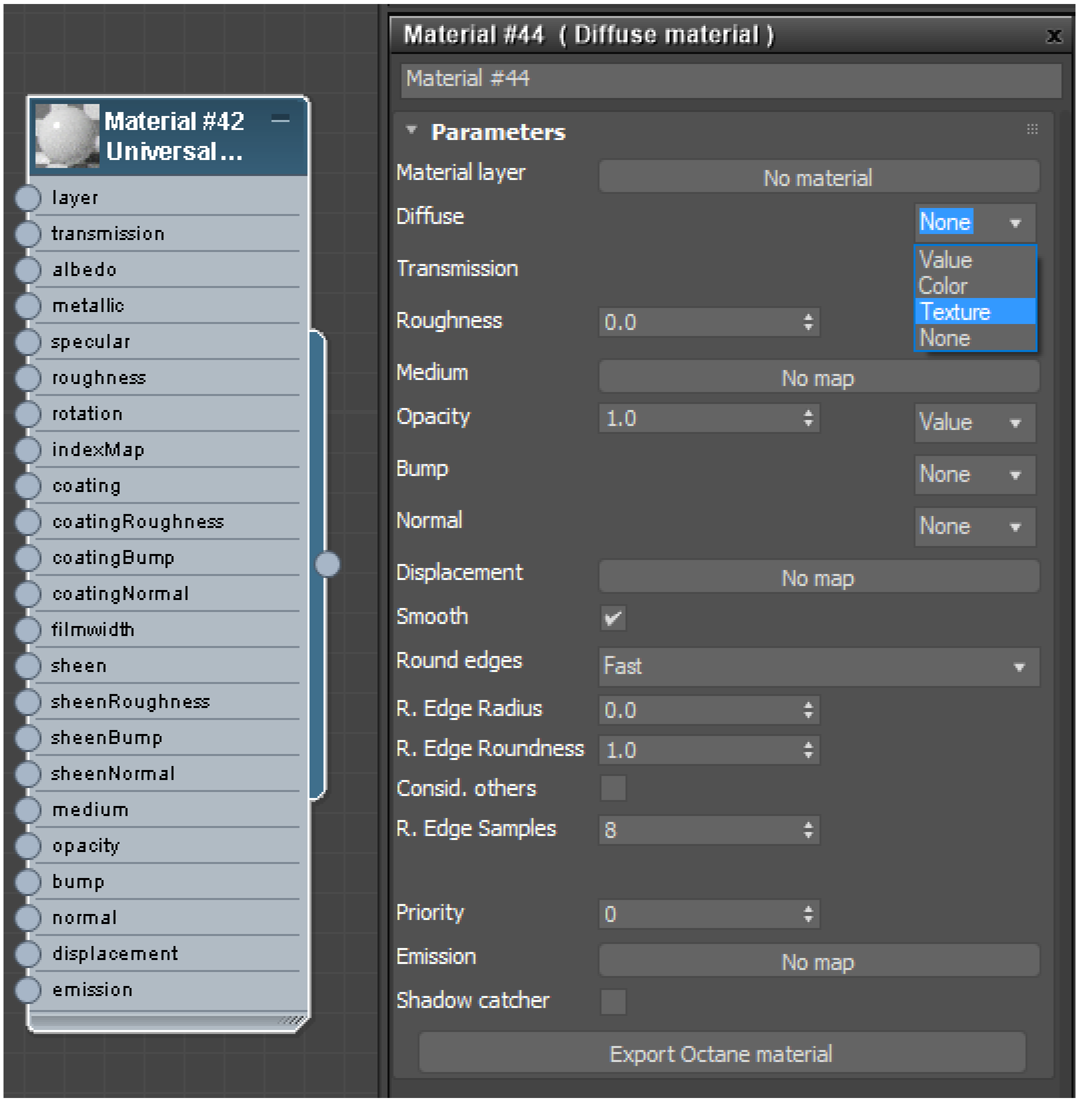


Figure 1: Accessing Map nodes in the Slate Material Editor

Black Body Emission

The **Black Body**[1] emission map lets **Materials**[2] emit light. **Emission** maps work with the **Diffuse**[3] material type and connect to the **Emission** slot. There are two type of Emission maps: the Black Body emission, and the **Texture** emission map (see the Texture Emission topic in this manual for more information).

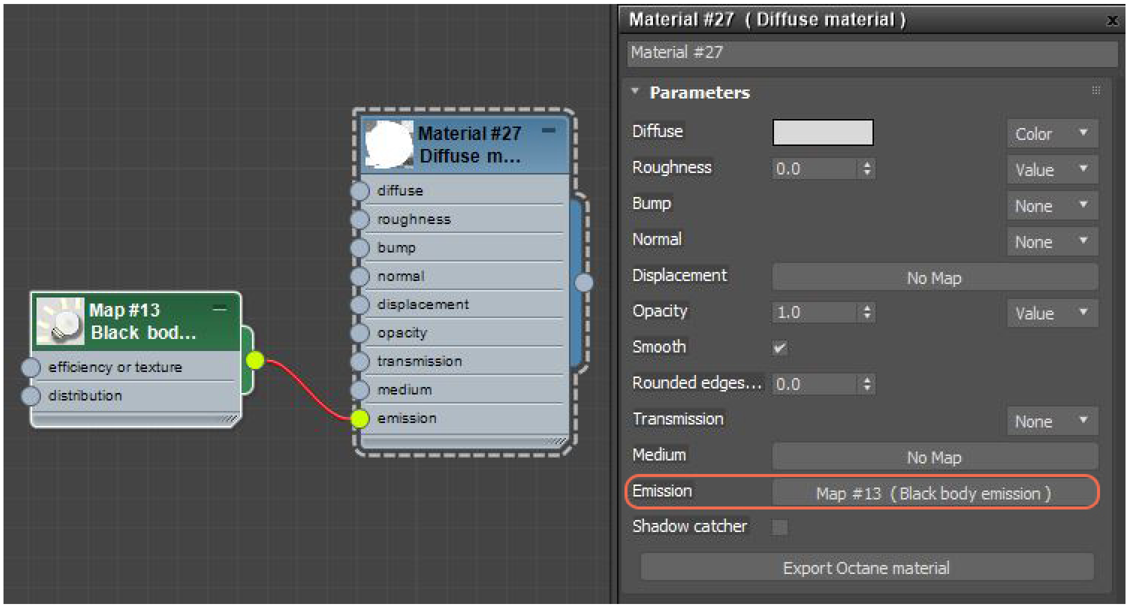


Figure 1: The Black Body emission node connected to a Diffuse material[4]'s Emission slot

The Black Body emission uses **Temperature** (in Kelvin) and **Power** to control the light's color and intensity.

---

[1]An opaque object that emits thermal radiation. In Octane, this is used to designate illumination properties for mesh emitters.
[2]A set of attributes or parameters that describe surface characteristics.
[3]Amount of diffusion, or the reflection of light photons at different angles from an uneven or granular surface. Used for dull, non-reflecting materials or mesh emitters.
[4]Used for dull, non-reflecting materials or mesh emitters.

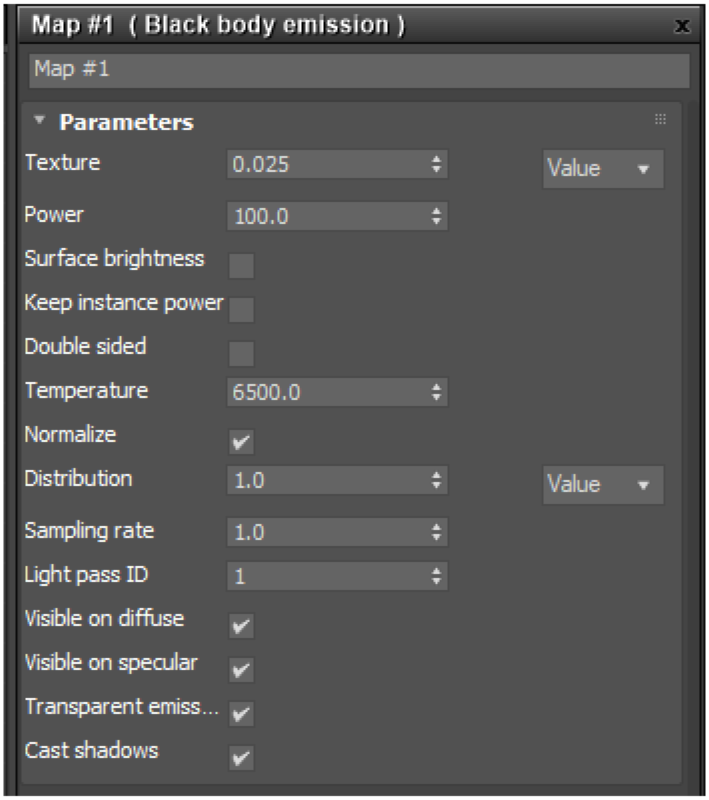


Figure 2: The Black Body emission node parameters

Black Body Parameters

Texture - Sets the light source's efficiency to either a value or texture. Keep in mind that real-world lights aren't 100% efficient at delivering power at their specified wattage - a 100-watt light bulb doesn't deliver 100 watts of light. This parameter enters the real-world values.

Power - The light source's wattage. You should set each light to their real-world wattage - for example, set a desk lamp to **25** watts, a ceiling lamp to **100** watts, and an LED light to **0.25** watts.

Surface Brightness - Causes emitters to keep a constant Surface Brightness, independent of the emitter's surface area.

Keep Instance Power - Enabling this option with Surface Brightness disabled and Uniform Scaling applied to the object causes Power to remain constant.

Double Sided - Allows emitters to emit light from the front and back sides.

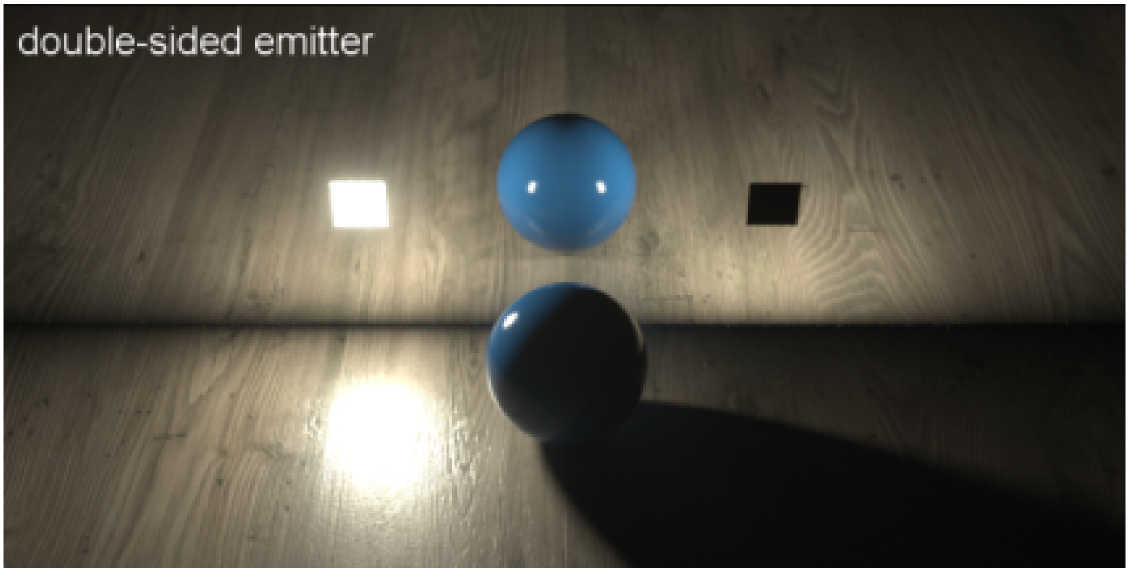


Figure 3: Comparison of the Double Sided parameter when enabled and disabled

Temperature - The temperature (in Kelvin) of the Black Body emission's emitted light.

Normalize - Ensures all the normal vectors have the same length for the Black Body emission to keep the luminance of the Black Body's emitted light constant if the temperature varies.

Distribution - Controls the light pattern. This can be set to an **Image** (**Grayscale** or **RGB**) so you can load an Image texture or **IES**[1] file. The Image texture's **Projection** map then adjusts the light's orientation or direction.

Sampling Rate - Choose what light sources receive more samples. Adjusting the light source Sampling Rates in the scene leads to a better balance between light sources. You can set the Sampling Rate to **0**, which means the direct light calculation excludes the emitter.

Light Pass ID - The ID for the light pass that captures the emitter's contribution. It works with the Light Pass ID render element.

Visible On Diffuse - Makes the light source visible on diffuse surfaces. This is enabled by default. You can enable or disable the Black Body or Texture emission's light sources from casting illumination or shadows on **Diffuse** objects. Disabling this option disables emission - it isn't visible in diffuse reflections, but it is visible in specular reflections. It is also excluded from the direct light calculation.

---

[1]An IES light is the lighting information representing the real-world lighting values for specific light fixtures. For more information, visit http://www.ies.org/lighting/.

Figure 4: The Visible On Diffuse option when enabled and disabled

Visible On Specular[1] - Makes the light source visible on specular surfaces. This is enabled by default. You can hide emitters on specular reflections or refractions only.

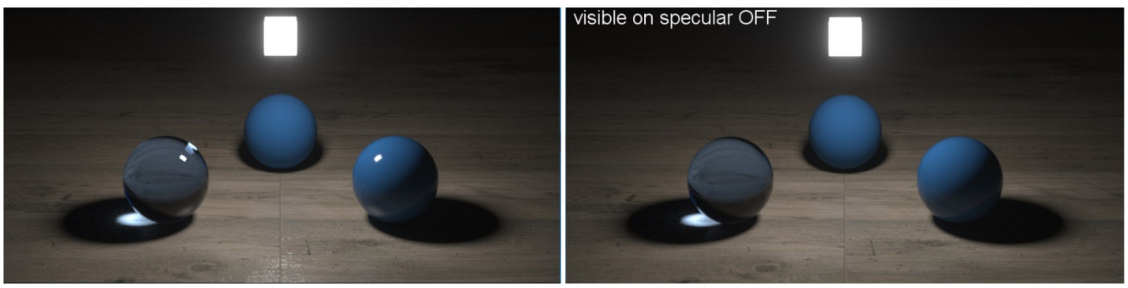


Figure 5: The Visible On Specular option when enabled and disabled

Transparent Emission - Lets the light source cast illumination on Diffuse objects, even if the light source is on a **Transparent** material.

Cast Shadows - Enables the light source to cast light and shadows on diffuse surfaces, and it disables direct light shadows for **Mesh** emitters. This option has an effect if the emitter is included in the direct light calculation, or when the Sampling Rate is greater than **0**. This option is enabled by default.

> **Note**: Most of the Emission parameters are common between Texture and Black Body emissions. The main difference is if the color comes from the Black Body temperature, or from the Diffuse material's texture settings.

[1] Amount of specular reflection, or the mirror-like reflection of light photons at the same angle. Used for transparent materials such as glass and water.

IES Texture

The **IES**[1] texture uses an IES profile with geometry as an illumination source. To create an IES light source using geometry:

1. Apply a **Diffuse**[2] material to the geometry.

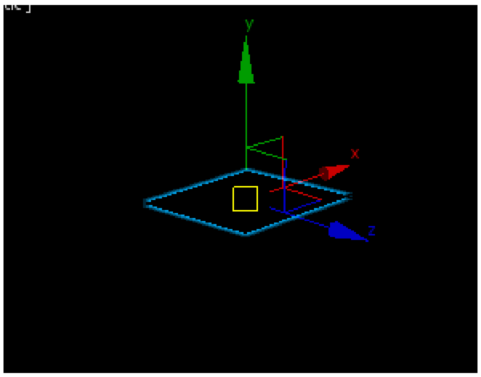


Figure 1: A Diffuse material[3] applied to a plane

2. Attach a **Texture** emission map to the Diffuse material's **Emission** parameter.

---

[1] An IES light is the lighting information representing the real-world lighting values for specific light fixtures. For more information, visit http://www.ies.org/lighting/.

[2] Amount of diffusion, or the reflection of light photons at different angles from an uneven or granular surface. Used for dull, non-reflecting materials or mesh emitters.

[3] Used for dull, non-reflecting materials or mesh emitters.

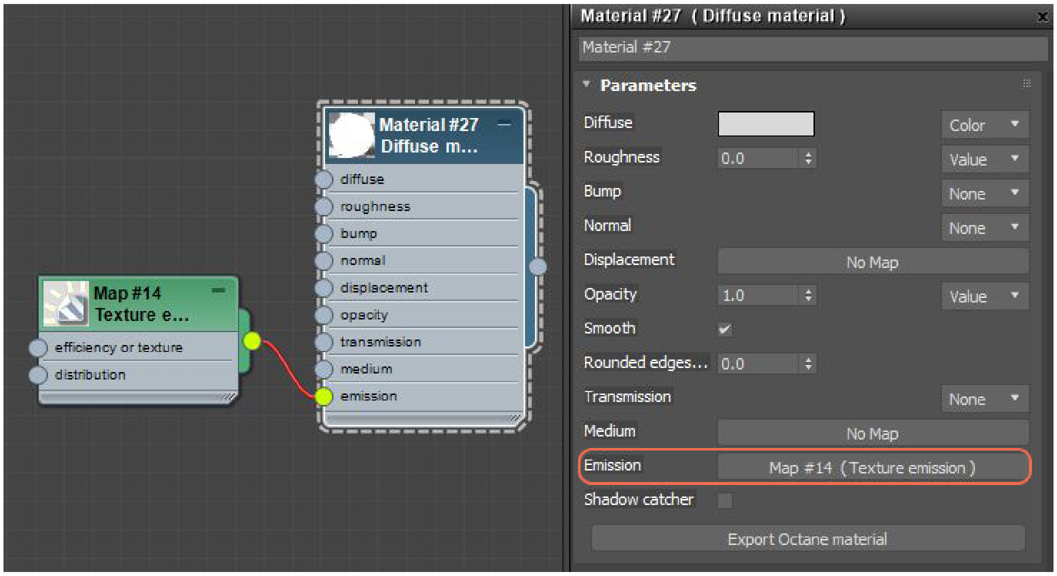


Figure 2: Connecting the Texture emission node

3. Attach an IES texture to the Texture emission's **Distribution** parameter and import an appropriate IES file in the IES texture node.

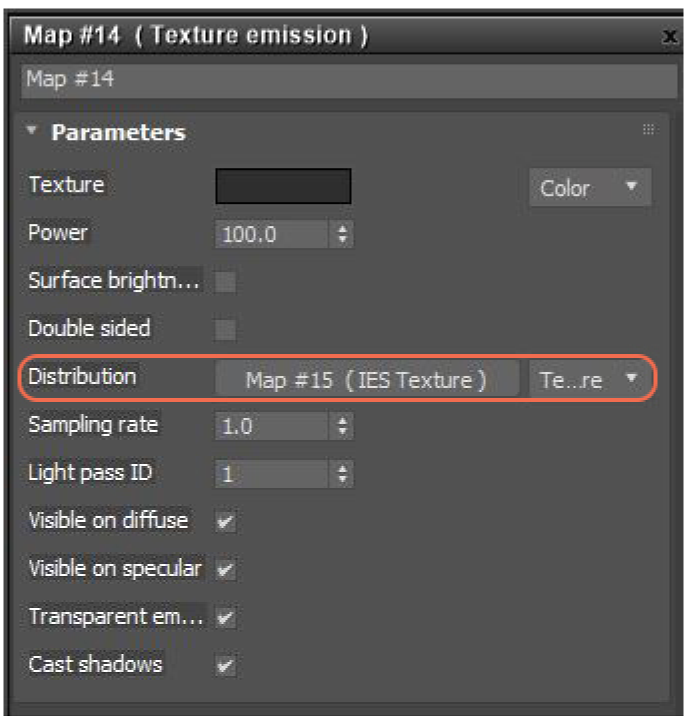


Figure 3: Adding an IES texture map to an Emission's Distribution slot

4. Select **Normal Space** as the coordinate space for the IES light projection.

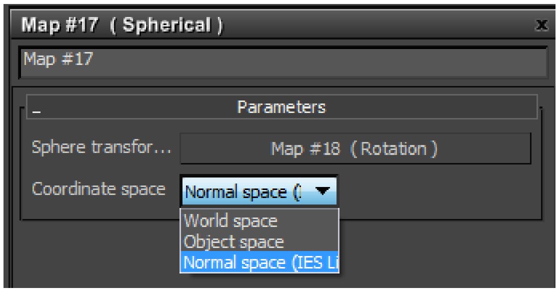


Figure 4: Selecting Normal Space in the Spherical projection node

Pictured below are the typical **Slate Editor** nodes and connections for a IES lighting setup.

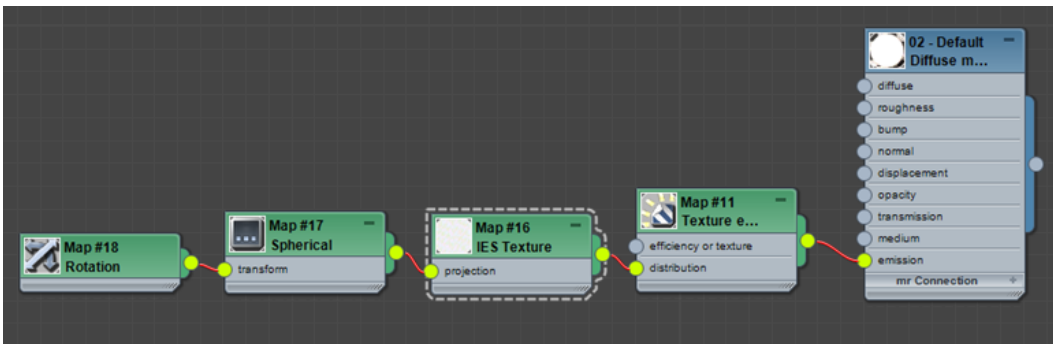


Figure 5: IES lighting node connections

Texture Emission

The **Texture** emission makes a **Mesh** behave as a light source. Use it with a **Diffuse**[1] material type connected to the **Emission** parameter. This allows any valid Texture type to set the light intensity. You can use this to create effects such as TV screens by using an **Image** texture as the source.

---

[1] Amount of diffusion, or the reflection of light photons at different angles from an uneven or granular surface. Used for dull, non-reflecting materials or mesh emitters.

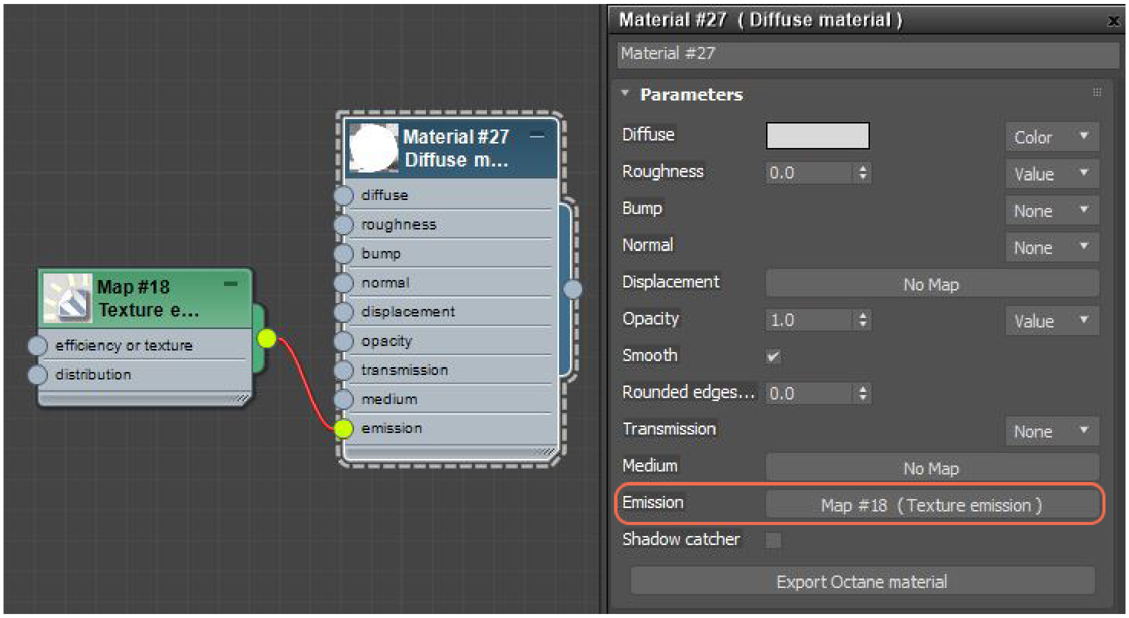


Figure 1: The Texture emission node connected to a Diffuse material[1]'s Emission slot

---

[1]Used for dull, non-reflecting materials or mesh emitters.

Texture Emission Parameters

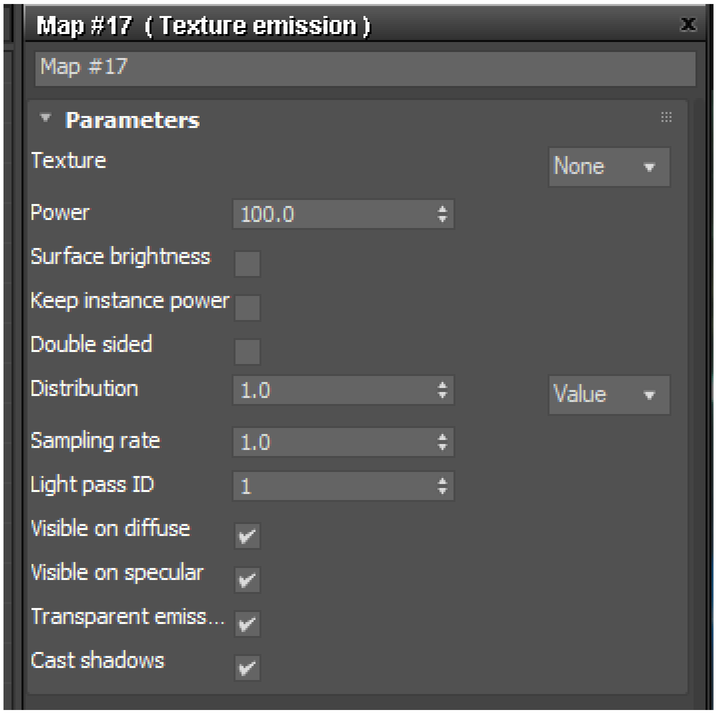


Figure 2: Texture emission parameters

Texture - Sets the light source's efficiency. No light is 100% efficient at delivering the power at the specified wattage. The efficiency setting accepts a value, **Color**, or Texture map.

Power - This is the wattage of the light source. Each light in the scene should be set to its real-world wattage. This power is multiplied by the Texture input, where **1.0** means 100% of the power. By default, **0.025** means 2.5%, which gives 2.5 watts of light).

Surface Brightness - Causes emitters to keep a constant Surface Brightness, independent of the emitter surface area.

Keep Instance Power - Enabling this option with Surface Brightness disabled and Uniform Scaling applied to the object causes Power to remain constant.

Double Sided - Makes the emitter emit light from the front and back sides.

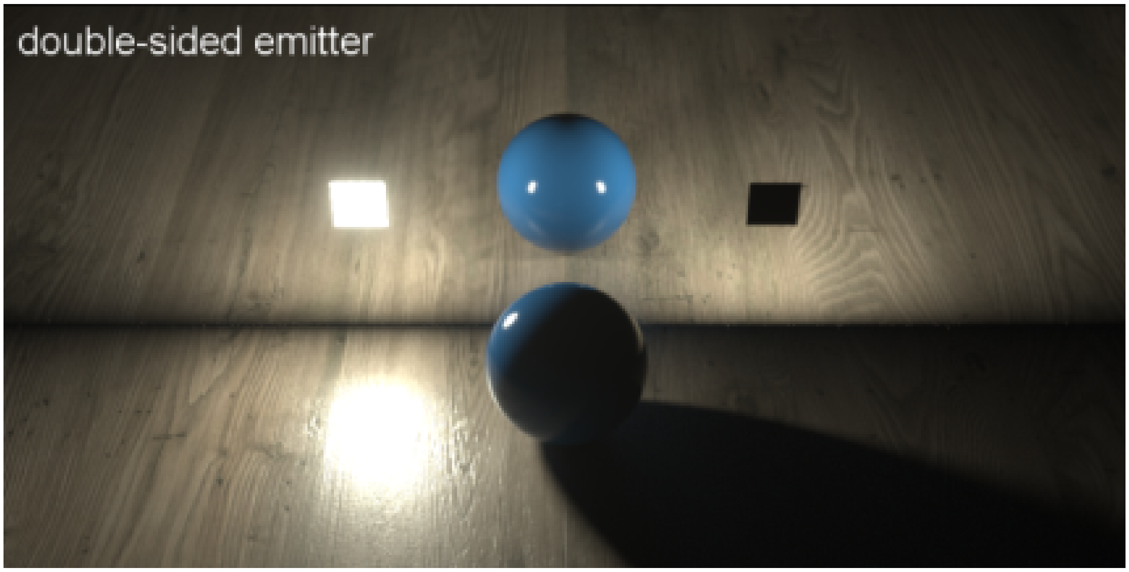


Figure 3: The Double Sided parameter when enabled and disabled

Distribution - Controls the light pattern. You can set this to an Image (**Grayscale** or **RGB**) so it can load an Image texture or **IES**[1] file. The Image texture's **Projection** map adjusts the light's orientation and direction.

Sampling Rate - Allows you to choose what light sources receive more samples. Adjusting the light source sampling rates in the scene leads to a better balance between light sources. You can set the **Sampling Rate** to **0**, which means the direct light calculation excludes the emitter.

Light Pass ID - The ID of the light pass that captures the emitter's contribution. It works with the Light Pass ID render element.

Visible On Diffuse - Makes the light source visible on diffuse surfaces. This is enabled by default. This allows you to enable or disable the **Black Body**[2] emission or Texture emission light sources from casting illumination or shadows on **Diffuse** objects. Disabling this option disables Emission, meaning it's visible in diffuse reflections, but is still be visible in specular reflections. It's also excluded from the direct light calculation.

[1] An IES light is the lighting information representing the real-world lighting values for specific light fixtures. For more information, visit http://www.ies.org/lighting/.
[2] An opaque object that emits thermal radiation. In Octane, this is used to designate illumination properties for mesh emitters.

Figure 4: The Visible On Diffuse parameter when enabled and disabled

Visible On Specular[1] - Makes the light source visible on specular surfaces. This is enabled by default. You can hide emitters on just specular reflections and refractions.

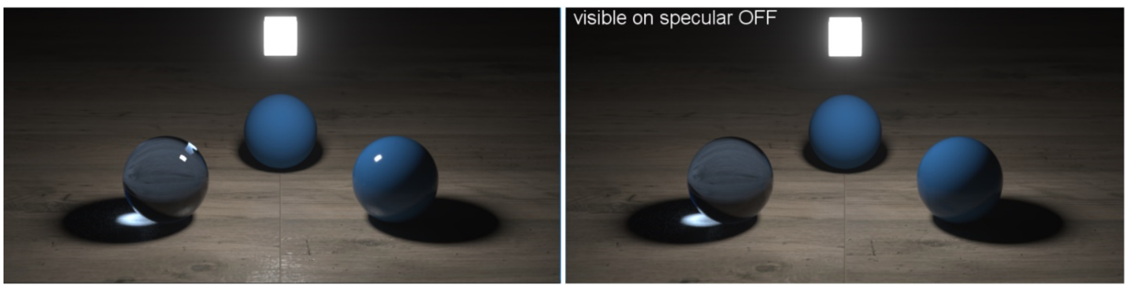


Figure 5: The Visible on Specular parameter when enabled and disabled

Transparent Emission - Lets the light source cast illumination on Diffuse objects even if the light source is on a **Transparent** material.

Cast Shadows - Lets the light source cast light and shadows on diffuse surfaces, and disables direct light shadows for Mesh emitters. This option has an effect if the emitter is included in the direct light calculation, or when the **Sampling Rate** is greater than **0**. This option is enabled by default.

Texture Environment

[1] Amount of specular reflection, or the mirror-like reflection of light photons at the same angle. Used for transparent materials such as glass and water.

The **Texture** environment affects the environment's illumination and color. This map adds an **HDRI**[1] environment texture to the scene for illumination. In the example below, the **Texture Environment** map is added to the **Environment Map** parameter in the 3DS Max® **Environment** panel.

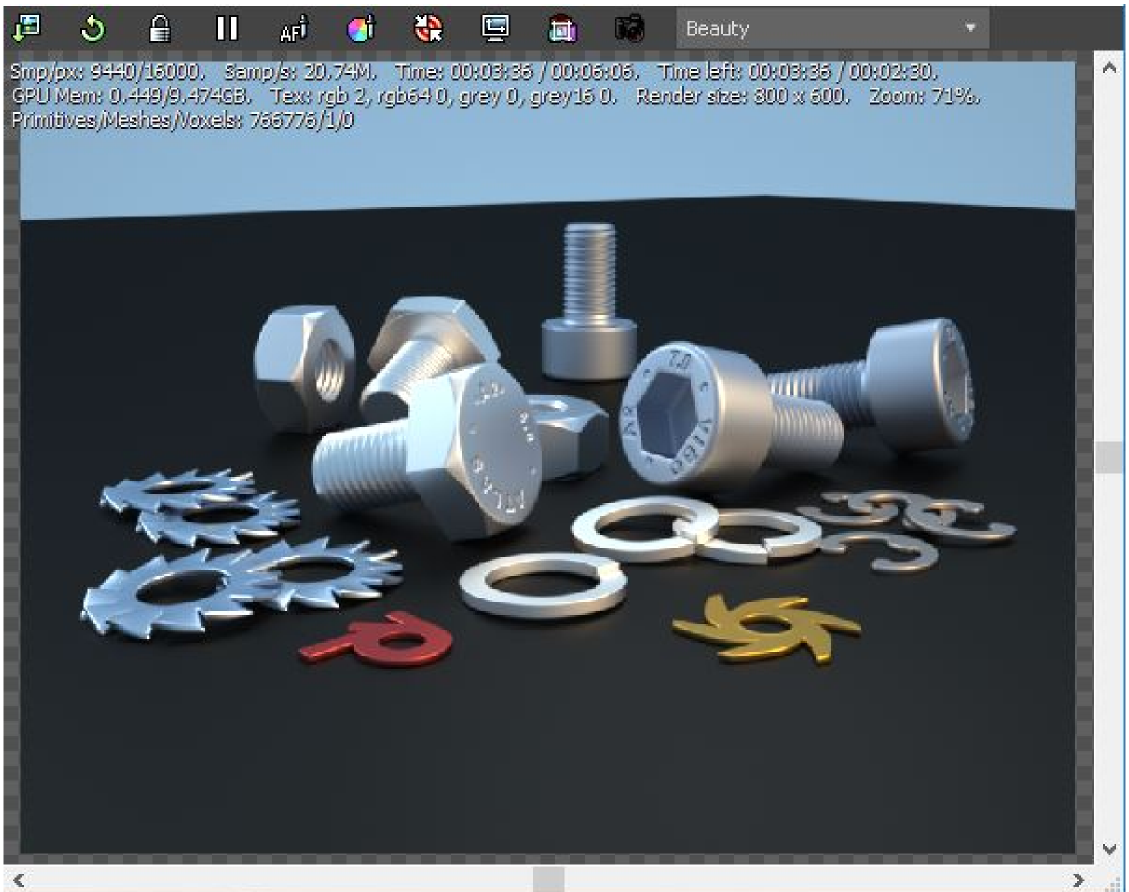


Figure 1: A simple color used for the Texture environment

The **Texture** parameter is set to a simple color, but you can also use a Texture map.

[1] An image which presents more than 8 bit per color channel unlike most common image formats.

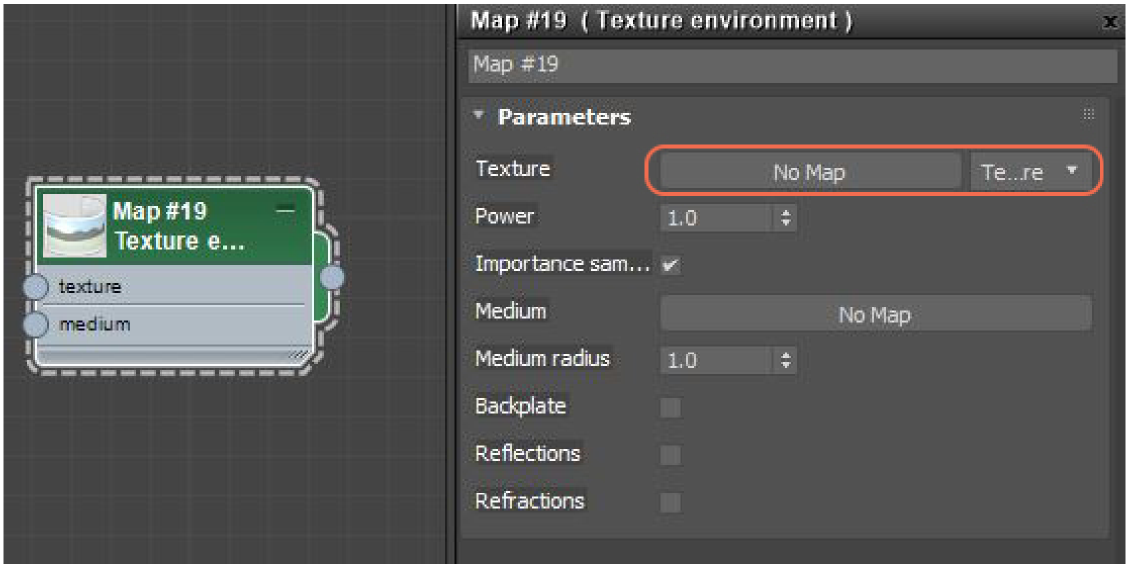


Figure 2: An HDRI texture map illuminates a scene by connecting it to a Texture environment's Texture slot

Texture Displacement

Texture **Displacement**[1] mapping utilizes a **2D** texture map to generate 3D surface relief. As opposed to **Bump** and **Normal** mapping, displacement mapping creates depth, and it displaces the actual geometric position of points over the textured surface. It typically goes into the **displacement** input of a material.

[1]The process of utilizing a 2D texture map to generate 3D surface relief. As opposed to bump and normal mapping, Displacement mapping does not only provide the illusion of depth but it effectively displaces the actual geometric position of points over the textured surface.

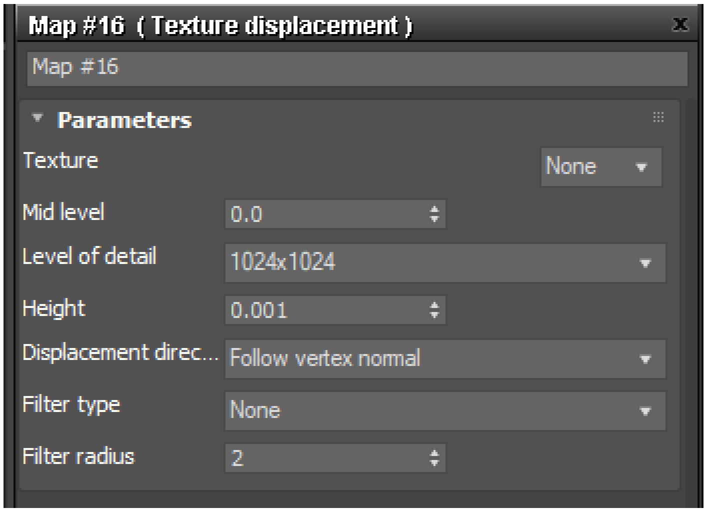


Figure 1: Texture Displacement Parameters

Displacement mapping requires **Objects** to have UV coordinates, either with a typical **Projection** type, or unwrapped. Displacement in OctaneRender® is not compatible with **Bump** or **Normal** mapping. You can't use Displacement together with a Normal or Bump map on the same **Material**[1]. Rendering a Material with Normal and Bump maps along with a Material that has the Displacement generates artifacts.

Displacement Parameters

Texture - This parameter accepts Displacement maps generated in other 3D programs.

Mid-Level - Defines the displacement shift in the Texture value range. For example, if a Zbrush® export's **0-Displacement** is **0.5**, set the Mid-Level to **0.5**.

Level Of Detail - Controls the Displacement map resolution.

[1]The representation of the surface or volume properties of an object.

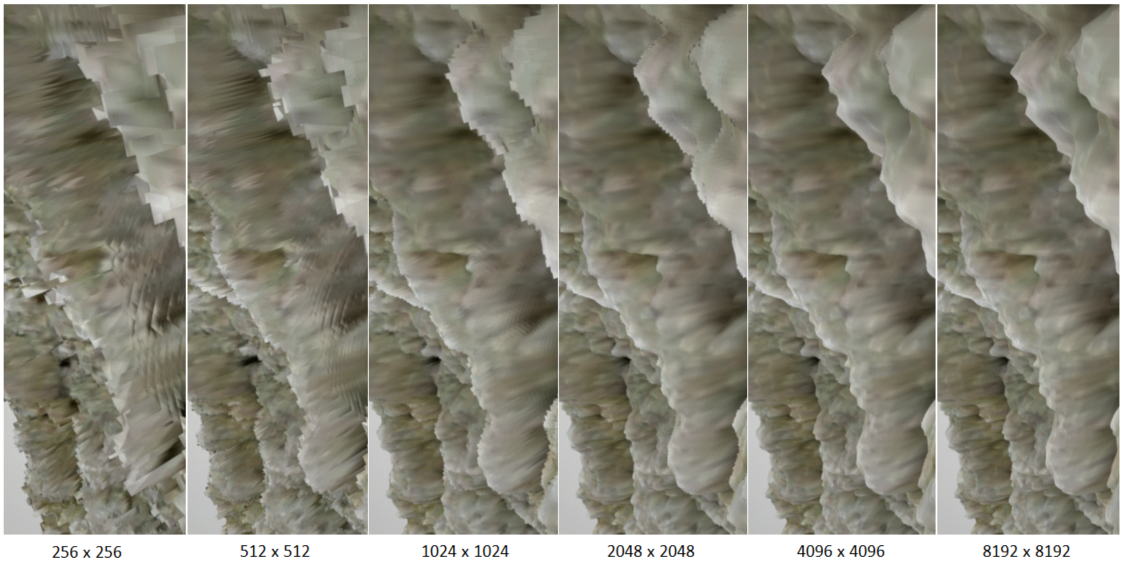


Figure 2: Level Of Detail comparisons

Height - This sets the Displacement's strength and amount.

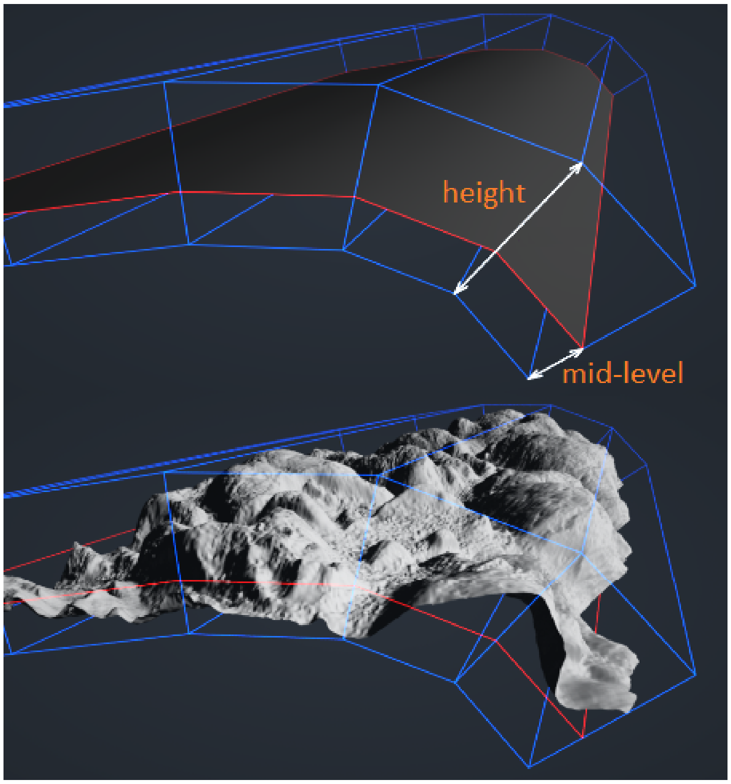


Figure 3: Height level illustration

Displacement Direction - Choose the Displacement vectors to use.

Filter Type - Determines the filter type to use on the displaced Texture map.

Filter Radius - Specifies the number of nearest pixels to use for the filtering. Higher values produce smoother displacement. This parameter is only valid if you select a **Filter Type**.

Image Tiles

The **Image Tiles** texture sets up a UV tile grid similar to UDIM image tile formats. These tile formats are generated in modeling and texturing applications.

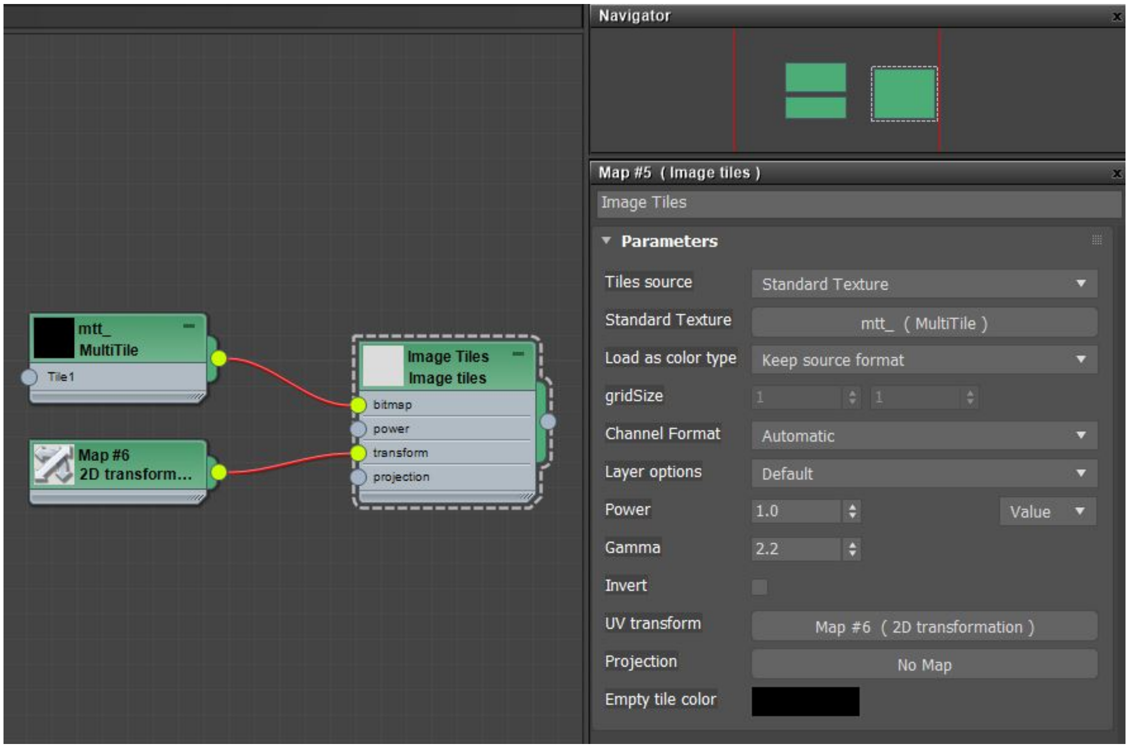


Figure 1: Creating an Image Tile node creates a Node network consisting of a MultiTile and 2D Transform node

Image Tile Parameters

Tiles Source - Reads the tile **Images** from the **MultiTile** or grid. This parameter synchronizes with the **Global Settings**.

Standard Texture - Selects the standard Texture to use as the tile source and to display in the **Viewport**.

Load As Color Type - Controls the color format to load the **Textures**[1].

Grid Size - Determines the Image tile grid's size. This parameter is active if **Tile Source** is set to **Tile Grid**.

Channel Format - Indicates the preferred channel format for loading images. Floating point options are ignored for 8-bit images. This parameter is disabled if **Tile Source** is set to **Tile Grid**.

Layer Options - Selects specific layers if the Image has multiple layers.

Power - Controls the Image's brightness. Lower values make the Image darker on the surface.

---

[1]Textures are used to add details to a surface. Textures can be procedural or imported raster files.

Gamma[1] - Controls the input Image's luminance, and can tune or color-correct the image if needed.

Invert - Inverts the Image's color values.

UV Transform - Accepts any **Transformation** node to control the surface texture's position, rotation, and scale.

Projection - This input accepts an OctaneRender® **Projection** node. If nothing is connected to this input, the Image texture uses the surface's **UV** texture coordinates by default.

Empty Tile Color - This color's use depends on the primary usage of the Texture map. For instance, set this parameter to white if the Texture map is used for **Opacity**.

Toon Ramp

The **Toon** ramp node controls the amount of detail in toon shading. It provides the mapping positions for a range of colors to the **Toon** material. The representation of the surface or **Volume** properties of an Object's **Diffuse**[2] or **Specular**[3] channels and the resulting color range is based on the hue set by that channel, respectively. You can add more positions to increase the number of colors in the range. The Toon ramp is applied to a Toon material's **Toon Diffuse** ramp or **Toon Specular** ramp.

---

[1] The function or attribute used to code or decode luminance for common displays. The computer graphics industry has set a standard gamma setting of 2.2 making it the most common default for 3D modelling and rendering applications.

[2] Amount of diffusion, or the reflection of light photons at different angles from an uneven or granular surface. Used for dull, non-reflecting materials or mesh emitters.

[3] Amount of specular reflection, or the mirror-like reflection of light photons at the same angle. Used for transparent materials such as glass and water.

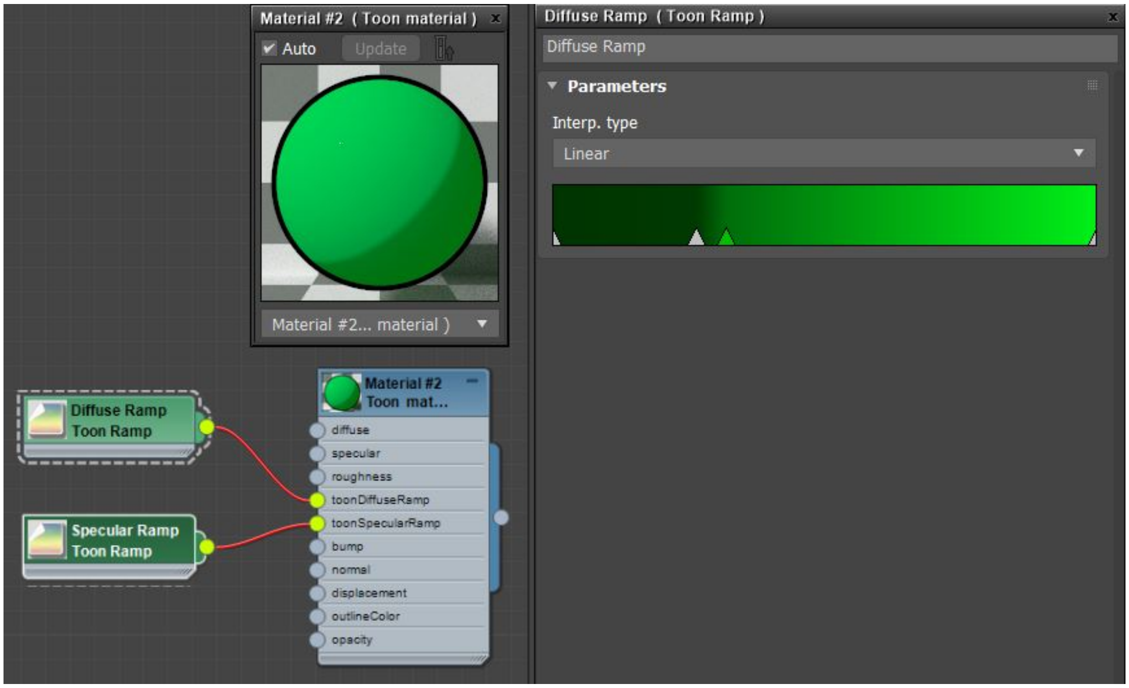
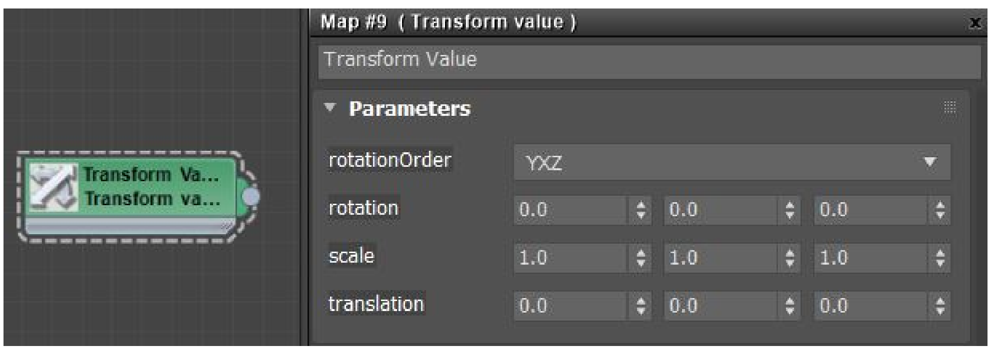


Figure 1: The Toon ramp applied to the Toon Diffuse and Toon Specular Ramp inputs

Transform Value

Transform Value sets the other **Textures**[1]' orientation. It connects to the **Transform** input pin on any given **Texture** map and provides true 3D positioning for Texture maps, unlike the **2D Transformation** node, which provides planar positioning for Texture maps.



[1]Textures are used to add details to a surface. Textures can be procedural or imported raster files.

Figure 1: Transform Value parameters

Mediums - Subsurface Scattering

OctaneRender® supports participating media inside objects. These settings are stored in **Medium** nodes, which are attached to the corresponding input pin of **Diffuse**[1] or **Specular**[2] material nodes.

There are three types of Medium nodes: **Absorption**, **Scattering**, and **Volume**.

- **Scattering**[3] - Has parameters for absorption, scattering light passing through the Medium, and emission inside the Medium.
- **Absorption**[4] - Is a simple version with **Absorption** parameters.
- **Volumes** - Volume mediums are used on volumetric surfaces such as smoke and clouds, and require a **VDB**[5] file to create the Volume objects. Volumes are described in more detail in the Effects Overview topic in this manual.

To render with Medium nodes, the **Path Tracing** or **PMC** render kernels are the best choice. It is possible to render **Mediums**[6] using the **Direct Light** kernel, but only if the Medium node is connected to a **Diffuse material**[7], and if **Diffuse Mode** is set to **GI**.

---

[1] Amount of diffusion, or the reflection of light photons at different angles from an uneven or granular surface. Used for dull, non-reflecting materials or mesh emitters.

[2] Amount of specular reflection, or the mirror-like reflection of light photons at the same angle. Used for transparent materials such as glass and water.

[3] Defines how fast light gets scattered when traveling through the medium.

[4] Defines how fast light is absorbed while passing through a medium.

[5] Dreamworks' open-source C++ library housing the data structures and tools implementation for storing and manipulating volume data, like smoke and other amorphous materials. The purpose of OpenVDB is mostly to have an efficient way to store volumetric data in memory and on disk. It has evolved into a more general toolkit that also lets you accomplish other things, such as fracturing volumes, converting meshes to volumes and vice versa. However, it does not include a computational fluid dynamics solver, and therefore it cannot procedurally generate smoke or fire. OpenVDB is fully integrated as a library in OctaneRender. For more information about OpenVDB, please see http://www.openvdb.org/.

[6] The behavior of light inside a surface volume described by scatter, absorption, and transmission characteristics.

[7] Used for dull, non-reflecting materials or mesh emitters.

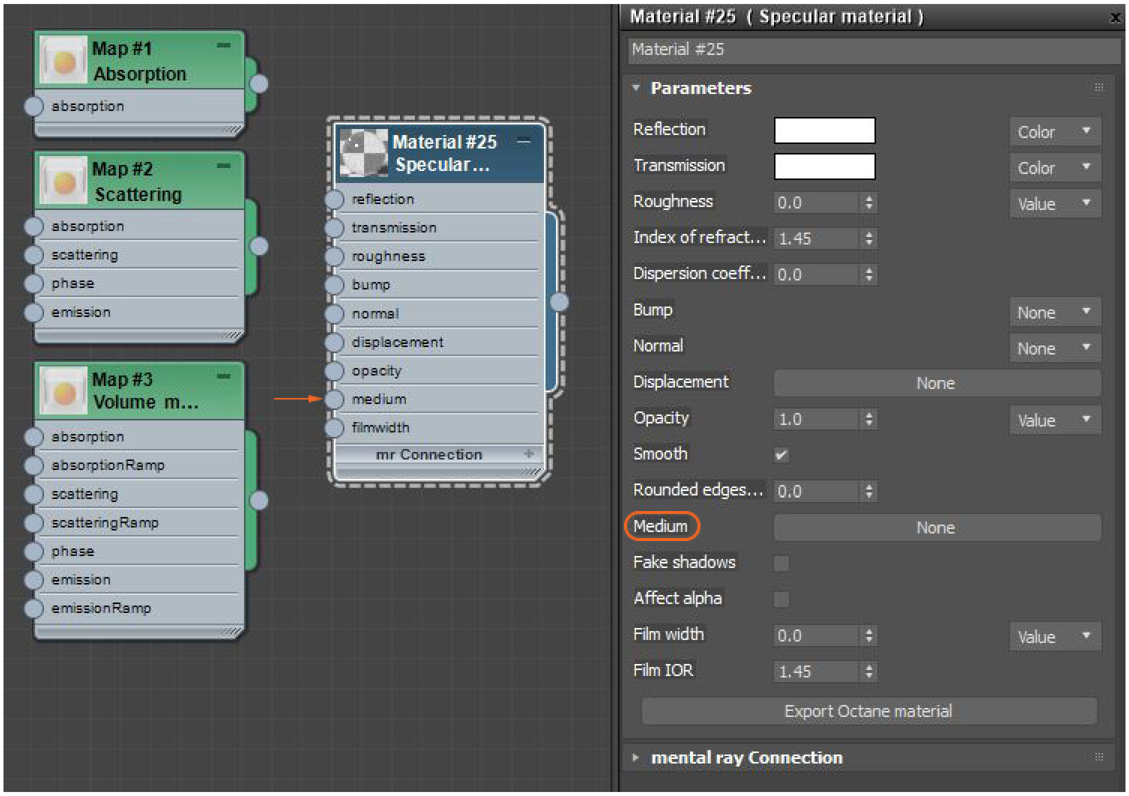


Figure 1: Medium nodes are connected using the Medium button in the Material[1]'s parameters

Add Medium nodes to **Materials**[2] applied to Meshes that define a closed volume. A single-sided plane will not work. For example, a plane representing a leaf will not work if you apply a Material with a Medium to it. The one exception is a plane representing the ground. OctaneRender® treats the ground plane as an endless, deep surface.

Specular materials are the best choice when using a Medium node. Set the **Transmission**[3] and **Reflection** parameters to a non-zero value, or a color other than black, or a Texture map.

---

[1]The representation of the surface or volume properties of an object.

[2]A set of attributes or parameters that describe surface characteristics.

[3]A surface characteristic that determines if light may pass through a surface volume.

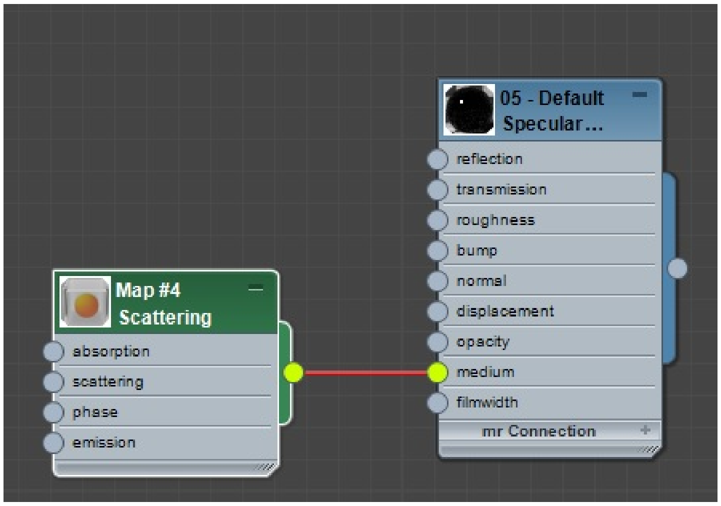


Figure 2 : A Scattering medium node is connected to a Specular material[1]

When connecting a Medium node to a Diffuse material, set the **Transmission** to a non-zero value, or a Color other than black, or a Texture map.

[1]Used for transparent materials such as glass and water.

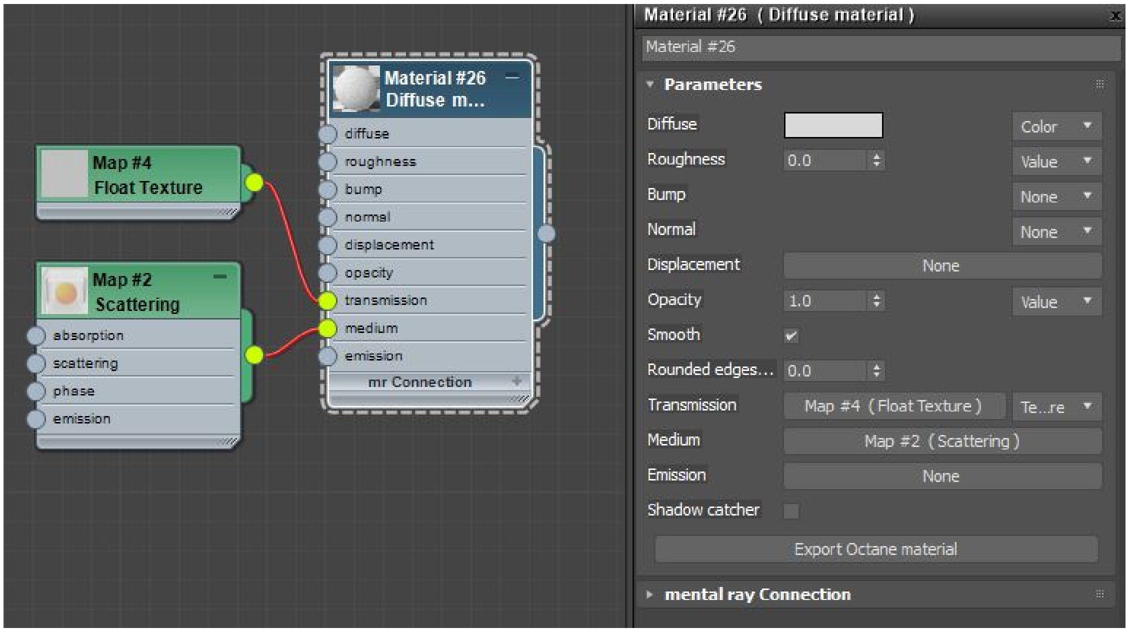


Figure 3 : A Scattering medium node is connected to an Octane Diffuse material, a Float texture is connected to the Transmission input

When using a Specular shader, set the **Reflection** parameter to a low value because the part of the spectrum that is not reflected can enter the object for scattering. If the reflection is set to **1.0**, all light gets reflected regardless of the Transmission value. If reflection is set to **0.0**, all light gets transmitted through the surface. However, the result is an unnatural appearance. Reflection values of **0.1** - **0.2** are good starting points.

If the **Reflection** parameter uses a color, the light transmitted through the surface is shaded as the complementary color (e.g., if the reflection is set to yellow, the transmitted light is bluish).

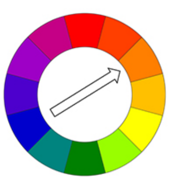


Figure 4 : A diagram shows that complementary colors are opposite from each other on the color wheel

Absorption

Absorption[1] is controlled with the **Absorption** texture, which defines how fast the Medium absorbs light passing through it. A setting of **0.0** means no absorption. The higher the value, the faster the Medium absorbs the light.

You can add the Absorption map to a **Material**[2] by clicking on the **None** button next to the **Medium** parameter.

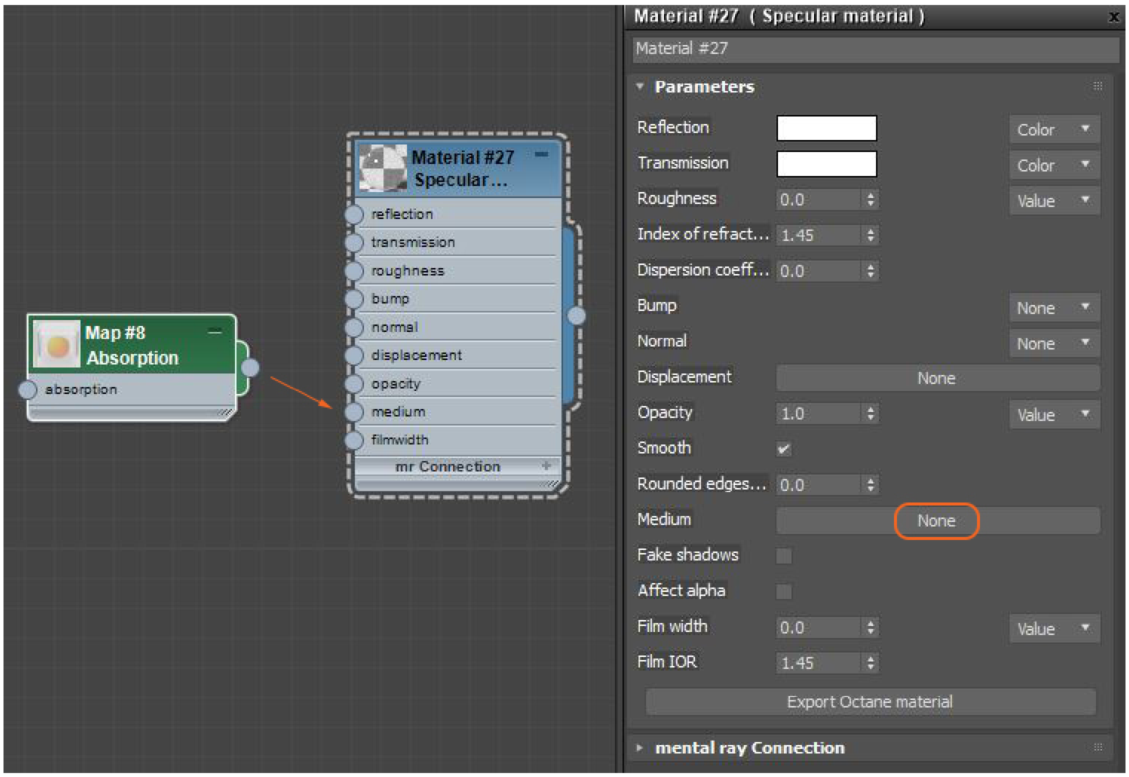


Figure 1: Add the Absorption texture using the Medium button

The Absorption texture can be a **Color**, value, or **Texture** map, and is multiplied by the **Density** parameter. This allows you to set a wide range of values in an easier way.

[1] Defines how fast light is absorbed while passing through a medium.

[2] The representation of the surface or volume properties of an object.

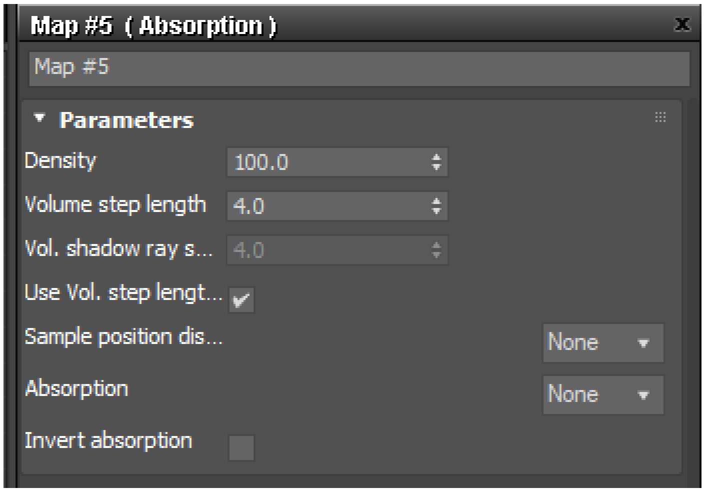


Figure 2: Absorption parameters

The color resulting from the Absorption is dependent on the distance light travels through the Material. With increased distance, it gets darker, and if the Absorption is colored, it becomes more saturated. It works in a subtractive manner in that the scattered color is the compliment of the color designated in the parameter.

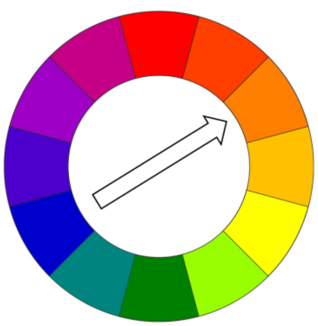


Figure 3: A diagram shows that complementary colors are opposite each other on the color wheel

Rendering an Absorption medium requires the **Path Tracing** or the **PMC** kernel with a large **Maxdepth** setting. For media inside **Diffuse**[1] transmitting **Materials**[2], you can use the **Direct Light** kernel with **Diffuse GI Mode** too.

Meshes - You should add participating media to Meshes that define a closed volume rather than planes. Using planes to model leaves of a plant with SSS will not work. Using a single plane as a ground plane will work since it is treated as an endless, deep Material. The Mesh can have opaque Objects nested inside, but nested participating media are not supported.

Absorption Paramaters

Absorption - Controlled by Absorption color, which defines how fast a Medium absorbs light passing through it. A 0.0 or black value means no absorption. Higher values result in faster light absorption. The specified color in the Absorption parameter produces its complimentary color in the rendering (Figure 1). The Absorption texture is multiplied by the Density parameter. This allows setting a wide range of values.

Volume Step Length - Depending on the surface, you may need to adjust this parameter. The default value is 4, but if the volume is smaller than this, you need to decrease the value. Decreasing this value decreases render speed, and increasing the value causes the ray marching algorithm to take longer steps. If the Volume Step Length exceeds the volume's dimensions, then the ray marching algorithm takes a single step through the whole volume. To get the most accurate results, keep Volume Step Length as small as possible.

Invert Absorption - Inverts the Absorption color so that the Absorption channel becomes a **Transparency** channel. This helps visualize the effect of the specified color, since a neutral background shining through the Medium appears in that approximate color.

Scattering

The **Scattering**[3] map is a **Medium** with single-scattering SSS as well as **Absorption**[4]. The **Scattering** parameter defines how fast light scatters when traveling through the Medium, similar to how Absorption is defined. A very high value means light scatters very fast, and a value of **0.0** means no scattering when the Scattering parameter is set to a value. The Scattering parameter also accepts a **Texture** map or a color.

---

[1] Amount of diffusion, or the reflection of light photons at different angles from an uneven or granular surface. Used for dull, non-reflecting materials or mesh emitters.

[2] A set of attributes or parameters that describe surface characteristics.

[3] Defines how fast light gets scattered when traveling through the medium.

[4] Defines how fast light is absorbed while passing through a medium.

You should add participating media to Meshes that define a closed Volume rather than planes. Using planes to model plant leaves with SSS will not work. Using a single plane as ground plane should be okay (it's treated as an endless, deep **Material**[1]). The Mesh can have opaque Objects nested inside, but nested participating media are not supported.

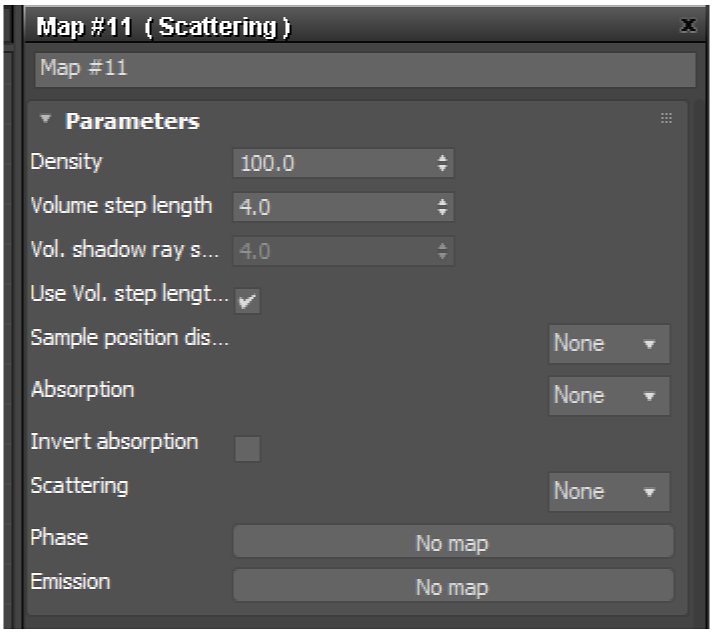


Figure 1: Scattering parameters

Scattering Parameters

Density - This parameter multiplies against **Scattering**.

Volume Step Length - Depending on the surface, you may need to adjust this parameter. The default value is **4**, but if the volume is smaller than this, you need to decrease the value. Decreasing this value decreases render speed, and increasing the value causes the ray marching algorithm to take longer steps. If Volume Step Length exceeds the volume's dimensions, then the ray marching algorithm takes a single step through the whole volume. To get the most accurate results, keep Volume Step Length as small as possible.

Absorption - Controlled by Absorption color, which defines how fast a medium absorbs light passing through it. A **0.0** or black value means no absorption. Higher values result in faster light absorption. The specified color

[1] The representation of the surface or volume properties of an object.

in the Absorption parameter produces its complimentary color in the rendering (Figure 1). The Absorption texture is multiplied by the **Density** parameter. This allows setting a wide range of values.

Scattering - Determines how fast light scatters as it moves through a surface. High values mean that light scatters sooner as it enters a surface, and low values mean that light passes deeper into the surface before scattering. A **0** value disables Scattering.

Phase - Controls light direction as it scatters through the surface. A value of **0** results in light scattering equally in all directions; a positive value results in forward scattering, where photons continue the same approximate direction as when they enter the surface; and a negative value results in backwards scattering, where light moves through the surface in the same direction, but opposite to the angle that they entered the surface. This is known as backscattering.

Emission - Attaches an **Emission** node to the **Emission** input pin. When you connect an Emission node to a Medium node, it defines emission inside the volume instead of on the object's surface. In this case, **Power** controls how fast a ray's radiance increases while traveling through the volume; it doesn't represent total power. It's not multiplied with the **Scale** parameter. This effect works best with large, not-too-bright objects - small, bright objects create lots of noise.

Volume Medium

The **Volume** medium applies independent **Absorption**[1], **Scattering**[2], and **Emission** color ramps. To use a ramp, you must specify a **Color** for the corresponding parameter. For example, to use the Absorption ramp, you must select a Color for **Absorption**.

---

[1]Defines how fast light is absorbed while passing through a medium.

[2]Defines how fast light gets scattered when traveling through the medium.

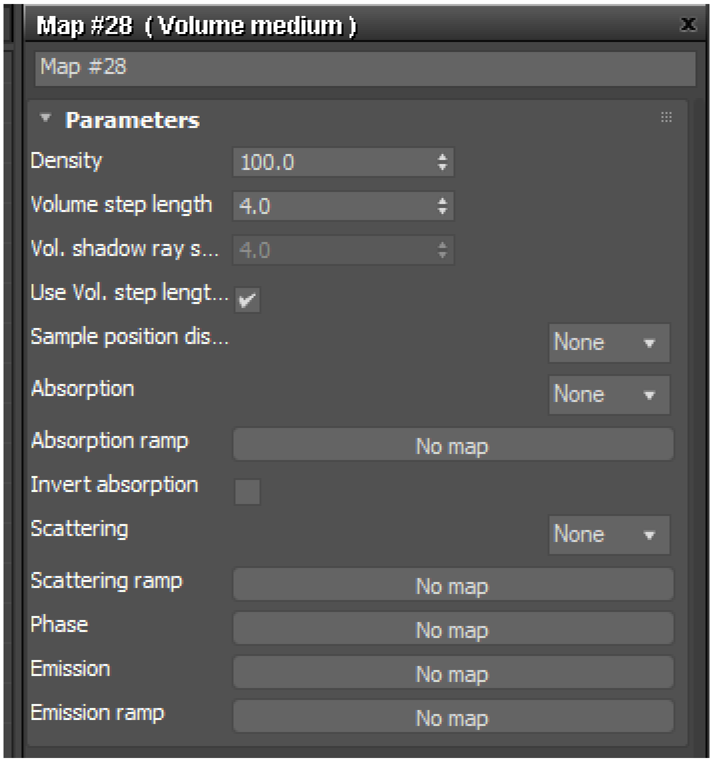


Figure 1: Volume medium parameters

Volume Medium Parameters

Density - Refers to the Absorption and Scattering scales.

Volume Step Length - This parameter determines the average distance between points where OctaneRender® samples the Absorption, Scattering, and Emission textures. The default value is **4.0**. If your Volume is smaller than this, decrease the Step Length. Decreasing this value reduces the render speed. Increasing this value causes the ray marching algorithm to take longer steps. You get the most accurate results when the Step Length is as small as possible. For simplifying your workflow, set the Volume Step Length to an acceptable value.

Vol. Shadow Step Length - Similar to Volume Step Length but defined for Shadows.

Use Vol. Step Length - Will Set the **Vol. Shadow Step Length** to the value of **Volume Step Length.**

Sample Position **Displacement**[1] - Allows a texture to control a volume's sample position displacement.

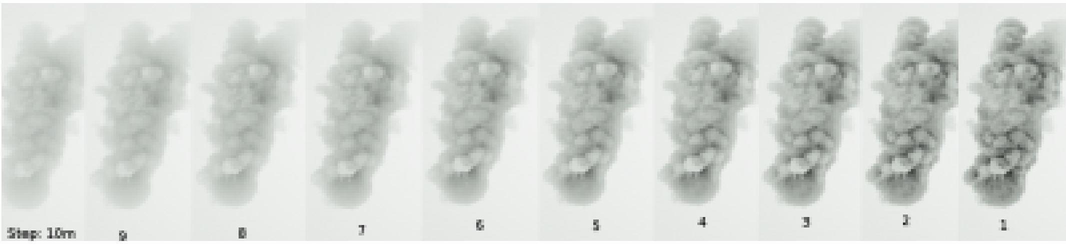


Figure 2: A comparison of Volumes rendered with different Volume Step Length settings

The Volume can't be too dense. We recommend reducing the Volume Step Length to an acceptable performance and accuracy level, and then reducing the **Volume Density**. Otherwise, you risk rendering a solid Object at a high Step Length and get deceiving results.

Volumes can also have their own Scattering, Emission, and Absorption ramps. These colors make a strong influence on the Volume's appearance. The **Phase** function also affects a Volume like a Medium node, and modifying the Volume's **Scale** value scales the Volume's Density values in a linear fashion. This can also increase emission as Absorption values are also used as particle density.

Figure 3: A comparison of Volumes rendered with different Phase settings

[1]The process of utilizing a 2D texture map to generate 3D surface relief. As opposed to bump and normal mapping, Displacement mapping does not only provide the illusion of depth but it effectively displaces the actual geometric position of points over the textured surface.

Volumes are rendered in an unbiased way and can scatter multiple times. They also cause self-shadowing effects. To reduce the maximum scatter events in a Volume, reduce the **Diffuse**[1] **Depth** in your **Kernel** node.

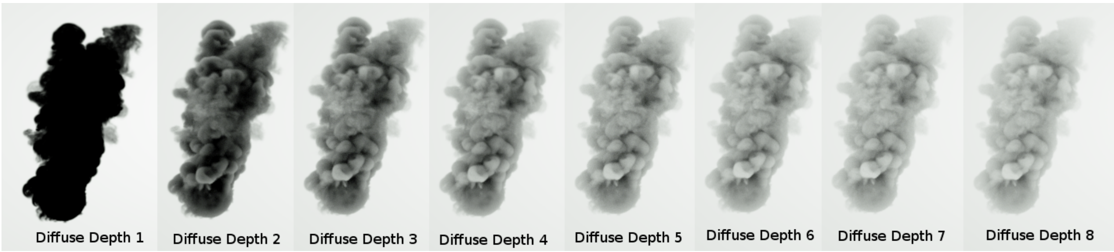


Figure 4: A comparison of Volumes rendered with different Diffuse Depth settings

Absorption - Specifies the color texture for absorption.

Absorption Ramp - Defines the color range. Absorption Ramp takes the grid value as input. In the color gradient, the colors near **0** on the left side map low grid values to a custom color. Higher grid values map to colors on the right side of the color gradient. Desaturated colors create less-pronounced absorption. **Emission** and **Scattering** ramps operate in the same way.

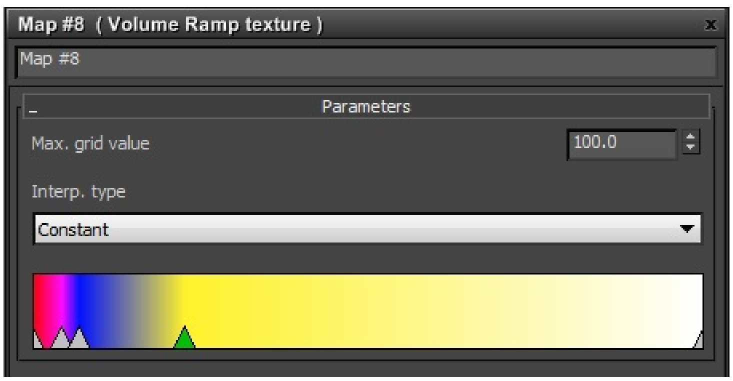


Figure 5: Absorption ramp parameters

[1]Amount of diffusion, or the reflection of light photons at different angles from an uneven or granular surface. Used for dull, non-reflecting materials or mesh emitters.

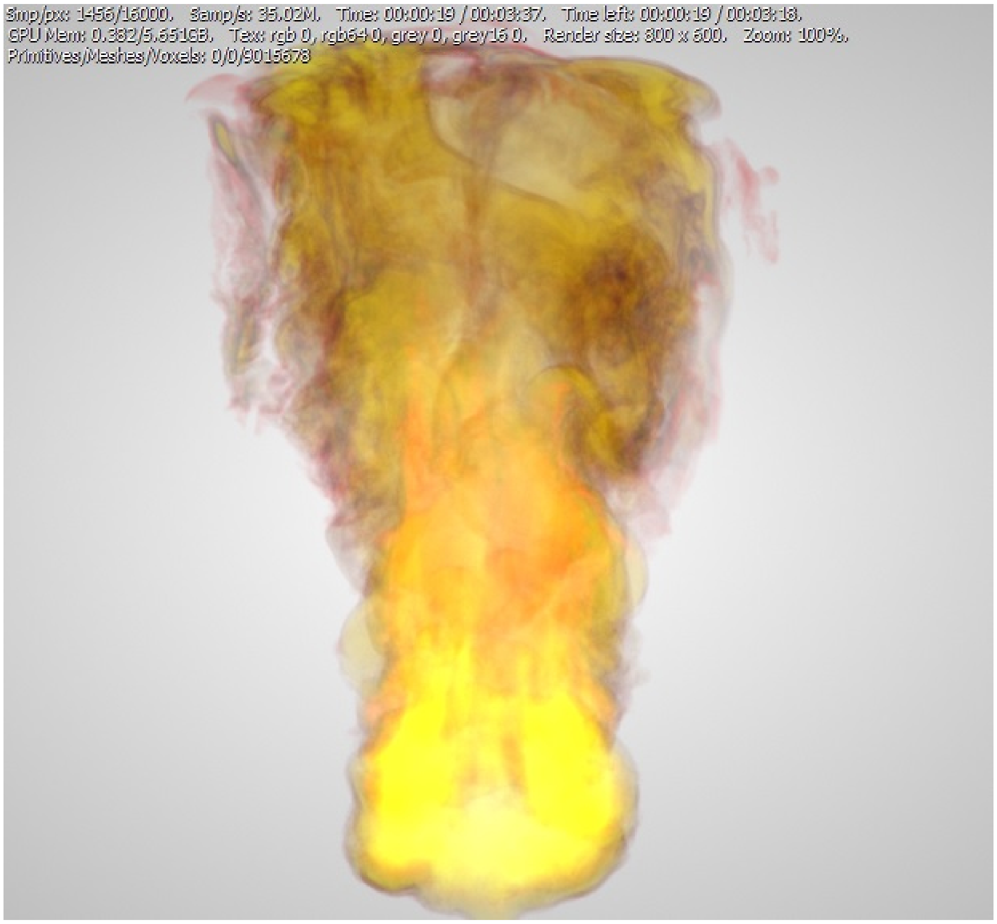


Figure 6: A render of a Volume using an Absorption ramp

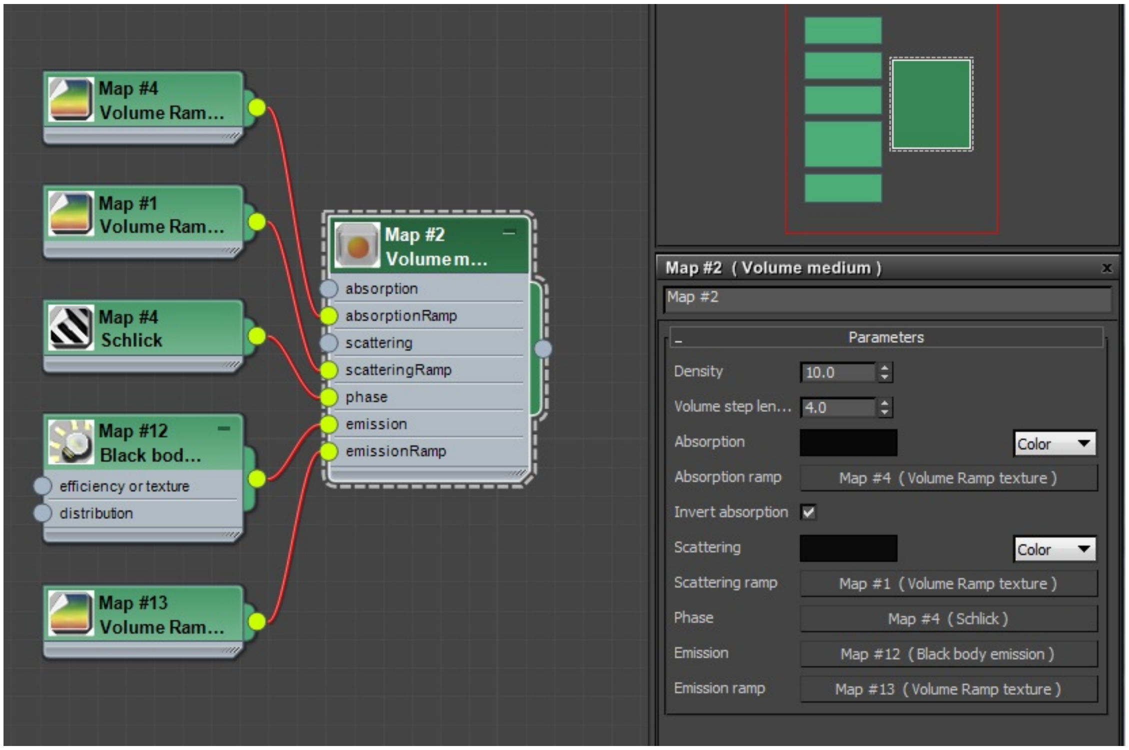


Figure 7: A graph of the Volume medium with incoming connections

Invert Absorption - Inverts the Absorption color so that the **Absorption** channel becomes a **Transparency** channel. This helps visualize the specified color's effect since a neutral background shining through the Medium appears close to that color.

Scattering - This is the scattering cross-section. This channel defines how much light is absorbed over the color range. It can accept a value, Color, or Texture map.

Scattering Ramp - Acts similar to the Absorption ramp, but instead it maps colors to the light as it scatters within the volume.

Phase - Affects a volume as it would affect a Medium node, and modifying the Volume's scale value scales the Volume's Density values in a linear fashion.

Emission - Controls the Volume's emission. For Emission, the Medium map can have either a **Blackbody** emission or a **Texture** emission.

Emission Ramp - This is the Emission color ramp.

Volume Ramp

The **Volume Ramp** map provides precise control for a **Volume** geometry's color, such as a flame. This map works with the **Volume** medium map discussed in the **Volume Medium**[1] topic in this manual.

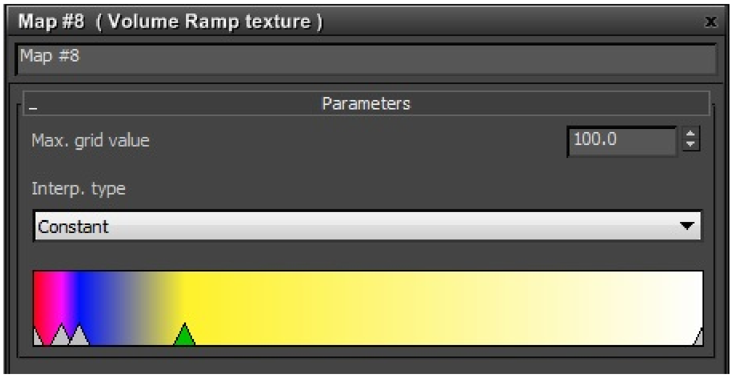


Figure 1: Volume Ramp parameters

Schlick

The **Schlick** map connects to a **Scattering**[2] map's **Phase** input. It has a single parameter, **Scattering Direction**. Negative values result in more backscattering, and positive values result in more forward scattering.

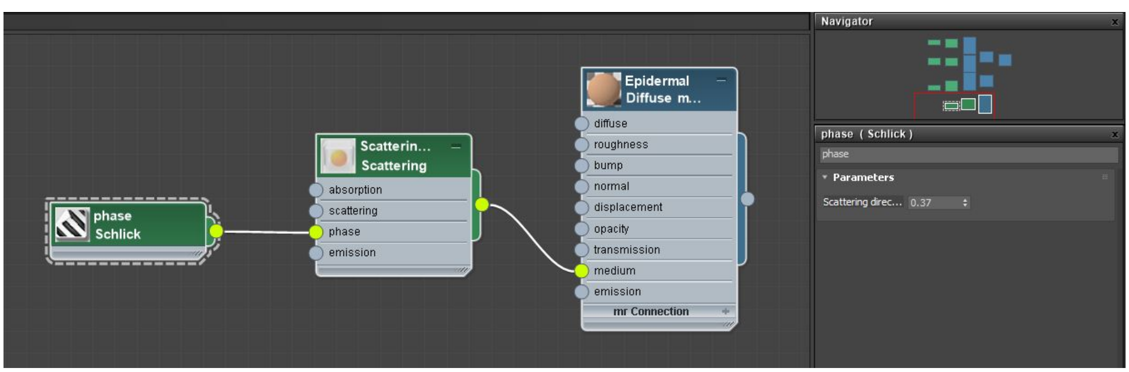


Figure 1: A Schlick map connected to a Scattering map

---

[1] A shading system designed to render volumes such as smoke and fog.

[2] Defines how fast light gets scattered when traveling through the medium.

Random Walk

The **Random Walk** node is a newer variant of subsurface scattering that utilizes a stochastic or random process for the scattering of light through an object. This provides the most realistic result when rendering scatter volumes. The **Random Walk** parameters are similar to the **Scattering**[1] node's parameters with a few exceptions (Figure 1).

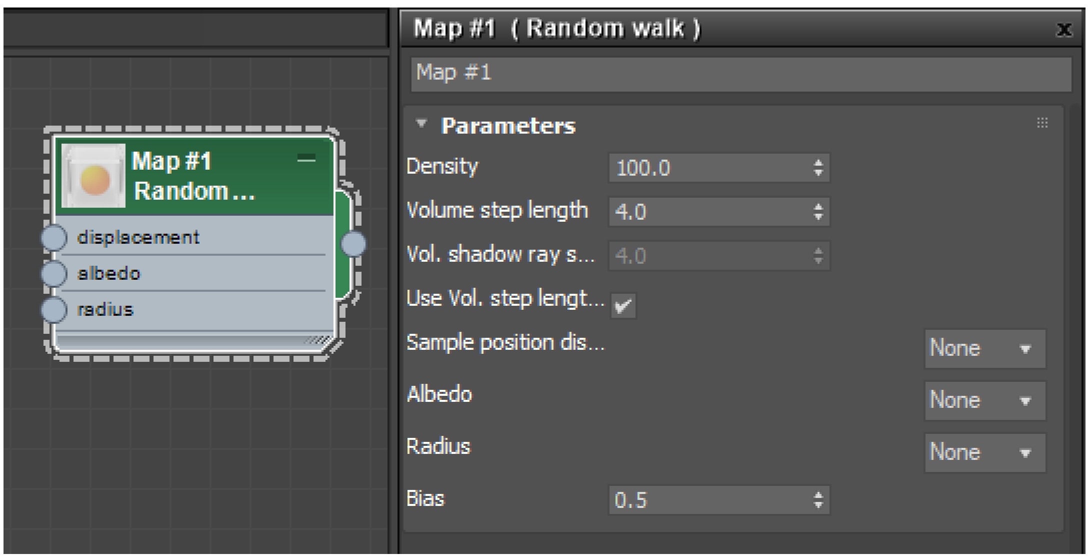


Figure 1: The Random Walk parameters.

Random Walk Parameters

Density - This parameter multiplies against **Scattering**.

Volume Step Length - Depending on the surface, you may need to adjust this parameter. The default value is **4**, but if the volume is smaller than this, you need to decrease the value. Decreasing this value decreases render speed, and increasing the value causes the ray marching algorithm to take longer steps. If Volume Step

[1]Defines how fast light gets scattered when traveling through the medium.

Length exceeds the volume's dimensions, then the ray marching algorithm takes a single step through the whole volume. To get the most accurate results, keep Volume Step Length as small as possible.

Volume Shadow Ray Step Length - The step length that is used by the shadow ray for marching through volumes.

Use Volume Step Length for Volume Shadow Ray Step Length - If active, uses Volume Step Length as Volume Shadow Ray Step as well.

Sample Position Displacement[1] - Allows a texture to control a volume's sample position displacement.

Albedo - The scattering albedo where the coefficients are determined for this parameter using the mean free path.

Radius - Determines the depth that the light scatters in the medium.

Bias - The bias of the subsurface scattering. Higher values use biased sampling, which usually yields better results with lower depth settings.

---

[1] The process of utilizing a 2D texture map to generate 3D surface relief. As opposed to bump and normal mapping, Displacement mapping does not only provide the illusion of depth but it effectively displaces the actual geometric position of points over the textured surface.

Lights And Cameras

This section will cover adding Octane lights and cameras into your scene.

Lighting

Cameras

Lighting

There are six **Light** types in the OctaneRender® for 3DS Max® plugin, which you can find in the **Create** Panel's **Lighting** category (Figure 1). You can also create Lights by using **Mesh** objects in the scene, and then adding indirect/environment lighting from the 3DS Max® **Environment And Effects** window (Figure 2).

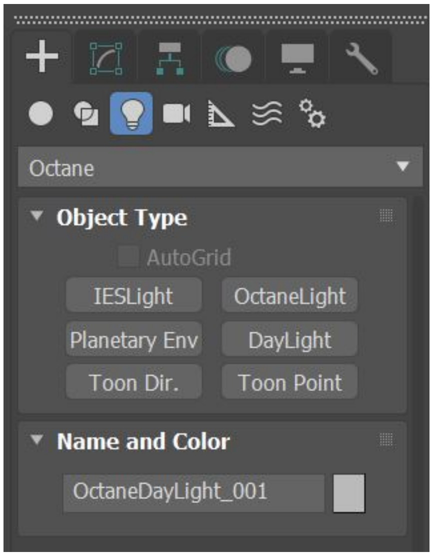


Figure 1: Accessing Lights from the Command Panel

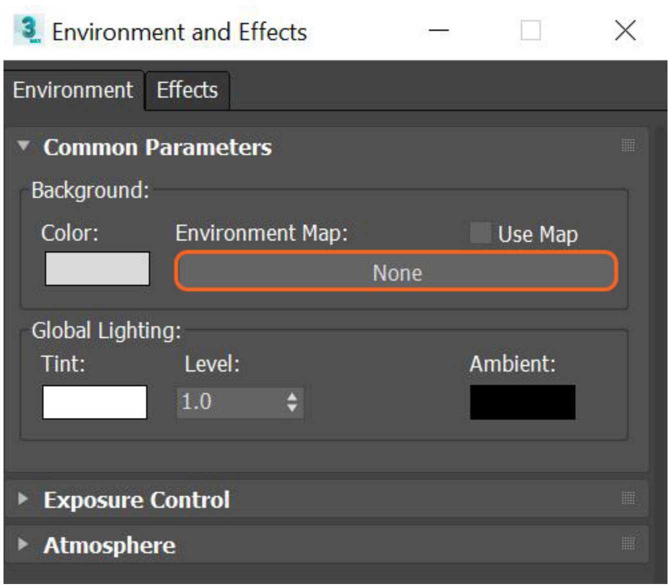


Figure 2: Accessing Environment lighting from the Environment And Effects window

Daylight

The OctaneRender® **Daylight** system simulates a complete Environment with an infinite **Direct** light (like the sun) and a **Sky**. To add the Daylight system to the scene, click the **Daylight** button in the **Octane Lights** category, then click-and-drag in an **Orthographic Viewport**. The light's origin is where you first click the mouse.

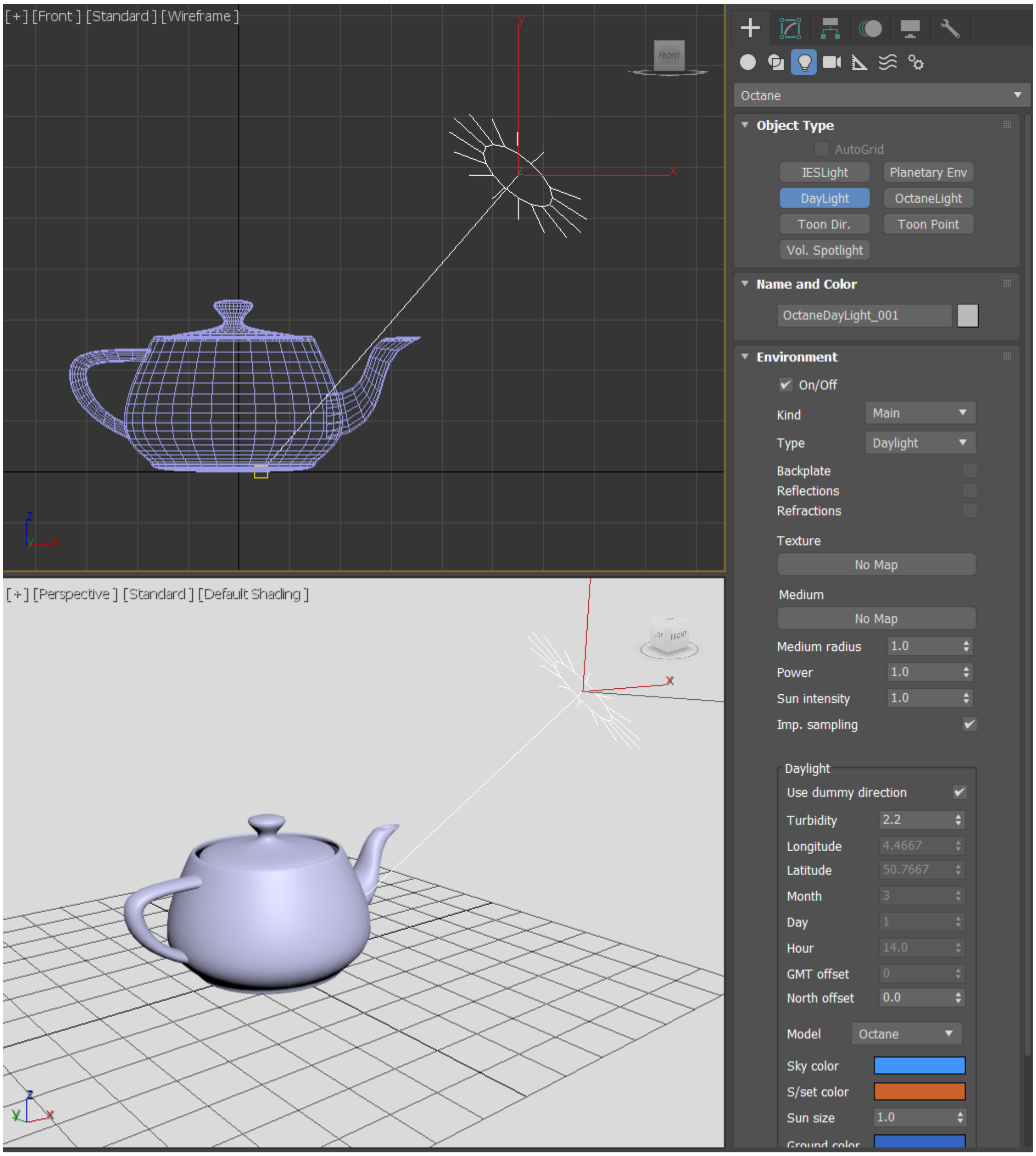


Figure 1: Daylight system

Octane Daylight Parameters

On/Off - Toggles the Environment for this light type on and off.

Kind - Determines if the light source contributes to just the Environment (**Visible**), or if it provides a directional light source in addition to an Environment (**Main**).

Type - There are two options for this parameter:

- **Daylight** - Used with the **Kind** parameter's **Main** option. It provides an Environment that consists of a Sky and **Ground** texture.
- **Texture** - Used with the **Kind** parameter's **Visible** option. It provides a consistent lit Environment and works best with a Texture environment (**HDRI**[1]).

The following three options are available when the Kind parameter is set to **Visible**.

- **Backplate** - Makes the Visible environment behave as a background plate.
- **Reflections** - Makes the Visible environment visible in **Material**[2] reflections.
- **Refractions** - Makes the Visible environment visible in the Material refractions.

Texture - Specify Environment maps as either a **Lighting** environment or a **Background** environment. Apart from having an HDR image to light the Environment, you can also specify a different Environment for the background that does not contribute to the scene's illumination, but is visible in reflections.

Medium - Creates an **Environment** medium. If specified, OctaneRender® applies the Medium to a virtual sphere around the camera.

Medium Radius - If you specify a Medium (either **Absorption** or **Scattering**), this controls the radius of the virtual sphere created around the camera when you apply the Medium.

Power - Adjusts the overall light's strength. This can affect the Image's overall contrast and exposure level.

Sun Intensity - Scale factor that adjust only the sun's strength.

Imp. Sampling - This toggles the **Sky** texture's Importance Sampling, similar to the Importance Sampling in the **Texture** environment. This enables quicker noise reduction for HRDI images. The ray tracing is more efficient because it doesn't waste time on rays that do not contribute much to the overall image quality. Due to the added complexity, this reduces the Ms/sec of rendering, but each Ms is more effective, resulting in higher image quality in less time. Importance Sampling is enabled by default.

Use Dummy Direction - Controls the sun's direction through the placement of the Light object in the scene. When activated, the **Longitude**, **Latitude**, **Month**, **Day**, and **Hour** options are disabled.

---

[1] An image which presents more than 8 bit per color channel unlike most common image formats.

[2] The representation of the surface or volume properties of an object.

Turbidity - Adjusts the sunlight shadows' sharpness. Low values create sharp shadows like on a sunny day, and high values diffuse the shadows like on a cloudy day.

Longitude/Latitude - Makes realistic sun settings for a specified geographic location.

Month/Day/GMT Offset/Local Time - Places the sun in the sky according to the date and time for the sun at the current longitude and latitude.

North Offset - Adjusts the scene's North direction. This is useful for architecture visualization to ensure the sun's direction is accurate to the scene.

Model - Specifies the Daylight model to use for the current Environment.

- **Octane** - This is the new default daylight model simulates full-spectrum daylight, providing more sky color variation as the sun moves along and bearing shorter rays as the sun moves closer to the normal plane.
- **Preetham** - This is the old daylight model that lights a scene with basic spectral radiance as the sun moves over the horizon at a relative distance from the object.
- **Nishita** - Implements atmospheric scattering based on the Nishita sky model, and it displays the color variations that are optical effects caused by the particles in the atmosphere.
- **Hosek Wilkie** - Produces more realistic and detailed results than other implementations specially in hazy conditions and near the horizon.

Sky Color/Sunset Color - The new Daylight model uses these settings to customize the light's spectral shade and affects the scene's overall mood.

Sun Size - Controls the sun's radius in the Daylight environment.

Ground Color - Determines the ground's base color. Ground Color works with the New and Nishita sky models.

Gnd Start - The angle below the horizon where the transition to the Ground Color starts. This only works with the New sky model.

Gnd Blend - The angle over which the sky color transitions to the ground color. This only works with the New sky model.

Enable Texture - Enables the Sky texture.

Octane Light

The **Octane Light** provides a light source **Object** to use for fast lighting setups without adding new **Geometry** and **Materials**[1]. This Object produces Geometry and Materials for rendering, and makes light setup much easier.

[1] A set of attributes or parameters that describe surface characteristics.

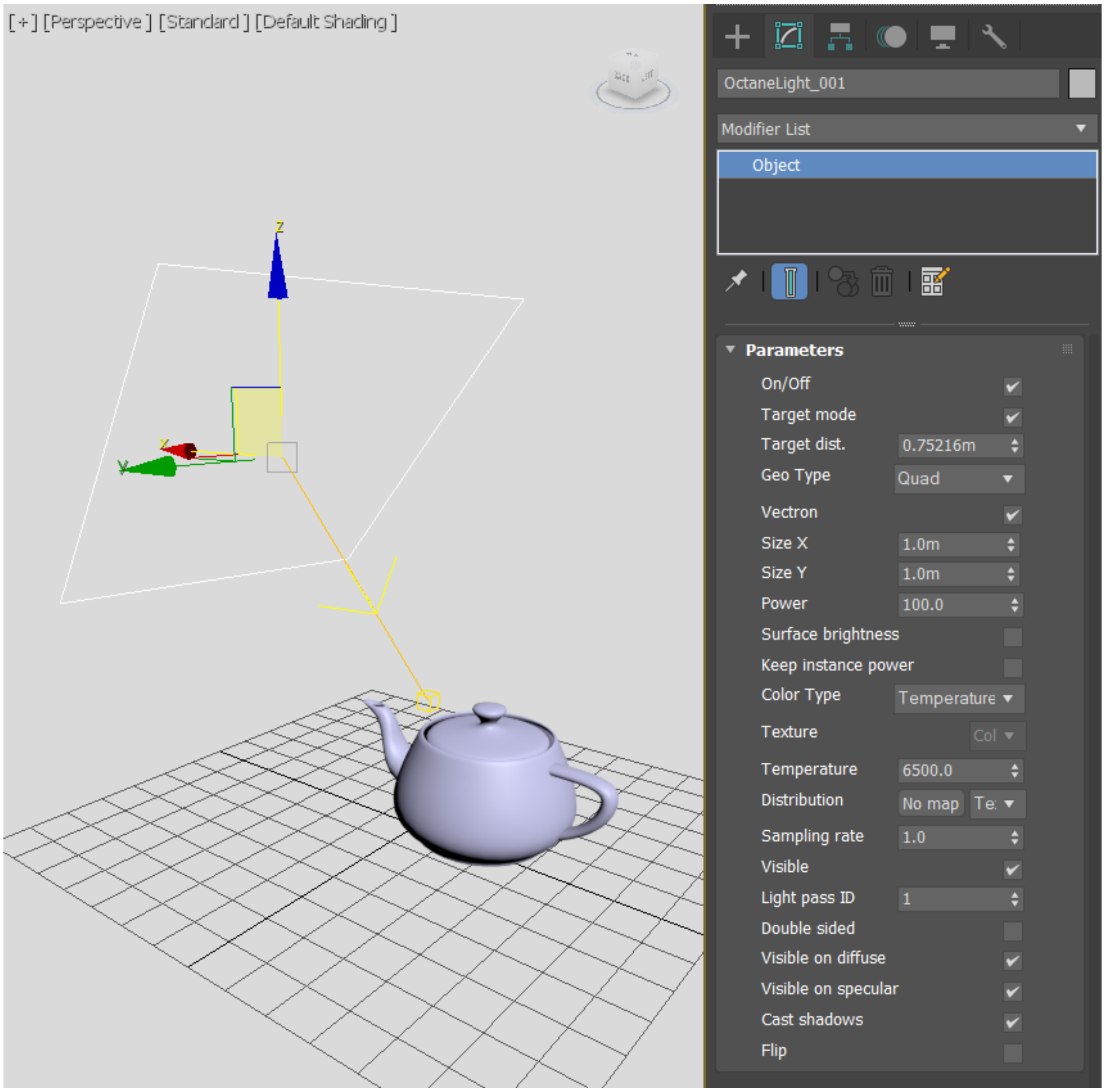


Figure 1: The Octane Light source

Octane Light Parameters

On/Off - Toggles this light type on and off.

Target mode - Adds a target item to help orient your light direction.

Target dist. - Set the target a specific distance from the light source.

Geo Type - Select from a **Plane**, **Disk**, or **Sphere** light shapes.

Vectron - Use procedural geometry for the light source.

Size X and Y - Determines the size of the Octane Light source. A larger area produces a softer result.

Power - The Light source's wattage. Set each light in the scene to its real-world wattage. For example, set a desk lamp's Power to **25**. Don't use this setting to balance the scene's lighting power. Power gives the total power that the light source is emitting. The default value is 100 watts.

Surface Brightness - The Light source's size contributes to its overall brightness.

Keep Instance Power - If this is enabled and Surface Brightness is disabled, the Power remains constant if uniform scaling is applied to the instance.

Color Type - The Light emission may be based on a color hue or a temperature, from warmer values around **500** to cooler values upwards of **12000**.

Texture - Define color values of the light source. This parameter is valid when the **Color Type** is set to **Color**.

Temperature - The Light emission's temperature (in Kelvin). This parameter is valid when the **Color Type** is set to **Temperature**.

Distribution - This option is used to define the Pattern of the light. It's also called "Gobo Lights" or "**IES**[1] Lights". These features will be create with the help of this option. You can also create various light patterns using an Alpha image.

Sampling rate - Choose what light sources receive more samples. The maximum sampling rate is **10000**. You can set the sampling rate to **0**, which means the direct light calculation excludes the emitter.

Visible - When this is disabled, the light source is invisible to the camera, reflections, and refractions, but it still illuminates the scene.

Light Pass ID - Light Pass ID numbers **1** - **8** capture the Octane Light emitter's contribution. Octane Light emitters (just like the **Texture** emitter and **Black Body**) have a **Light Pass ID** that assigns the Emitter to a Light Pass. You can assign multiple Emitters to the same Light Pass. If nothing is configured, all emitters contribute to Light Pass ID **1** by default.

Double Sided - The Emitter emits light from the front and back sides.

---

[1] An IES light is the lighting information representing the real-world lighting values for specific light fixtures. For more information, visit http://www.ies.org/lighting/.

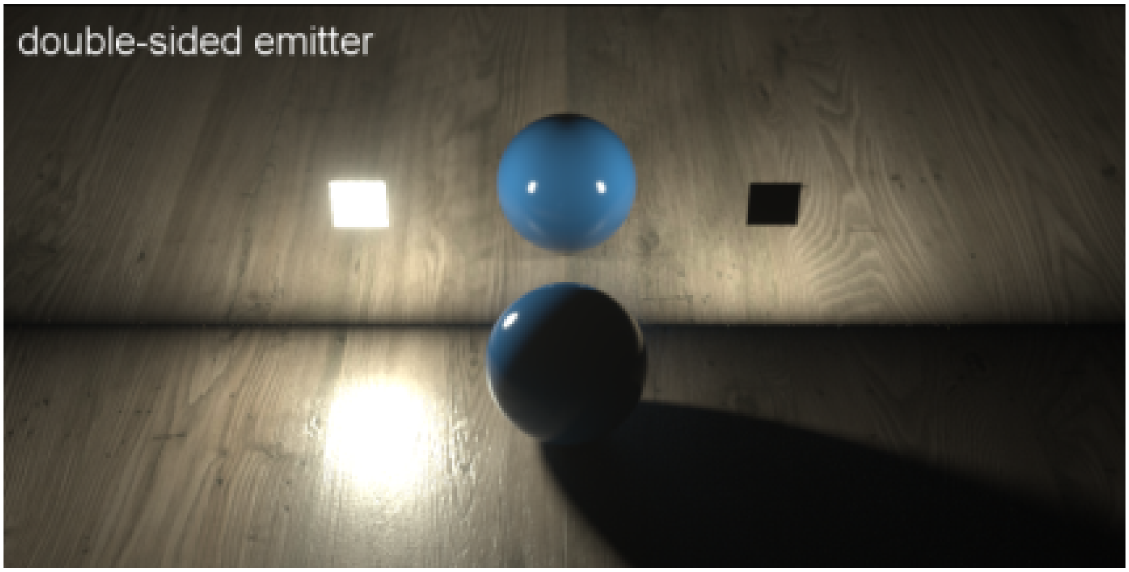


Figure 3: The Double Sided parameter when enabled and disabled

Visible On Diffuse[1] - Makes the light source visible on diffuse surfaces. This is enabled by default. You can enable or disable the **Black Body**[2] or Texture emission's light sources from casting illumination or shadows on Diffuse objects. Disabling this option disables emission - it's not visible in diffuse reflections, but it is visible in specular reflections. It's also excluded from the Direct Light calculation.

Figure 4: The Visible On Diffuse parameter when enabled and disabled

---

[1] Amount of diffusion, or the reflection of light photons at different angles from an uneven or granular surface. Used for dull, non-reflecting materials or mesh emitters.
[2] An opaque object that emits thermal radiation. In Octane, this is used to designate illumination properties for mesh emitters.

Visible On Specular[1] - Makes the light source visible on specular surfaces, and it lets you hide Emitters just on specular reflections and refractions. This is enabled by default.

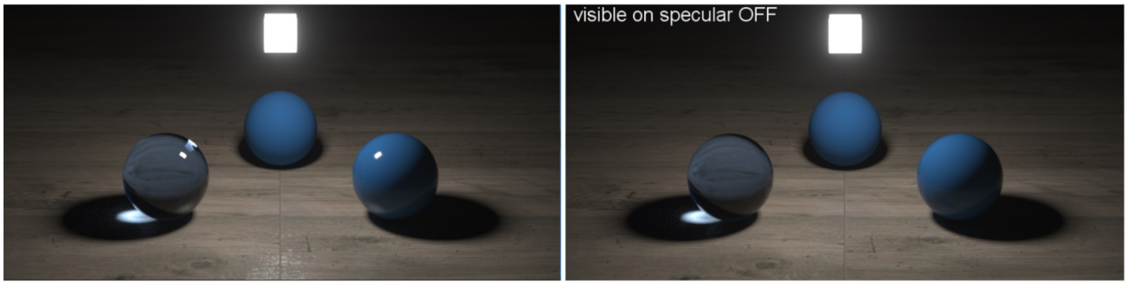


Figure 5: The Visible On Specular parameter when enabled and disabled

Cast Shadows - Enables the light source to cast light and shadows on Diffuse surfaces, and lets you disable Direct Light shadows for Mesh emitters. This option takes effect if the emitter is included in the Direct Light calculation, such as if the Sampling Rate value is greater than **0**. This option is enabled by default.

Flip - Reverses the light source's direction.

Octane IES Light

The **Octane IES**[2] light source is similar to **Octane Light**, but it can accept IES light profiles. The IES light does not have light shape options - the only shape is spherical.

[1]Amount of specular reflection, or the mirror-like reflection of light photons at the same angle. Used for transparent materials such as glass and water.
[2]An IES light is the lighting information representing the real-world lighting values for specific light fixtures. For more information, visit http://www.ies.org/lighting/.

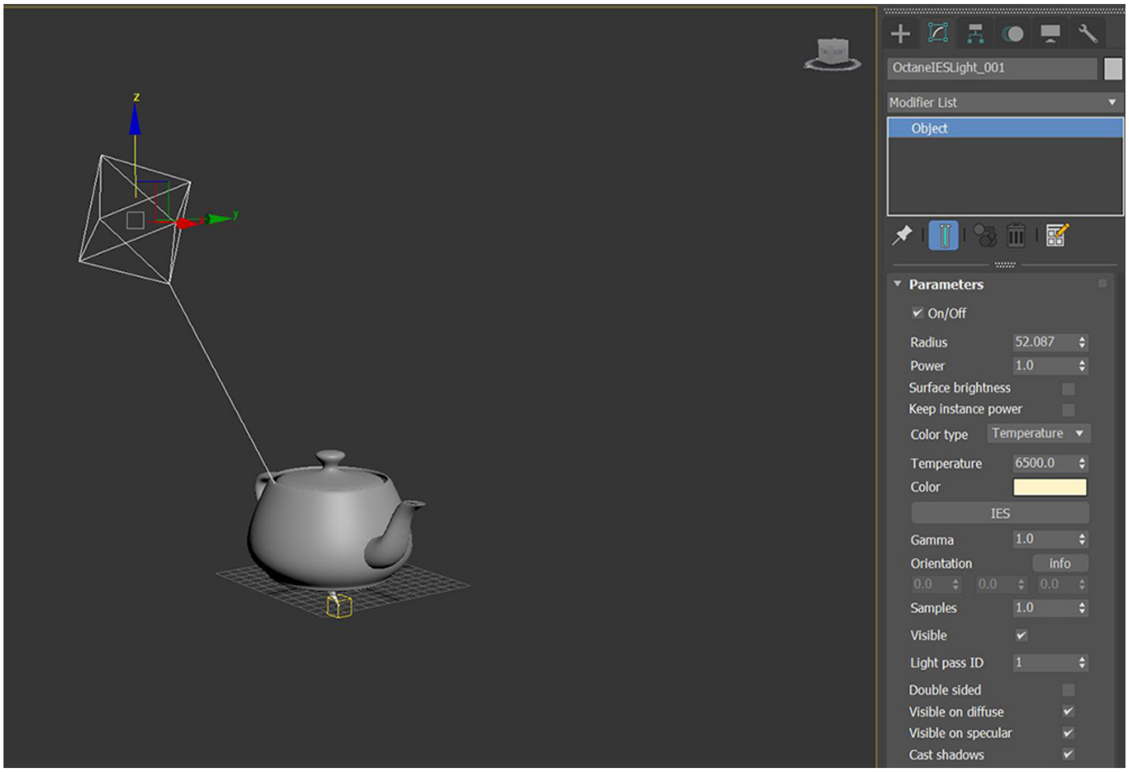


Figure 1: Octane IES light parameters

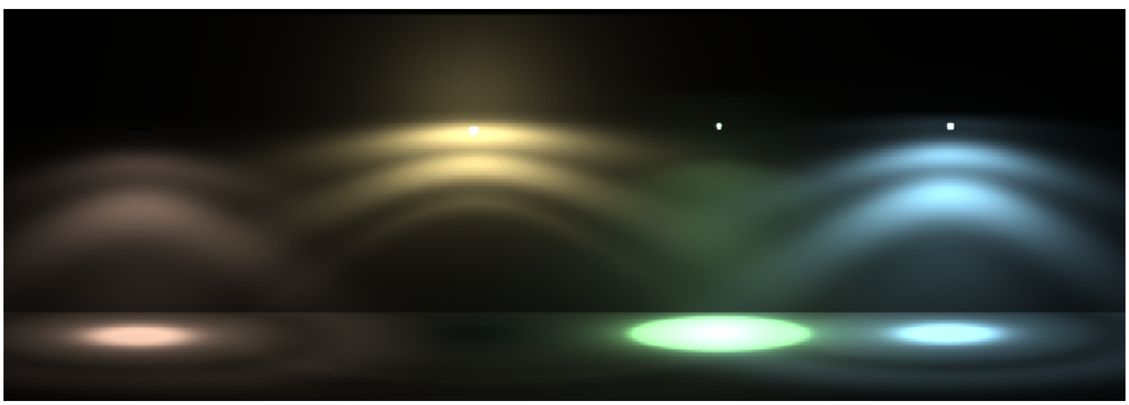


Figure 2: Octane IES light

Octane IES Light Parameters

On/Off - Toggles this light type on and off.

Radius - Sets the sphere's radius.

Power - Represents the total power the light source emits in wattage. You should set each light in the scene to its real-world wattage. For example, to simulate a desk lamp, set this parameter to **25**. By default, this value is **1**, or 100 watts.

Surface Brightness - The light source's size contributes to its overall brightness.

Keep Instance Power - If this is enabled and Surface Brightness is disabled, Power remains constant if you apply uniform scaling to the instance.

Color Type - The light emission is based on a color hue or a temperature, from warmer values of **500** to cooler values of **12000**.

Temperature - The light emission's temperature (in Kelvin).

Color - This defines the color of the light, when Color Type is set to Color.

Gamma[1] - Controls the IES file's gamma.

Orientation - When you load an IES file, this attribute adjusts the light's orientation or direction.

Samples - Choose what light sources receive more samples. Adjusting the Emission's light source sampling rates in the scene leads to a better balance between light sources. You can set the sampling rate to **0**, which means the **Direct Light** calculation excludes the Emitter.

Visible - When this is disabled, the light source is invisible to the camera but still visible in reflections and refractions.

Light Pass ID - Light Pass ID numbers **1** - **8** capture the contribution from the IES Light source emitter. IES Light emitters (just like the **Texture** emitter and **Black Body**[2]) have a **Light Pass ID** pin to assign the Emitter to a light pass. It is possible to assign multiple Emitters to the same light pass. If nothing is configured, all Emitters contribute to Light Pass ID **1** by default.

Double Sided - Makes the Emitter emit light from the front and back sides.

Visible On Diffuse[3] - Makes the light source visible on diffuse surfaces. This is enabled by default. You can enable or disable the Black Body or Texture emission's light sources from casting illumination or shadows on

---

[1] The function or attribute used to code or decode luminance for common displays. The computer graphics industry has set a standard gamma setting of 2.2 making it the most common default for 3D modelling and rendering applications.

[2] An opaque object that emits thermal radiation. In Octane, this is used to designate illumination properties for mesh emitters.

[3] Amount of diffusion, or the reflection of light photons at different angles from an uneven or granular surface. Used for dull, non-reflecting materials or mesh emitters.

Diffuse objects. Disabling this option disables the Emission - it's not visible in diffuse reflections, but it is visible in specular reflections. It's also excluded from the Direct Light calculation.

Visible On Specular[1] - Makes the light source visible on Specular surfaces and hides Emitters just on specular reflections and refractions. This is enabled by default.

Cast Shadows - Enables the light source to cast light and shadows on diffuse surfaces, and lets you disable Direct Light shadows for Mesh emitters. This option takes effect if the Emitter is included in the Direct Light calculation, like when the Sampling Rate value is greater than **0**. This option is enabled by default.

Planetary Environment

The **Planetary** environment type is a flexible Nishita sky model. It is most useful when rendering scenes as they are seen from outer space. For its effects to be visible, the camera has to have a very high **Altitude** value as it moves to outer space to view the expansive horizon of the planetary body. It takes into account the conditions within and beyond the atmosphere of a planetary body (like Earth) and its surroundings in space. Instead of a single ground color and a sky/sunset color, there is a planetary surface that reflects and emits light. This **Node** extends the Environment's **Medium** (volume rendering and subsurface scattering) with an atmospheric scattering through the planetary body's atmosphere. Here, the atmosphere is perceived as a layer of gas surrounding a planetary mass, and it is held in place because of gravity so as the light travels into atmosphere either from the outer layer to the ground or from a light source within the atmosphere, then the atmosphere's density is sampled along the ray at regular intervals, resulting in an amount of scattering based on the atmosphere's density.

---

[1] Amount of specular reflection, or the mirror-like reflection of light photons at the same angle. Used for transparent materials such as glass and water.

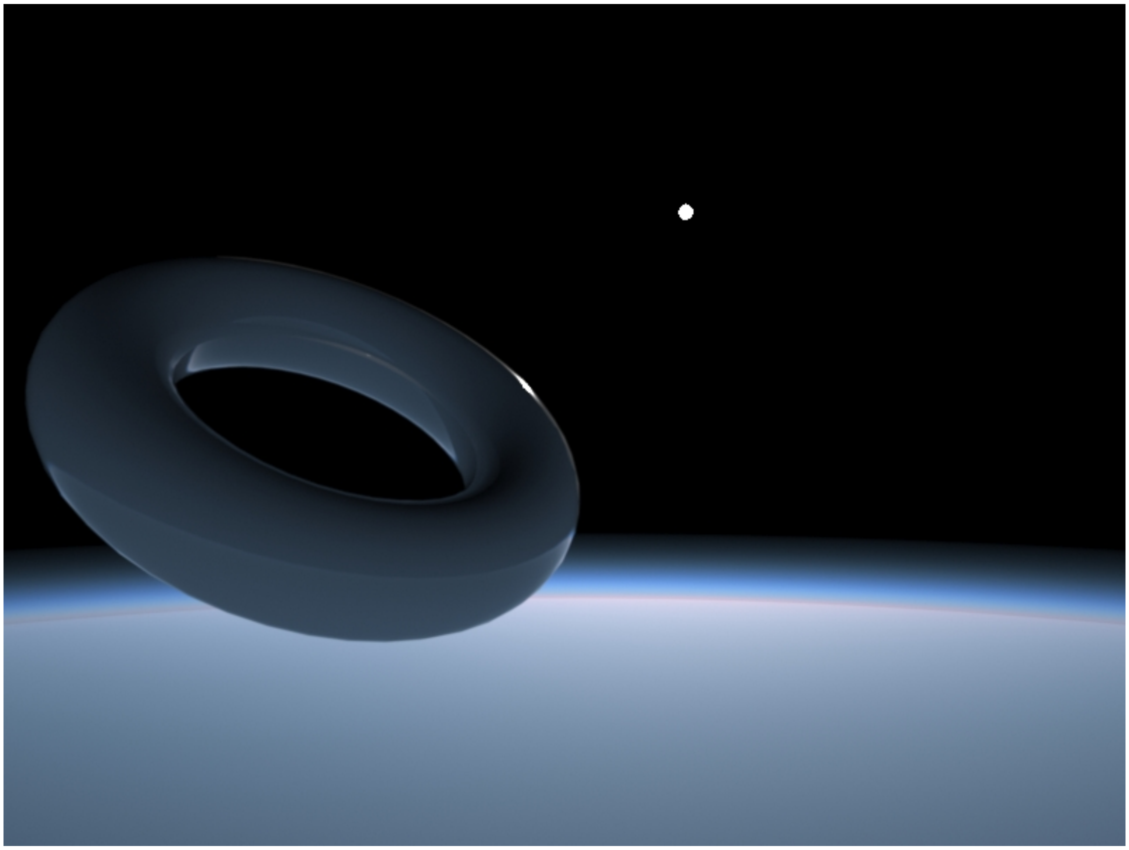


Figure 1: A scene rendered using the Planetary environment, where the camera is set at a very high altitude

Planetary Environment Parameters

Visible Environment

- **On/Off** - Toggles this light type on and off.
- **Kind** - The main option is the default option, and it renders everything in the environment. The **Visible** option activates the **Backplate**, **Reflections**, and **Refractions** parameters. You can use this option to separate foreground and background elements for compositing.
- **Backplate** - Generates a cutout rendering where foreground elements are positioned in the scene.
- **Reflections** - Generates the Planetary environment in scene object reflections.
- **Refractions** - Generates the Planetary environment in scene object refractions.

Sun Direction

- **Use Dummy Direction** - When active, the Planetary environment uses the dummy **Sun** object in the **Viewport** to determine the sun's direction.
- **Longitude/Latitude** - These options become available when you disable **Use Dummy Direction**. These work with the geo-based longitude and latitude values to determine the sun's direction.
- **Month/Day/Hour/GMT Offset** - These options become available when you disable **Use Dummy Direction**. These work with the Longitude/Latitude settings to determine the sun's direction.

Sun

- **Turbidity** - Adjusts the sharpness of the sun light's shadows. Low values create sharp shadows like on a sunny day, and high values diffus the shadows like on a cloudy day.
- **Power** - Adjusts the light's strength. This affects the image's overall contrast and exposure levels.
- **Sun Intensity** - Scale factor that adjust only the sun's strength.
- **North Offset** - Adjusts the scene's actual North direction. This is useful for architecture visualization to ensure the sun's direction to the scene is accurate.
- **Sun Size** - Controls the size of the sun disk that's visible in the scene.
- **Altitude** - The camera's altitude. Set this to a very high value in order to view the expansive horizon of the planetary body.
- **Star Field** - Conveys star fields behind the planet.
- **Importance Sampling** - This toggles the importance sampling of the **Sky** texture, similar to the importance sampling in the **Texture** environment.
- **Medium** - This parameter accepts an **Absorption**[1], **Scattering**[2], or **Volume** medium node to create volume or fog effects across the scene. More information can be found in the Volume **Mediums**[3] topic of the Effects category in this manual.
- **Medium Radius** - Adjusts the Medium's scale.
- **Longitude/Latitude** - Coordinates where we are currently positioned.

Planetary Surface

- **Ground Albedo** - The planet's Surface texture map.
- **Ground Reflection** - The planet's **Specular**[4] texture map.
- **Ground Glossiness** - The planetary glossiness.

---

[1] Defines how fast light is absorbed while passing through a medium.

[2] Defines how fast light gets scattered when traveling through the medium.

[3] The behavior of light inside a surface volume described by scatter, absorption, and transmission characteristics.

[4] Amount of specular reflection, or the mirror-like reflection of light photons at the same angle. Used for transparent materials such as glass and water.

- **Ground Emission** - The planet's Surface texture map at nighttime.
- **Ground Normal Map** - The planet's Normal map.
- **Ground Elevation** - The planet's Elevation map.

Planetary Atmosphere

- **Planetary Axis** - The planet's rotational axis running through the North and South Poles.
- **Planetary Angle** - The rotation around the planetary axis.

Toon Directional

The **Toon Directional** light works with **Toon** materials - it was developed only for this **Material**[1] type. This adds a **Directional** light to the scene, much like the native Direct Light in 3DS Max®, and is suitable for global lighting. Rotation is important, and position does not have any effect on the illumination. The **Size** and **Power** parameters have no effect on the light source due to the stylistic nature of toon shading. The **Efficiency** parameter controls the light source's brightness when the **Color Type** is set to **Value**. If the Color Type is set to a **Color** or **Texture**, the Efficiency parameter has no effect.

---

[1] The representation of the surface or volume properties of an object.

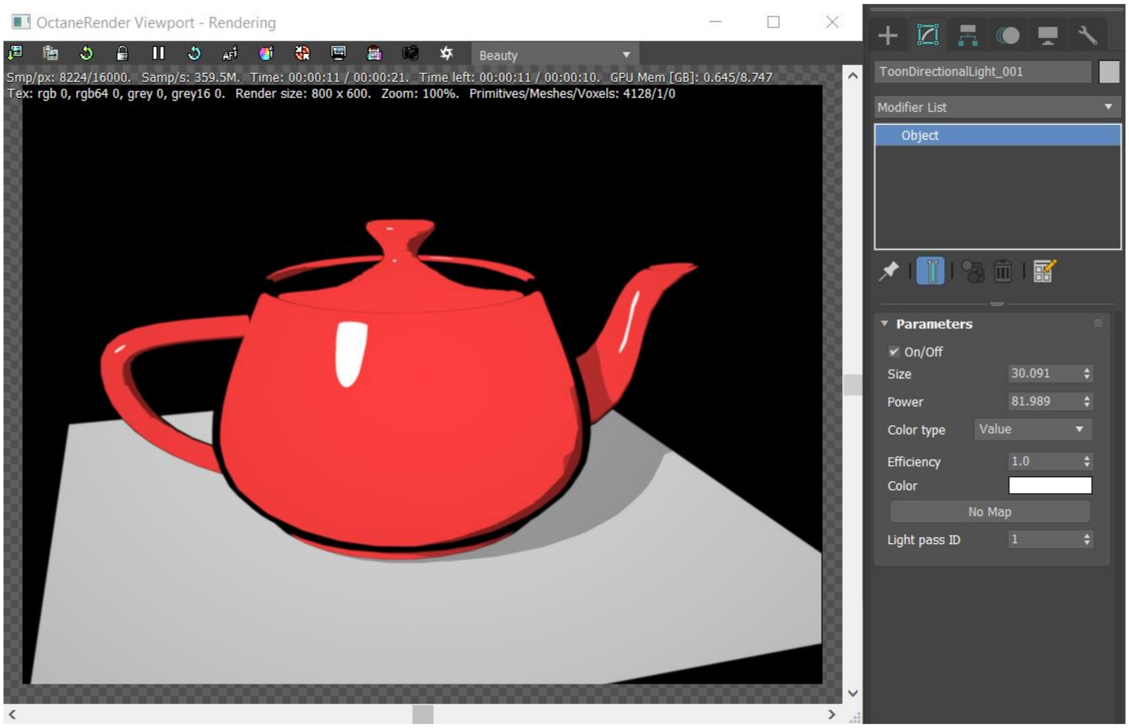


Figure 1: The Toon Directional light

Toon Directional Parameters

On/Off - Toggles this light type on and off.

Size - Defines the light object size.

Power - Strength of the light source.

Color Type - Defines how to color the light source.

Efficiency - When **Color Type** is set to **Value**, this defines a gray scale value for the light source betwen 0 and 1.

Color - When **Color Type** is set to **Color**, this defines a color value for the light source.

Texture - When **Color Type** is set to **Texture**, this defines a texture value for the light source. Initially set to **No Map**.

Light Pass ID - Light Pass ID numbers **1** - **8** capture the Octane Light emitter's contribution. Octane Light emitters (just like the **Texture** emitter and **Black Body**) have a **Light Pass ID** that assigns the Emitter to a

Light Pass. You can assign multiple Emitters to the same Light Pass. If nothing is configured, all emitters contribute to Light Pass ID **1** by default.

Toon Point

The **Toon Point** light works with **Toon** materials, as it was developed only for this **Material**[1] type. This adds a point light to the scene, much like the native point light in 3DS Max®. Position is important, but rotation has no effect as this light source type illuminates in all directions. The **Size** and **Power** parameters have no effect on the light source due to the stylistic nature of toon shading. The **Efficiency** parameter controls the light source's overall brightness when the **Color Type** is set to **Value**. If the Color Type is set to a color or texture, the Efficiency parameter has no effect.

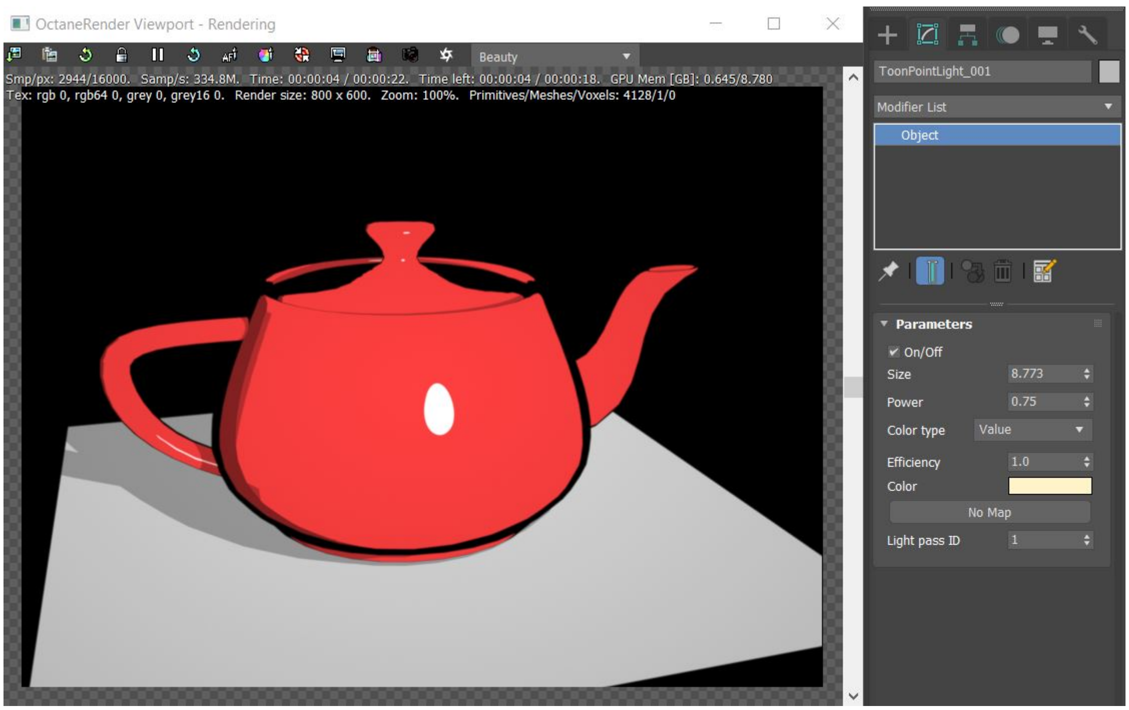


Figure 1: The Toon Point light

Toon Point Parameters

On/Off - Toggles this light type on and off.

[1]The representation of the surface or volume properties of an object.

Size - Defines the light object size.

Power - Strength of the light source.

Color Type - Defines how to color the light source.

Efficiency - When **Color Type** is set to **Value**, this defines a gray scale value for the light source betwen 0 and 1.

Color - When **Color Type** is set to **Color**, this defines a color value for the light source.

Texture - When **Color Type** is set to **Texture**, this defines a texture value for the light source. Initially set to **No Map**.

Light Pass ID - Light Pass ID numbers **1** - **8** capture the Octane Light emitter's contribution. Octane Light emitters (just like the **Texture** emitter and **Black Body**) have a **Light Pass ID** that assigns the Emitter to a Light Pass. You can assign multiple Emitters to the same Light Pass. If nothing is configured, all emitters contribute to Light Pass ID **1** by default.

Volume Spotlight

This is a Spotlight that uses the **Spectron** procedural lighting system, with blockers, barn doors, gels (in the **Distribution** pin) and more. Spectron is exposed as a **Procedural** light node type, which you can use for quick volumetric effects and spotlight generation.

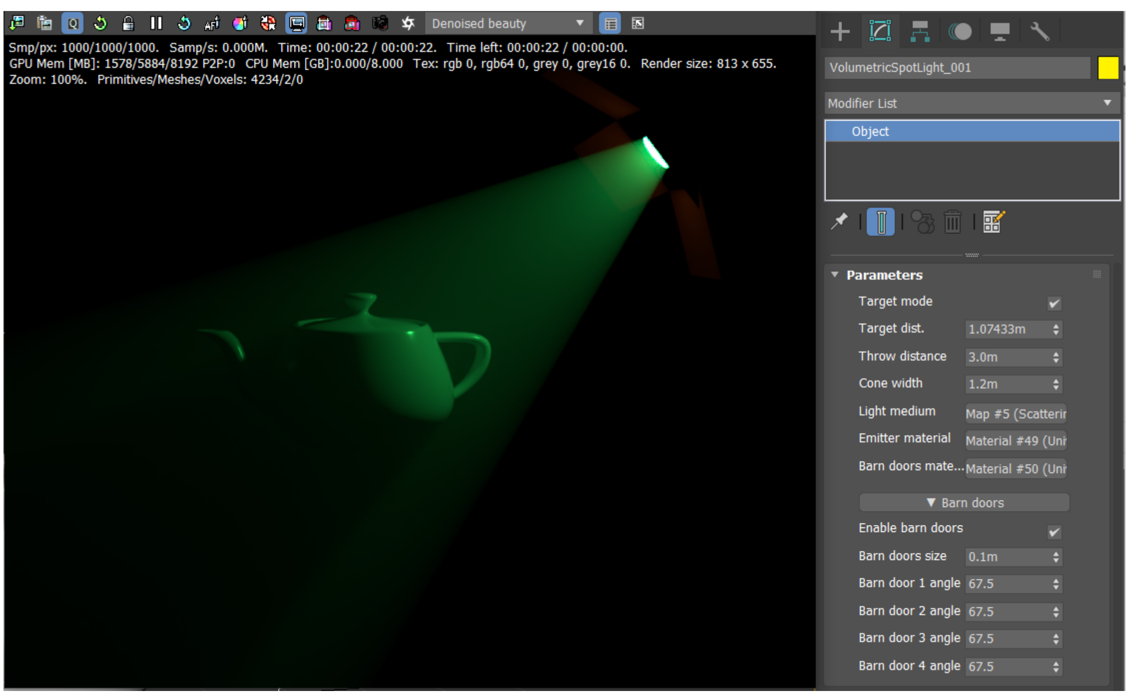


Figure 1: Volume Spotlight

Volume Spotlight Parameters

Target Mode - Creates a target to orient the spotlight.

Target dist. - Set the target a specific distance from the light source.

Throw Distance - Max distance the spotlight will affect.

Cone Width - Controls the spread of the light.

Light Medium - Lets you use a specific medium.

Emitter Material[1] - Define a material with emission properties to control the light source.

Barn Doors Material - Define a material for the barn doors.

Enable Barn Doors - Toggle the barns doors on or off.

Barn Doors Size - Scales the barn doors.

Barn Door Angle - Controls the barn door angle to control light occlusion.

Mesh Emitters

In order to use a **Mesh** as a light source, first connect a **Diffuse**[2] or **Universal** material to the Mesh. Next, connect a **Black Body**[3] or a **Texture** emission node to the **Diffuse** or **Universal** material's **Emission** slot. The Black Body and Texture emission nodes are covered in more detail in their respective topics in this manual.

---

[1]The representation of the surface or volume properties of an object.

[2]Amount of diffusion, or the reflection of light photons at different angles from an uneven or granular surface. Used for dull, non-reflecting materials or mesh emitters.

[3]An opaque object that emits thermal radiation. In Octane, this is used to designate illumination properties for mesh emitters.

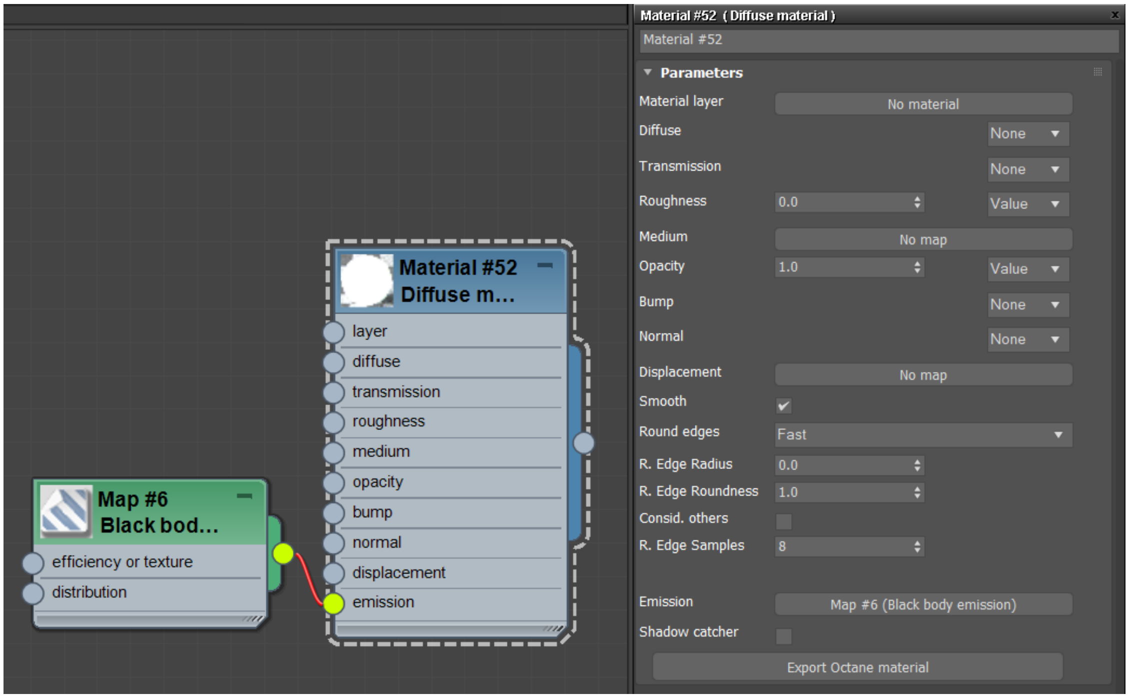


Figure 1: The Black Body emission node connected to a Diffuse or Universal material's Emission slot

Emission Types

Black Body Emission - Uses **Color Temperature** (in Kelvin) and **Power** to control the light's color and intensity.

Texture Emission - Any valid Texture type can set the light intensity. This creates effects such as TV screens by using a TV image texture as the source.

Cameras

OctaneRender® for 3DS Max® provides three basic types of **Cameras**:

- OSL Baking
- OSL Camera

- Camera
- Universal Camera

Camera contains options for

- Standard
- Baking
- Panoramic - Spherical, Cylindrical, and Cube

Universal Camera contains options for

- Thin Lens
- Orthographic
- Fisheye
- Equirectangular
- Cubemap

You can access these camera types from the **Create** panel's **Camera** category.

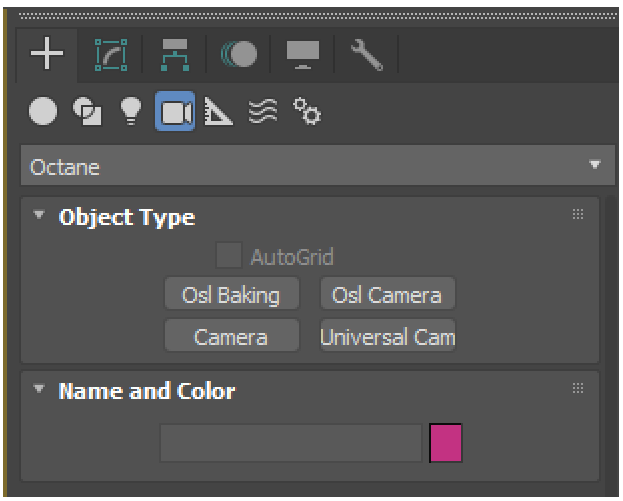


Figure 1: Accessing cameras from the Create panel

Camera Settings

You can access OctaneRender® **Camera** settings in the **Modify** panel when you add an OctaneRender® camera to the scene and make it the current selection. The Camera settings are identical for the various types of Cameras available (**Standard**, **Baking**, **Spherical**, **Cylindrical**, **Cube Map**).

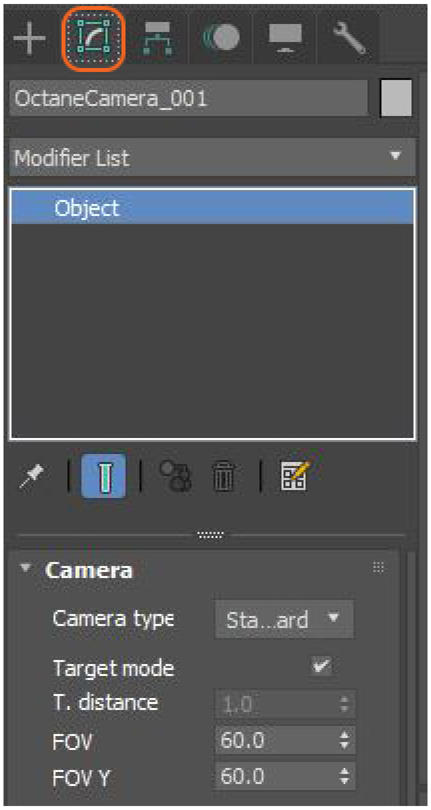


Figure 1: Accessing Octane Camera settings from the Modify panel

You can also access OctaneRender's camera settings from the **Camera** tab in the **Render Setup** window when no OctaneRender® cameras are present in the scene. These parameters provide camera controls for the active **Viewport**.

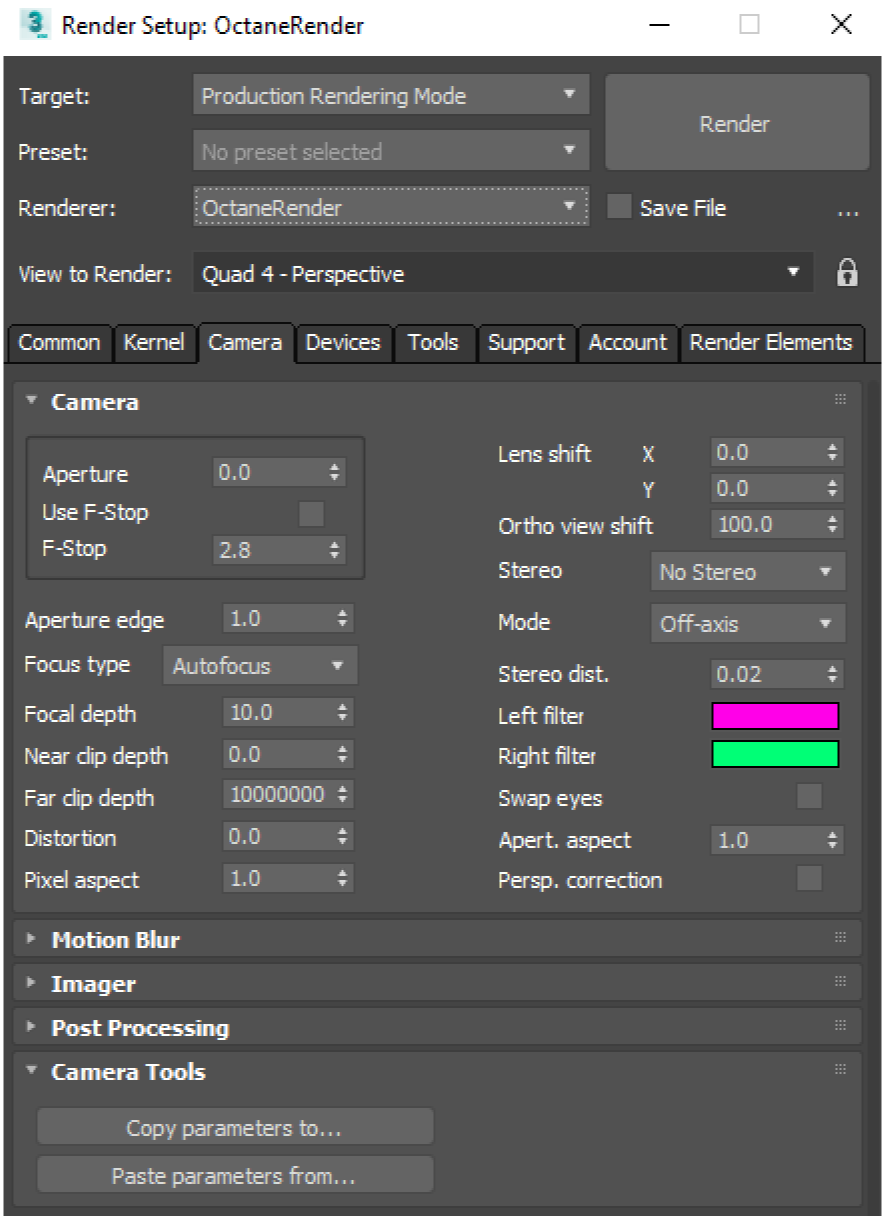


Figure 2: Octane Camera settings in the Render Setup window

OctaneRender® Camera Parameters

Camera Type - You can select from Standard, Baking, Spherical, Cylindrical, and Cube Map Cameras

Target mode - Adds a target item to help orient your light direction.

T. Distance. - Set the target a specific distance from the source.

FOV[1] - Controls the horizontal camera view based on angle.

FOV Y - Controls the vertical camera view based on angle, but only affects Spherical and Cylindrical camera types.

Aperture[2] - The camera lens opening's radius, measured in centimeters. Low values create a deep depth-of-field, where everything is in focus. High values create a shallow depth-of-field, where objects in the foreground or background are out-of-focus.

Use F-Stop And F-Stop - Controls the field-of-view and depth-of-field, similar to a real-world camera. The **F-Stop** value is the aperture-to-focal-length ratio.

Aperture Edge - Controls aperture edge detection at all points within the aperture. Lower values give more pronounced edges to out-of-focus objects affected by a shallow depth-of-field. Aperture Edge modifies the depth-of-field's bokeh effect. High values increase the contrast towards the edge.

Distortion - Adjusts the spherical and cylindrical distortion. The rendered image displays the entire sphere and uses equidistant cylindrical projection.

Focus Type - Select the type of focus to use for the camera. If **AutoFocus** is enabled, the focus is kept on the closest visible surface at the center of the image, regardless of the Aperture, Aperture Edge, or the Focal Depth values.

Focal Depth - The depth of the plane in focus, measured in meters.

Stereo - This specifies the output rendered in stereo.

- **Left** - Renders the image for the left eye.
- **Right** - Renders the image for the right eye.
- **Side-By-Side** - Renders the scene as a pair of two-dimensional images.
- **Anaglyphic** - Makes the render viewable with red/blue 3D glasses.
- **Over-Under** - The pair of two-dimensional images is placed one above the other for special viewers.

Mode - When you choose a Stereo mode, you can choose **Off-Axis** or **Parallel** stereo camera projections.

Stereo Distance - The distance between the left and the right eye in Stereo mode, measured in meters. The stereo distance is also referred to the IPD (Inter Pupillary Distance), and is often exchanged with terms like stereo interocular distance or eye distance. For realistic depth, use values between **0.055** and **0.075**.

Stereo Distance Falloff - Controls how fast the eye distance reduces towards the poles. This reduces eye strain at the poles when the panorama is viewed through a head-mounted display. A value of **1** reduces the

---

[1] The area that is visible to a camera lens usually measured in millimeters. A wide angle lens provides a larger field of view and a telephoto lens provides a narrow field of view.
[2] Determines how much light enters a camera lens. A large aperture produces a narrow depth of field and a small aperture produces a wide depth of field.

eye distance from equator to the poles, which creates a relaxed viewing experience. This also causes flat surfaces to appear curved. A value smaller than **1** keeps the eye distance constant for a larger latitude range above and below the horizon, but it rapidly reduces the eye distance near the poles. This keeps a flat surface, but causes more eye strain near the poles, which you can reduce again by setting the pano cutoff latitude to a value less than 90 degrees.

Left/Right Stereo Filter - The left and right filter colors adjust the colors for the anaglyphic stereo effect in the render.

Swap Eyes - Swaps the left and right eye images.

Lens Shift - This is useful for architectural rendering, where you want to render images of tall buildings or structures from a similar height as the human eye, but keeping the vertical lines parallel.

Near Clip Depth - Distance from the camera to the near clipping plane, measured in meters. The main purpose is for interior scenes, where you can get a good shot of the whole room but not without a very large **field of view**[1], because the camera needs to stay inside the room.

With camera clipping (near plane), you can position the camera outside the room - lower the field of view and increase the clipping plane distance in front until the closest walls are clipped out. The geometry is not altered, but the camera clipping is altered, which means that shadows, reflections, and refractions are still affected by the clipped geometry.

Far Clip Depth - The distance from the camera, where objects farther than the specified distance aren't seen from the camera.

Pixel Aspect - This makes the pixel ratio non-square to accommodate older formats like NTSC or PAL.

Aperture Aspect Ratio - Stretches or squashes the depth-of-field disc.

Blackout Latitude - This is the +/- latitude where the panorama cuts off when Stereo Rendering is enabled. This defines the minimum latitude (in spherical camera coordinates) where the rendering is blacked out. The area with higher latitudes is blacked out.

Keep Upright - When enabled, the panoramic camera is always oriented towards the horizon, and the up-vector stays in its default direction (vertical).

Orthographic - Makes the camera show an orthographic view. If disabled, the camera shows a perspective view.

Perspective Correction - If the up-vector is vertical, enabling this option keeps vertical lines parallel. This is useful for architectural rendering, when you want to render images of tall buildings from a similar height as the human eye, but keep the vertical lines parallel.

---

[1] The area that is visible to a camera lens usually measured in millimeters. A wide angle lens provides a larger field of view and a telephoto lens provides a narrow field of view.

Baking Submenu - These parameters are covered in greater detail in the **Texture Baking**[1] topic under the Rendering section in this manual.

Bokeh Rot - Adjusts the quality of highlights when depth-of-field blurring is apparent. Raising the Aperture value increases depth-of-field blurring. **Bokeh Rotation** rotates the shape of the blurred highlights. This becomes more obvious when the **Bokeh Roundness** is lowered.

Bokeh Round - Keeps blurred highlights rounded. Lowering this value reduces the roundness and increases the appearance of edges on the highlights.

Bokeh Sides - Sets the number of edges on blurred highlights.

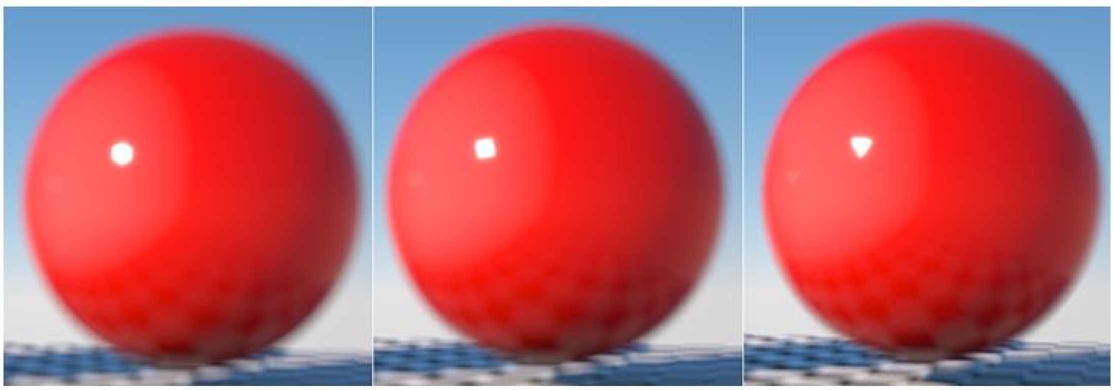


Figure 3: From left to right, the sphere is rendered with a Bokeh roundness set to 6, 4, and 3

OSL Baking

The **OSL Baking** camera is a scriptable camera. You can create custom camera types for any purpose (such as **VR**[2] warping) with OSL (**Open Shader Language**[3]) scripts. It is a very flexible camera used to match the rendering to the existing footage. To learn about the generic OSL standard, read the OSL Readme and PDF documentation.

---

[1]A process in which scene lighting is "baked" into a texture map based on an object's UV texture coordinates. The resulting texture can then be mapped back onto the surface to create realistic lighting in a real-time rendering environment. This technique is frequently used in game engines and virtual reality for creating realistic environments with minimal rendering overhead.
[2]Immersively engaging and experiencing depth perception in a three dimensional scene through stereo vision goggles and head-mounted displays.
[3]A shading language developed by Sony Pictures Imageworks. There are multiple render engines that utilize OSL as it is particularly suited for physically-based renderers.

The OSL cameras work in conjunction with other OSL features like the **OSL** texture node. While the OSL camera's parameters are identical to the **Standard** camera, the OSL Baking camera contains a unique set of camera parameters used for controlling the baking process. These parameters are covered in more detail in the **Texture Baking**[1] topic under the Rendering section of this manual.

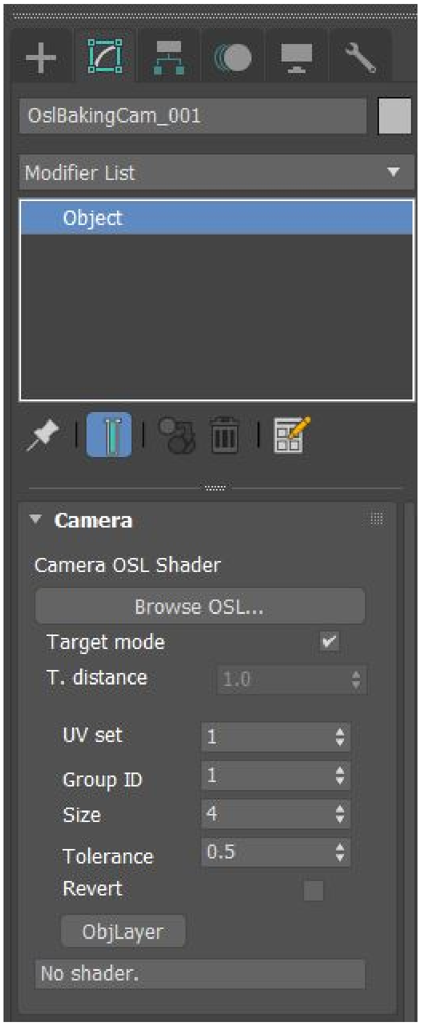


Figure 1: The OSL Baking camera parameters

---

[1] A process in which scene lighting is "baked" into a texture map based on an object's UV texture coordinates. The resulting texture can then be mapped back onto the surface to create realistic lighting in a real-time rendering environment. This technique is frequently used in game engines and virtual reality for creating realistic environments with minimal rendering overhead.

OSL Camera

The **OSL** camera is a scriptable camera. You can create custom **Camera** types for any purpose (such as **VR**[1] warping) with OSL (**Open Shader Language**[2]) scripts. It is a very flexible camera that matches the rendering to the existing footage. To learn about the OSL standard, refer to the OSL Readme and PDF documentation. The OSL cameras are designed to work with other OSL features, such as the **OSL** texture node.

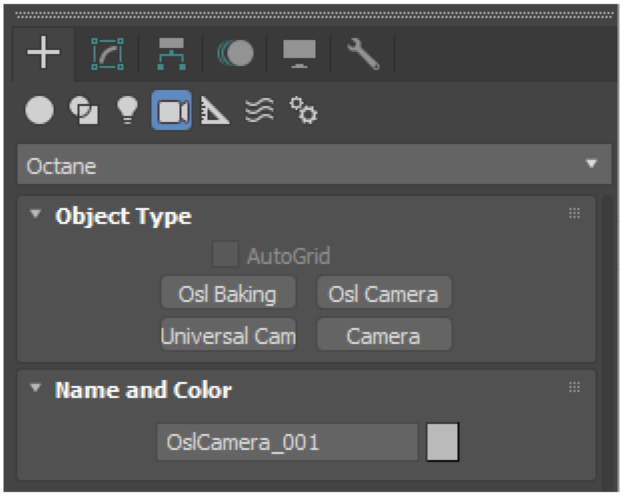


Figure 1: Accessing the OSL cameras from the Create panel

Standard Camera

Thin Lens

The **Standard** camera type is accessible from the **Camera Type** parameter's dropdown in the **Modify** panel. The Standard camera is the typical **Camera** type used for most rendering scenarios.

---

[1]Immersively engaging and experiencing depth perception in a three dimensional scene through stereo vision goggles and head-mounted displays.
[2]A shading language developed by Sony Pictures Imageworks. There are multiple render engines that utilize OSL as it is particularly suited for physically-based renderers.

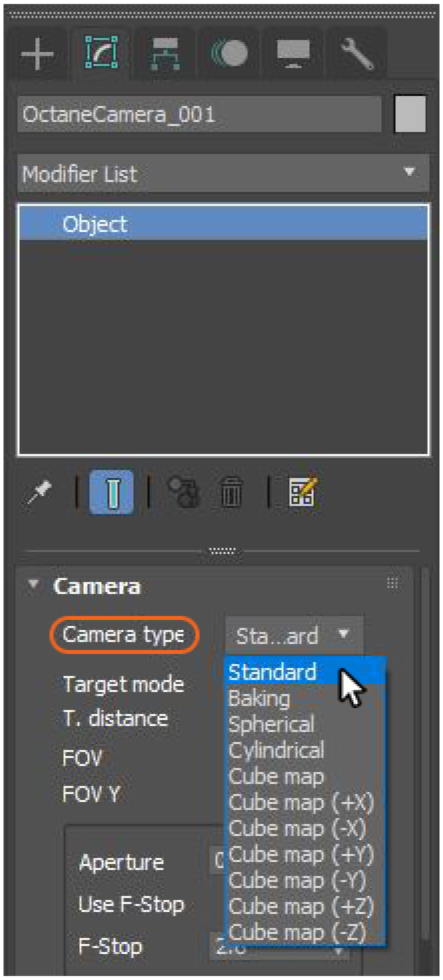


Figure 1: Selecting the Standard (Thin Lens) camera type from the Modify panel

The Standard camera parameters are covered in more detail in the Camera Settings topic in this manual.

Baking

The **Baking** camera type renders lighting, texture, and material data directly into **Texture** maps. This process is covered in greater detail in **Texture Baking**[1] topic under the Rendering section in this manual.

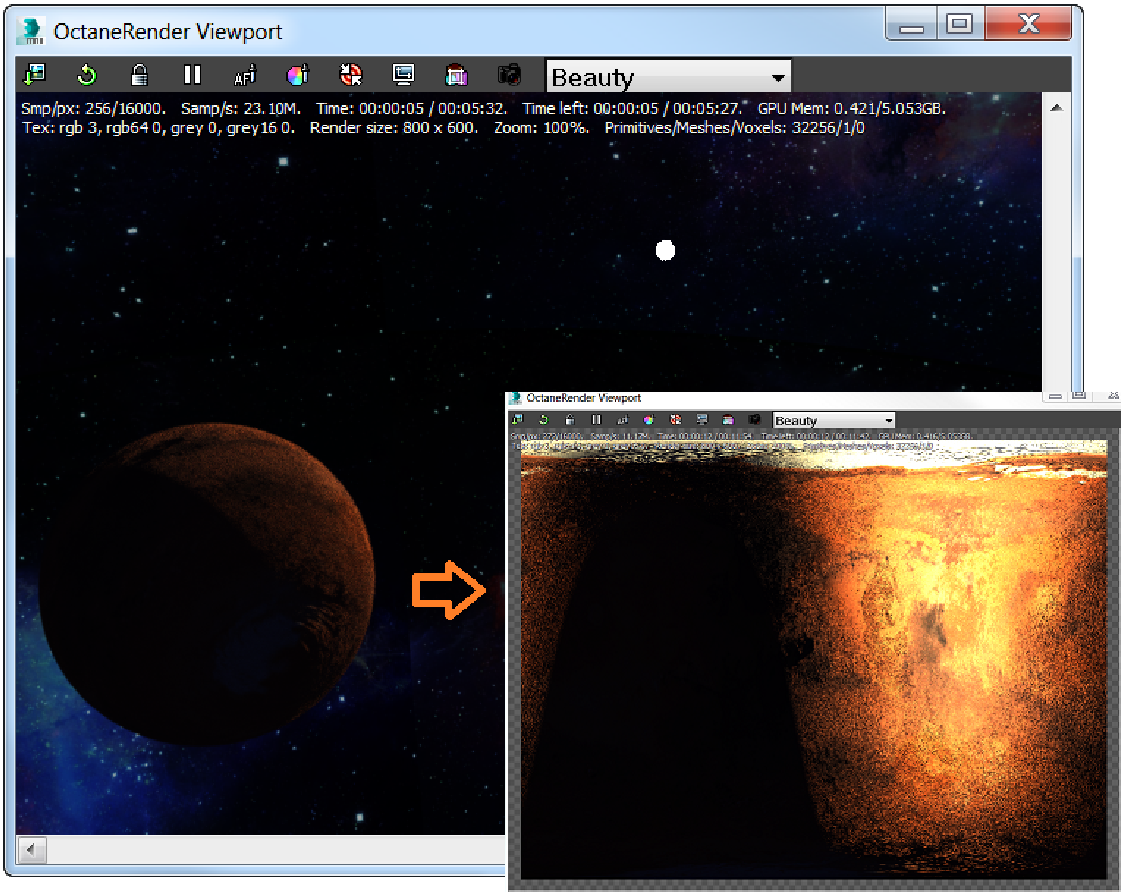


Figure 1: Lighting and material data bake to a Texture map using the Baking camera

Panoramic

[1]A process in which scene lighting is "baked" into a texture map based on an object's UV texture coordinates. The resulting texture can then be mapped back onto the surface to create realistic lighting in a real-time rendering environment. This technique is frequently used in game engines and virtual reality for creating realistic environments with minimal rendering overhead.

The **Panoramic** camera type is accessible from the **Camera Type** parameter dropdown in the **Modify** panel. The Panoramic camera is used for rendering **VR**[1]-related images. There are three types of Panoramic cameras available: **Spherical**, **Cylindrical**, and various **Cube Map** types (**+x**, **-x**, **+y**, **-y**, **+z**, **-z**).

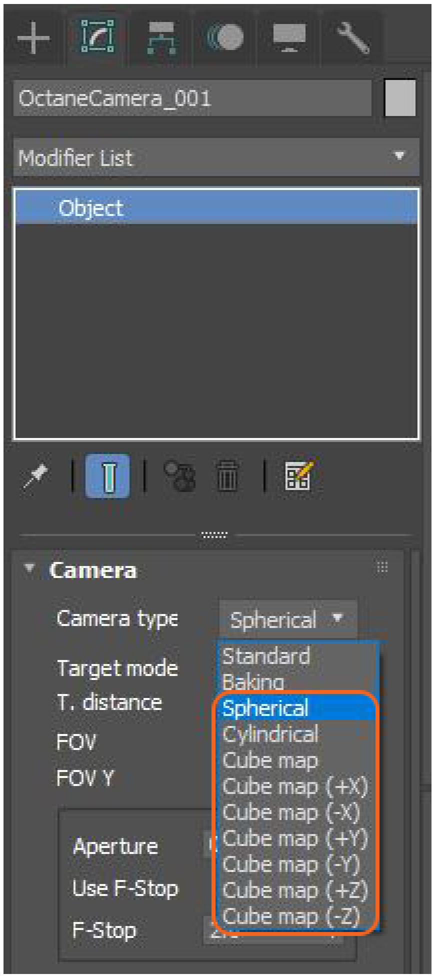


Figure 1: Accessing the Panoramic camera types from the Modify panel

Panoramic camera parameters are covered in more detail in the Camera Settings topic in this manual.

[1] Immersively engaging and experiencing depth perception in a three dimensional scene through stereo vision goggles and head-mounted displays.

Universal Camera

The **Universal** camera is a full-featured camera, with support for five different camera types:

- **Thin lens**
- **Orthographic**
- **Fisheye**
- **Equirectangular**
- **Cubmap**

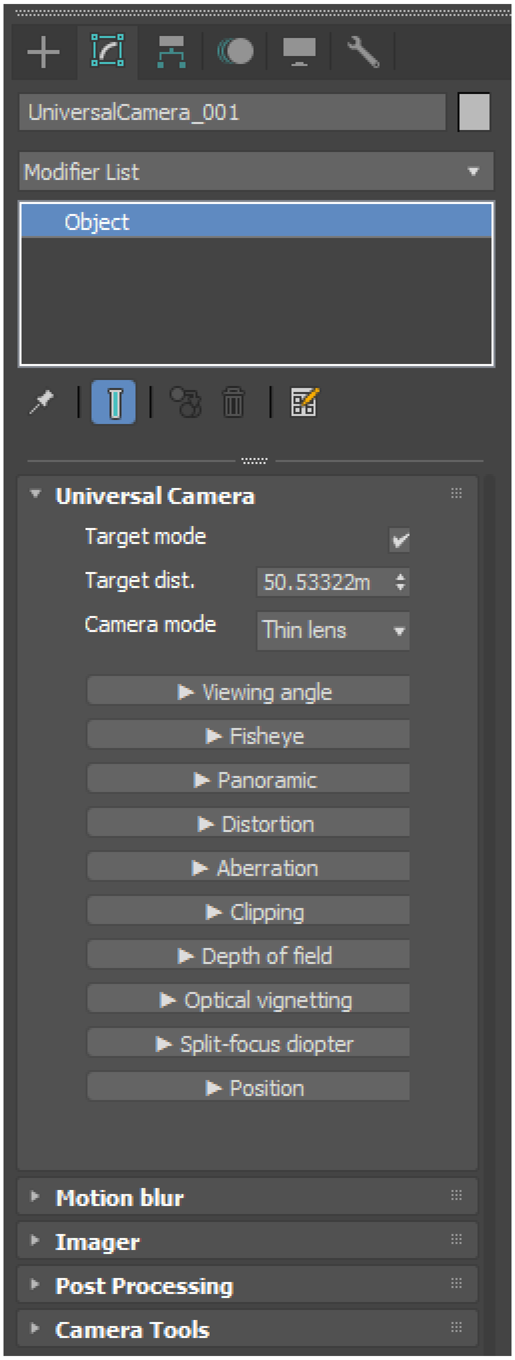


Figure 1: Universal Camera Parameters

Universal Lens Parameters

Target mode - Adds a target item to help orient your light direction.

Target Dist. - Set the target a specific distance from the source.

Camera Mode - Options to set your camera type.

Viewing Angle

- **Field Of View**[1] - The horizontal field-of-view, measured in degrees.
- **Lens Shift X** - The lens shift on X, as a factor of the image width.
- **Lens Shift Y** – The lens shift on Y, as a factor of the image height.
- **Lens Shift Z** – The lens shift on Z, as a factor of the image depth.
- **Pixel Aspect Ratio** - The pixels' X:Y aspect ratio.

Fisheye

- **Field Of View** - The camera's field of view, measured in degrees.
- **Fisheye Type** - Choose between covering the lens circle in the sensor, or covering the whole sensor.
- **Hard Vignette** - Renders the lens (Circular fisheye only).
- **Fisheye Projection** - The projection function used for the fisheye.

Panoramic

- **Horizontal Field of View** - The horizontal field of view, in degrees. This sets the X-coordinate for the camera's horizontal field of view in the scene. This is ignored when cube mapping is used.
- **Vertical Field of View** - The vertical field of view, in degrees. This sets the Y-coordinate for the camera's vertical field of view in the scene. This is ignored when cube mapping is used.
- **Cubemap Layout** - Determines the configuration for laying out the cubemap.
- **Equi-angular Cubemap** - Activates an equi-angular cubemap projection.

Distortion

- **Use Distortion Texture** - Enables the distortion texture.
- **Distortion Texture** - The Distortion texture input.
- **Spherical Distortion** - The amount of spherical distortion.
- **Barrel** - Straight lines appear curved.
- **Barrel (Corners)** - Straight lines appear curved, affecting corners.

Aberration

- **Spherical** - Rays hitting the edge of the lens focus closer to the lens.
- **Coma** - Rays hitting the lens edge have a larger field of view.
- **Astigmatism** - Sagittal and tangential rays focus at different distances from the lens.
- **Field Curvature** - The curvature of the plane in focus.

---

[1] The area that is visible to a camera lens usually measured in millimeters. A wide angle lens provides a larger field of view and a telephoto lens provides a narrow field of view.

Clipping

- **Near Clip Depth** - Distance from the camera to the nearest clipping plane, measured in meters.
- **Far Clip Depth** - Distance from the camera to the farthest clipping plane, measured in meters.

Depth of Field[1]

- **Focus Type**
  - **Auto-Focus** - Keeps the focus on the closest visible surface at the center of the image. This setting is on by default.
  - **Focal Depth** - Define the focus distance.
  - **Target Focus** - Use the camera target to define focus.
- **Focal Depth** - The depth of the plane in focus, measured in meters. If you are having trouble seeing a result when you adjust this setting, double-check to make sure that Auto-Focus is enabled. Auto-Focus overrides the Focal Depth setting.
- **Use F-Stop And F-Stop** - Controls the field-of-view and depth-of-field, similar to a real-world camera. The **F-Stop** value is the aperture-to-focal-length ratio.
- **Aperture**[2] - The radius of the camera's lens opening, measured in centimeters. Low values have a wide depth-of-field, where everything is in focus. High values have a shallow depth-of-field, where objects in the foreground and background will be out of focus.
- **Aperture Aspect Ratio** - This allows users to squash and stretch the depth-of-field disc.
- **Aperture Shape** - Controls the shape of the aperture.
- **Aperture Edge** - Modifies the relative distribution of rays across the aperture, impacting the hardness of the edges of bokeh shapes. Higher values increase the contrast towards the edge. Values between 0 and 1 simulate an apodization filter.
- **Aperture Blade Count** - The number of blades forming the iris diaphragm.
- **Aperture Rotation** - The rotation of the aperture shape in degrees.
- **Aperture Roundness** - The roundness of the blades forming the iris diaphragm.
- **Central Obstruction** - Simulates the obstruction from the secondary mirror of a catadioptric system. This option is only enabled on circular apertures.
- **Notch Position** - Determines the position of the notch on the blades.
- **Notch Scale** - Scale of the notch.
- **Custom Aperture** - Sets the custom aperture opacity map. The projection type must be set to OSL Delayed UV.

Optical Vignetting

---

[1]The distance between the nearest and farthest objects in a scene that appear acceptably sharp in an image. Although a lens can precisely focus at only one distance at a time, the decrease in sharpness is gradual on each side of the focused distance, so that within the DOF, the unsharpness is imperceptible under normal viewing conditions. source: wikipedia (https://en.wikipedia.org/wiki/Depth_of_field)
[2]Determines how much light enters a camera lens. A large aperture produces a narrow depth of field and a small aperture produces a wide depth of field.

- **Optical Vignetting Distance** - The distance between the lens and the opening of the lens barrel.
- **Optical Vignetting Scale** - The scale of the opening of the lens barrel relative to the aperture.

Split-Focus Diopter

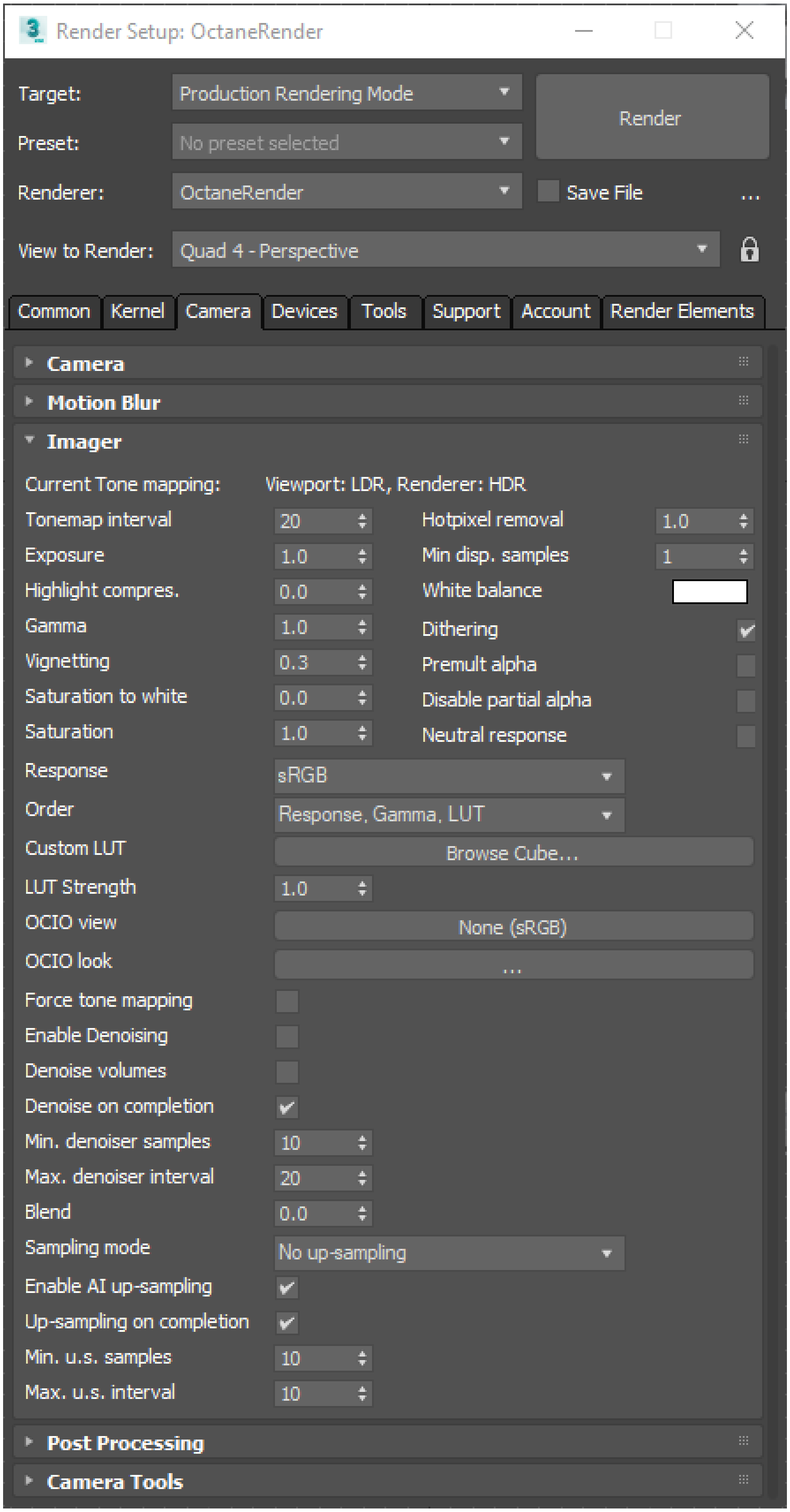
- **Enable** - Enables the split-focus diopter.
- **Diopter Focal Depth** - Depth of the plane in focus measured in meters.
- **Diopter Rotation** - Rotation of the split-focus diopter in degrees.
- **Diopter Translation** - Translation of the split-focus diopter.
- **Diopter Boundary Width** - Width of the boundary between the two fields.
- **Diopter Boundary Falloff** - Controls how quickly the Split-Focus diopter focal depth blends into the main focal depth.
- **Show Diopter Guide** - Displays guide lines, toggling this option on or off will restart the render.

Position

- **Keep Upright** - The panoramic camera always orients towards the horizon, and the up-vector stays in its default vertical direction (0, 1, 0).

Camera Imager Settings

You can access the **Camera Imager Settings** from the **Camera** tab in the **Render Setup** window, or from the **Modify** panel if you added an OctaneRender® **Camera** to the scene.



Render Setup: OctaneRender
Target:
Production Rendering Mode
Preset:
No preset selected
Renderer:
OctaneRender
Render
Save File
View to Render:
Quad 4 - Perspective
Common
Kernel
Camera
Devices
Tools
Support
Account
Render Elements
Camera
Motion Blur
Imager
Current Tone mapping:
Viewport: LDR, Renderer: HDR
Tonemap interval
20
Hotpixel removal
1.0
Exposure
1.0
Min disp. samples
1
Highlight compres.
0.0
White balance
Gamma
1.0
Dithering
Vignetting
0.3
Premult alpha
Saturation to white
0.0
Disable partial alpha
Saturation
1.0
Neutral response
Response
sRGB
Order
Response, Gamma, LUT
Custom LUT
Browse Cube...
LUT Strength
1.0
OCIO view
None (sRGB)
OCIO look
...
Force tone mapping
Enable Denoising
Denoise volumes
Denoise on completion
Min. denoiser samples
10
Max. denoiser interval
20
Blend
0.0
Sampling mode
No up-sampling
Enable AI up-sampling
Up-sampling on completion
Min. u.s. samples
10
Max. u.s. interval
10
Post Processing
Camera Tools

Figure 1: Accessing the Camera Imager settings from the Render Setup window

The **Modify** panel for Octane Camera and Universal Camera have similar **Imager** settings.

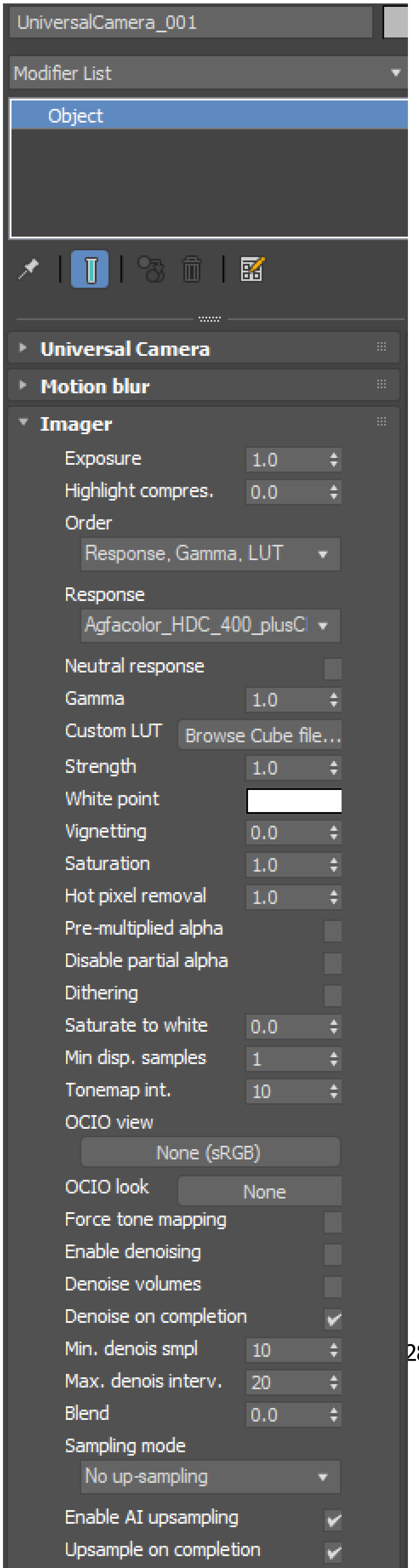

UniversalCamera_001
Modifier List
Object
Universal Camera
Motion blur
Imager
Exposure 1.0
Highlight compres. 0.0
Order
Response, Gamma, LUT
Response
Agfacolor_HDC_400_plusC
Neutral response
Gamma 1.0
Custom LUT Browse Cube file...
Strength 1.0
White point
Vignetting 0.0
Saturation 1.0
Hot pixel removal 1.0
Pre-multiplied alpha
Disable partial alpha
Dithering
Saturate to white 0.0
Min disp. samples 1
Tonemap int. 10
OCIO view
None (sRGB)
OCIO look None
Force tone mapping
Enable denoising
Denoise volumes
Denoise on completion
Min. denois smpl 10
Max. denois interv. 20
Blend 0.0
Sampling mode
No up-sampling
Enable AI upsampling
Upsample on completion

Figure 2: Camera Imager settings from the Modify panel.

Camera Imager Parameters

Current Tone Mapping[1] - Display the dynamic range settings used for the viewport and renderer.

Tonemap Interval - Maximum interval between tonemapping operations (in seconds).

Exposure - Controls the scene's exposure. Low values create a dark scene, while high values brighten the scene. Exposure has no effect on any of the render layer passes.

Highlight Compression - Reduces burned out highlights by compressing them and reducing their contrast.

Gamma[2] - Adjusts the render's gamma and controls the image's overall brightness.

Vignetting - Darkens the corners of the render, which increases the render's realism. OctaneRender® doesn't apply vignetting to any of the beauty passes except the main pass.

Saturation To White - Creates multicolored reflections when the sun is too bright. Increasing this value changes the colors to white. This is also applicable to all sources of light.

Saturation - Adjusts the amount of color saturation for the render.

Hot Pixel Removal - Removes bright pixels (known as fireflies) during the rendering process. Many pixels can disappear if the render progresses, so this feature removes the bright pixels at a much lower sample per pixel.

Minimum Display Samples - The minimum amount of samples that OctaneRender® calculates before displaying the image. This feature reduces the noise when navigating, and it's useful for real-time walkthroughs. When using multiple GPUs, we recommend setting this value as a multiple of the number of available GPUs for rendering - if you're rendering with four GPUs, set this value to **4** or **8**.

White Balance - Specify the color to adjust the tint for producing and simulating the relative temperature cast throughout the image by different light sources. The white point is white by default.

Dithering - Adds random noise, which removes banding in very clean images.

Pre-Multiplied Alpha - Multiplies any transparency value of the output pixel by the pixel color.

Disable Partial Alpha - Makes semi-transparent pixels opaque (where the **Alpha** is greater than **0**).

Neutral Response - The camera response curve doesn't tint the render result.

---

[1] Refers to applying a curve to an image to reduce dynamic range

[2] The function or attribute used to code or decode luminance for common displays. The computer graphics industry has set a standard gamma setting of 2.2 making it the most common default for 3D modelling and rendering applications.

Response - Select the use of measured camera response curves, which provides various pre-defined color grades to a rendering.

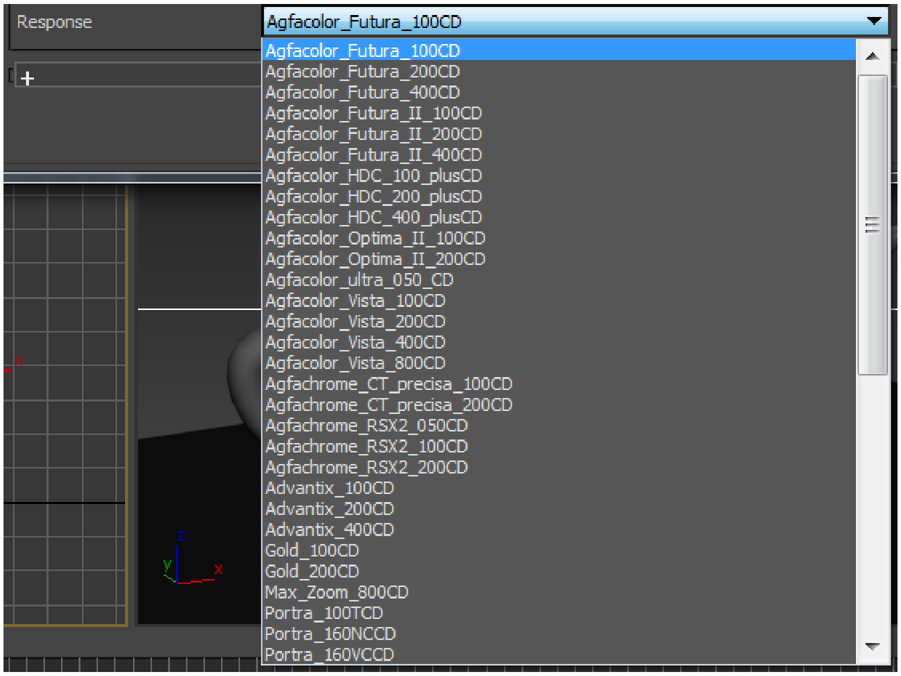


Figure 2: Listing the various types of response curves available in OctaneRender®

Order - Determines the order that OctaneRender®applies the Response, Gamma, and **Custom LUT** (Custom Look Up Table) to the scene. 3D LUTs are defined for sRGB input values - you should apply the custom LUT last, but if there are 3D LUTs for linear input data, then apply the Custom LUT first.

Custom LUT - Choose a custom LUT.

LUT Strength - Controls the influence of the LUT.

OCIO View - OCIO View to use when displaying in the **Render Viewport**. OCIO Config file is set in Octane preferences.

OCIO Look - OCIO Look to apply when displaying in the **Render Viewport**, if using an OCIO view. OCIO Config file is set in Octane preferences.

Force Tone Mapping - Whether to apply Octane's built-in tone mapping (before applying any OCIO look(s)) when using an OCIO view. This may produce undesirable results due to an intermediate reduction to the sRGB color space.

Enable Denoising - Enables the spectral AI denoiser. This applies denoise to some beauty passes, including the main beauty pass, and writes the outputs into separate denoiser render passes.

Denoise Volumes - The spectral AI denoiser denoises **Volumes** in the scene.

Denoise On Completion - The beauty pass denoises once at the end of a render. Disable this option while rendering with the **Interactive Region** tool.

Min. Denoiser Samples - The minimum number of samples taken per pixel before the denoiser begins processing.

Max. Denoiser Interval - The maximum interval between denoiser runs (in seconds).

Blend - Accepts a value between **0** and **1** to blend the original image with the denoised results. A value of **0** results in a total denoised image, and a value of **1** results in an image without any denoising.

Sampling Mode - Selects the upsampler mode for rendering. The image renders at a lower resolution divided by the sampling mode, then it upscales to the final resolution.

Enable AI Up-Sampling - When you have an Upsampler Mode selection made and you enable this option, the render scales by using AI upsampling. Otherwise, scaling is done using traditional methods.

Up-Sampling on Completion - Beauty passes upsample once at the end of a render.

Min/Max U.S. Samples - The minimum and maximum number of samples per pixel until the upsampler activates. This parameter does not apply if you select No Upsampling in Upsampler Mode.

Camera Motion Blur Settings

The **Motion Blur**[1] parameters controls the shutter interval. The value is relative to the frame time. Shown below are the **Motion Blur** parameters under the **Render Setup** window, but similar settings are found under the **Modify** Panel of **Octane Camera** and **Universal Camera**.

---

[1] An optical phenomenon that occurs when a camera’s shutter opens and closes too slowly to capture movement without recording a blurring of the subject.

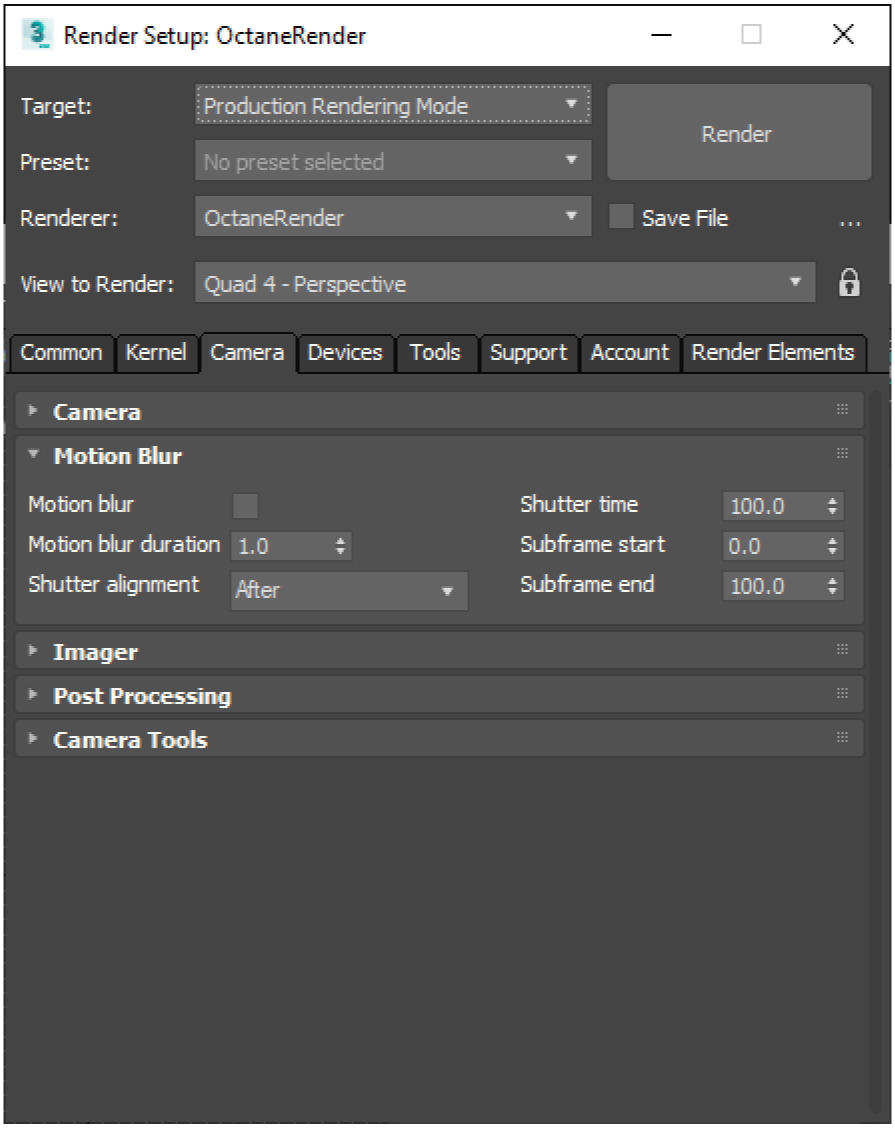


Figure 1: Octane Motion Blur settings in the Render Setup window

Motion Blur Parameters

Override Motion Blur - Replaces the global settings with its own camera motion blur settings. The global settings can be round in Render Setup > Camera Tab > Motion Blur.

Motion Blur - Enables or disables motion blur for the camera.

Motion Blur Duration - Overall control of motion blur settings.

Shutter Alignment - Specifies how the shutter interval aligns to the current time, which determines when the camera shutter is triggered. The options are **Before**, **Symmetrical**, or **After**, and they apply to each frame thereafter relative to the given frame rate.

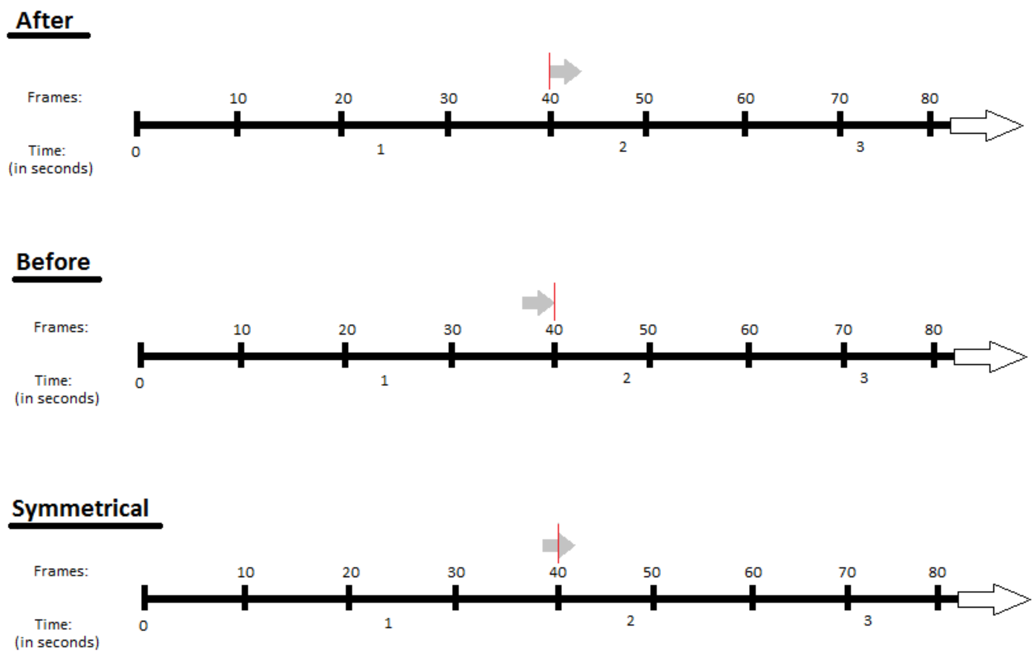


Figure 2: Illustrating the After, Before, and Symmetrical Shutter Alignment

Shutter Time - The shutter time percentage relative to the duration of a single frame, which controls how much time the shutter stays open. You can set this parameter to any value above **100%**.

Subframe Start/Subframe End - Specifies the approach, in terms of proportion (%) to simulate the camera's shutter speed for that particular frame. OctaneRender uses Subframe Start and End percentages to render only a portion of a particular frame. If the scene has a lot of motion blur, OctaneRender® uses these parameters to render a piece of that motion blur. The default Start and End values of **0%** and **100%**, respectively, render the whole frame.

> **Note:** Motion Blur with **Displacement**[1] is currently not supported.

---

[1] The process of utilizing a 2D texture map to generate 3D surface relief. As opposed to bump and normal mapping, Displacement mapping does not only provide the illusion of depth but it effectively displaces the actual geometric position of points over the textured surface.

Camera Post Processing Settings

The **Post-Processing** provides lens effects to help mimic a real camera. Shown below are the **Post-Processing** parameters under the **Render Setup** window, but similar settings are found under the **Modify** Panel of **Octane Camera** and **Universal Camera**.

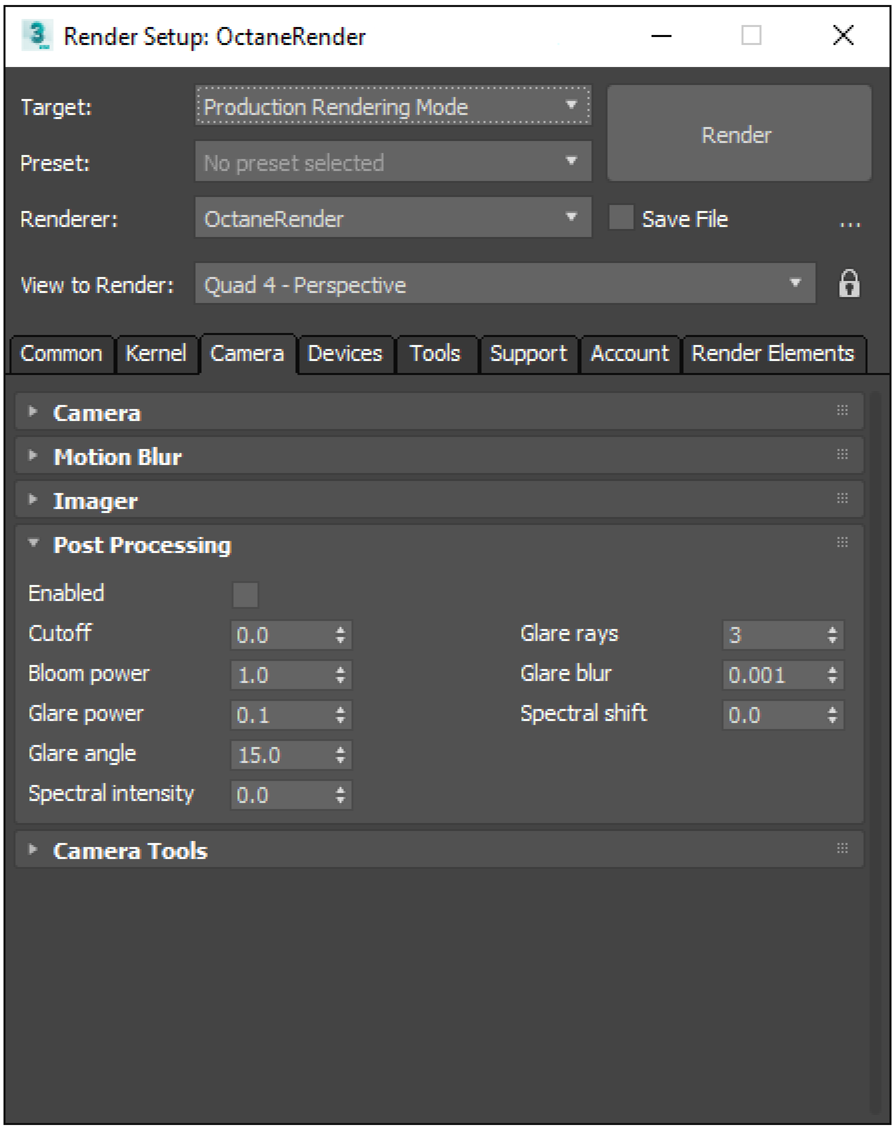


Figure 1: Octane Post Processing[1] settings in the Render Setup window

Post Processing Parameters

Enable - Enables post-processing effects on the render. Post-processing is disabled by default.

Cutoff - Applies bloom/glare to pixel values above the Cutoff value.

Bloom Power - Controls the size and intensity of the halo around the sun, light source, or reflective **Glossy**[2] materials.

Glare Power - Controls the size and intensity of the glare originating from reflective Glossy materials.

Glare Angle - Adjusts the glare rotation relative to the object. A glare angle of **-90** and **90** results to one main horizontal glare, and a glare angle of **0** results to one main vertical glare.

Spectral Intensity - Adjusts the intensity distribution of the rays across a source. This affects the radiant energy's brightness.

Glare Rays - Controls the number of visible rays radiated or reflected.

Glare Blur - Controls the glare sharpness. Smaller values result in a crisp linear glare, and this is softened with higher values.

Spectral Shift - Adjusts the spectrum displacement as the source's emitted light frequency changes. The shift is evident by a color change, similar to the doppler effect.

---

[1] Effects such as Bloom and Glare that are applied after a scene has been rendered.

[2] The measure of how well light is reflected from a surface in the specular direction, the amount and way in which the light is spread around the specular direction, and the change in specular reflection as the specular angle changes. Used for shiny materials such as plastics or metals.

Sample Images With Post-Processing Applied

(Bloom and Glare on a ring)

(Bloom effect)

(Sun glare)

Figure 2: Renders using post-process effects

Camera Tool Settings

Camera Tool can be used to copy and paste settings between Cameras. Shown below are the **Camera Tool** parameters under the **Render Setup** window, but similar settings are found under the **Modify** Panel of **Octane Camera** and **Universal Camera**.

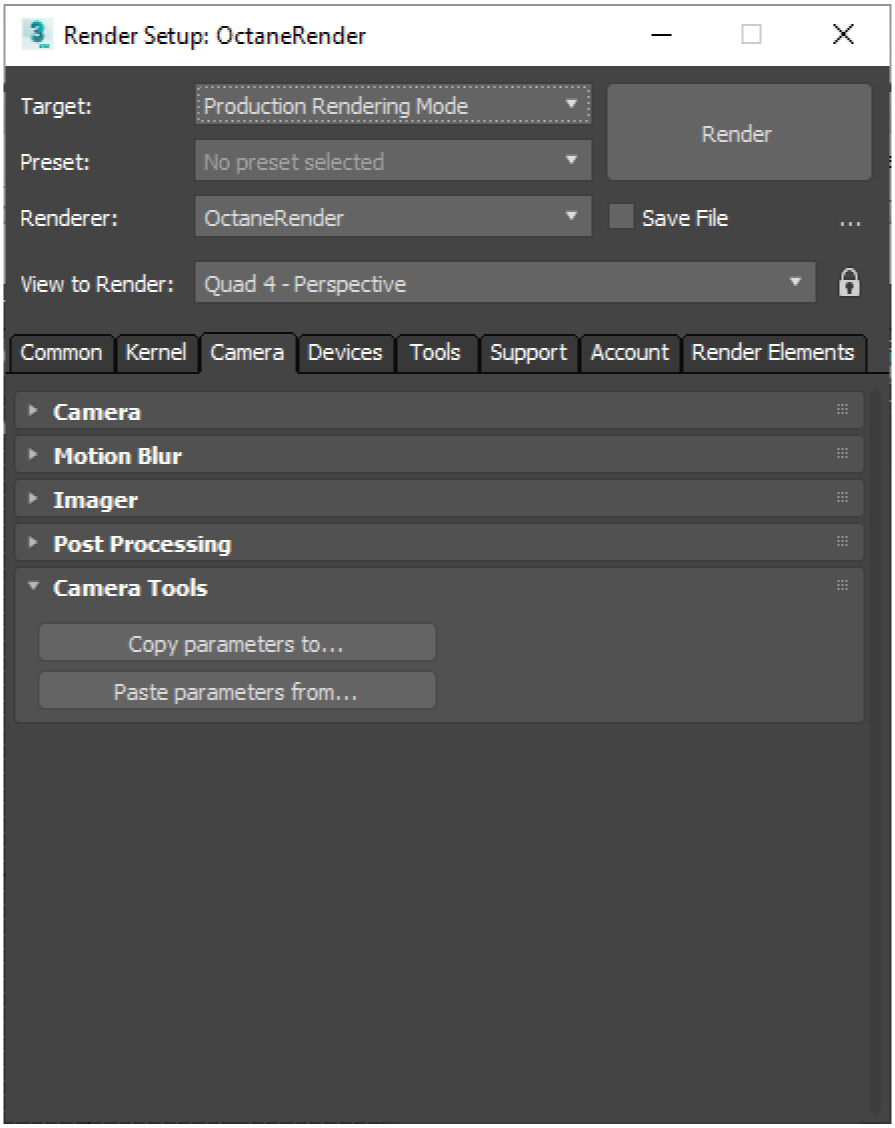


Figure 1: Octane Camera Tools settings in the Render Setup window

Camera Tool Parameters

Copy parameters to... - Copy camera settings to viewport camera or clipboard.

Paste parameters from... - Copy camera settings from viewport camera or clipboard.

Render Setup

Rendering in 3DS Max® for OctaneRender® consists of utilizing the **OctaneRender Viewport** in conjunction with the standard 3DS Max® frame buffer to produce final renders or animations. The typical rendering workflow is to conduct test renderings with the **OctaneRender Viewport** window to develop the look, tune the lighting and shaders, and determine the amount of samples needed to eliminate noise. This sample number is then sent to **Kernel Type - Max. Samples** under the **Kernel tab** in the **Render Setup** window.

The 3DS Max® batch rendering process can then render out stills or a sequence of images using OctaneRender® based on these settings. You can also save still images from the OctaneRender Viewport window, so it's not always necessary to set up a batch render if you need just a single frame.

The OctaneRender Viewport updates changes to **Materials**[1] and **Lights** (parameter adjustments only) without having to reload the scene onto the **GPU**[2]. Most transformations made to **Objects** and **Camera** views in the scene are updated in the OctaneRender Viewport window as the image is tuned. Sometimes you need to refresh the OctaneRender Viewport render to reflect updates made to the scene. Figure 1 shows the Render Setup window with OctaneRender® options.

Note: If you're rendering stills, you can do this with the OctaneRender Viewport without rendering in the 3DS Max® frame buffer.

If transformations (move, scale, rotate) are made to scene geometry including lights, then you need to refresh the Viewport (reload geometry onto the GPU) to see these changes or set the objects as **Movable Proxy**[3].

---

[1]A set of attributes or parameters that describe surface characteristics.

[2]The GPU is responsible for displaying graphical elements on a computer display. The GPU plays a key role in the Octane rendering process as the CUDA cores are utilized during the rendering process.

[3]An object saved as a separate file with the purpose of being reused in larger scenes. This is used to minimize any addition to the total polygon count in the scene, especially if the scene requires the same object to appear several times. If used in conjunction with instancing, Proxies help keep very large scenes from reaching polygon limits and also keeps the relative file size of the main project file manageable.

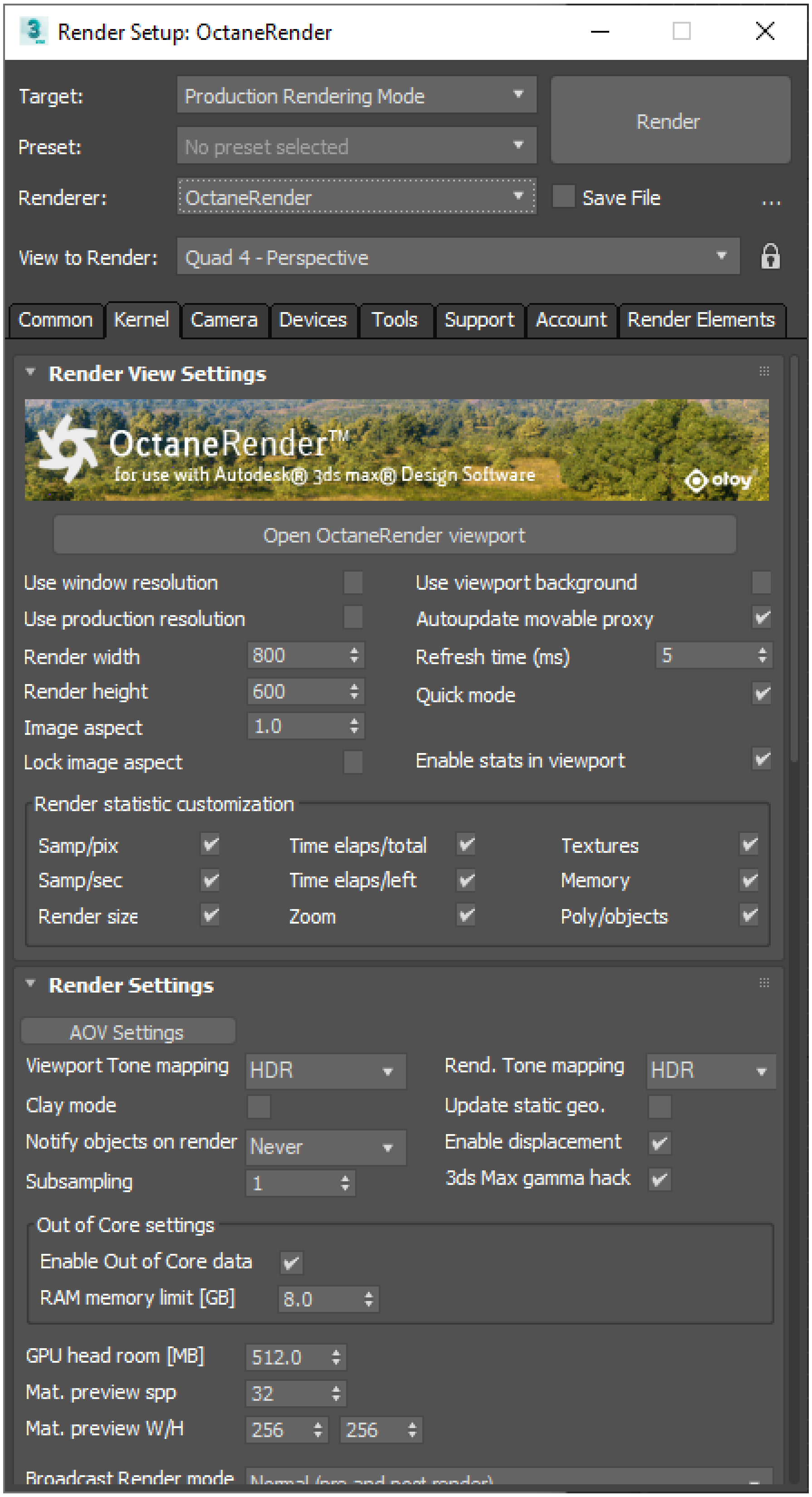

Render Setup: OctaneRender
Target:
Production Rendering Mode
Render
Preset:
No preset selected
Renderer:
OctaneRender
Save File
View to Render:
Quad 4 - Perspective
Common
Kernel
Camera
Devices
Tools
Support
Account
Render Elements
Render View Settings
OctaneRender™
for use with Autodesk® 3ds max® Design Software
otoy
Open OctaneRender viewport
Use window resolution
Use viewport background
Use production resolution
Autoupdate movable proxy
Render width
800
Refresh time (ms)
5
Render height
600
Quick mode
Image aspect
1.0
Lock image aspect
Enable stats in viewport
Render statistic customization
Samp/pix
Time elaps/total
Textures
Samp/sec
Time elaps/left
Memory
Render size
Zoom
Poly/objects
Render Settings
AOV Settings
Viewport Tone mapping
HDR
Rend. Tone mapping
HDR
Clay mode
Update static geo.
Notify objects on render
Never
Enable displacement
Subsampling
1
3ds Max gamma hack
Out of Core settings
Enable Out of Core data
RAM memory limit [GB]
8.0
GPU head room [MB]
512.0
Mat. preview spp
32
Mat. preview W/H
256
256

Figure 1: The Render Setup window

Kernel

The Kernel tab under **Render Setup** provide options to control the optimization and quality of a render. Under this section, you will find the following options:

- **Render View Settings**
- **Render Settings**
- **Kernel Type**
  - Direct Lighting
  - Path Tracing
  - PMC
  - Info Channel

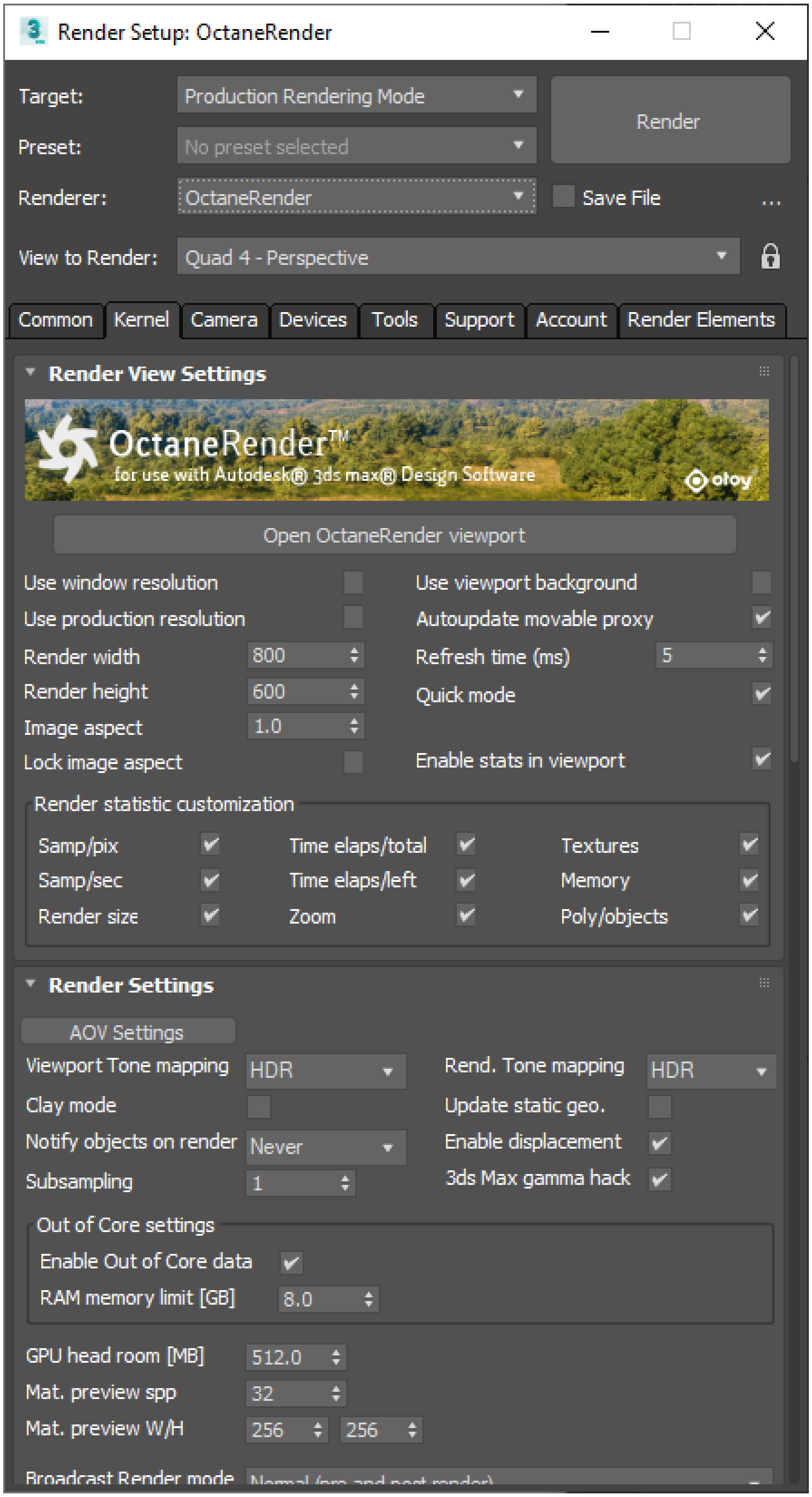

Render Setup: OctaneRender
Target:
Production Rendering Mode
Render
Preset:
No preset selected
Renderer:
OctaneRender
Save File
View to Render:
Quad 4 - Perspective
Common
Kernel
Camera
Devices
Tools
Support
Account
Render Elements
Render View Settings
OctaneRender™
for use with Autodesk® 3ds max® Design Software
otoy
Open OctaneRender viewport
Use window resolution
Use viewport background
Use production resolution
Autoupdate movable proxy
Render width
800
Refresh time (ms)
5
Render height
600
Quick mode
Image aspect
1.0
Lock image aspect
Enable stats in viewport
Render statistic customization
Samp/pix
Time elaps/total
Textures
Samp/sec
Time elaps/left
Memory
Render size
Zoom
Poly/objects
Render Settings
AOV Settings
Viewport Tone mapping
HDR
Rend. Tone mapping
HDR
Clay mode
Update static geo.
Notify objects on render
Never
Enable displacement
Subsampling
1
3ds Max gamma hack
Out of Core settings
Enable Out of Core data
RAM memory limit [GB]
8.0
GPU head room [MB]
512.0
Mat. preview spp
32
Mat. preview W/H
256
256
Broadcast Render mode

Figure 1: Render Setup Kernel Tab from the Render Setup window

Render View Settings

This section provide controls to define the resolution and information display on the OctaneRender Viewport.

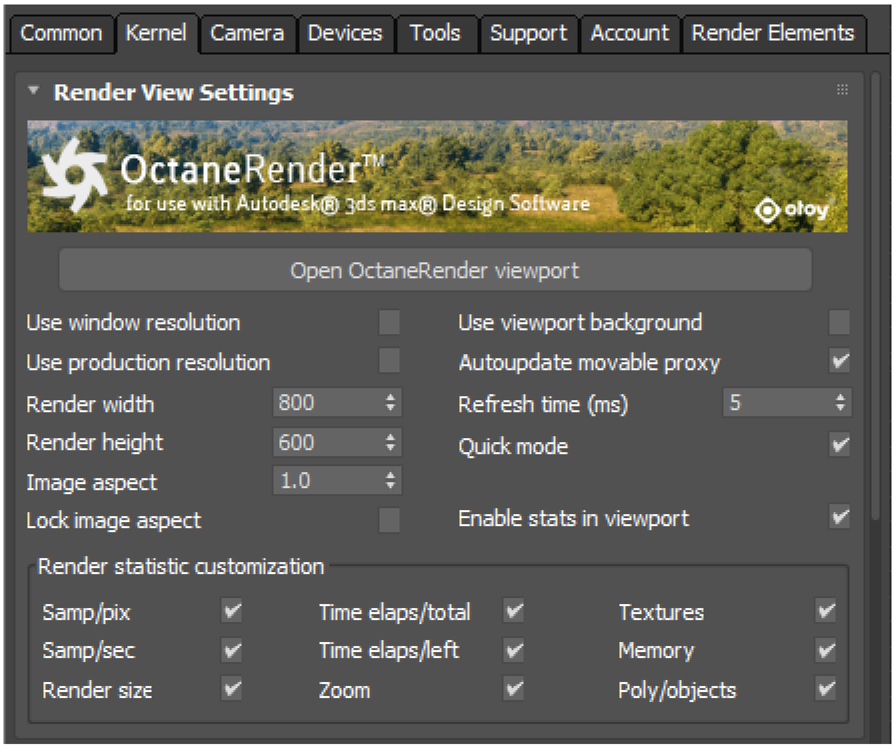


Figure 1: Render View Settings

Render View Setting Parameters

Use Window Resolution - Use the OctaneRender Viewport Window Size.

Use Production Resolution - Use the **Output Size** define from the **Common** tab.

Use Viewport Background - Use the image set as background in 3DS Max viewport. The image will be adjusted to the resolution and will be visible if the **Alpha channel** is enabled in **Kernel Type** settings.

Autoupdate movable proxy - Update mesh of **Movable Proxy**[1] when the geometry changes.

Render Width - The OctaneRender Viewport width when both **Windows** and **Production Resolution** are unchecked

Render Height - The OctaneRender Viewport height when both **Windows** and **Production Resolution** are unchecked

Image Aspect - The OctaneRender Viewport aspect ratio, render height will be automatically adjusted to match the ratio.

Refresh Time (ms) - How often to check for updates. This helps balance between viewport responsiveness and giving more time to render.

Quick Mode - Will speed up interactivity by skipping certain update checks.

Lock Image Aspect - Will adjust **Render Width** or **Height** to maintain image aspect ratio.

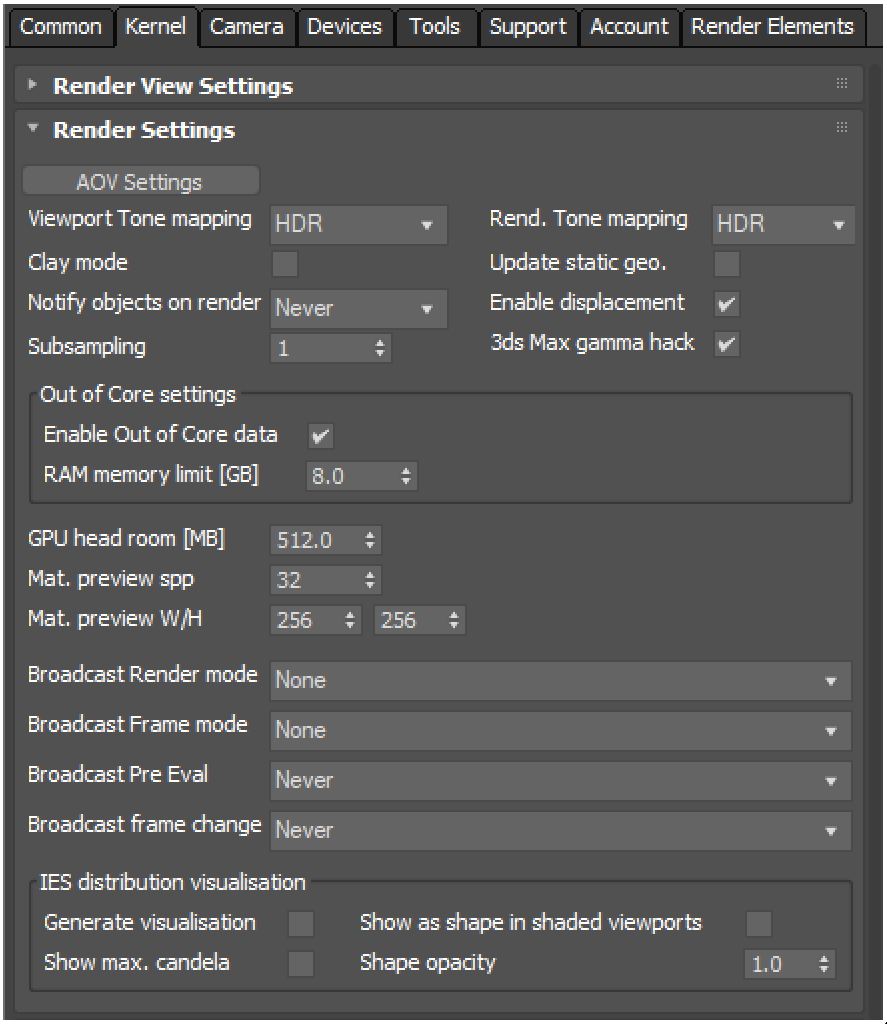
Enable Stats in Viewport - Display render statistics.

Render Statistic Customization - Customize options to display render statistics.

Render Settings

This section will provide additional miscellaneous render settings.

[1] An object saved as a separate file with the purpose of being reused in larger scenes. This is used to minimize any addition to the total polygon count in the scene, especially if the scene requires the same object to appear several times. If used in conjunction with instancing, Proxies help keep very large scenes from reaching polygon limits and also keeps the relative file size of the main project file manageable.



q

Figure 1: Render Settings

Render Setting Parameters

AOV Settings - A layer system used for compositing AOVs.

Viewport Tone Mapping[1] - Defines the color depth to display in the OctaneRender Viewport. Options are **LDR**[2]: 8-bit, HDR: 32-bit, and HDR Half: 16-bit.

Render Tone Mapping - Defines the color depth for final render output (Render Setup Render Button). Options are LDR: 8-bit, HDR: 32-bit, and HDR Half: 16-bit.

Clay Mode - Grey color override for all materials.

Update Static Geo - Update static geometry (non movable proxy) if it's animated. Note that if one static object is animated, all objects must be re-created (reduce performance).

Notify Objects on Render - Setup objects for rendering. Can be needed for some modifiers or plugin objects when 2 mesh can be created, one simplified for the viewport, and one for rendering. Note that this can be very slow if enabled for all objects. When needed, please use 'Per object' instead, enable it in the Octane Object properties, only for those objects.

Enable Displacement[3] - Enable or disable displacement maps in all materials.

Subsampling - Sets the sub-sampling mode. Valid values are 1: None, 2: 2x2, 4: 4x4.

3DS Max Gamma[4] Hack -

Out of Core Settings

- **Enable Out of Core data** - Enable Out of Core Memory. This allows using your RAM memory to store any kind of out of core data, like textures and geometry.
- **RAM Memory Limit (GB)** - Maximum amount of RAM memory to use (in Gigabytes).

GPU[5] Head Room [MB] 512 - Amount of GPU memory to leave free, to allow for the overhead of running the kernels (in Megabytes).

Material[6] Preview SPP - Quality of the preview images in Material Editor (samples per pixel).

Material Preview W/H - Texture size used in 3DS Max Viewports - See also 3DS Max settings in 'Configure Viewports' > 'Display Performance'.

---

[1] Refers to applying a curve to an image to reduce dynamic range

[2] Image formats that have 8 bits per color channel such as the common image formats JPEG, PNG, GIF among others.

[3] The process of utilizing a 2D texture map to generate 3D surface relief. As opposed to bump and normal mapping, Displacement mapping does not only provide the illusion of depth but it effectively displaces the actual geometric position of points over the textured surface.

[4] The function or attribute used to code or decode luminance for common displays. The computer graphics industry has set a standard gamma setting of 2.2 making it the most common default for 3D modelling and rendering applications.

[5] The GPU is responsible for displaying graphical elements on a computer display. The GPU plays a key role in the Octane rendering process as the CUDA cores are utilized during the rendering process.

[6] The representation of the surface or volume properties of an object.

Broadcast Render Mode - Notifications to other plugins when rendering starts and ends, which may be needed to update geometry.

Broadcast Frame Mode - Notifications to other plugins when rendering a frame starts and ends, which may be needed to update geometry.

Broadcast Pre Eval - Notifications to other plugins on scene Pre-evaluation, which may be needed to update geometry.

Broadcast Frame Change - Notifications to other plugins on frame change, which may be needed to update geometry.

IES[1] Distribution Visualisation

- **Generate Visualisation** - Display distribution of IES lights in 3DS Max viewport.
- **Show as Shape in Shaded Viewports** - Display distribution as shape or wireframe.
- **Show Max Candela** - Display the maximum candela value from the IES file.
- **Shape Opacity** - Opacity amount, when displaying distribution as a shape.

Kernel Type

A **Kernel** in OctaneRender® is the central part of the rendering engine that interfaces with the rendering hardware (graphics card or **GPU**[2]). There are four major rendering **Kernels**[3] in OctaneRender®:

- **Direct Lighting**
- **Path Tracing**
- **PMC**
- **Info Channel**

---

[1] An IES light is the lighting information representing the real-world lighting values for specific light fixtures. For more information, visit http://www.ies.org/lighting/.

[2] The GPU is responsible for displaying graphical elements on a computer display. The GPU plays a key role in the Octane rendering process as the CUDA cores are utilized during the rendering process.

[3] By definition, this is the central or most important part of something. In Octane, the Kernels are the heart of the render engine.

Each Kernel has a set of parameters to adjust.

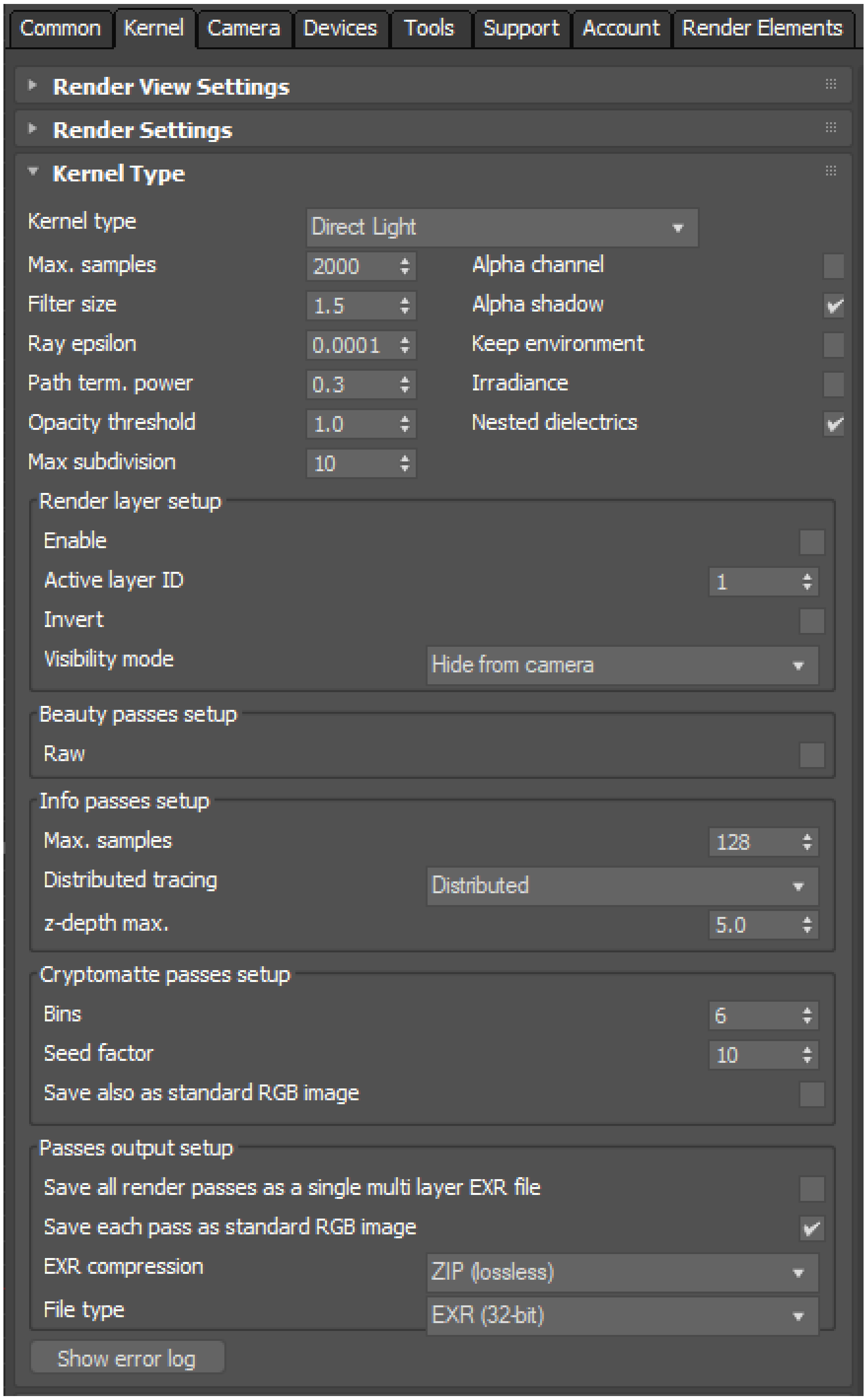


Figure 1: Kernel Type

Kernel Type Parameters

Kernel Type - Select between 4 Kernel Types: Direct Light, Path Tracing, PMC, and Info Channel.

Maximum Samples - This sets the maximum number of samples per pixel before the rendering process stops. Higher values create cleaner renders. There is no rule as to how many samples per pixel create a good render, it is dependent on the scene.

Filter Size - This sets the render filter's pixel size. This improves aliasing artifacts in the render. If the value is too high, the image becomes blurry.

Ray Epsilon - The distance to offset new rays so they don’t intersect with the originating geometry. We recommend leaving this value at the default setting.

Path Termination Power - High values increase render speed, but increase noise in the dark areas.

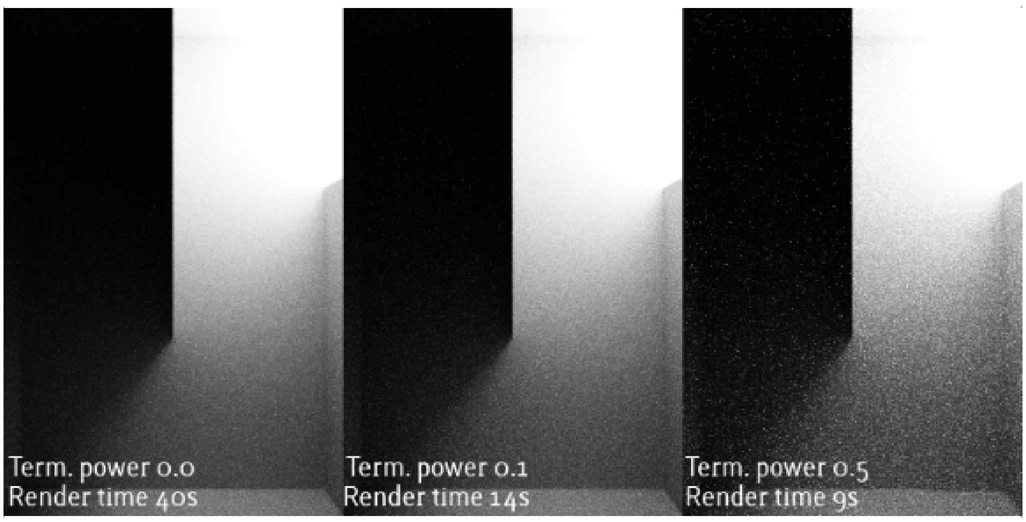


Figure 2: A comparison of Path Termination powers

Opacity Threshold - Geometry with an **Opacity** value that's greater than or equal to this value is treated as totally opaque.

Max Subdivision - The maximum subdivision level applied to geometry in the scene. A value of **0** disables this parameter.

Alpha Channel[1] - This option removes the background and renders it as transparent (zero alpha). This is useful if you want to composite the render over another image and don't want the background to be present.

[1]A greyscale image used to determine which areas of a texture map are opaque and which areas are transparent.

Alpha Shadow - This setting allows any **Object** with transparency (**Specular**[1] materials, **Materials**[2] with **Opacity** settings and **Alpha Channels**) to cast a shadow instead of behaving as a solid object.

Keep Environment - Use this option in conjunction with the Alpha Channel setting. The background renders with zero alpha, but is still visible in the final render. This allows more flexibility in compositing images.

Irradiance - Displays the rendering with light energy, but no **Material**[3] characteristics.

Nested Dielectrics - Enables nested dielectrics. If disabled, the surface IORs and priorities are ignored.

Render Layer Setup

- **Enable** - On and off toggle.
- **Active Layer ID** - Isolate all passes to render only the object(s) with a specific **Object Layer ID**. This is assigned from the **Octane Object Properties** window.
- **Invert** - Isolate all objects that does not have the assigned **Active Layer ID**.
- **Visibility Mode**
    - **Normal** - The beauty passes contain the active layer only and the render layer passes (shadows, reflections...) record the side-effects of the active render layer for those samples/pixels that are not obstructed by the active render layer. Beauty passes will be transparent for those pixels which are covered by objects on the inactive layers, even if there is an object on the active layer behind the foreground object.
    - **Hide inactive layers** - All geometry that is not on an active layer will be made invisible. No side effects will be recorded in the render layer passes, i.e. the render layer passes will be empty.
    - **Only side effects** - Similar to **Normal**, with the exception that the active layer will be made invisible to the camera, i.e. the beauty passes will be empty. The render layer passes (shadows, reflections...) still record the side-effects of the active render layer. This is useful to capture all side-effects without having the active layer obstructing those.
    - **Hide from camera** - Similar to **Hide inactive layers**, all geometry that is not on an active layer will be made invisible. Side effects(shadows, reflections...) will be recorded in the render layer passes.

Beauty Passes Setup

- **Raw** - Make the beauty pass raw render passes by dividing out the color of the BxDF of the surface hit by the camera ray.

Info Passes Setup

---

[1] Amount of specular reflection, or the mirror-like reflection of light photons at the same angle. Used for transparent materials such as glass and water.

[2] A set of attributes or parameters that describe surface characteristics.

[3] The representation of the surface or volume properties of an object.

- **Max. Samples** - The maximum number of samples for the info passes.
- **Distributed Tracing**
  - **Distributed rays** - Enables motion blur and **DOF**[1], and also enables pixel filtering.
  - **Nan-distributed with pixel filtering** - Disables motion blur and DOF, but leaves pixel filtering enabled.
  - **Nan-distributed without pixel filtering** - Disables motion blur and DOF, and disables pixel filtering for all render passes except for render layer mask and ambient occlusion.
- **Z-Depth**[2] **Max.** - The maximum Z-depth value. Background pixels will get this value and and any foreground depths will be clamped at this value. This applies With or without tone mapping, but tone mapping will map the maximum Z-depth to white (0 is mapped to black).

Cryptomatte Passes Setup

- **Bins** - Amount of cryptomatte bins to render.
- **Seed Factor** - Amount of samples used for seeding cryptomatte. This gets multiplied with the amount of bins. Low values result in pitting artifacts at feathered edges, while large values can result in artifacts in places with coverage for lots of different IDs.
- **Save also as standard RGB image** - Enable saving cryptomatte passes as standard rgb image. They are always saved as **EXR**[3] file, so the rgb output may not be needed. Note that simplified colored images are produced when using **LDR**[4] tone mapping in the Render settings.

Passes Output Setup

- **Save All Render Passes**[5] **as a Single Multi-Layer EXR File** - Enable Octane to handle outputting multi-layer exr files.
- **Save Each Pass as Standard RGB Image** - Enable saving all passes as Standard RGB image information.
- **EXR Compression** - Select the compression algorithm used for EXRs.
- **File Type** - Select between 16 or 32-bit file types.

Show Error Log - Error log window.

---

[1] The distance between the nearest and farthest objects in a scene that appear acceptably sharp in an image. Although a lens can precisely focus at only one distance at a time, the decrease in sharpness is gradual on each side of the focused distance, so that within the DOF, the unsharpness is imperceptible under normal viewing conditions. source: wikipedia (https://en.wikipedia.org/wiki/Depth_of_field)

[2] A measure of object distances from the camera typically represented as a grayscale image.

[3] Also known as OpenEXR. This image file format was developed by Industrial Light & Magic and provides a High Dynamic Range image capable of storing deep image data on a frame-by-frame basis.

[4] Image formats that have 8 bits per color channel such as the common image formats JPEG, PNG, GIF among others.

[5] Render passes allow a rendered frame to be further broken down beyond the capabilities of Render Layers. Render Passes vary among render engines but typically they allow an image to be separated into its fundamental visual components such as diffuse, ambient, specular, etc..

Direct Lighting

The **Direct Light** kernel renders previews faster. Direct lighting is not unbiased, and is not meant for photorealism. This **Kernel** is good for creating quick animations or renders.

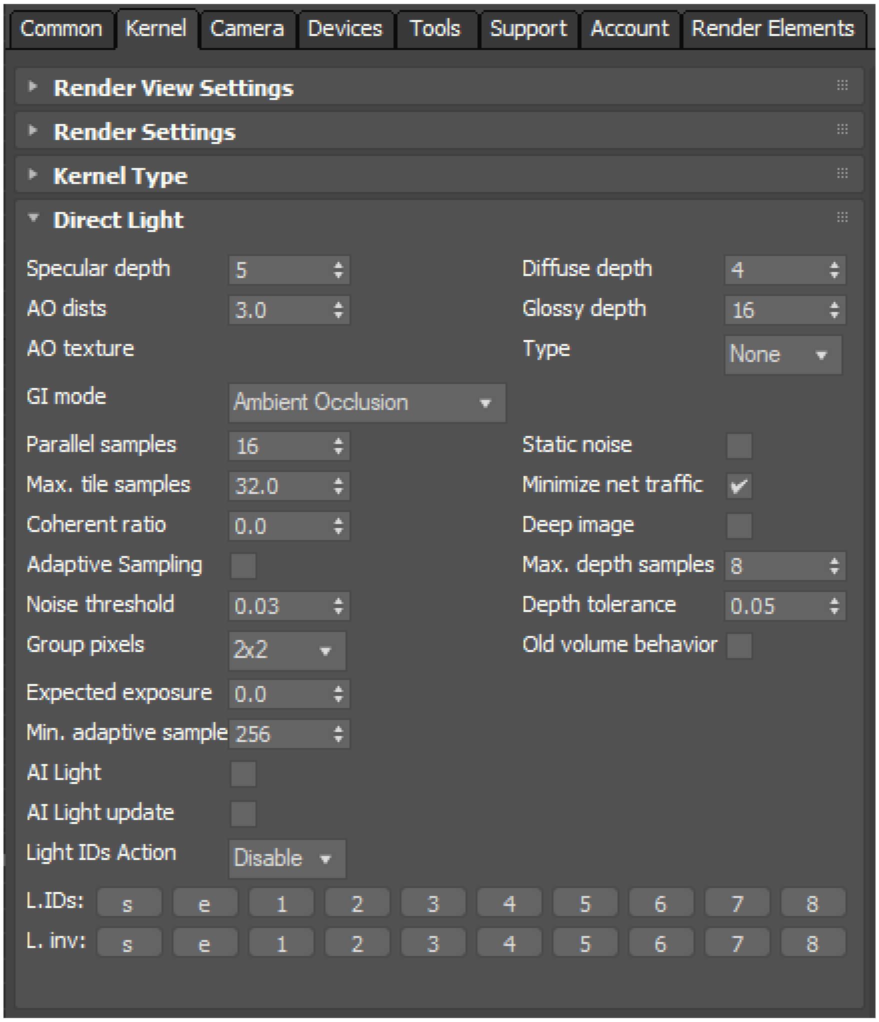


Figure 1: The Direct Light kernel parameters

Direct Light Kernel Parameters

Specular[1] **Depth** - Controls the number of times a ray refracts before dying. Higher values mean higher render times, but more color bleeding and more details in transparent **Materials**[2]. Low numbers introduce artifacts or turn some refractions into pure black.

AO Distance - The ambient occlusion distance in units. Always check if the amount is correct relative to the scene scale. For example, you don't need **3** units if your object is a small toy. However, if your model is a house or something large, then increase the value.

AO Texture - Specifies an ambient occlusion color or value. If you select **None**, OctaneRender® uses the **Environment** instead.

GI Mode - There are three different Global Illumination Modes in the Direct Light kernel:

- **None** - Includes just the direct lighting from area lights. Areas with indirect lighting get no contribution and become black.
- **Ambient Occlusion** - Standard ambient occlusion.
- **Diffuse**[3] - An indirect diffusion with a configuration to set the number of indirect diffuse bounces. This gives a GI quality that is in-between ambient occlusion and path tracing, without caustics and with a decent, realistic quality that's better than AO, but much faster than path tracing/PMC. It's very good for quick final renders and animations. It's similar to brute force indirect GI in other engines.

Parallel Samples - Controls how many samples OctaneRender® calculates in parallel. Low values require less memory to store the sample's state, but rendering is slower. High values need more graphics memory, making rendering faster. The change in performance depends on the scene and the **GPU**[4] architecture.

Maximum Tile Samples - Controls the number of samples per pixel that OctaneRender® will render until it takes the result and stores it in the film buffer. Higher values mean results arrive less often in the film buffer.

Coherent Ratio - Increasing this value increases the render speed, but introduces low-frequency noise (blotches), which may require a few hundred or even a few thousand samples per pixel to remove, depending on the scene.

---

[1] Amount of specular reflection, or the mirror-like reflection of light photons at the same angle. Used for transparent materials such as glass and water.

[2] A set of attributes or parameters that describe surface characteristics.

[3] Amount of diffusion, or the reflection of light photons at different angles from an uneven or granular surface. Used for dull, non-reflecting materials or mesh emitters.

[4] The GPU is responsible for displaying graphical elements on a computer display. The GPU plays a key role in the Octane rendering process as the CUDA cores are utilized during the rendering process.

Adaptive Sampling[1] - Stops sampling pixels that reach a specified noise threshold. This lets the kernel focus its processing on areas that still need refinement.

Noise Threshold - When Adaptive Sampling is enabled, Noise Threshold specifies the smallest relative noise level. When a pixel's noise estimate is less than this value, OctaneRender® turns off sampling for this pixel. Good values are in the range of **0.01** - **0.03**. The default is **0.02**, which is pretty clean.

Group Pixels - When Adaptive Sampling is enabled, Group Pixels specifies the number of pixels that are handled together. If all pixels of a group reach the noise level, sampling stops for all of these pixels.

Expected Exposure - This parameter's value should be about the same value as the image exposure, otherwise set it to **0** (the default value) to ignore these settings. Adaptive Sampling uses this parameter to determine what pixels are bright and dark. This depends on the exposure setting in the Octane Imager. If the value is not **0**, Adaptive Sampling adjusts the noise estimate of very dark areas of the image. It also increases the **Min. Adaptive Samples** limit for very dark areas, because they tend to find paths to light sources irregularly, resulting in an over-optimistic noise estimate.

Min. Adaptive Samples - Specifies the minimum number of samples to calculate before Adaptive Sampling kicks in. The pixel's noise estimate has a large initial error. If you set the noise threshold to a high value, then you should set Min. Adaptive Samples to a high value to avoid artifacts.

Diffuse Depth - Gives the maximum number of diffuse reflections if GI Mode is set to **Diffuse**.

Glossy[2] **Depth** - Controls the number of times a ray reflects before dying. Higher values mean higher render times. Values lower than **4** can introduce artifacts or turn some reflections into pure black.

Type - Specifies the Ambient Occlusion environment texture type for use with the AO rays. If set to **None**, the **Environment** is used instead.

Static Noise - Makes noise static - it doesn't change between frames.

Minimize Net Traffic - OctaneRender® distributes the same tile to the net render nodes until it reaches the max samples/pixel for that tile, and then it distributes the next tile to render nodes. Work done by local GPUs is not affected by this option. A render node can merge all of its results into the same cached tile until the Primary Render Node switches to a different tile.

Deep Image[3] - Enables rendering deep pixel images used for deep image compositing.

Max. Depth Samples - Used when Deep Image is enabled. This sets the maximum number of depth samples per pixel. For more information, see the Deep Image Rendering topic in this manual.

---

[1] A method of sampling that determines if areas of a rendering require more sampling than other areas instead of sampling the entire rendering equally.

[2] The measure of how well light is reflected from a surface in the specular direction, the amount and way in which the light is spread around the specular direction, and the change in specular reflection as the specular angle changes. Used for shiny materials such as plastics or metals.

[3] Renders frames with multiple depth samples in addition to typical color and opacity channels.

Depth Tolerance - Used when deep image rendering is enabled. OctaneRender® merges the depth samples whose relative depth difference falls below this tolerance value together. This is covered in the Deep Image Rendering section in this manual.

Old Volume Behavior - Emulates the emission and scattering behavior from previous versions of OctaneRender®.

AI Light - Enables AI lights. AI light functionality learns from the scene, and rendering becomes more efficient as more samples are rendered. When used with **Adaptive Sampling**, AI Light becomes even more effective as it learns pixel and light importance in a scene, and some pixels are no longer sampled.

AI Light Update - Enables dynamic updates to the AI lighting.

Light IDs Action - This parameter determines whether the **L.IDs** (Light IDs) and **L. Inv** (Light Inverse) buttons enable or disable lights with matching **Light Pass ID** numbers.

Path Tracing

The **Path Tracing** kernel is best for realistic renderings, although the render times are slower than the **Direct Lighting** kernel. Path Tracing has difficulties with small light sources and proper caustics, for which **PMC** is a better option.

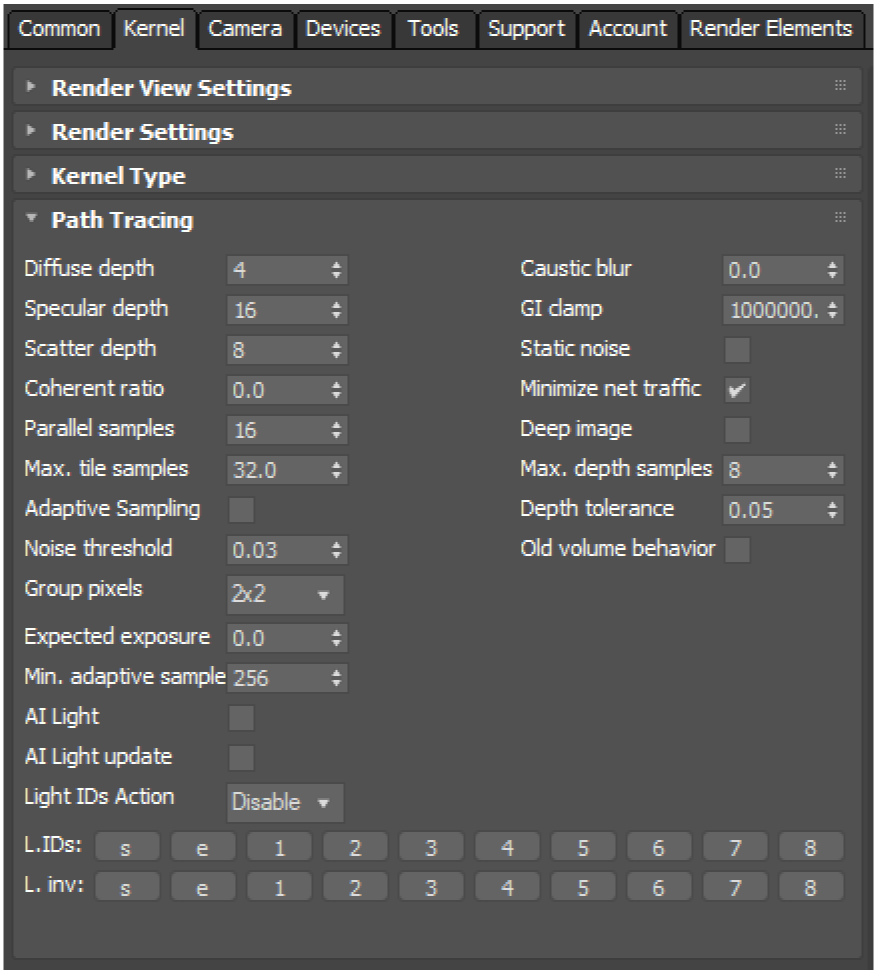


Figure 1: Path Tracing kernel parameters

Path Tracing Parameters

Diffuse[1] **Depth** - The maximum number of times a ray bounces, reflects, or refracts on a surface with high **Roughness** or **Diffuse** settings. Higher values mean slower render speeds, but more realistic results. For outdoor scenes, set this parameter to around **4.** For lighting interiors with natural light, use settings of **8** or higher. While high values are possible, in reality, rays won't go beyond 16 ray bounces.

---

[1] Amount of diffusion, or the reflection of light photons at different angles from an uneven or granular surface. Used for dull, non-reflecting materials or mesh emitters.

Specular[1] **Depth** - Determines the maximum path depth for specular reflections and refractions.

Scatter Depth - The maximum path depth to allow scattering.

Coherent Ratio - Increasing this value increases the render speed, but it introduces low-frequency noise (blotches), which require a few hundred or even a few thousand samples per pixel to remove, depending on the scene.

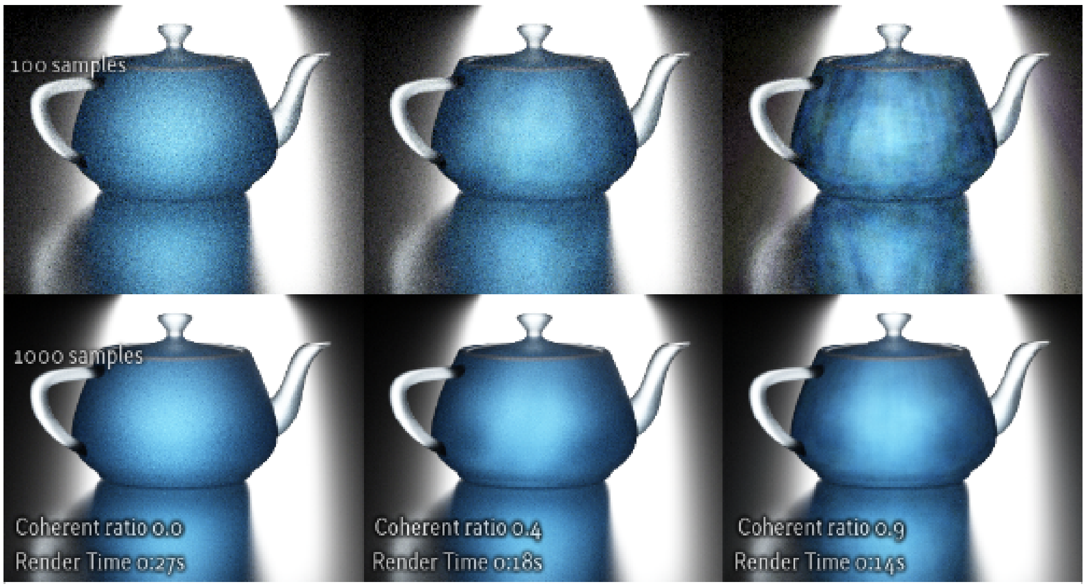


Figure 2: A comparison of different Coherent Ration settings

Parallel Samples - Controls how many samples OctaneRender® calculates in parallel. Small values require less memory to store the sample's state, but renders are slower. High values use more graphics memory, and rendering is faster. The change in performance depends on the scene and the **GPU**[2] architecture.

Maximum Tile Samples - Controls the number of samples per pixel that OctaneRender® will render until it takes the result and stores it in the film buffer. Higher values generate results less often in the film buffer.

---

[1] Amount of specular reflection, or the mirror-like reflection of light photons at the same angle. Used for transparent materials such as glass and water.

[2] The GPU is responsible for displaying graphical elements on a computer display. The GPU plays a key role in the Octane rendering process as the CUDA cores are utilized during the rendering process.

Adaptive Sampling[1] - Stops sampling pixels that reach a specified noise threshold, which allows the kernel to focus its processing on areas that still need refinement.

Noise Threshold - When Adaptive Sampling is enabled, Noise Threshold specifies the smallest relative noise level. When the noise estimate of a pixel becomes less than this value, OctaneRender® disables sampling for this pixel. Good values are in the range of **0.01** - **0.03**. The default is **0.02**, which is pretty clean.

Group Pixels - When Adaptive Sampling is enabled, Group Pixels specifies the number of pixels that are handled together. When all of a group's pixels reach the noise level, sampling stops for all of these pixels.

Expected Exposure - This value should match the image exposure, or set it to **0** (the default value) to ignore these settings. Adaptive Sampling uses this parameter to determine the bright and dark pixels, which depends on the exposure setting in the **Octane Imager**. If the value is not **0**, Adaptive Sampling adjusts the noise estimate of very dark areas of the image. It also increases the **Min. Adaptive Samples** limit for very dark areas since they find paths to light sources irregularly, resulting in over-optimistic noise estimates.

Min. Adaptive Samples - Specifies the minimum amount of samples to calculate before Adaptive Sampling kicks in. A pixel's noise estimate is just an estimate with a large initial error. If you set Noise Threshold to a high value, then set this parameter to a high value as well to avoid artifacts.

Caustic Blur - Higher values result in less caustic noise.

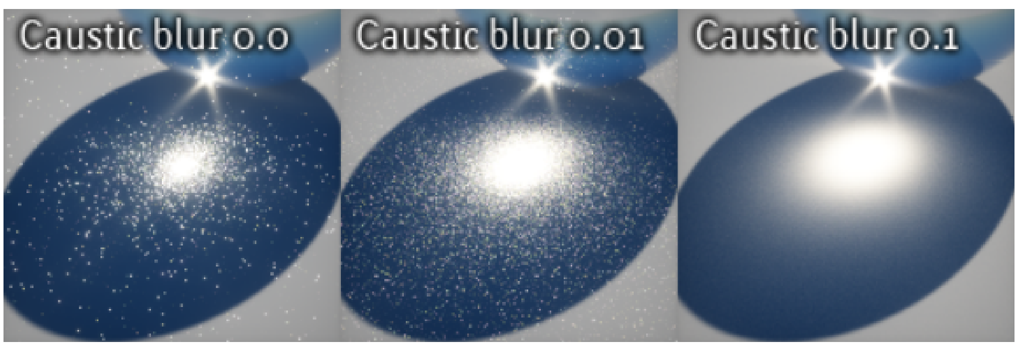


Figure 3: A comparison of various Caustic Blur values

GI Clamp - Clamps each path's contribution to the specified value. Reducing this value can reduce the amount of fireflies caused by sparse but very strong contributing paths. Reducing this value also reduces noise by removing energy.

---

[1] A method of sampling that determines if areas of a rendering require more sampling than other areas instead of sampling the entire rendering equally.

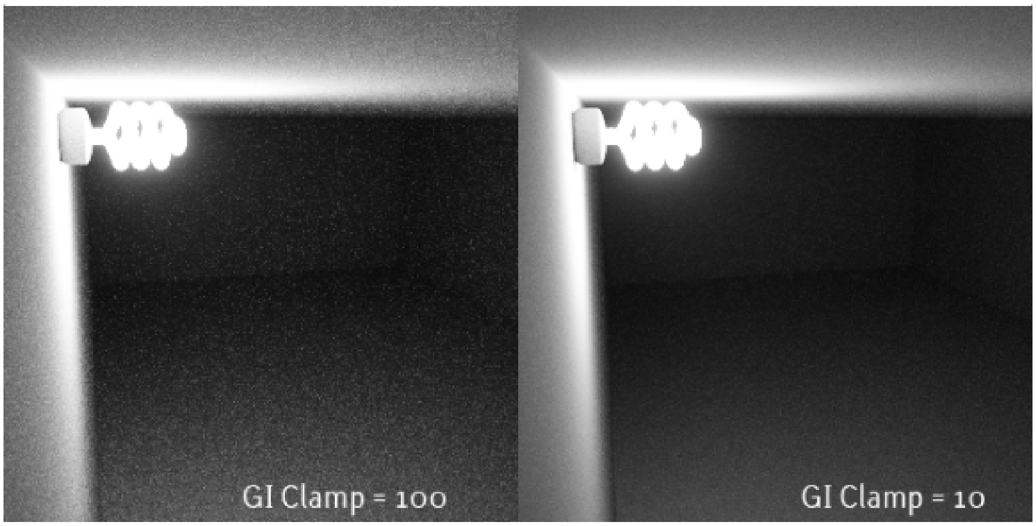


Figure 3: A comparison of various GI Clamp values

Static Noise - Makes the noise static - it doesn't change between frames.

Minimize Net Traffic - Makes OctaneRender® distribute the same tile to the net render nodes until the tile reaches the max number of depth samples per pixel, and then OctaneRender® distributes the next tile to the render nodes. This option doesn't affect work done by local GPUs. This way, a render node can merge all of its results into the same cached tile until the Primary Render Node switches to a different tile.

Deep Image[1] - Enables rendering deep pixel images used for deep image compositing.

Maximum Depth Samples - When you enable Deep Image, this sets the maximum number of depth samples per pixel.

Depth Tolerance - When you enable Deep Image rendering, this option merges depth samples with relative depth differences that fall below this tolerance value.

Old Volume Behavior - Emulates the behavior of emission and scattering from previous versions of OctaneRender®.

AI Light - Enables AI lights. AI light functionality learns from the scene, and rendering becomes more efficient as you render more samples. When used with Adaptive Sampling, AI Light becomes even more effective, as it learns pixel and light importance in a scene, and some pixels are no longer sampled.

AI Light Update - Enables dynamic updates to the AI lighting.

---

[1]Renders frames with multiple depth samples in addition to typical color and opacity channels.

Light IDs Action - Determines whether the **L.IDs** (Light IDs) and **L. Inv** (Light Inverse) buttons enable or disable lights with matching **Light Pass ID** numbers.

PMC Kernel

The **PMC** kernel is a custom mutating unbiased kernel written for GPUs. It lets OctaneRender® resolve complex caustics and lighting.

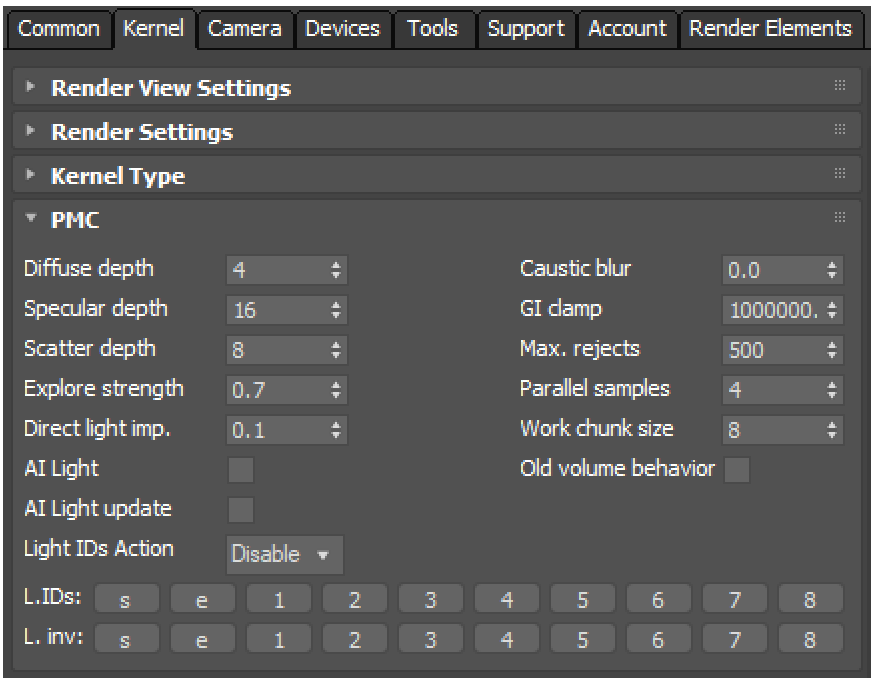


Figure 1: The PMC kernel parameters

PMC Kernel Parameters

Diffuse[1] **Depth** - The maximum number of times a ray can bounce, reflect, or refract on a high roughness/diffuse surface. Higher values mean higher render times, but more realistic results. For outdoor scenes, a

[1] Amount of diffusion, or the reflection of light photons at different angles from an uneven or granular surface. Used for dull, non-reflecting materials or mesh emitters.

good value is around **4**. For lighting interiors with natural light, you need higher values such as **8** or more. While high values are possible, in reality, rays won't go beyond 16 ray bounces.

Specular[1] Depth - Controls the number of times a ray refracts before dying. Higher values mean higher render times, but more color bleeding and more details in transparent materials. Low values introduce artifacts or turn some refractions into pure black.

Scatter Depth - The maximum path depth that allows scattering.

Exploration Strength - Specifies how long the kernel investigates good paths before it tries to find a new path. Low values create a noisy image, while high values create a splotchy image.

Direct Light Importance - Makes the kernel focus more on paths with indirect light. For example, imagine sunlight through a window, which creates a bright spot on the floor. If Direct Light Importance is set to **1**, the kernel focuses its sampling on this area. If Direct Light Importance is reduced, the kernel reduces its efforts to sample that area, and focuses more on more tricky areas that are harder for light rays to reach.

Caustic Blur - Increasing this parameter's value results in less caustic noise.

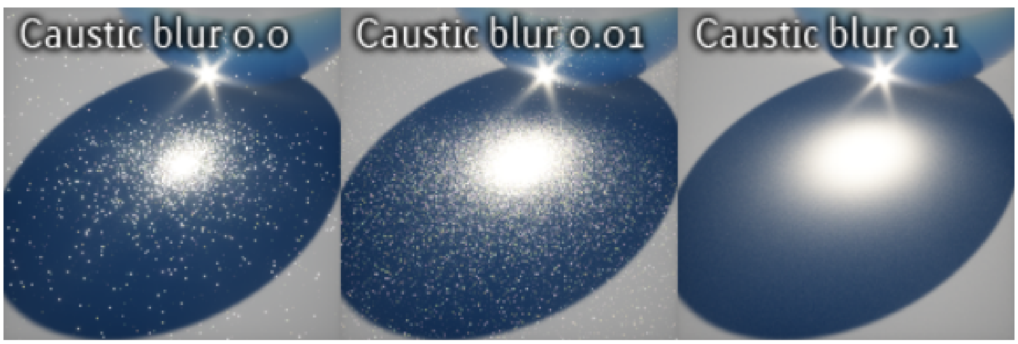


Figure 2: A comparison of various Caustic Blur values

GI Clamp - This clamps each path's contribution to the specified value. Reducing this value reduces the amount of fireflies caused by sparse but strong contributing paths. Reducing this value also reduces noise by removing energy.

[1] Amount of specular reflection, or the mirror-like reflection of light photons at the same angle. Used for transparent materials such as glass and water.

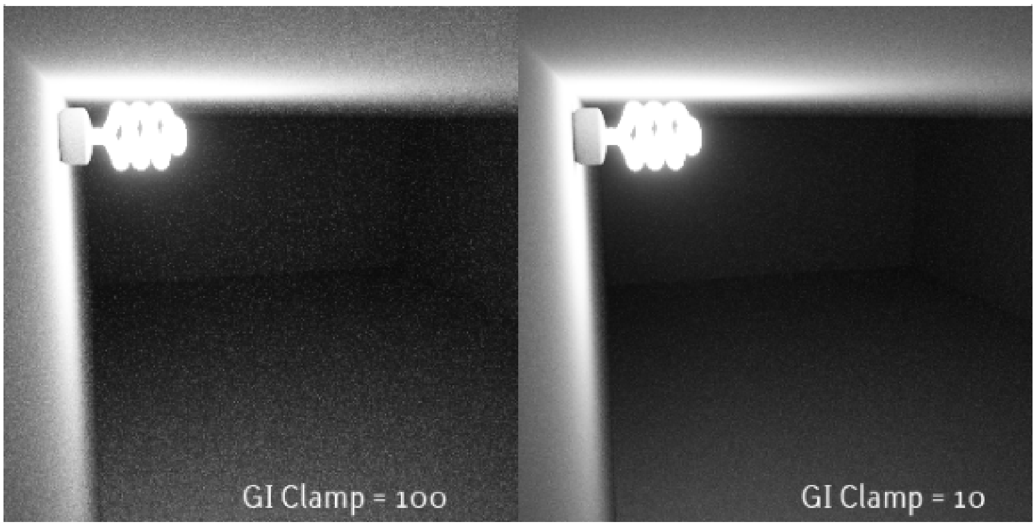


Figure 3: A comparison of various GI Clamp values

Max Rejects - Controls the render's bias. Reducing the value creates more biased results along with shorter render times.

Parallel Samples - Controls how many samples OctaneRender® calculates in parallel. Small values require less memory to store the sample's state, but renders are slower. High values use more graphics memory, and renders are faster. The change in performance depends on the scene, the **GPU**[1] architecture, and the number of shader processors on the GPU.

Work Chunk Size - The number of work blocks done per kernel run. Increasing this value also increases the memory requirement on the system, but it does not affect memory usage, and may increase render speed.

Old Volume Behavior - Emulates the behavior of emission and scattering from previous versions of OctaneRender®.

AI Light - Enables AI lights. AI light functionality learns from the scene, and rendering becomes more efficient as more samples are rendered. When used with **Adaptive Sampling**[2], AI Light becomes even more effective as it learns pixel and light importance in a scene, and some pixels are no longer sampled.

AI Light Update - Enables dynamic updates to the AI lighting.

[1]The GPU is responsible for displaying graphical elements on a computer display. The GPU plays a key role in the Octane rendering process as the CUDA cores are utilized during the rendering process.
[2]A method of sampling that determines if areas of a rendering require more sampling than other areas instead of sampling the entire rendering equally.

Light IDs Action - Determines whether the **L.IDs** (Light IDs) and **L. Inv** (Light Inverse) buttons enable or disable lights with matching **Light Pass ID** numbers.

Info Channel

The **Info Channel** kernel evaluates scene data and renders the data as color images that you can use in post processes for compositing.

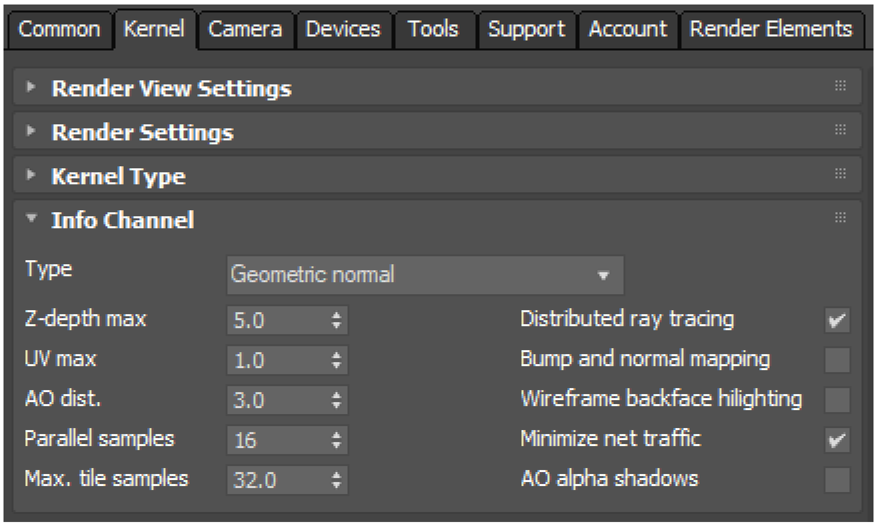


Figure 1: The Info Channel kernel parameters

Info Channel Kernel Parameters

Channel - The following settings are available with some of the most often-used channels defined.

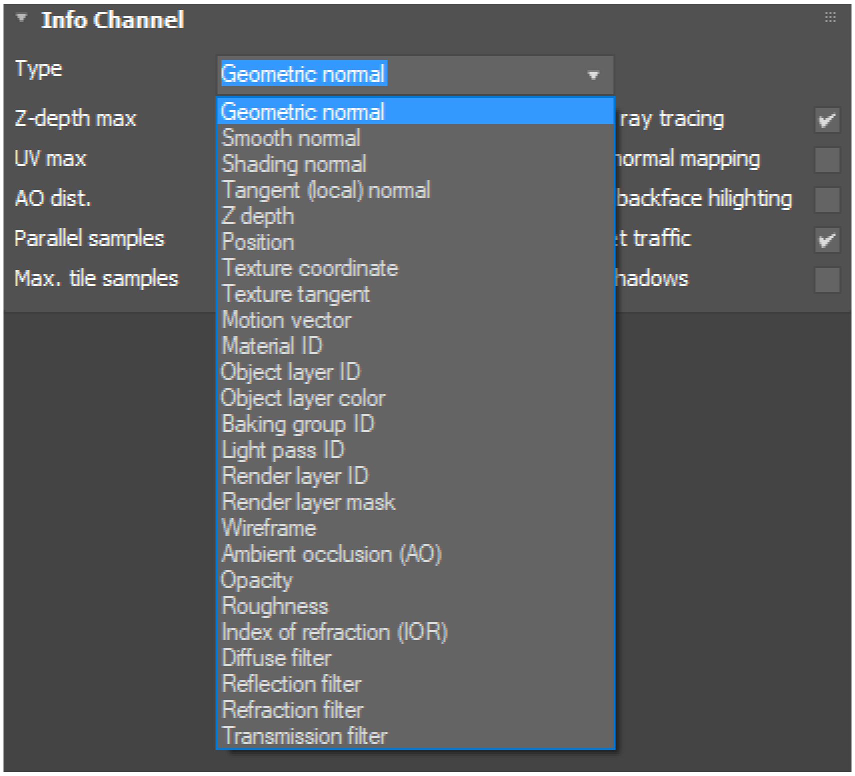


Figure 2: The available Info Channel types

- **Geometric Normals** - The vectors perpendicular to the mesh's triangle faces.
- **Smooth Normals** - Shows information on the integrity of the model's geometry in terms of the normals perpendicular to the mesh's smooth underlying surface.
- **Shading Normals** - The interpolated normals used for shading. This does not take into account the object's Bump map. The result is less faceted and smoother than Geometric Normals.
- **Tangent (Local) Normals** - A color shows the Tangent (Local) Normal in tangent space at the position hit by the camera ray.
- **Z-depth** - An image that's shaded based on the distance between the objects in the scene and the position of the rendering camera.
- **Position** - A color-coded image that shows the position of the objects in the scene, often used in compositing to help position 3D-rendered images from different renders.
- **Texture Coordinates** - A color-coded image showing a Gradient map based on the direction of the object's UV texture coordinates.
- **Texture Tangent** - The first tangent vector. This determines the Normal map distortion's orientation.
- **Motion Vector** - This renders the 2D motion vector in screen space. The X-coordinate shows pixels set in motion to the right (stored in the Red channel), while the Y-coordinate shows pixels in the up motion (stored in the Green channel).

- **Material[1] ID** - Every material assigned in the scene is represented as a separate color.
- **Object Layer ID** - A color-coded image, each object is colored based on their Object Layer ID settings. The Layer ID setting is found in the Octane Attributes section in the object's Shape node tab.
- **Object Layer Color** - Shows the color specified in the Object Layer node.
- **Baking Group ID** - Every Baking Group ID assigned in the scene is represented as a separate color.
- **Light Pass ID** - Every Light Pass ID assigned in the scene is represented as a separate color.
- **Render Layer ID** - A color-coded image, each object is colored based on their Object Layer ID settings. The Layer ID setting is found in the Octane Attributes section in the object's Shape node tab. For more information, see the **Render Layers**[2] topic in this manual.
- **Render Layer Mask** - A mask that's rendered based on an object's Layer ID and render layer membership. For more information, see the Render Layers topic in this manual.
- **Wireframe** - Triangles outlined in black represent the mesh.
- **Ambient Occlusion (AO)** - A render that's shaded using ambient occlusion calculations. Recessed areas of the surfaces are shaded darker than their surroundings.
- **Opacity** - An Opacity render mask that's based on the object's Opacity map.
- **Roughness** - Based on the material roughness at the camera ray's hit point.
- **Index Of Refraction (IOR)** - Based on the material Index Of Refraction at the camera ray's hit point.
- **Diffuse[3] Filter** - Shows the diffuse texture color of the scene's Diffuse and **Glossy**[4] materials.
- **Reflection Filter** - Shows the reflection texture color of the scene's **Specular**[5] and Glossy materials.
- **Refraction Filter** - Shows the refraction texture color of the scene's Specular materials.
- **Transmission[6] Filter** - Shows the transmission texture color of the scene's Diffuse materials.

Z-Depth[7] Maximum - Sets the maximum z-depth that OctaneRender® shows.

UV Maximum - Sets the maximum value that OctaneRender® shows for the Texture coordinates.

AO Distance - Specifies the Ambient Occlusion distance.

---

[1]The representation of the surface or volume properties of an object.

[2]Render layers allow users to separate their scene geometry into parts, where one part is meant to be visible and the rest of the other parts "capture" the side effects of the visible geometry. The layers allow different objects to be rendered into separate images where, in turn, some normal render passes may be applied. The Render layers are meant for compositing and not to hide parts of the scene.

[3]Amount of diffusion, or the reflection of light photons at different angles from an uneven or granular surface. Used for dull, non-reflecting materials or mesh emitters.

[4]The measure of how well light is reflected from a surface in the specular direction, the amount and way in which the light is spread around the specular direction, and the change in specular reflection as the specular angle changes. Used for shiny materials such as plastics or metals.

[5]Amount of specular reflection, or the mirror-like reflection of light photons at the same angle. Used for transparent materials such as glass and water.

[6]A surface characteristic that determines if light may pass through a surface volume.

[7]A measure of object distances from the camera typically represented as a grayscale image.

Parallel Samples - Controls how many samples OctaneRender® calculates in parallel. Small values require less memory to store the sample's state, but rendering is slower. High values need more graphics memory, but rendering is faster. The change in performance depends on the scene, the **GPU**[1] architecture, and the number of shader processors in the GPU.

Maximum Tile Samples - Controls the number of samples per pixel to render until OctaneRender® takes the result and stores it in the film buffer. High values generate results less often at the film buffer.

Distributed Ray Tracing - Shows distributed ray tracing. This enabled by default.

Bump And Normal Mapping - Shows or hides the **Bump** and **Normal** maps.

Wireframe Backface Highlighting - Highlights backfaces in the **Wireframe** channel.

Minimize Net Traffic - Distributes the same tile to the net render nodes until it reaches the max samples/pixel for that tile, and then it distributes the next tile to render nodes. This option doesn't affect work done by local GPUs. This way, a render node can merge all of its results into the same cached tile until the Primary Render Node switches to a different tile.

AO Alpha Shadows - Takes Opacity into account in the AO calculation, and render passes can use this to specify if OctaneRender® should render shadows cast by Ambient Occlusion as transparent (zero alpha). This is useful if the you want to composite the render over another image and don't want the AO shadows to be present.

Adaptive Sampling

Adaptive Sampling[2] disables sampling for pixels that reach a specified noise threshold, which allows the **Kernel** to focus its processing on areas that still need refinement. You can find the Adaptive Sampling options in the attributes for the **Direct Light** and **Path Tracing** kernels.

---

[1] The GPU is responsible for displaying graphical elements on a computer display. The GPU plays a key role in the Octane rendering process as the CUDA cores are utilized during the rendering process.
[2] A method of sampling that determines if areas of a rendering require more sampling than other areas instead of sampling the entire rendering equally.

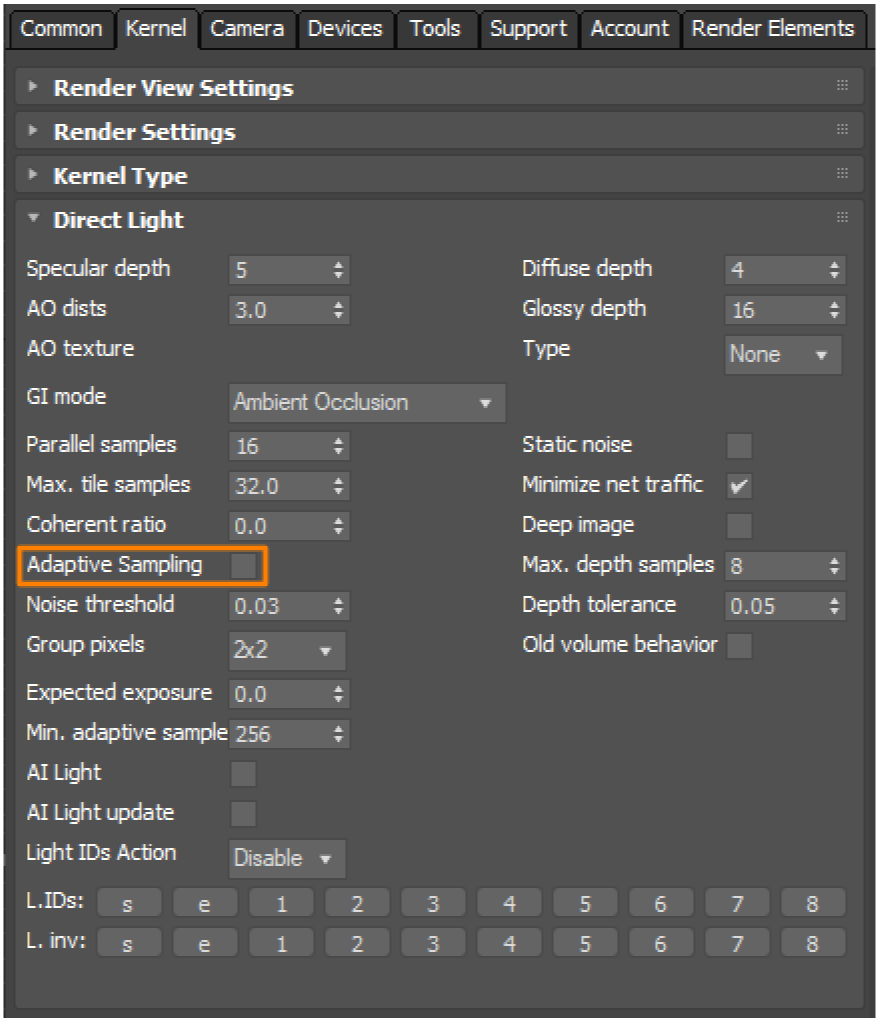


Figure 1: The Adaptive Sampling parameter of the Direct Light kernel

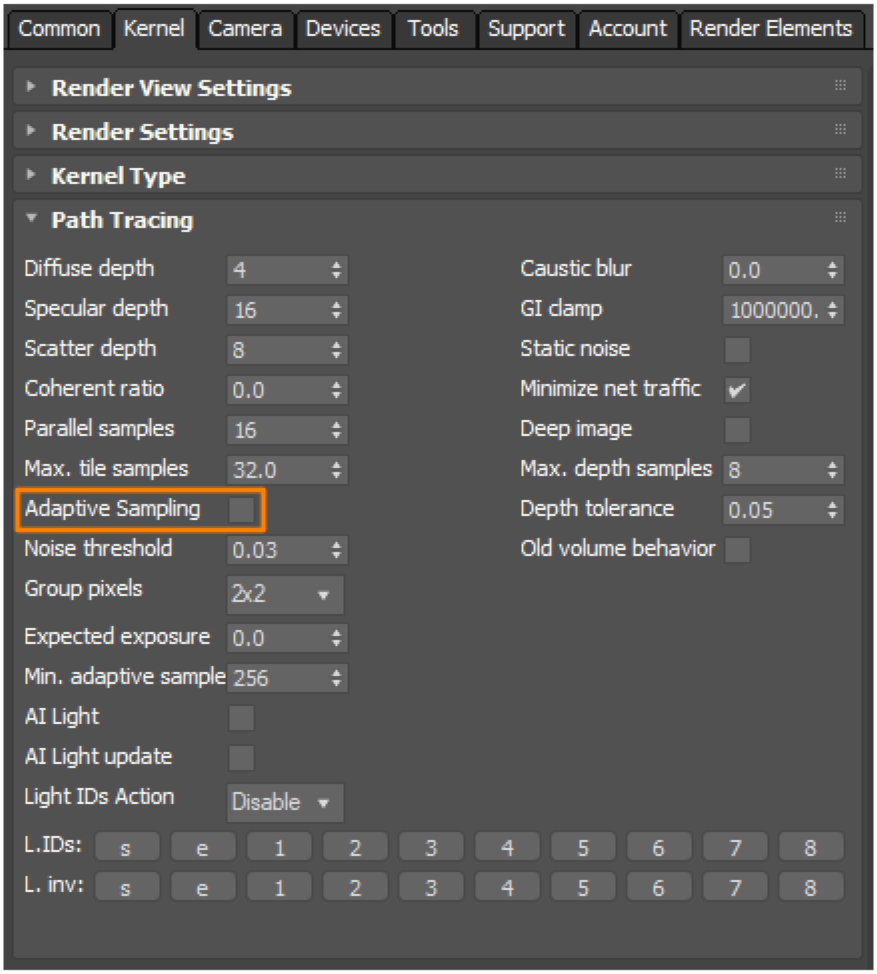


Figure 2: The Adaptive Sampling parameter of the Path Tracing kernel

Adaptive Sampling Parameters

Adaptive Sampling - Stops sampling pixels that reach a specified noise threshold, allowing the Kernel to focus its processing on areas that still need refinement.

Noise Threshold - When Adaptive Sampling is enabled, Noise Threshold specifies the smallest relative noise level. When a pixel's noise estimate is less than this value, OctaneRender® switches off sampling for this pixel. Good values to use are in the range of **0.01** - **0.03**. The default is **0.02**, which is pretty clean.

Group Pixels - When Adaptive Sampling is enabled, Group Pixels specifies the number of pixels that are handled together. If all pixels of a group reach the noise level, sampling stops for all of these pixels.

Expected Exposure - You should set Expected Exposure to be about the same value as the image exposure, otherwise set it to **0** (the default value) to ignore these settings. Adaptive Sampling uses this parameter to determine what pixels are bright and dark - this depends on the exposure setting in the **Octane Imager**. If the value is not **0**, Adaptive Sampling adjusts the noise estimate of very dark areas. It also increases the minimum adaptive samples limit for very dark areas, because very dark areas tend to find paths to light sources irregularly, resulting in an otherwise over-optimistic noise estimate.

Min. Adaptive Samples - Specifies the minimum samples to calculate before Adaptive Sampling kicks in. The noise estimate of a pixel is just an estimate with a large initial error. If you set Noise Threshold to a high value, you should also set **Min. Samples** to a high value to avoid artifacts.

Camera

You can also access OctaneRender's camera settings from the **Camera** tab in the **Render Setup** window when no OctaneRender® cameras are present in the scene. These parameters provide camera controls for the active **Viewport**.

A detailed explanation can be found in the **Lights and Cameras > Camera** section of the documentation.

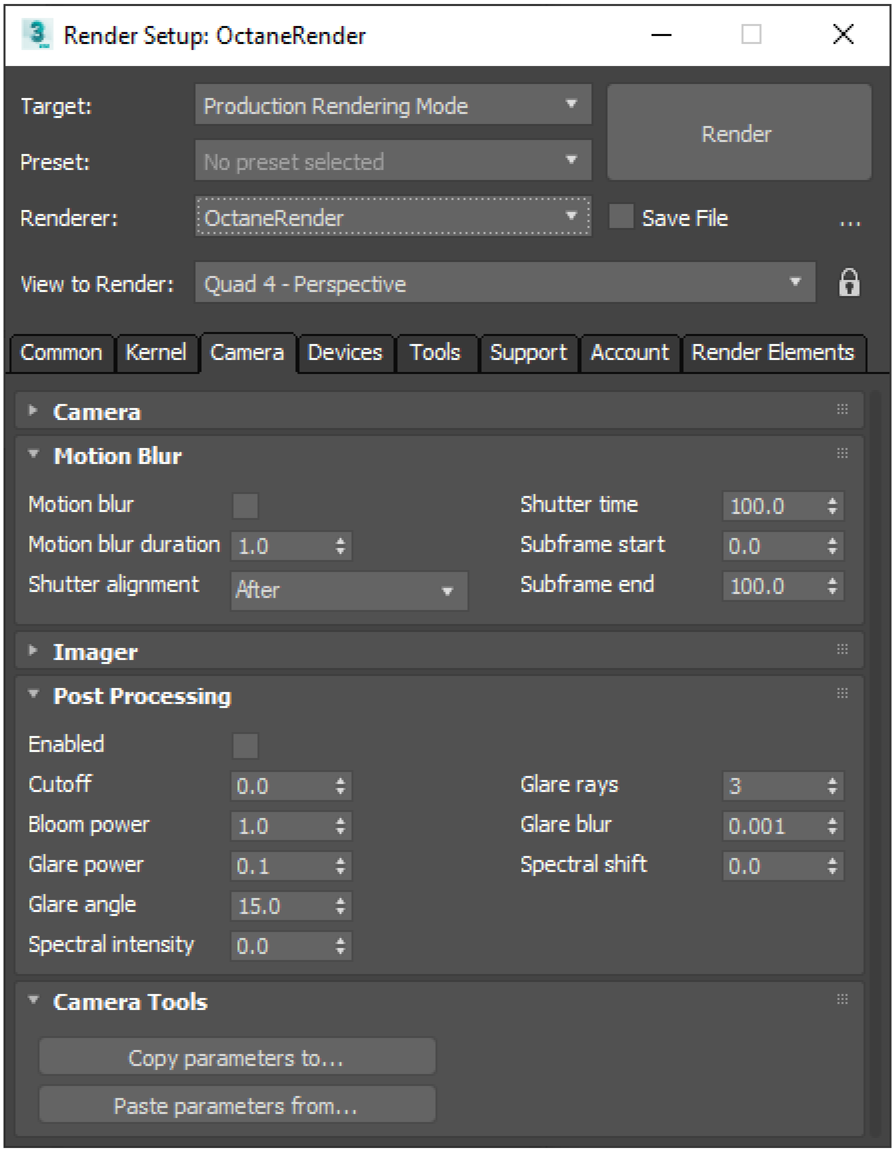


Figure 1: The Camera Tab from the Render Setup window

Devices

OctaneRender® is a **GPU**[1]-based render engine. It is important to manage the GPUs in the system that are used for rendering. This is done from the **Devices** tab under **Render Setup**. Under this tab, the checkboxes for unsupported GPUs are not shown, and you can enable just the GPUs with a supported compute model.

Under this section you'll find:

- **GPU Config**
- **Network Config**

[1] The GPU is responsible for displaying graphical elements on a computer display. The GPU plays a key role in the Octane rendering process as the CUDA cores are utilized during the rendering process.

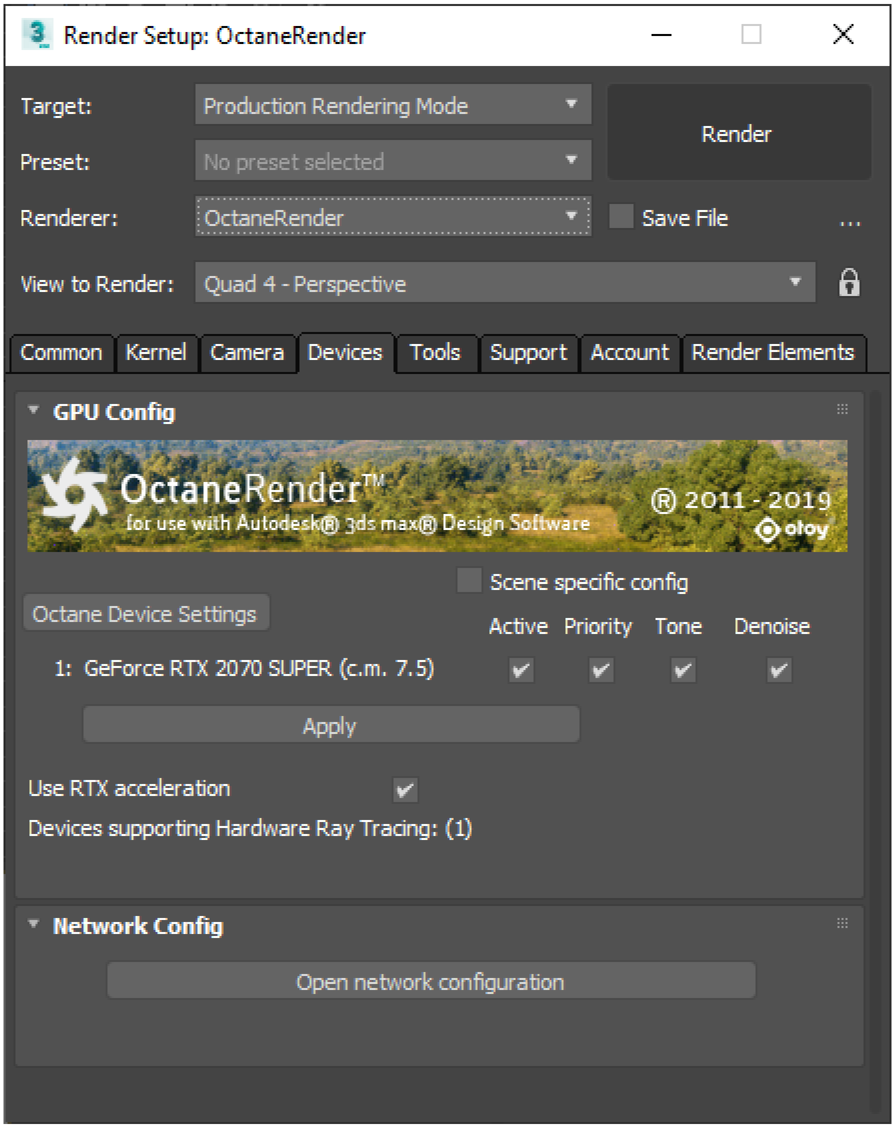


Figure 1: The Devices Tab from the Render Setup window

GPU Config

The **GPU**[1] **Config** allows you to select support GPU hardware(s) used for Octane.

---

[1]The GPU is responsible for displaying graphical elements on a computer display. The GPU plays a key role in the Octane rendering process as the CUDA cores are utilized during the rendering process.

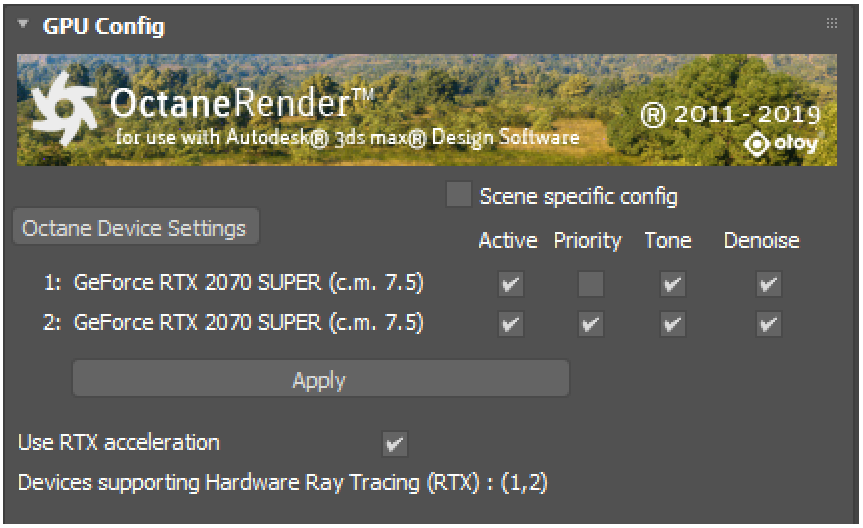


Figure 1: GPU Config

GPU Config Parameters

Scene Specific Config - Setting will be specific for each scene file.

Active - Select GPUs to use for rendering if more than one GPU is installed.

Priority - This shows whether the device will use the priority indicated at the Render Viewport's Render Priority setting. The Use Priority option throttles down rendering on one or more GPUs to improve system responsiveness, especially when rendering on a GPU used for the display.

Tone - Enables the specific GPU to be used for tonemapping.

Denoise - Enables the specific GPU to be used for denoising.

Use RTX Acceleration - Enable ray-tracing hardware.

Octane Device Settings

- **CUDA Driver** - This shows the current CUDA® driver and runtime versions.
- **Render** - Select GPUs to use for rendering if more than one GPU is installed.
- **Use Priority** - This shows whether the device will use the priority indicated at the Render Viewport's Render Priority setting. The Use Priority option throttles down rendering on one or more GPUs to improve system responsiveness, especially when rendering on a GPU used for the display.
- **Tonemap** - Enables the specific GPU to be used for tonemapping.
- **Denoise** - Enables the specific GPU to be used for denoising.
- **Device Info** - Shows the selected device's specifications.

- **Device Memory Usage** - This shows how the video card memory is allocated based on the current scene's geometry, textures, render target, etc.
- **GPU Headroom** - Determines the amount of GPU memory to leave free on each graphics card when storing image textures or geometry data. VRAM is faster than RAM, therefore GPU Head Room tends to be set to a minimal level since it is practical to have the maximum amount of texture and geometry data fitted into VRAM.
- **Use RTX Acceleration** - Enable ray-tracing hardware.

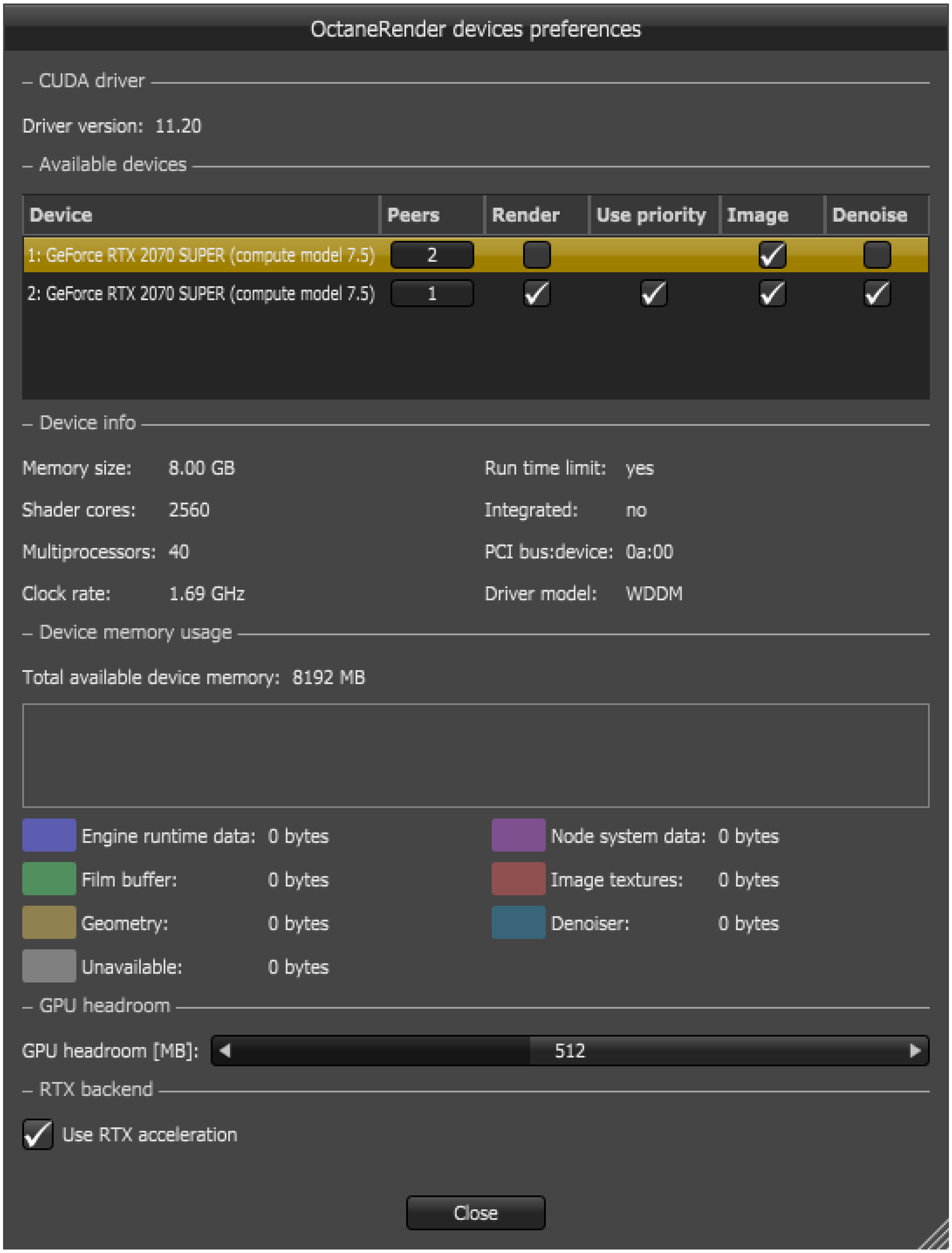


Figure 2: Octane Device Settings

Network Config

Network rendering is dependent on the Standalone Edition's connectivity to other machines. The 3DS Max® plugin needs to connect to this feature through the Standalone Edition. For this reason, each machine that houses GPUs used for distributed rendering should have its own unique Standalone Edition license key.

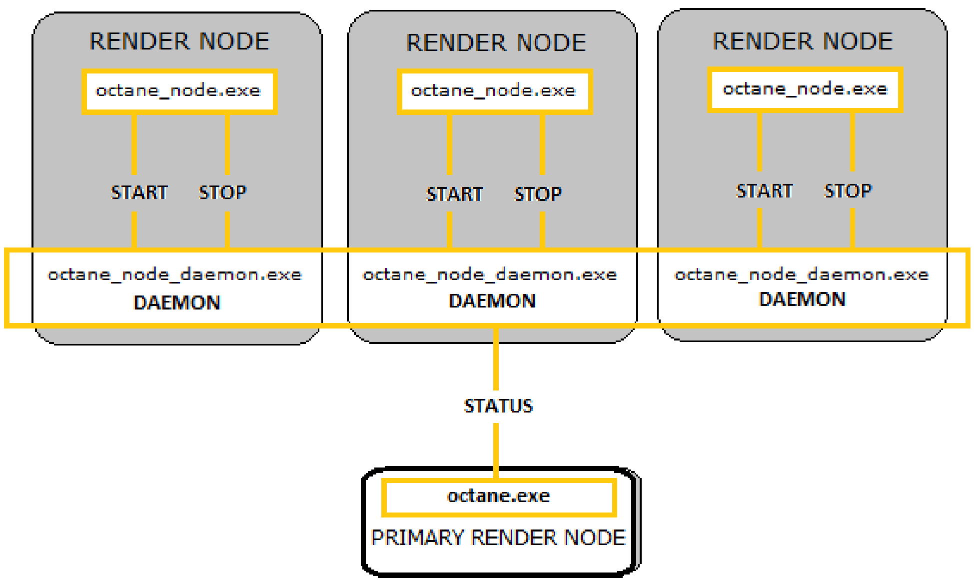


Figure 1: Connectivity for network rendering

To use the plugin's network rendering feature, you need the Enterprise Render Node installed on each of the external computers. Refer to the Standalone manual to learn how to do this.

The network render nodes must have the same version as the current plugin's engine version, otherwise you will not be able to use the render nodes.

Once you install, authenticate, and set up the OctaneRender® Enterprise Render Nodes, go back to the Primary Render Node machine and click on **Render Setup**, then the **Network** tab, and then click on **Open Network Configuration**. Click the **Open Network Configuration** button.

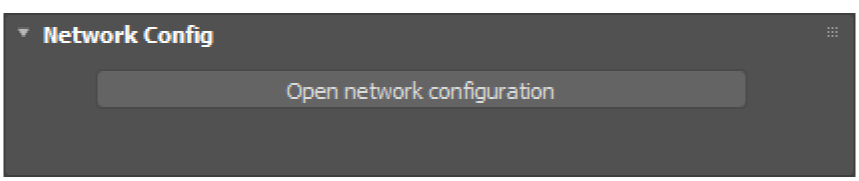
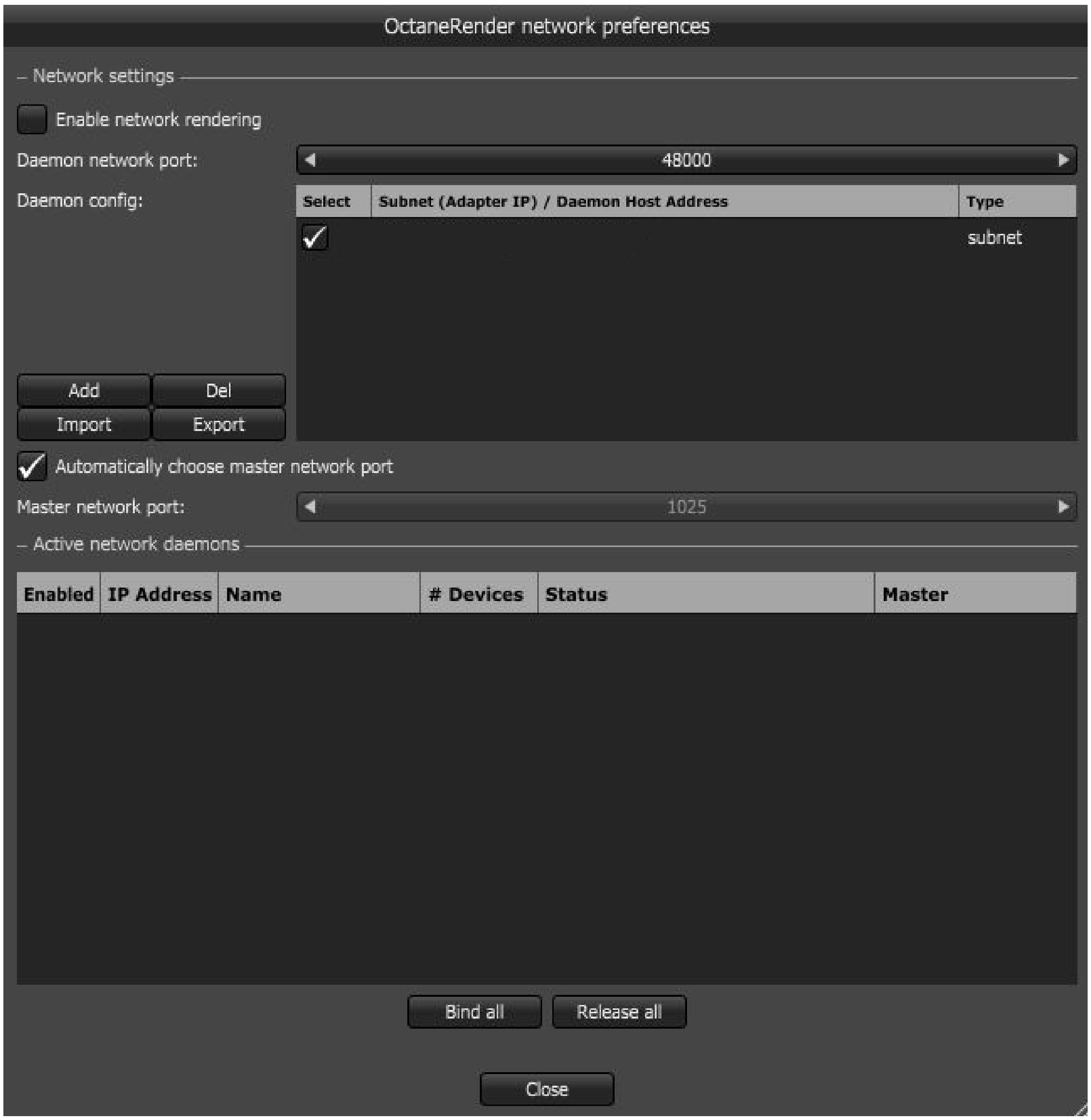


Figure 2: Opening the Network Configuration window

The **Network Rendering**[1] preferences work a bit different compared to the other settings, as you can only edit these preferences when Network Rendering is disabled. Likewise, any changes on the network preferences are updated when you re-enable Network Rendering. The Cancel button does not have an effect in this case.



[1] The utilization of multiple CPUs or GPUs over a network to complete the rendering process.

Figure 3: OctaneRender Network Preferences

Network Rendering Overview

Network rendering lets you utilize additional GPUs in other computers to render images. OctaneRender® distributes compiled render data and not scene data, so no file management is required. It is similar to working with additional GPUs by allowing the distributed rendering of single images over multiple computers connected through a fast local area network. Network rendering requires a Primary Render Node and one or more helper render nodes on different computers. The OctaneRender® instance that drives the rendering is the Primary Render Node, and the OctaneRender® instances that help are the render nodes.

The Primary Render Node requires an OctaneRender®Enterprise license while the render nodes will each require an OctaneRender®Enterprise Render Node license. Make sure that the Primary Render Node and the helper render nodes are not blocked by any firewalls in the network or operating system.

Primary Node, Render Nodes, And Daemons

The Standalone version or the octane.exe act as Primary Node and a special console version of OctaneRender®, octane_node.exe, can run on other computers as Render Nodes. They should all be on different computers, or they would have to share the same GPUs.

The OctaneRender® Primary Node does all the render data processing. The Render Node does not need to have a powerful CPU, but the Render Node needs enough RAM to store the render data plus some render results. The Render Node's operating systems can also be different since the communication between the machines is cross-platform. No data is stored on the Render Node’s discs, it's all stored in memory.

Each time network rendering is required, the Render Node process has to launch on the Render Node machines. The Render Node daemon makes the control of the Render Nodes more practical, as it can launch at startup on each machine in the network. The daemon is the little program that starts a Render Node process on the machine on request by a Primary Node, monitors it, and stops it on request by a Primary Node. Monitoring means making sure that a running Render Node sends a regular heartbeat to the daemon, and if that doesn’t happen, it first tries to stop the Render Node, and then it kills the process as a last resort if necessary. The daemon runs all the time, and starts/stops a Render Node process if a Primary Node requests it. The daemon also listens for the heartbeat of the Render Node to check if the Render Node process is still running. This Render Node daemon eliminates needing to launch the Render Node process manually on each computer each time rendering is required on the Render Node.

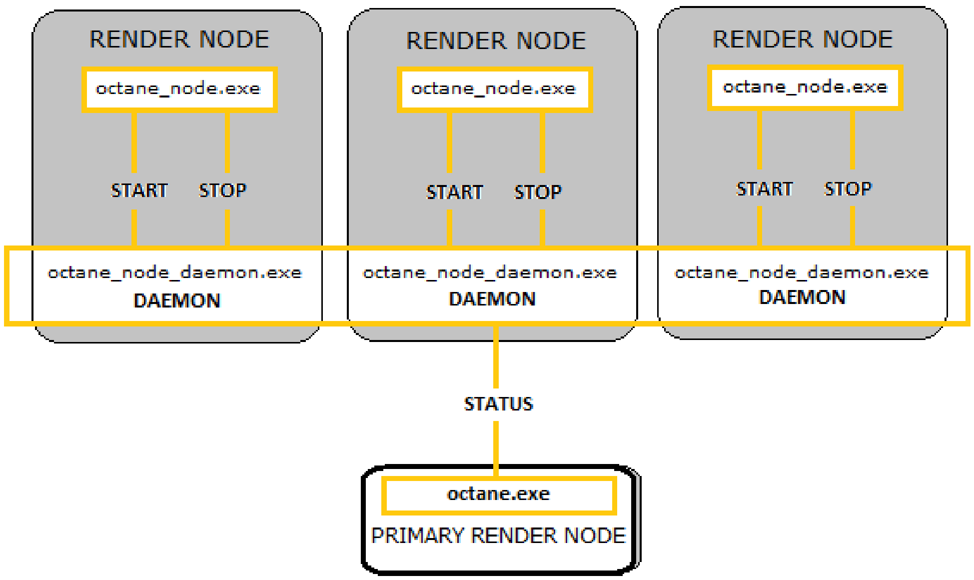


Figure 4: Primary Node - Render Node configuration

The Standalone Edition's network rendering feature is also useful while using a plugin. OctaneRender's licenses are assigned per machine, so any machine can become a Primary Node or Render Node.

Setting Up The Render Node Daemon

To set up the daemon, run the batch script _install_daemon.bat on the Render Node computer. During the setup, OctaneRender® asks you to choose a port for Primary Node requests. After that, the daemon resides on that machine, active at all times.

When a Primary Node invokes a daemon, the Render Node launches to get some information about the number of GPUs, version, bitness, etc., and closes again. After that there is no Render Node process running, so the daemon waits for Primary Nodes to detect it by scanning the complete local network in regular intervals. The daemon should appear in the daemon list of the network preferences of the Primary Nodes. If it does not, it could be because:

- The network rendering in the Primary Node is not enabled.
- The daemon is listening on a different port than the Primary node is scanning.
- The daemon is in a different subnet than the Primary Node is scanning. If you have one ethernet adapter on the daemon and Primary Node PCs, you can ignore this case.

- The Windows® firewall keeps the Primary Node from connecting to the daemon, or the daemon from responding to the Primary Node. To verify it, disable the firewall on both PCs. If the daemon is now detected (this can take up to 20 seconds), you can try enabling one firewall after the other to see which one is causing trouble.

When you enable a daemon in the Primary Node's settings, the Render Node launches and appears in the Primary Node's status bar. One Primary Node can activate one daemon at a time. If daemon is occupied by another Primary Node, you will see the daemon state change accordingly. The automatic port configuration is an option on the Primary Node that enables the same computer to use multiple Primary Nodes.

When network rendering is disabled, the local network is not scanned for daemons. The Primary Node scans for daemons only when network rendering is enabled.

Maximum Number Of GPUs

OctaneRender® (through the Primary Node) may use the networked GPUs as long as the number of GPUs do not exceed the limit set by the OctaneRender® license for that Primary Node. The OctaneRender® Enterprise license lets a Primary node render with up to 200 GPUs at a timeincluding the Primary Node's GPUs in the machine. For Studio licenses, a machine can use up to two GPUs at a time and has no support for network rendering.

In versions 3.05.0 and earlier, OctaneRender® looks for GPUs in Render Node nodes within the same subnet, so OctaneRender® treats every **GPU**[1] in the Primary Node and Render Nodes as if these are installed in the Primary Node machine. Each local and remote GPU in the Render Nodes is just another GPU, so as long as it is available (not used by other renderers or the OS or any other application) and it's exposed as a CUDA® GPU, OctaneRender® will pick those GPUs in that subnet until it reaches the GPU limit, if the network has more than the GPUs limit then the rest of the available GPUs will just not be used.

In OctaneRender® v3.06.x and later, the native Octane Network Rendering feature has integrated support for multiple subnets and considers hostnames and IP addresses. The improved Network Render feature considers the per-Render Node systems that determine how GPUs are allocated to applications and the overall network system configuration that affect how networked nodes allocate processing capabilities between nodes. Depending on how the mix of configurations play out, there are some cases where a network allows OctaneRender® to use up to the GPU limit across the network, regardless what Render Nodes the GPUs are installed in (per GPU regardless of the Render Node), and there are also other cases where the network cuts off the other remaining Render Nodes when a Render Node will go over the GPU limit (so per Render Node rather than per GPU, this means it will use all GPUs in that Render Node or not use that Render Node at all) - effectively reducing the total rendering GPUs to significantly less than the limit.

---

[1]The GPU is responsible for displaying graphical elements on a computer display. The GPU plays a key role in the Octane rendering process as the CUDA cores are utilized during the rendering process.

In all versions of OctaneRender®, there is always some hardware considerations, such as the node/GPU power density and heat loads within the Render Nodes that affect the availability and usability of each installed GPU.

Tools

Octane provides additional tools to deal with **Materials**[1], file Exports, and support for older versions of Octane. Here you will find:

- Live DB
- **Material**[2] Converter
- Octane Export
- Export Settings
- 1.9 Version Support
- Octane Preferences

---

[1] A set of attributes or parameters that describe surface characteristics.

[2] The representation of the surface or volume properties of an object.

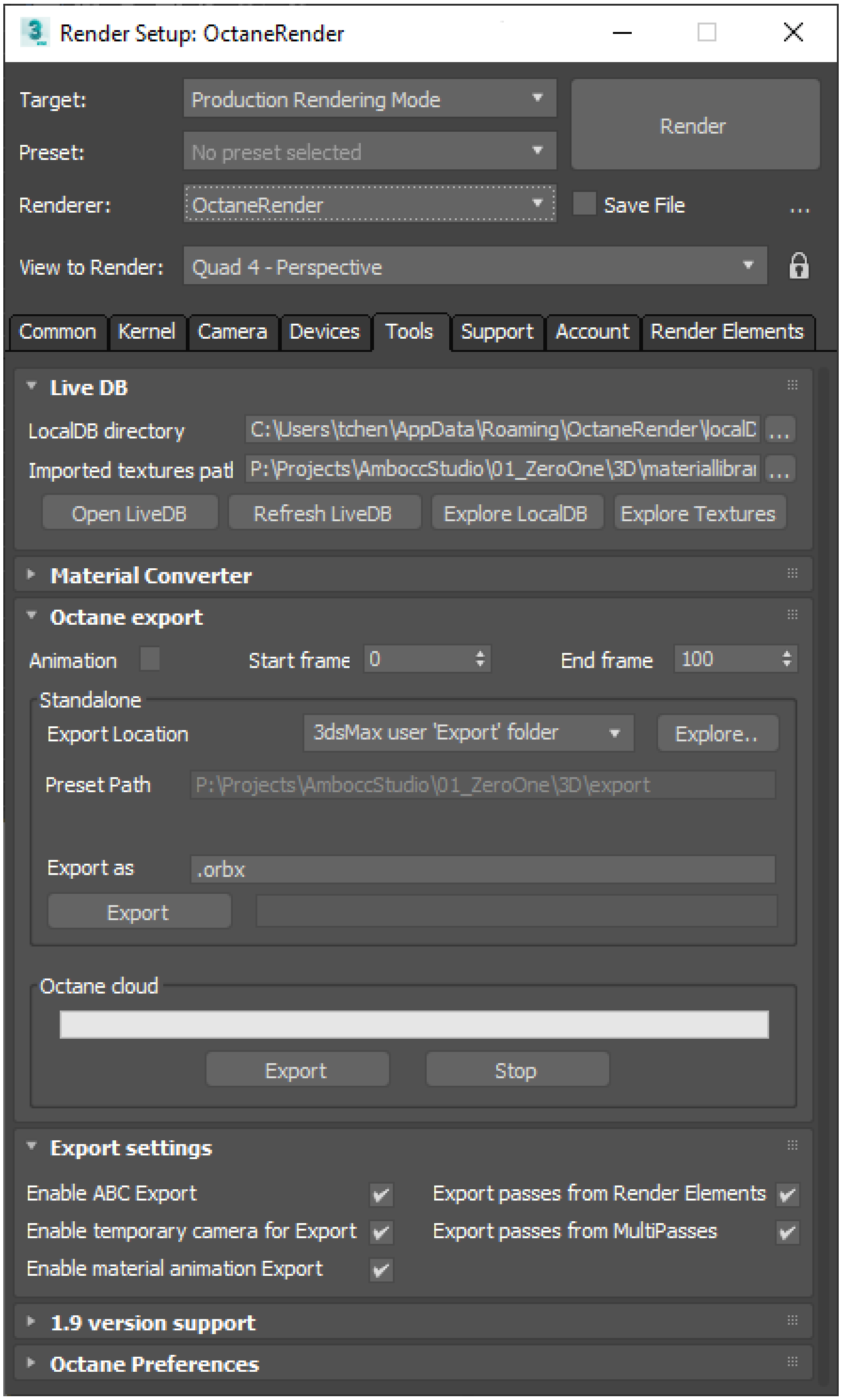

Render Setup: OctaneRender
Target:
Production Rendering Mode
Preset:
No preset selected
Renderer:
OctaneRender
Render
Save File
View to Render:
Quad 4 - Perspective
Common
Kernel
Camera
Devices
Tools
Support
Account
Render Elements
Live DB
LocalDB directory
C:\Users\tchen\AppData\Roaming\OctaneRender\localD
Imported textures patl
P:\Projects\AmboccStudio\01_ZeroOne\3D\materiallibrai
Open LiveDB
Refresh LiveDB
Explore LocalDB
Explore Textures
Material Converter
Octane export
Animation
Start frame
0
End frame
100
Standalone
Export Location
3dsMax user 'Export' folder
Explore..
Preset Path
P:\Projects\AmboccStudio\01_ZeroOne\3D\export
Export as
.orbx
Export
Octane cloud
Export
Stop
Export settings
Enable ABC Export
Enable temporary camera for Export
Enable material animation Export
Export passes from Render Elements
Export passes from MultiPasses
1.9 version support
Octane Preferences

Figure 1: The Tools Tab from the Render Setup window

Live DB

The 3DS Max® plugin has access to the OctaneRender® online Live DB, which is OctaneRender's asset database. It stores **Materials**[1], groups of **Nodes**, and whole scenes shared by the OctaneRender® community and the OctaneRender® team. The asset database makes it easier to move groups of Nodes, scenes, and assets across the OctaneRender® plugins and the Standalone edition.

You can access the Live DB from the **Octane** menu (Figure 1), or by opening the **Render Setup** window, then clicking on the **Tools** tab and clicking on the **Open LiveDB** button (Figure 2).

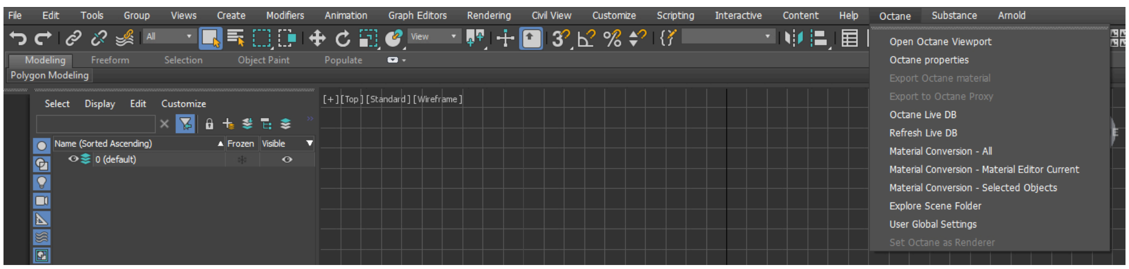


Figure 1: Accessing the Octane Live DB from the Octane menu

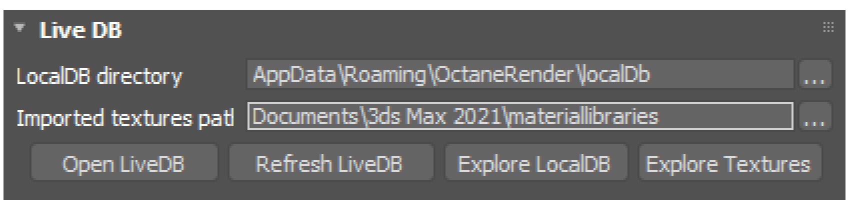


Figure 2: Live DB

Many of the Materials contain **Texture** maps. 3DS Max® cannot store these Texture maps within a scene, so 3DS Max® saves the Texture maps to disk under a specified folder and adds the **Material**[2] name to the path. You can customize the path in **Imported Textures**[3] **Path** directory.

To use a Material in the Live DB, right-click on the Material thumbnail and click **Import**.

---

[1] A set of attributes or parameters that describe surface characteristics.

[2] The representation of the surface or volume properties of an object.

[3] Textures are used to add details to a surface. Textures can be procedural or imported raster files.

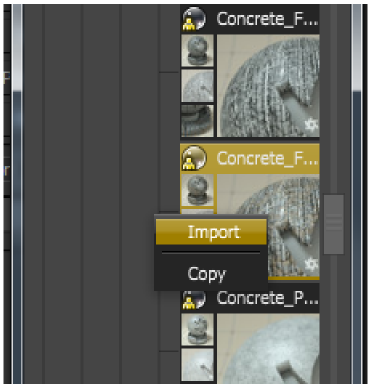


Figure 3: Importing a Material

Live DB Parameters

LocalDB Directory - Select the **LocalDB** directory. This should match the directory set in **OctaneRender Standalone Preferences**. See options to move files in **Octane Preferences > Material** tab.

Imported Textures Path - Defines the folder to save the textures from LiveDB.

Open LiveDB - Open the **LiveDB** tool.

Refresh LiveDB - Manual refresh the content of the **LiveDB** tool.

Explore LocalDB - Open Windows Explorer to the **LocalDB** folder.

Explore Textures - Opens Windows Explorer to the **Imported Textures** folder.

Local DB

OctaneRender® includes a Local DB that houses **Materials**[1] and **Nodes**, similar to the Live DB. You can save these assets on a local drive and organize them in folders with thumbnails for easy identification. To access the Local DB, click on **Open LiveDB**, then proceed to the **LocalDB** tab.

---

[1] A set of attributes or parameters that describe surface characteristics.

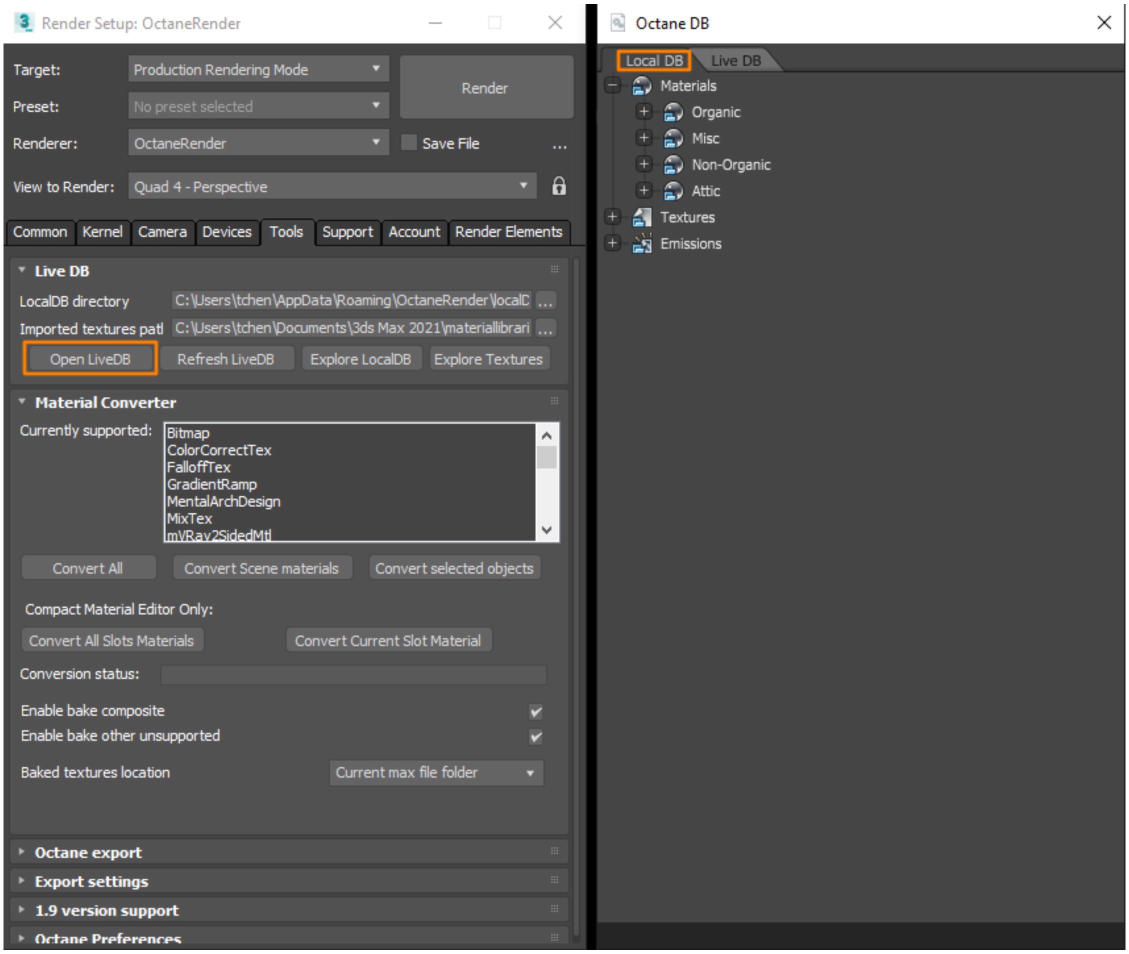


Figure 1: Local DB

Right-click and import to use a **Material**[1] or other Local DB asset.

---

[1]The representation of the surface or volume properties of an object.

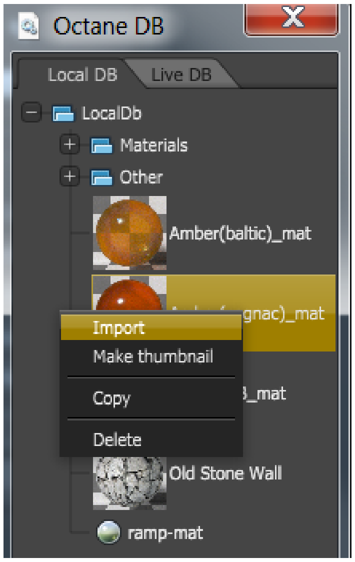


Figure 2: Importing Materials from the Local DB into 3DS Max®

Material Converter

In some situations, you'll need to convert **Materials**[1] from other render engines into OctaneRender® Materials in order to render them in OctaneRender® for 3DS Max®. To accomplish this, go to the **Render Setup** window and click the **Tools** tab. Next, expand the **Material**[2] **Converter** rollout and click the **Convert All** button. OctaneRender® converts all **Object** types listed in the **Currently Supported** window.

---

[1] A set of attributes or parameters that describe surface characteristics.
[2] The representation of the surface or volume properties of an object.

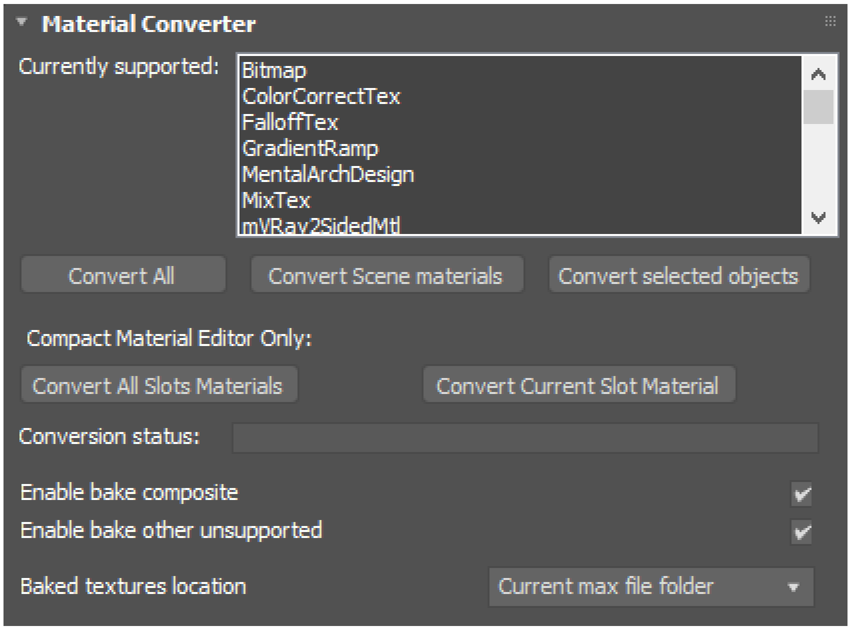


Figure 1: Material Conversion

Certain Objects, such as light sources, may need adjustments once the conversion process is complete.

Material Converter Parameters

Currently Supported - defines a supported list of object types that can be converted to OctaneRender®.

Convert All - Convert all supported object types.

Convert Scene Materials - Convert only the current scene materials.

Convert Selected Objects - convert only the objects that are selected.

Convert All Slot Materials - Convert all materials in the Compact Material Editor's slots.

Convert Current Slot Material - Convert only the currently selected Compact Material Editor's slot.

Conversion Status - Shows the progress of the conversion.

Enable Bake Composite - Bake out textures from combined textures for OctaneRender®.

Enable Bake Other Unsupported - Bake out textures from unsupported features for OctaneRender®.

Bake Textures[1] Location - Defines the directory to place the baked textures.

Octane Export

OctaneRender® creates packages containing macro nodes otherwise known as nodegraphs. **ORBX**[2] packages store all the geometry, materials, animation data, textures and everything else related to the scene into a single archive file. This feature replaces the older method of embedding images in .ocs files. Each .orbx package can then be stored locally or even uploaded to the OctaneLive Database as a shared resource for other Octane users to access. The .orbx file format can also be exported from applications such as Maya, Cinema 4D, or 3DS Max that have the Octane plugin installed and licensed. This makes it possible to move scenes and materials from host applications to Octane Standalone and the Octane Render Cloud.

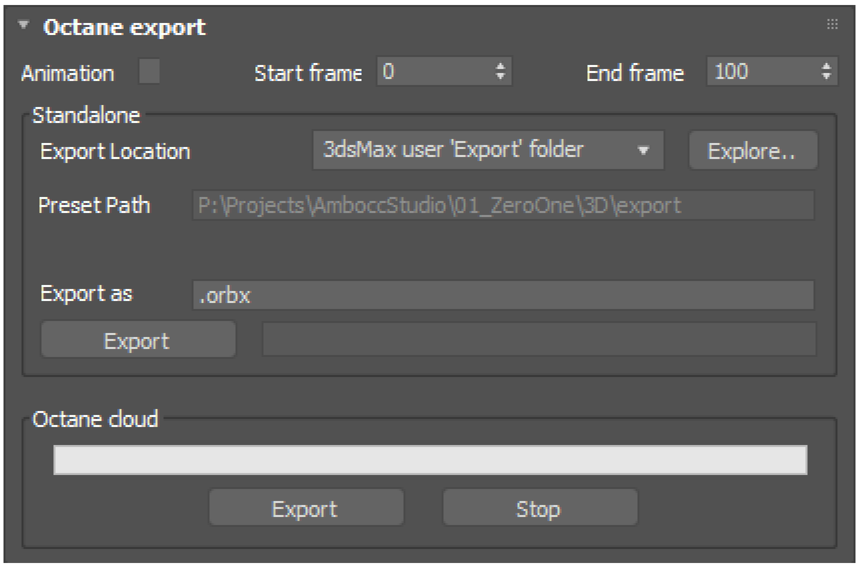


Figure 1: Octane Export

---

[1] Textures are used to add details to a surface. Textures can be procedural or imported raster files.

[2] The ORBX file format is the best way to transfer scene files from 3D Authoring software programs that use the Octane Plug-in such as Octane for Maya, Octane for Cinema 4D, or OctaneRender Standalone. This format is more efficient than FBX when working with Octane specific data as it provides a flexible, application independent format. ORBX is a container format that includes all animation data, models, textures etc. that is needed to transfer an Octane scene from one application to another.

Octane Export Parameters

Animation - Enables animation export.

Start Frame - Defines the first frame of the animation.

End Frame - Defines the last frame of the animation.

Standalone

- **Export Location** - Defines the type of folder path for the export.
- **Preset Path** - Show the folder path for the export
- **User Path** - Show the user path for the export.
- **Export as** - Defines the type of export. Currently only supports ORBX.
- **Export** - Export the scene.

Octane Cloud - Allows you to export the scene to the Octane Cloud server (Requires log-in).

Export Settings

Provides additional option for exporting your **ORBX**[1] file.

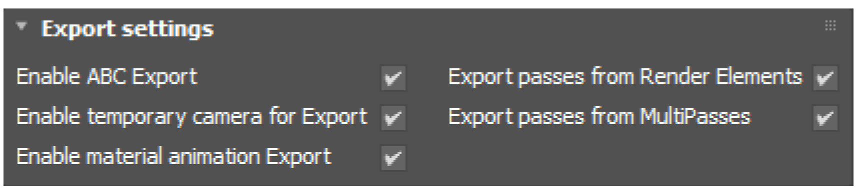


Figure 1: Export Settings

[1]The ORBX file format is the best way to transfer scene files from 3D Authoring software programs that use the Octane Plug-in such as Octane for Maya, Octane for Cinema 4D, or OctaneRender Standalone. This format is more efficient than FBX when working with Octane specific data as it provides a flexible, application independent format. ORBX is a container format that includes all animation data, models, textures etc. that is needed to transfer an Octane scene from one application to another.

Export Setting Parameters

Enable ABC Export - Enable saving an **Alembic**[1] file (.abc) while exporting to ORBX. Can be disabled with no animation to view details of geometry (export with animation always uses Alembic file).

Enable Temporary Camera for Export - Enable post and pre export script to create a camera, when there is no active camera in the scene.

Enable Material[2] Animation Export - Enable exporting all animated material values to ORBX.

Export Passes from Render Elements - Enable exporting Render Element passes.

Export Passes from MultiPasses - Enable exporting Multipass settings from OctaneRender Viewport.

1.9 Version Support

This section provides conversion support for older versions of OctaneRender®.

---

[1] An open format used to bake animated scenes for easy transfer between digital content creation tools.

[2] The representation of the surface or volume properties of an object.

Figure 1: 1.9 Version Support

1.9 Version Support Parameters

Support Scenes from 2.23.2 - 1.9 and Earlier

- **Activate** - When deactivate, loading an old Max file will result in grey materials. Check in **Current Scene** box that old items are detected. You can then activate the conversion.

- **Auto Set Gamma[1] Hack** - This will enable the **3DS Max Gamma Hack** when old max files are loaded.
- **Disable temporary Conversion of XRef Materials[2]** - XRef .max files with v1.9 materials must be converted separately. Enabling conversion may result in an unstable scene and is not recommended. However, it provides a way to have a quick look at what the fully converted scene will look like.

Current Scene - Scene Octane version statistics.

Octane Preferences

The **Octane Preferences** window provide behaviors when launching and running Octane for 3DS Max. You'll also find settings for dealing with common commercial plugins used for 3DS Max.

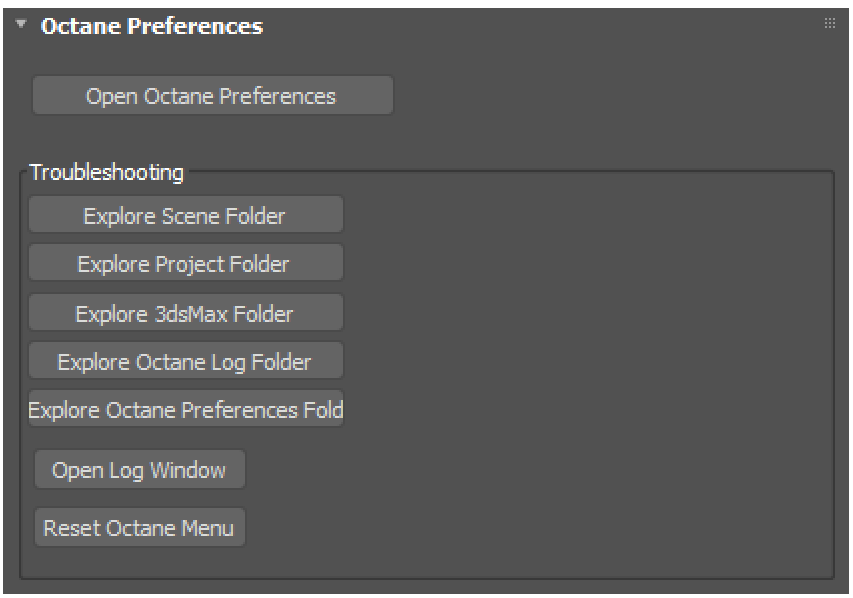


Figure 1: Octane Preferences

---

[1]The function or attribute used to code or decode luminance for common displays. The computer graphics industry has set a standard gamma setting of 2.2 making it the most common default for 3D modelling and rendering applications.
[2]A set of attributes or parameters that describe surface characteristics.

Octane Preference Parameters

Open Octane Preferences - Provides settings for the following:

- Main Settings
- Viewport
- **Material**[1]
- Export
- Previous Versions
- Vray, Corona, Standard
- Pflow, Standard
- Phoenix, Forest Pack, tyflow, Hair and Fur
- Troubleshooting
- Kernel Default
- User Interface

Troubleshooting

- **Explore Scene Folder** - Opens Windows Explorer to the file folder.
- **Explore Project Folder** - Opens Windows Explorer to the project folder.
- **Explore 3DS Max Folder** - Opens Windows Explorer to the 3DS Max install folder.
- **Explore Octane Log Folder** - Opens Windows Explorer to the Octane Log folder.
- **Explore Octane Preferences Folder** - Opens Windows Explorer to the Octane Preference folder.
- **Open Log Window** - Opens the Octane Log Viewer.
- **Reset Octane Menu** - When changing **Workspaces**, your **Octane Menu** might disappear. This will reinstall the menu to the new workspace.

Support

Support provides Octane with instructions on handling geometry and common commercial plugins for 3DS Max. Under the Support tab are:

- Common Scene Collection Settings
- Forest Pack

---

[1]The representation of the surface or volume properties of an object.

- RailClone
- Standard Particles
- PFlow
- tyFlow
- Phoenix FD
- Ornatrix
- Hair & Fur

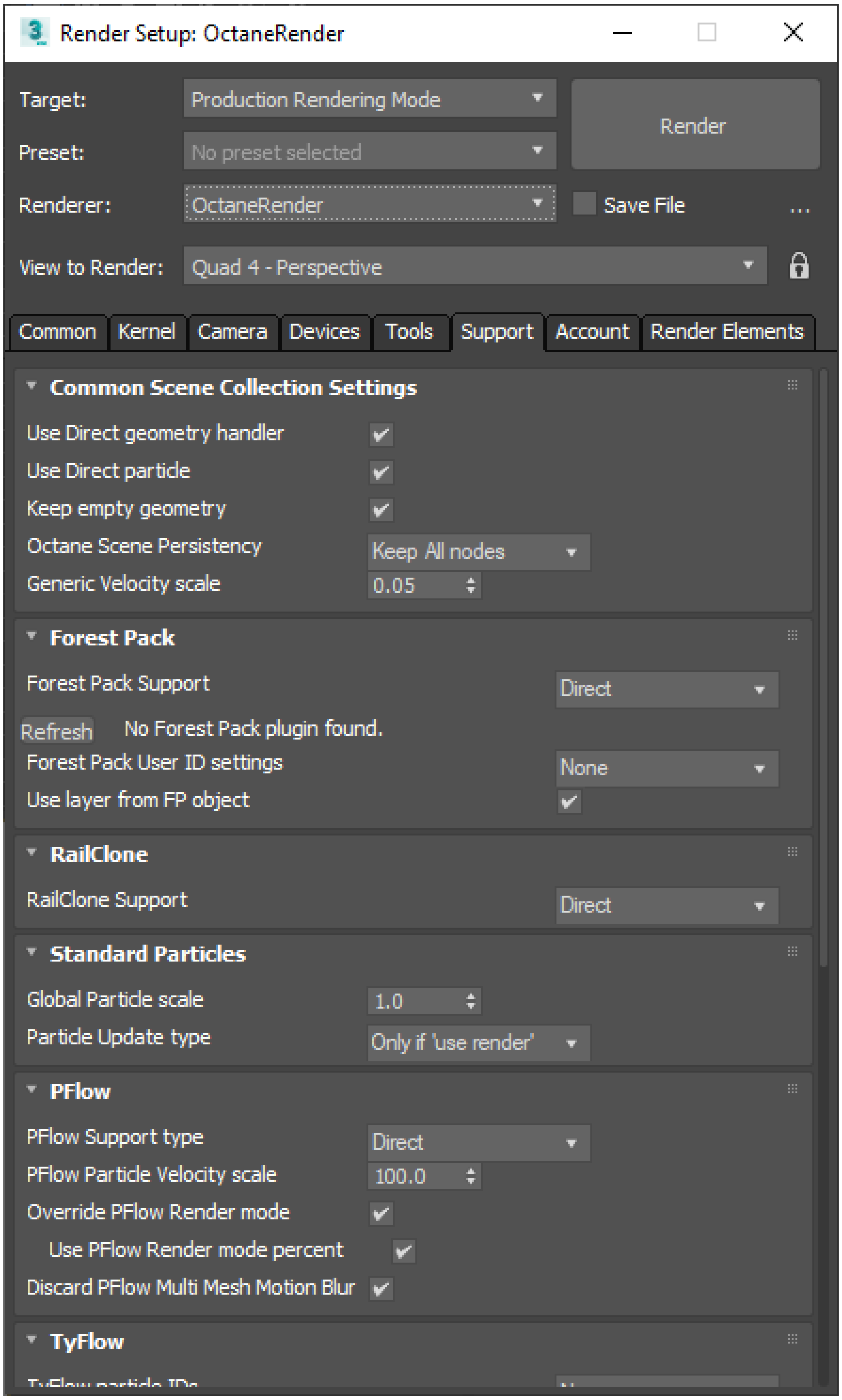

Render Setup: OctaneRender
Target:
Production Rendering Mode
Preset:
No preset selected
Render
Renderer:
OctaneRender
Save File
View to Render:
Quad 4 - Perspective
Common
Kernel
Camera
Devices
Tools
Support
Account
Render Elements
Common Scene Collection Settings
Use Direct geometry handler
Use Direct particle
Keep empty geometry
Octane Scene Persistency
Keep All nodes
Generic Velocity scale
0.05
Forest Pack
Forest Pack Support
Direct
Refresh
No Forest Pack plugin found.
Forest Pack User ID settings
None
Use layer from FP object
RailClone
RailClone Support
Direct
Standard Particles
Global Particle scale
1.0
Particle Update type
Only if 'use render'
PFlow
PFlow Support type
Direct
PFlow Particle Velocity scale
100.0
Override PFlow Render mode
Use PFlow Render mode percent
Discard PFlow Multi Mesh Motion Blur
TyFlow

Figure 1: Support Tab from the Render Setup window

Common Scene Collection Settings

Settings for handling Geometry.

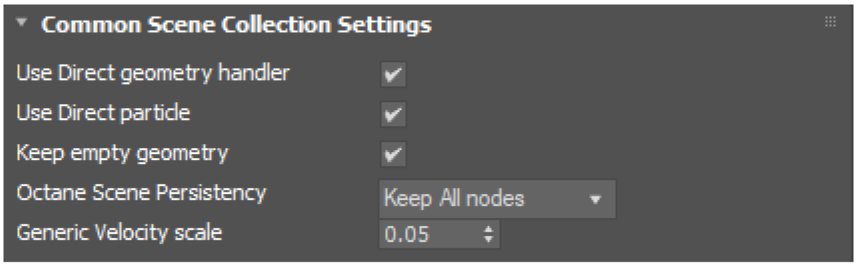


Figure 1: Common Scene Collection Settings

Common Scene Collection Setting Parameters

Use Direct Geometry Handler - Use a new geometry handler. This creates Octane nodes matching closely 3DS Max geometry created by other plugins.

Use Direct Particle - Use particle geometry instances when possible instead of a single mesh for all particles.

Keep Empty Geometry - Always create Octane nodes for each 3DS Max node of the scene, even if no geometry is available on the first rendered frame. This is needed for some animated geometry. Note that this can be changed per object, in the Octane Geometry modifier.

Octane Scene Persistency - Select the types of Octane nodes to keep in memory. This avoid re-process nodes (e.g. texture compression), and makes the work flow faster (avoiding re compiling geometry that didn't change).

Generic Velocity Scale - When **Motion Blur**[1] is used with vertex velocity from a map (**Velocity Channel ID**), velocity are scaled with this value.

Forest Pack

Settings for handling Forest Pack. An instance scattering tool from iToo Software.

---

[1] An optical phenomenon that occurs when a camera's shutter opens and closes too slowly to capture movement without recording a blurring of the subject.

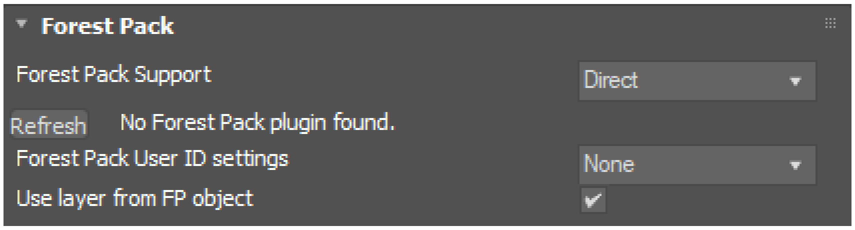


Figure 1: Forest Pack

Forest Pack Parameters

Forest Pack Support - Support Forest Pack object types.

- **Direct** - Can render both instances and mesh render type (as set in FP Display / Render / Mode: 'Automatic' or 'Meshes').
- **Legacy** - Support for older versions.
- **Standard** - Can render only 'Meshes' mode.
- **Discard** - Discard support.

Refresh - Detects if the Forest Pack plugin was installed.

Forest Pack User ID Settings - How to set the Octane User IDs from Forest Pack instances.

User Layer from FP Object - Use layer settings from Forest Pack object (Octane properties), instead of from instance object(s) properties.

RailClone

Settings for RailClone parametric object creation tools from iToo Software.

Figure 1: RailClone

RailClone Parameters

RailClone Support - Support RailClone Object types.

- **Direct** - Can render both instances and mesh render type.
- **Standard** - Can render only 'Meshes' mode.

- **Discard** - Discard support.

Standard Particles

Settings for 3DS Max non-event driven standard particle system.

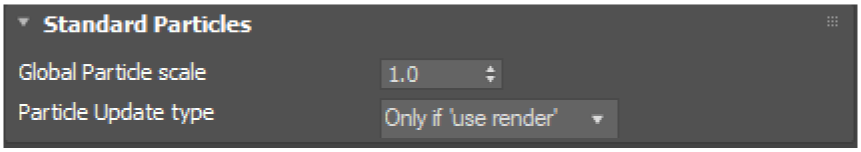


Figure 1: Standard Particles

Standard Particle Parameters

Global Particle Scale - For use with the 'Extended Geometry' render mode in object properties.

Particle Update Type - Defines how to perform a particle update on frame changes.

PFlow

Settings for 3DS Max event driven Particle Flow system.

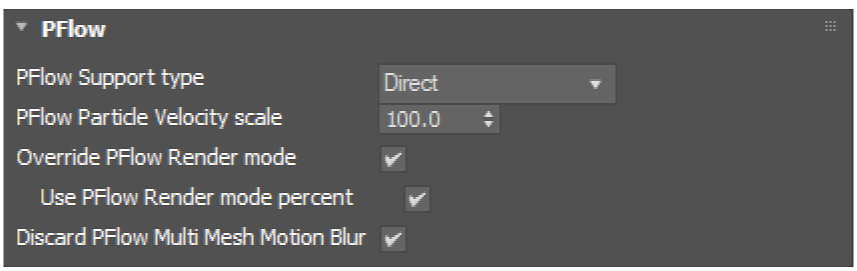


Figure 1: PFlow

PFlow Parameters

PFlow Support Type - Support PFlow object types.

- **Direct** - Can render both instances and mesh render type.
- **Legacy** - Support for older versions.

- **Standard** - Can render only 'Meshes' mode.
- **Discard** - Discard support.

PFlow Particle Velocity Scale - For **Motion Blur**[1] effect.

Override PFlow Render Mode - When disabled, uses the Render mode VS Viewport mode setting from the Kernel tab ('Notify object on render'). When enabled, uses the mode for all PFlow objects as set in 'Use PFlow Render mode percent'.

- **Use PFlow Render Mode Percent** - Use the corresponding percent of particles, as set in PF Source Quantity Multiplier.

Discard PFlow Multi Mesh Motion Blur - Avoid multiple evaluations using particle velocity if available.

tyFlow

An event driven particle simulation plugin suite for 3DS Max by Tyson Ibele.

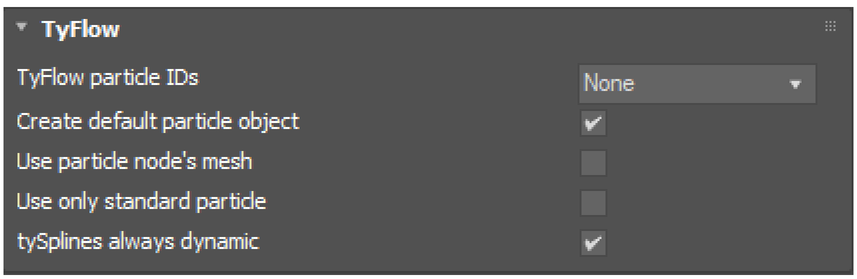


Figure 1: tyFlow

tyFlow Parameters

tyFlow Particle IDs - Applies to tyFlow particle simulators without an Octane Geometry Modifier.

Create Default Particle Object - If a tyFlow particle doesn't use a geometry (e.g. sprite), it creates and uses a dummy sphere instead. This allows the system to see the generated particles by adding shapes if needed.

Use Particle Node's Mesh - Use both the particles and the mesh of tyFlow's node. This should be used only for some special tyFlow objects.

---

[1]An optical phenomenon that occurs when a camera's shutter opens and closes too slowly to capture movement without recording a blurring of the subject.

Use Only Standard Particle - Use only standard particle interface. Can be used for some particle graph when no Shape or Mesh are set. However, it is preferable to add Shape and Mesh to fully use tyFlow particle interface. Note that the scene must be re-loaded for the change to take effect.

tySplines Always Dynamic - Force tySplines object to be dynamic (Movable **Proxy**[1]).

Phoenix FD

An all-in-one solution for fluid dynamics.

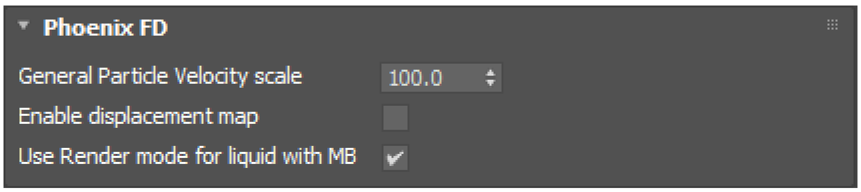


Figure 1: Phoenix FD

Phoenix FD Parameters

General Particle Velocity Scale - Scale motion blur intensity.

Enable Displacement[2] Map - Toggle use of displacement maps on or off.

Use Render Mode for Liquid with MB - Use Render Mode for liquid simulator object when **Motion Blur**[3] is enabled. This will override 'Notify objects on render' in kernel setting and 'Use render mode' in object properties.

Ornatrix

A hair, fur, and feather creation plugin with hair physics. Created by Ephere Inc.

---

[1]An object saved as a separate file with the purpose of being reused in larger scenes. This is used to minimize any addition to the total polygon count in the scene, especially if the scene requires the same object to appear several times. If used in conjunction with instancing, Proxies help keep very large scenes from reaching polygon limits and also keeps the relative file size of the main project file manageable.
[2]The process of utilizing a 2D texture map to generate 3D surface relief. As opposed to bump and normal mapping, Displacement mapping does not only provide the illusion of depth but it effectively displaces the actual geometric position of points over the textured surface.
[3]An optical phenomenon that occurs when a camera’s shutter opens and closes too slowly to capture movement without recording a blurring of the subject.

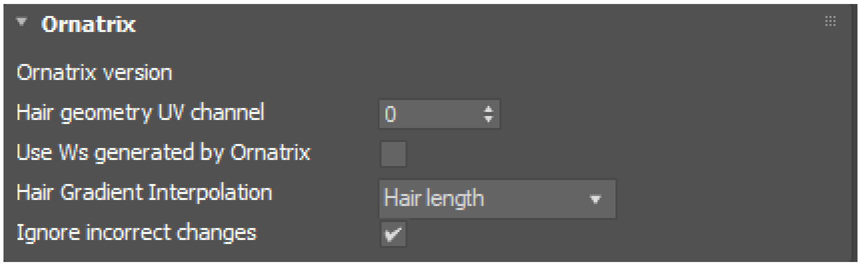


Figure 1: Ornatrix

Ornatrix Parameters

Hair Geometry UV Channel - UV channel generated by Ornatrix.

Use Ws Generated by Ornatrix - When the 'W coordinate' map is set as an input of a gradient texture for hair material, this will use Ornatrix generated Ws instead of automatic Ws.

Hair Gradient Interpolation - When using automatic Ws (generated by Octane), this specifies how the hair segment W coordinates are calculated from the gradient texture.

Ignore Incorrect Changes - Workaround to avoid repeated Viewport updates when some Ornatrix objects are selected.

Hair & Fur

A 3DS Max Modifier for creating hair and fur.

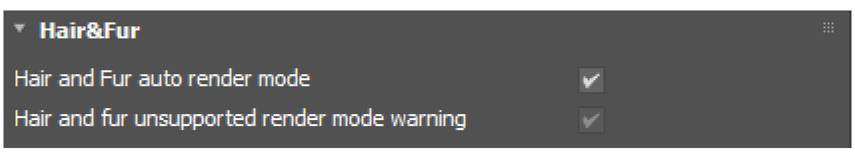


Figure 1: Hair & Fur

Hair & Fur Parameters

Hair and Fur Auto Render Mode - Automatically change Hair and Fur Effect render mode settings to supported mode ('Geometry').

Hair and Fur Unsupported Render Mode Warning - Display a warning if Hair and Fur Effect render mode setting is not supported (currently only 'Geometry' or 'Buffer' are supported).

Account

OctaneRender® requires authentication with its designated license key and requires internet access on its initial launch. OctaneRender® requests your OTOY® credentials and attempts to retrieve an available license from the OctaneRender Live™ server. Under the Account tab, you will find:

- OctaneLive Account

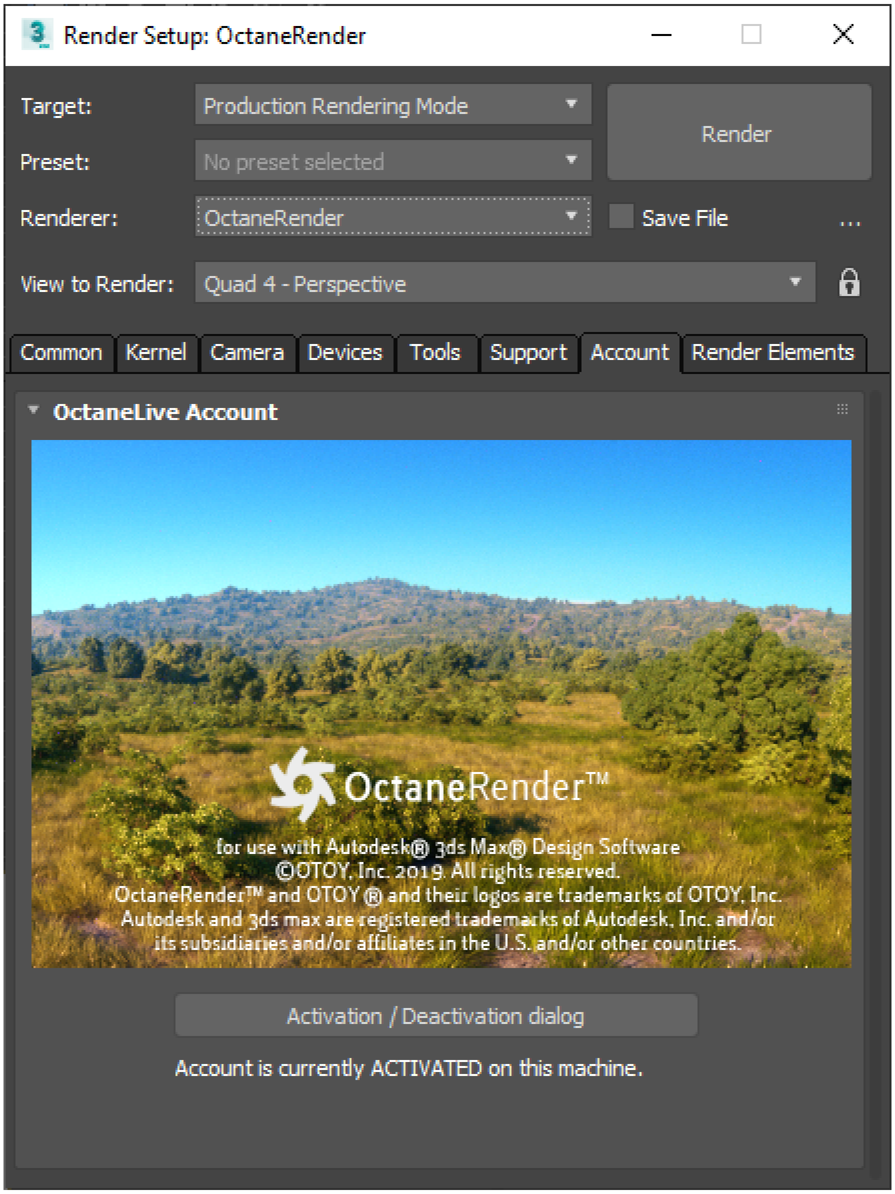


Figure 1: Account Tab and OctaneLive Account from the Render Setup window

OctaneRender® requires one available Standalone license on OctaneRender Live™, while plugins require one available standalone license plus one available license for that specific plugin. Standalone licenses are bound to one machine, which means you can share the Standalone license across multiple plugins running on that machine. You can also run multiple instances of Standalone or a plugin on a single machine using the same license.

Closing the application releases the OctaneRender® license, similar to a floating license scheme. Standalone edition just releases the standalone license, while plugins release both Standalone and their respective license. In either case, licenses are released if there is not another instance of Standalone or a plugin making use of that specific license. Note the distinctions below between just closing the applications and signing out of the applications.

Standalone Edition

- **Exiting Or Closing The Application**
  - Releases the Standalone license key, except when there is a plugin edition that is also open and still bound to that Standalone license key.
- **Signing Out**
  - Releases all OctaneRender® license keys bound on that machine. If other OctaneRender® instances are still running, you will be asked to close them before it can sign out and release all of the licenses.

Plugin Edition

- **Exiting Or Closing The Application**
  - Releases the license keys bound to the plugin. This includes the Standalone license Key, unless the Standalone edition is open or other plugins are open and their keys are still bound to the same Standalone license key on the same machine.
- **Signing Out**
  - Releases all OctaneRender® license keys bound on that machine. If there are other instances of OctaneRender® still running, you will be asked to exit those before it can sign out and release all OctaneRender® licenses.

Deactivating from the Octane live licenses administration page is not necessary as this is done automatically by the application. This lets you use OctaneRender® somewhere else without deactivating any licenses. Licenses in use by older versions have the Deactivate button next to them if you need to release the license.

If the application didn't close properly from a crash or other circumstances, there is a chance the license isn't released. If the same machine accesses the same keys, this is not a problem as the same keys are still bound. The problem arises when you use OctaneRender® on another machine, as the keys are still bound to the previous machine. In such cases, failsafe web deactivation unbinds the keys.

OctaneLive Account Parameter

Activation/Deactivation Dialog - Opens an Activation Status window to allow you to either activate or deactivate an account. If activating, an additional sign-in window will appear.

Render Elements

Render elements/passes are processed differently by different render engines. Since OctaneRender® is a separate render engine, it also uses its own set of render elements.

Using The Passes

To view and save render passes, go to **Rendering** > **Render Setup** > **Render Elements** and add the passes to include in the render.

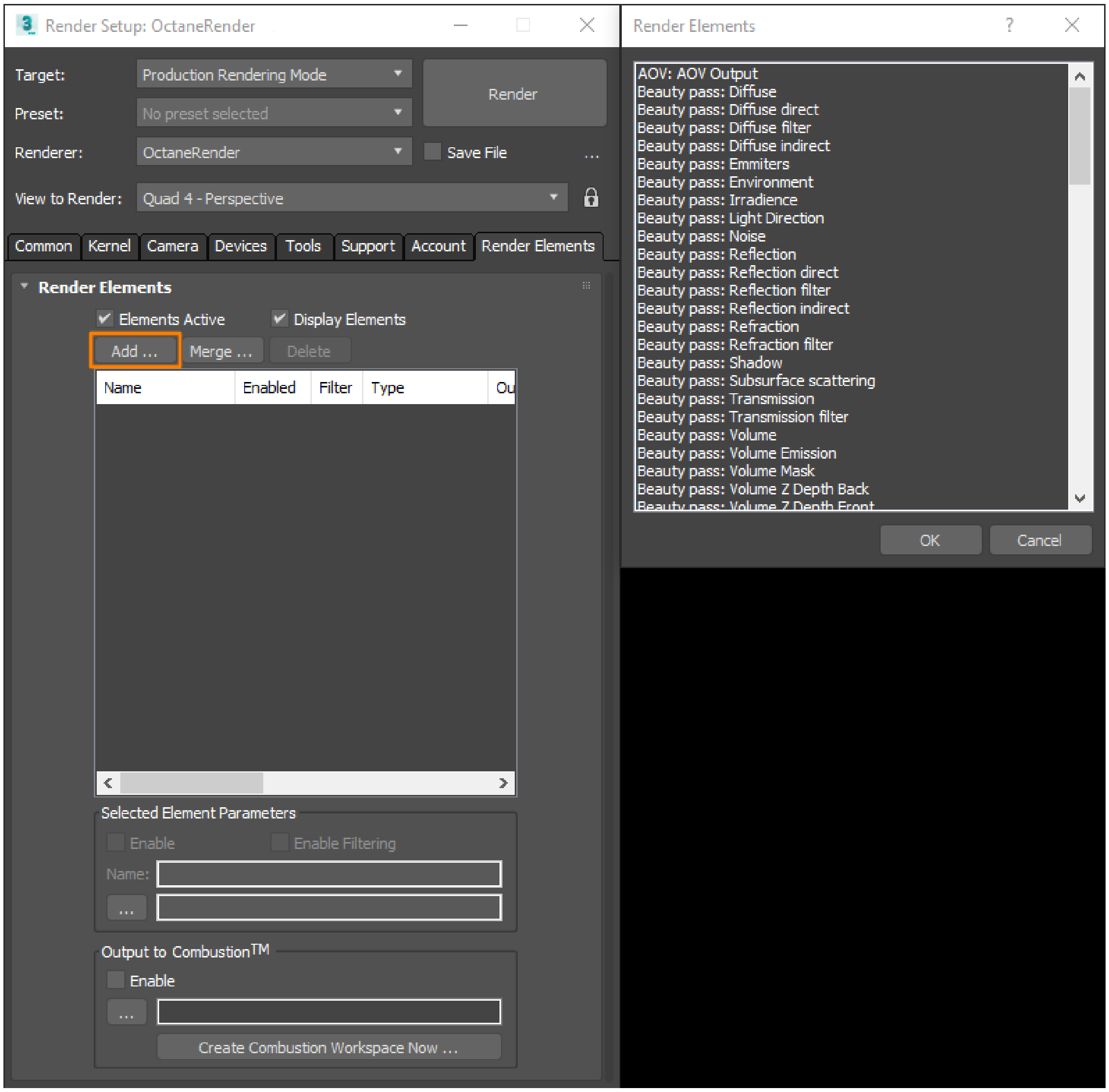


Figure 1: Accessing the OctaneRender® Render Elements

Make sure that the selected passes are enabled.

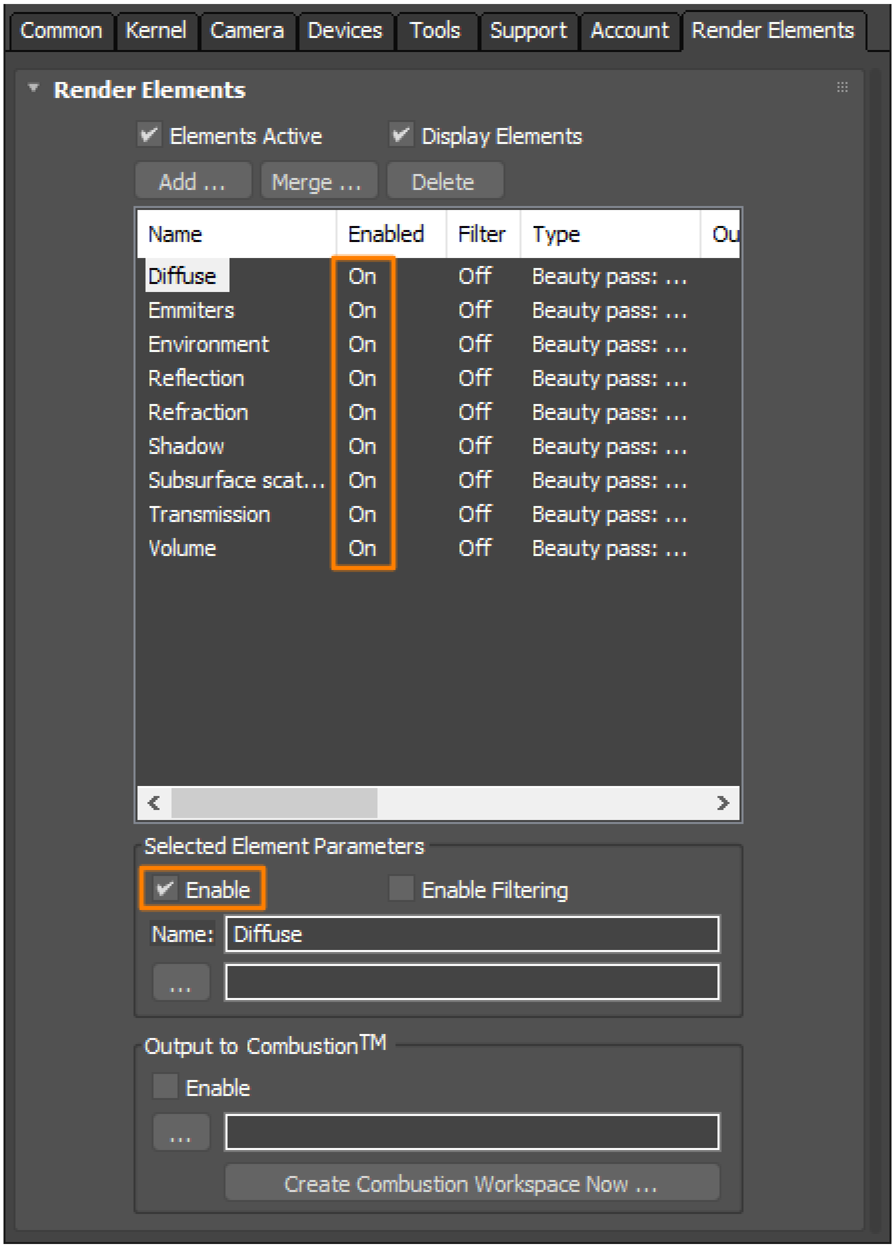


Figure 2: Enabling the Render Elements

Go back to the **Kernel** window and make sure to enable the **Alpha Channel**.

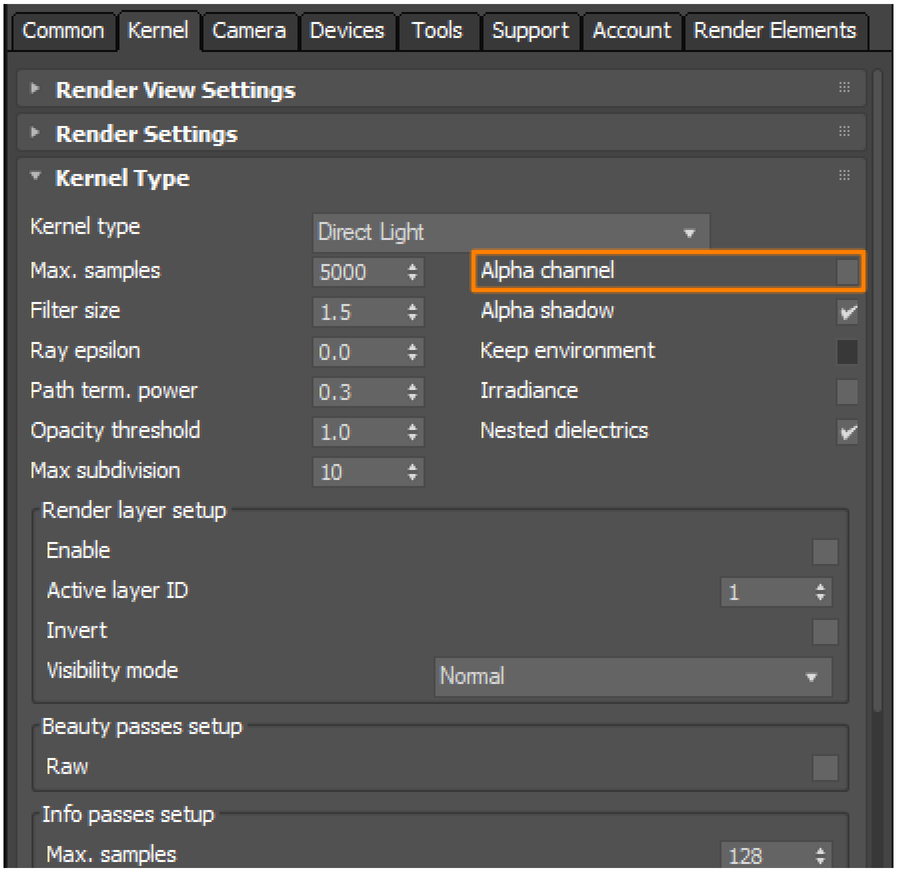


Figure 3: Activating the Alpha Channel[1] parameter in the Kernel settings

Next, open **OctaneRender Viewport** and use the **Select Multiple Passes** button to **Import Passes Selection from RenderElements.** You can also add or subtract other render passes to the viewport here. Keep in mind for file output renders, you need to make sure you export your passes back to **Render Setup > Render Elements**. You can easily do this with the **Export Passes Selection to Render Elements**.

[1] A greyscale image used to determine which areas of a texture map are opaque and which areas are transparent.

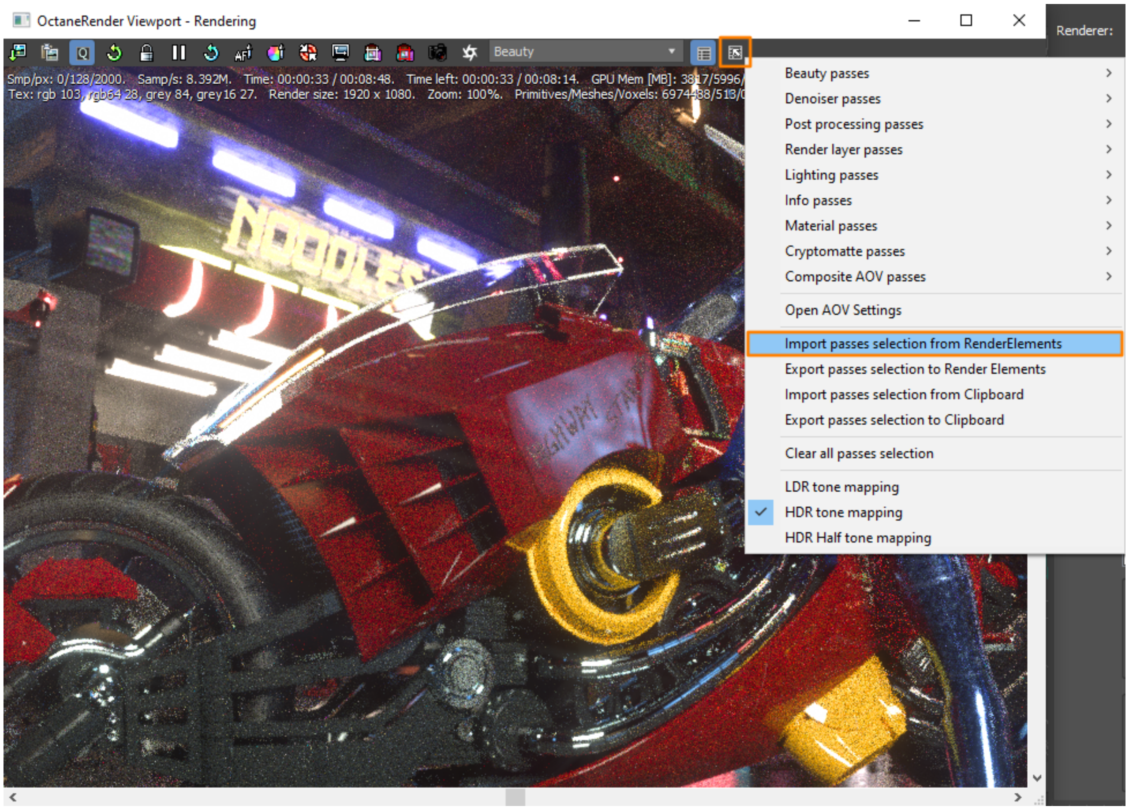


Figure 4: Import Render Elements via the OctaneRender Viewport

Use the **Passes Dropbox** to select and view each render pass. If you want to save out each pass manually, you can use the **Port To Max Framebuffer** button.

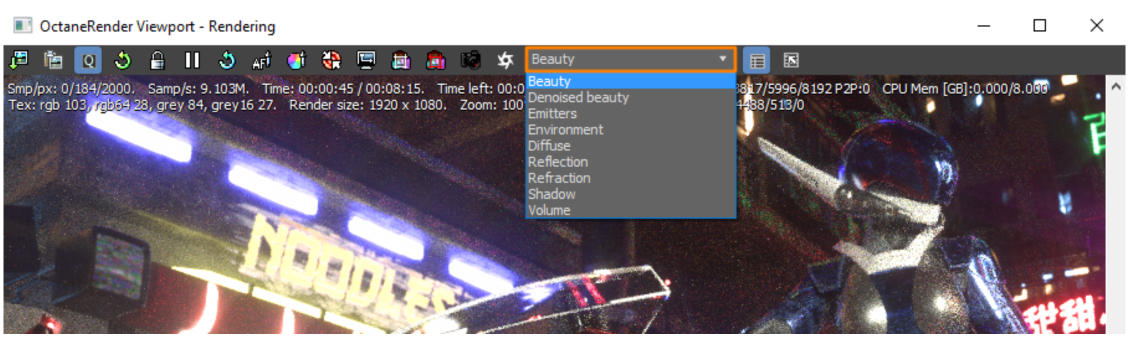


Figure 5: Passes Dropbox used to view each pass

Rendering And Effects

Additional rendering tools are explained in this section to help you control settings related to:

- Object Visibility
- **Batch Rendering**[1]
- Rendering for **VR**[2]
- **Texture Baking**[3]
- **Deep Image**[4] Rendering
- **Render Layers**[5]

Effects include a large range of features and tools. The OctaneRender® for 3DS Max® plugin include tools for:

- **Post Processing**[6] Effects
- Hair and Fur
- **Displacement**[7]
- **Instancing**[8]
- Octane **Proxy**[9]
- **Motion Blur**[10]

---

[1]The process of assigning sequential portions of frames to be rendered across multiple systems.

[2]Immersively engaging and experiencing depth perception in a three dimensional scene through stereo vision goggles and head-mounted displays.

[3]A process in which scene lighting is "baked" into a texture map based on an object's UV texture coordinates. The resulting texture can then be mapped back onto the surface to create realistic lighting in a real-time rendering environment. This technique is frequently used in game engines and virtual reality for creating realistic environments with minimal rendering overhead.

[4]Renders frames with multiple depth samples in addition to typical color and opacity channels.

[5]Render layers allow users to separate their scene geometry into parts, where one part is meant to be visible and the rest of the other parts “capture” the side effects of the visible geometry. The layers allow different objects to be rendered into separate images where, in turn, some normal render passes may be applied. The Render layers are meant for compositing and not to hide parts of the scene.

[6]Effects such as Bloom and Glare that are applied after a scene has been rendered.

[7]The process of utilizing a 2D texture map to generate 3D surface relief. As opposed to bump and normal mapping, Displacement mapping does not only provide the illusion of depth but it effectively displaces the actual geometric position of points over the textured surface.

[8]Instancing an object means taking a single imported mesh object, such as an OBJ or an FBX and making multiple copies, each of which can be placed in different parts of the scene. This saves an enormous amount of computational resources because only a single object is loaded into the scene.

- **ORBX**[1] Proxy
- Octane Vectron

Effects

Effects include a large range of features and tools. The OctaneRender® for 3DS Max® plugin categorizes **Volume** mediums, hair and fur, instancing, and motion blur as effects components.

Figure 1: Volume Effect

[1]The ORBX file format is the best way to transfer scene files from 3D Authoring software programs that use the Octane Plug-in such as Octane for Maya, Octane for Cinema 4D, or OctaneRender Standalone. This format is more efficient than FBX when working with Octane specific data as it provides a flexible, application independent format. ORBX is a container format that includes all animation data, models, textures etc. that is needed to transfer an Octane scene from one application to another.

Figure 2: Instances and Scattering[1]

Post-Processing Effects

You can access the post-processing effects in OctaneRender® from the **Camera** tab in the **Render Setup** window, or from the **Modify** panel if you add an OctaneRender® **Camera** to the scene.

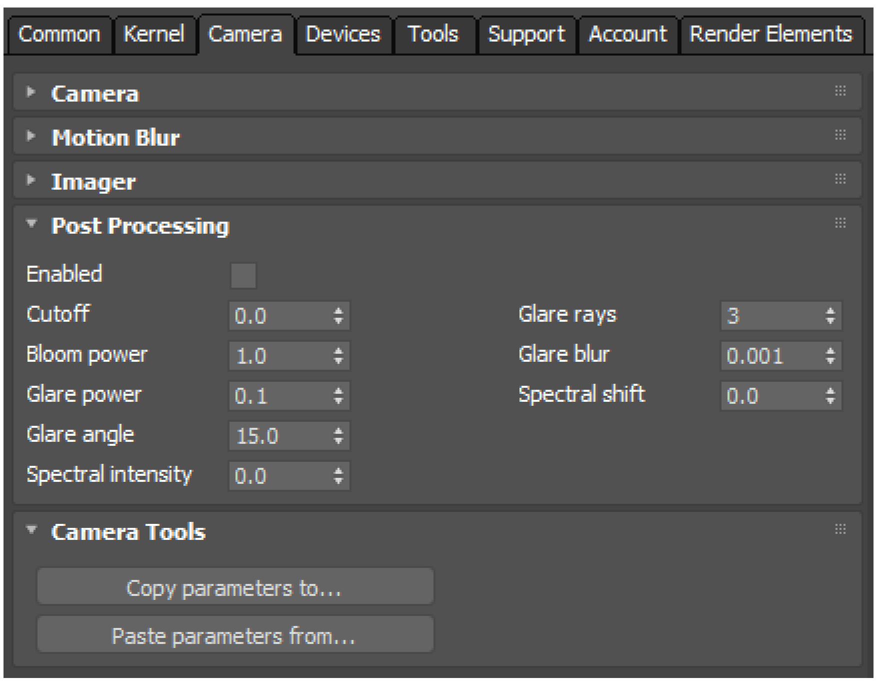
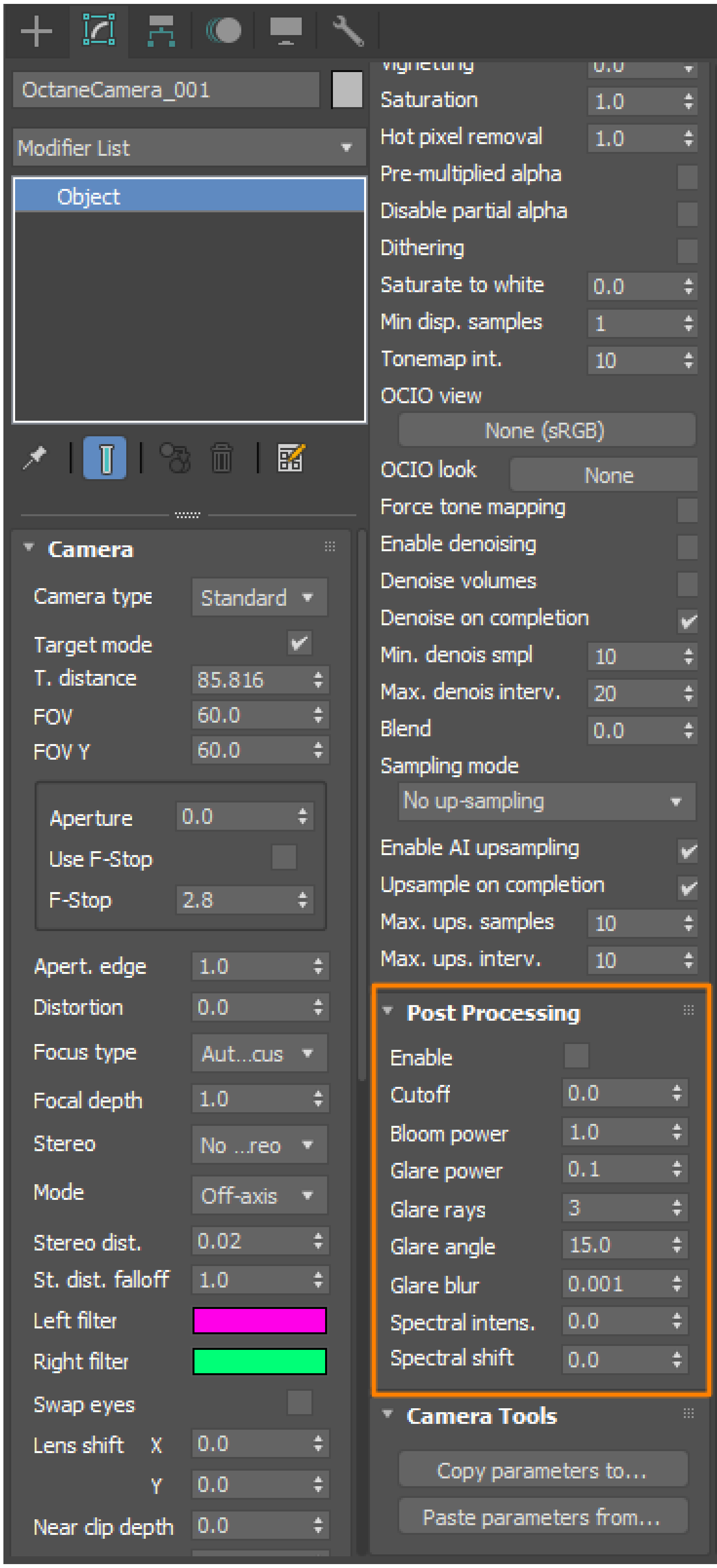


Figure 1: Render Setup window

---

[1] Defines how fast light gets scattered when traveling through the medium.


OctaneCamera_001
Modifier List
Object
Camera
Camera type
Standard
Target mode
T. distance
85.816
FOV
60.0
FOV Y
60.0
Aperture
0.0
Use F-Stop
F-Stop
2.8
Apert. edge
1.0
Distortion
0.0
Focus type
Aut...cus
Focal depth
1.0
Stereo
No ...reo
Mode
Off-axis
Stereo dist.
0.02
St. dist. falloff
1.0
Left filter
Right filter
Swap eyes
Lens shift X
0.0
Y
0.0
Near clip depth
0.0
Saturation
1.0
Hot pixel removal
1.0
Pre-multiplied alpha
Disable partial alpha
Dithering
Saturate to white
0.0
Min disp. samples
1
Tonemap int.
10
OCIO view
None (sRGB)
OCIO look
None
Force tone mapping
Enable denoising
Denoise volumes
Denoise on completion
Min. denois smpl
10
Max. denois interv.
20
Blend
0.0
Sampling mode
No up-sampling
Enable AI upsampling
Upsample on completion
Max. ups. samples
10
Max. ups. interv.
10
Post Processing
Enable
Cutoff
0.0
Bloom power
1.0
Glare power
0.1
Glare rays
3
Glare angle
15.0
Glare blur
0.001
Spectral intens.
0.0
Spectral shift
0.0
Camera Tools
Copy parameters to...
Paste parameters from...

fects

Figure 2: Octane Camera Modify panel

Post-Processing Parameters

Enable - Enable post-processing effects. Post-processing is disabled by default.

Figure 3: No post-processing

Cutoff - Set the threshold for the effect.

Bloom Power - Controls the size of the emitter's glow and the size of the sun's or concentrated light's halo on reflective **Glossy**[1] materials.

[1]The measure of how well light is reflected from a surface in the specular direction, the amount and way in which the light is spread around the specular direction, and the change in specular reflection as the specular angle changes. Used for shiny materials such as plastics or metals.

Figure 4: Bloom Power

Glare Power - Controls the size of the emitter's visible rays and the glare from reflective Glossy materials.

Figure 5: Glow and Glare Power

Glare Rays - Controls the number of radiated or reflected visible rays.

Glare Angle - Adjusts the glare direction relative to the **Object**.

Glare Blur - Controls the glare's sharpness. Smaller values result in a crisp linear glare, and larger values with result in a softer glare.

Figure 6: Glare Blur

Spectral Intensity - Adjusts the intensity distribution of the rays across a source, affecting the radiant energy's brightness.

Spectral Shift - Adjusts the spectrum's displacement as the soure's emitted light frequency changes. The shift is evident by a color change, similar to the Doppler effect, as the distance traveled by the ray from its source increases or decreases.

Figure 7: Glow, Glare, and Spectral

Volume Mediums

The best way to work with **Volumes** in 3DS Max® is to use **VDB**[1] files. VDB is a generic volume format used to create effects such as smoke, fog, vapor, and similar gaseous objects. VDBs are generated and exported from other 3D software packages such as Houdini®. VDB files are available for download online at www.openvdb.org/download. VDBs can be a single frame or an animated file sequence.

---

[1] Dreamworks’ open-source C++ library housing the data structures and tools implementation for storing and manipulating volume data, like smoke and other amorphous materials. The purpose of OpenVDB is mostly to have an efficient way to store volumetric data in memory and on disk. It has evolved into a more general toolkit that also lets you accomplish other things, such as fracturing volumes, converting meshes to volumes and vice versa. However, it does not include a computational fluid dynamics solver, and therefore it cannot procedurally generate smoke or fire. OpenVDB is fully integrated as a library in OctaneRender. For more information about OpenVDB, please see http://www.openvdb.org/.

To use a VDB in OctaneRender® for 3DS Max®, add an **Octane Volume** to the scene. You can find OctaneRender® Volumes by clicking on the **Octane** sub-menu and going to the **Create** panel.

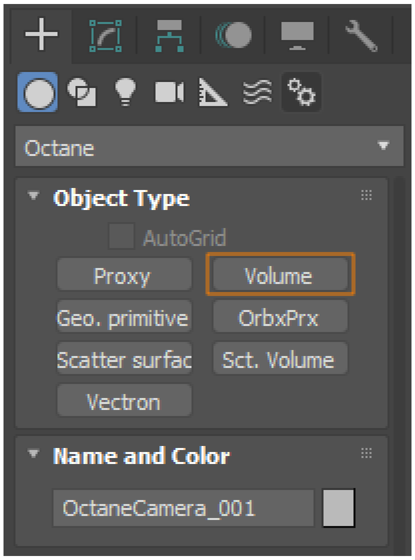


Figure 1: Accessing the Volume from the Create panel

After you add a Volume to the scene, you can import a VDB file into the **File** parameter of the Volume.

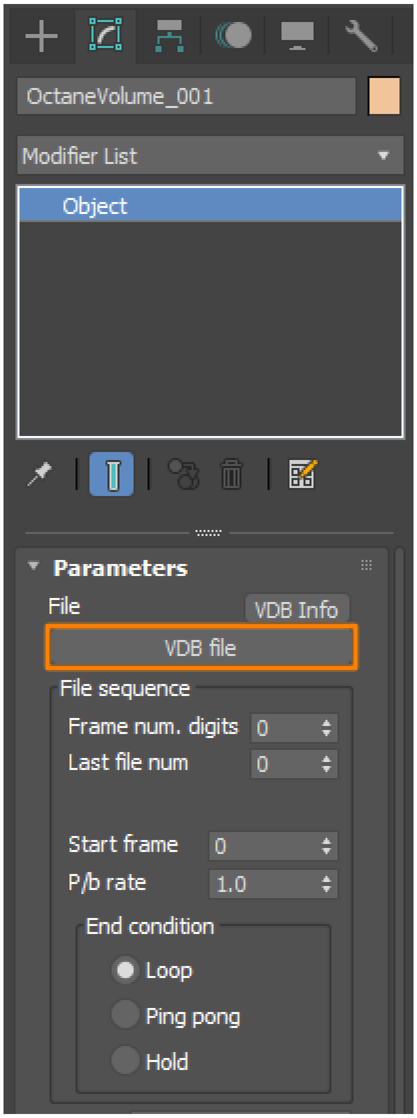


Figure 2: Adding a VDB file to the Volume

In order to gain maximum control over the Volume's look, add a Volume medium to the OctaneRender® Volume's **Medium** parameter. You can find the Volume medium in the **Slate Material**[1] **Editor** under the OctaneRender® **Maps** category.

[1]The representation of the surface or volume properties of an object.

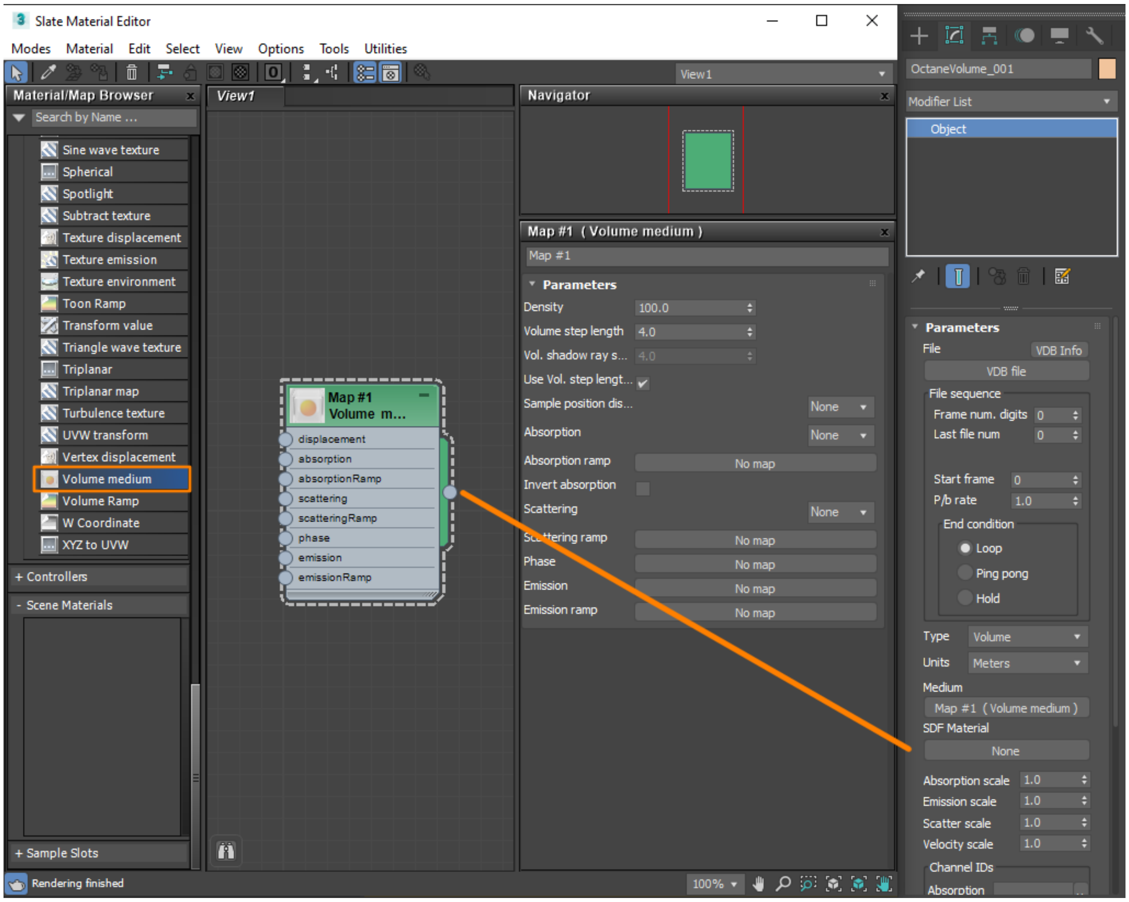


Figure 3: Adding a Volume medium to the Octane Volume's Medium parameter

The Volume medium parameters are covered in more detail in the **Volume Medium**[1] topic under the **Materials**[2] section in this manual.

Octane Volume Parameters

File - Select a VDB file to import.

File Sequence - Controls an animated VDB file.

---

[1]A shading system designed to render volumes such as smoke and fog.

[2]A set of attributes or parameters that describe surface characteristics.

Type - Defines your VDB file as either Volume or Volume SDF.

Units - Controls the scale of an imported VDB volume.

Medium - Connects a Volume medium to control the Volume's look.

SDF Material - Provides support for surface material rendering on Volumes.

Absorption[1]**, Emission, Scatter, Velocity Scale** - These parameters act as multipliers for the corresponding parameters in a Volume medium.

Channel IDs - Accepts any attached data from the imported VDB file.

Dummy Size - Determines the size of the Octane Volume placeholder visible in the scene.

Note: OctaneRender® for 3DS Max® also provides support for PhoenixFD volumes. You must add the Octane Volume modifier on top of the PhoenixFD object in the Modifier stack.

Hair And Fur

OctaneRender® for 3DS Max® can render the 3DS Max®-native **Hair And Fur** modifier. The hair renders when the OctaneRender® **Material**[2] is applied to the underlying **Object**.

[1] Defines how fast light is absorbed while passing through a medium.

[2] The representation of the surface or volume properties of an object.

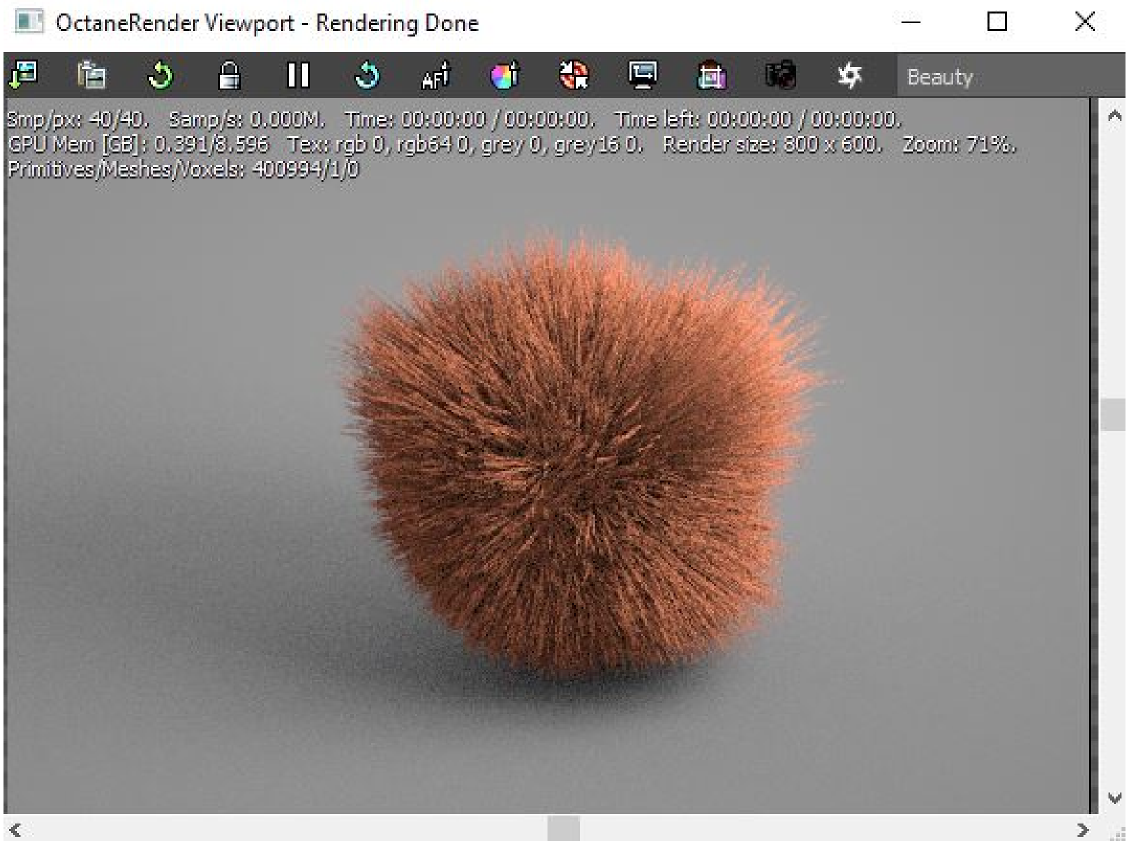


Figure 1: The Hair And Fur modifier applied to a sphere with an orange-colored Diffuse material[1]

At first, it appears as one big **Mesh**, similar to what we would get if we convert this Object to a Mesh from the Hair And Fur's **Tools** section.

To get the best results with rendering hair and fur in OctaneRender®, we recommend converting Hair And Fur to a Mesh.

[1] Used for dull, non-reflecting materials or mesh emitters.

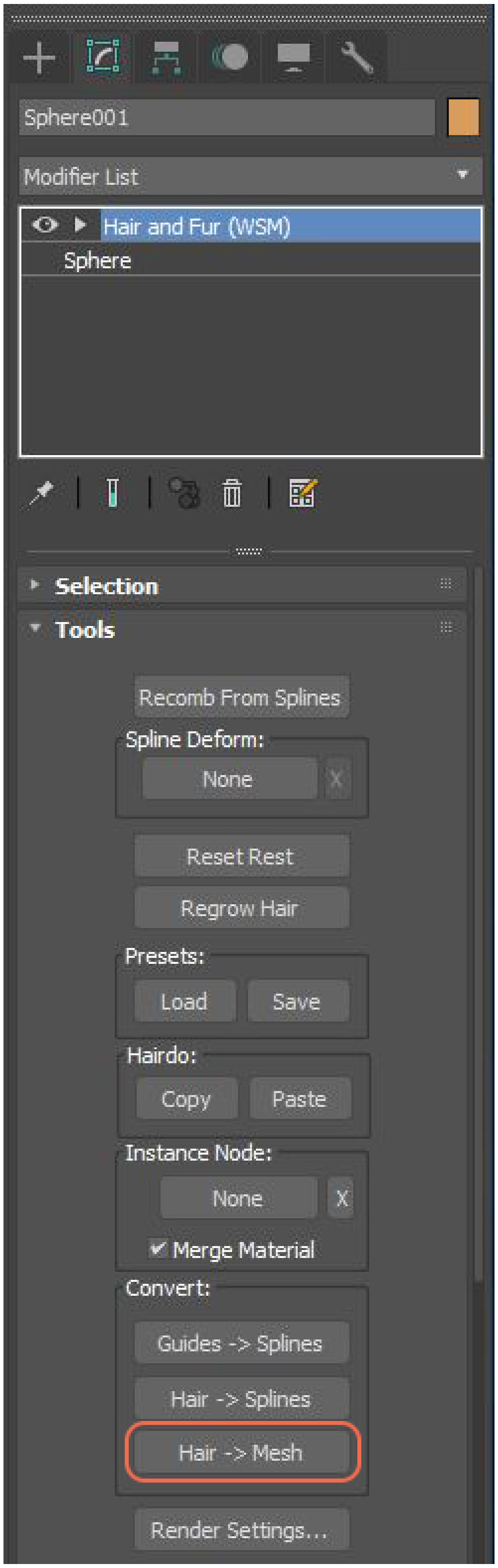

Sphere001
Modifier List
Hair and Fur (WSM)
Sphere
Selection
Tools
Recomb From Splines
Spline Deform:
None
X
Reset Rest
Regrow Hair
Presets:
Load
Save
Hairdo:
Copy
Paste
Instance Node:
None
X
Merge Material
Convert:
Guides -> Splines
Hair -> Splines
Hair -> Mesh
Render Settings...

Figure 2: Converting Hair And Fur to a Mesh

Converting hair to a Mesh lets you apply a **Mapping** texture to the hair or fur. This is done by connecting the Mapping texture to the **Diffuse**[1] slot of the **Material** applied to the Mesh. The converted Hair mesh object then takes on the pattern or appearance of the mapped texture, like an **RGB** image texture or a **Sine Wave** texture.

You can also use a **Gradient** texture for the **Diffuse** slot of an OctaneRender® material. However, you have to generate the Hair geometry with a third-party plugin other than the 3DS Max® Hair And Fur modifier. Using a third-party plugin with the ability to set custom Ws is necessary. Deploying in such a way lets you use the OctaneRender® **Gradient** texture with a **W Coordinate** texture, and connect the W Coordinate to the Gradient texture's **Input** slot. This orients the Gradient across the length of the hair strands. You can find more information about the W Coordinate in the **"W Coordinate" on page 193** topic in this manual under the Texture Modifiers section.

Note: OctaneRender® for 3DS Max® also supports third-party hair plugins, which provide easier access to modeling hair and fur, like Ornatrix® and Hair Farm, and Forest Pack® for grass rendering.

Displacement

Displacement[2](Texture Displacement) mapping utilizes a **2D** texture map to generate 3D surface relief. As opposed to **Bump** and **Normal** mapping, Displacement mapping provides the illusion of depth, and it displaces the actual geometric position of points over the textured surface.

Displacement mapping is covered in more detail under the **Material**[3] **Editor > Textures**[4] section in this manual.

---

[1]Amount of diffusion, or the reflection of light photons at different angles from an uneven or granular surface. Used for dull, non-reflecting materials or mesh emitters.

[2]The process of utilizing a 2D texture map to generate 3D surface relief. As opposed to bump and normal mapping, Displacement mapping does not only provide the illusion of depth but it effectively displaces the actual geometric position of points over the textured surface.

[3]The representation of the surface or volume properties of an object.

[4]Textures are used to add details to a surface. Textures can be procedural or imported raster files.

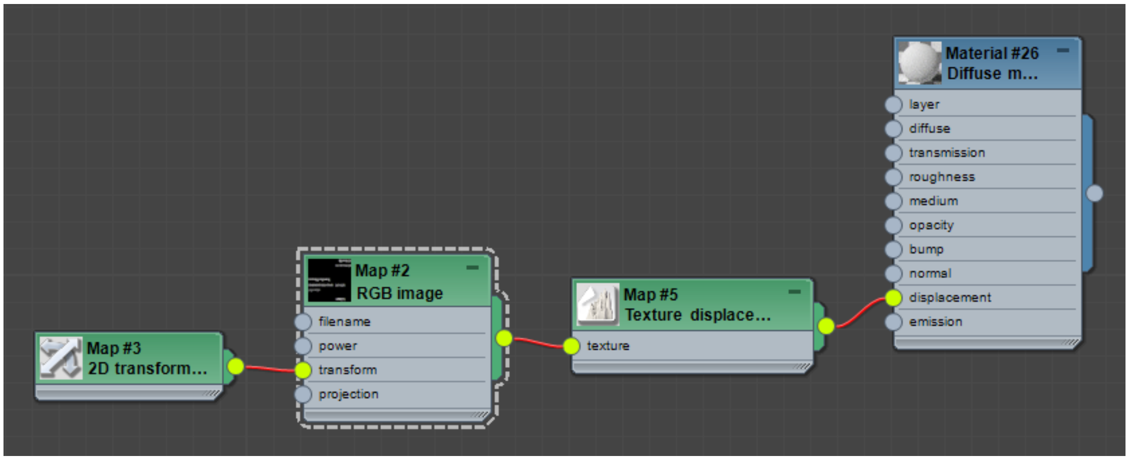


Figure 1: Displacement(Texture Displacement) mapping chain

Instancing

OctaneRender® supports two types of geometry:

Regular - This is the default geometry type for all **Objects** in a scene.

Instancing[1] Geometry (Movable Proxy[2]) - If the scene contains several Objects with the same geometry, OctaneRender® keeps one copy in memory and uses it as reference for the other copies. The renderer reduces memory consumption in exchange for rendering speed. Instancing allows OctaneRender® to transform geometry in real-time without geometry recalculation, which is useful when creating animations.

You can change the geometry type from the **Octane Object Properties** window.

---

[1] Instancing an object means taking a single imported mesh object, such as an OBJ or an FBX and making multiple copies, each of which can be placed in different parts of the scene. This saves an enormous amount of computational resources because only a single object is loaded into the scene.

[2] An object saved as a separate file with the purpose of being reused in larger scenes. This is used to minimize any addition to the total polygon count in the scene, especially if the scene requires the same object to appear several times. If used in conjunction with instancing, Proxies help keep very large scenes from reaching polygon limits and also keeps the relative file size of the main project file manageable.

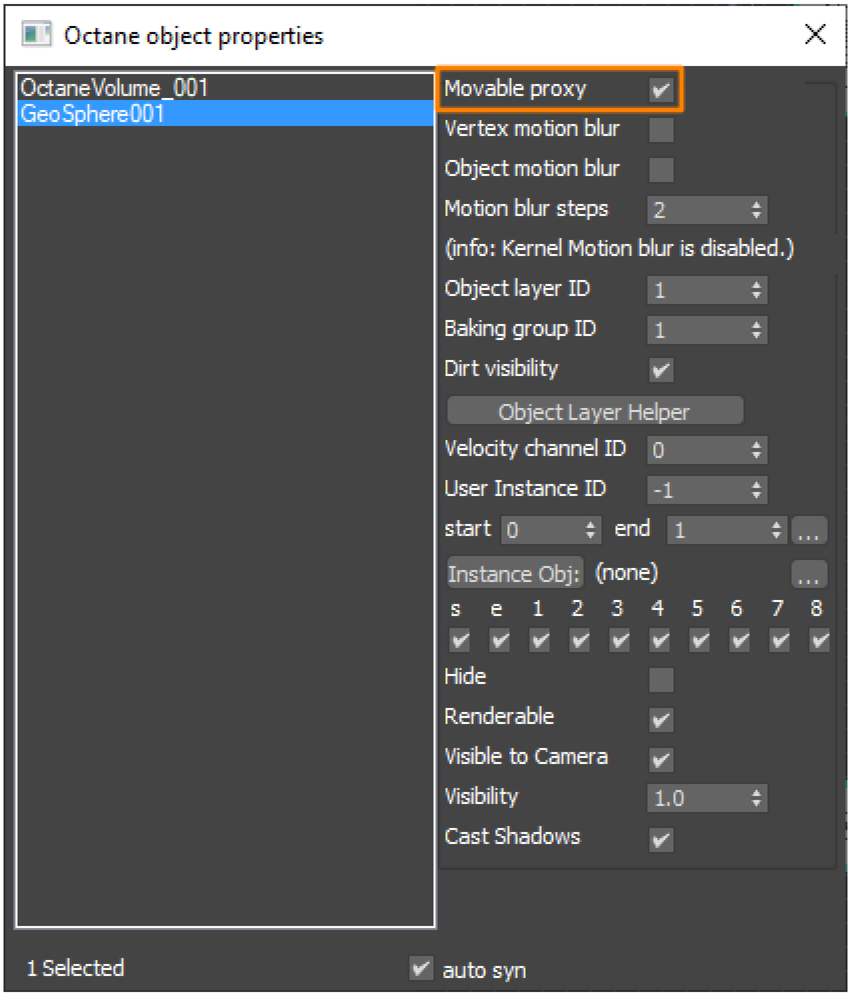


Figure 1: Changing the geometry type to a Movable Proxy in order to create instances

Octane Proxy

An **Octane Proxy**[1] is an **Object** saved as a separate file for reuse in larger scenes. This minimizes any additions to the scene's total polygon count, especially if the scene requires the same Object to appear several times. When used with instancing in 3DS Max®, Octane Proxies help keep large scenes from reaching polygon limits, and they also keep the relative file size of the main project file manageable.

To start using an Octane Proxy, create the Object, then export it with the **Octane Proxy Exporter**.

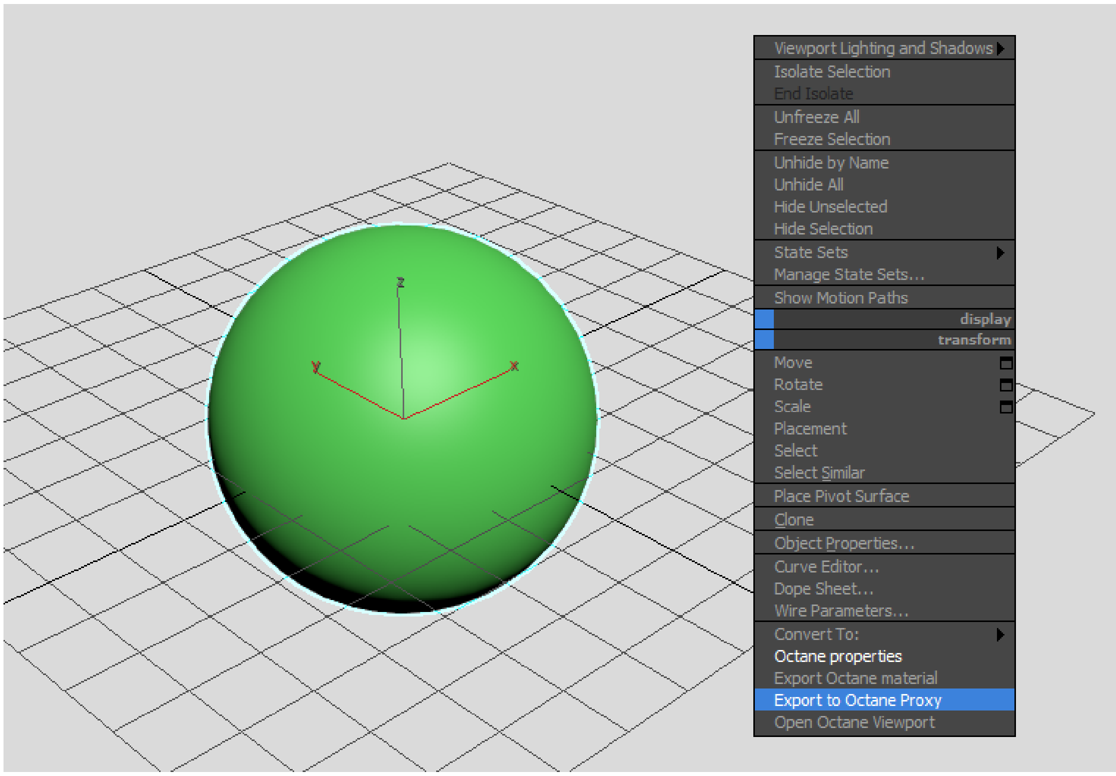


Figure 1: Creating an Octane Proxy from the Quad menu

[1] An object saved as a separate file with the purpose of being reused in larger scenes. This is used to minimize any addition to the total polygon count in the scene, especially if the scene requires the same object to appear several times. If used in conjunction with instancing, Proxies help keep very large scenes from reaching polygon limits and also keeps the relative file size of the main project file manageable.

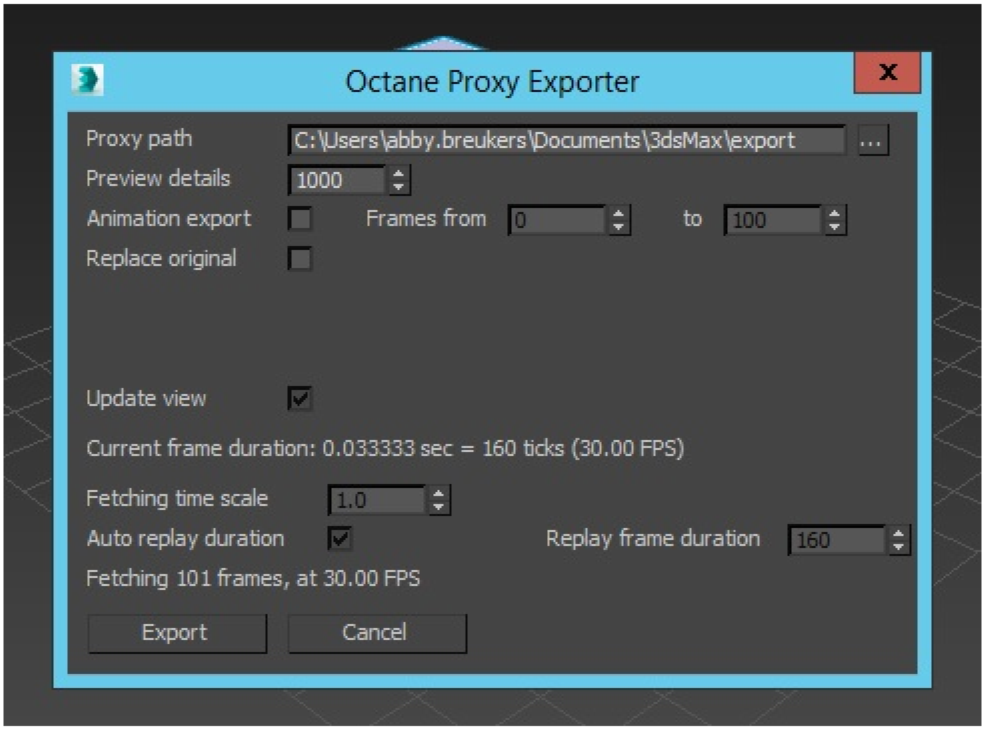


Figure 2: The Octane Proxy export options

An Octane Proxy is saved as an OCTPRX file inside the specified path's folder. After creating the Octane Proxy, you can use it in the main scene.

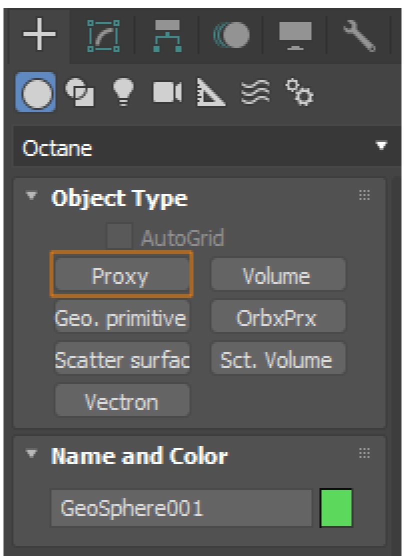


Figure 3: Creating an Octane Proxy

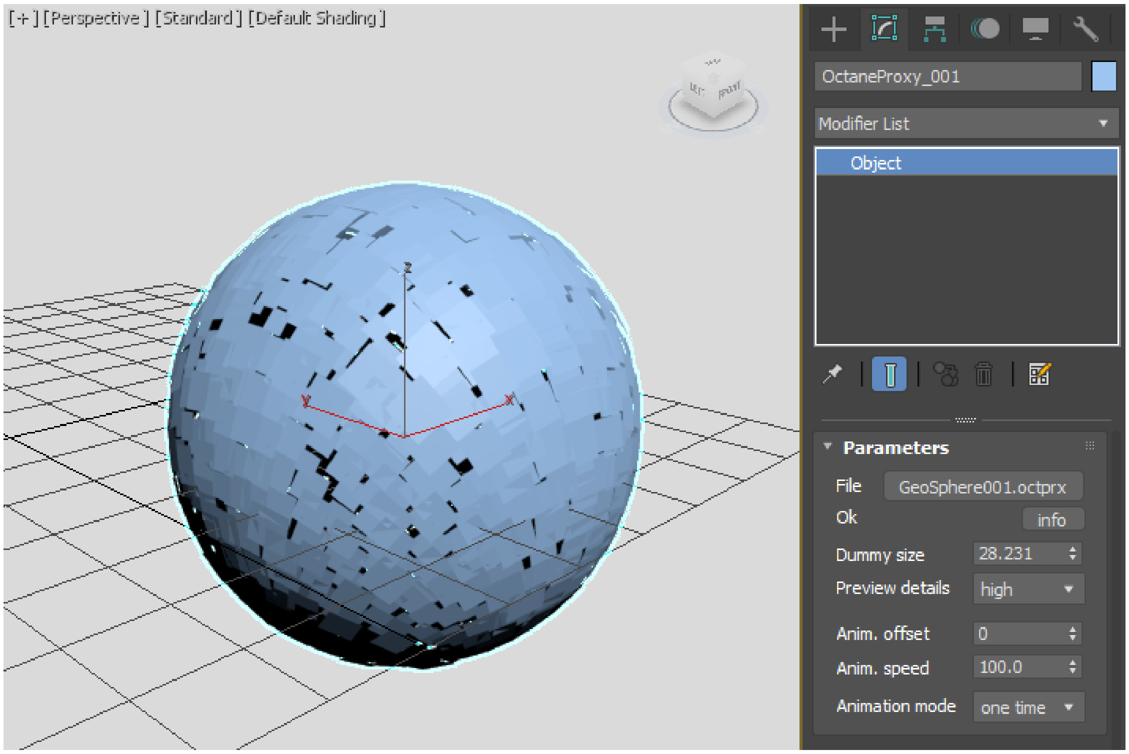


Figure 4: Loading an Octane Proxy back into the scene

From this point, you can instance the Octane Proxy. Make sure the Octane Proxy is set up as a movable proxy by right-clicking on it to open the **Octane Properties** dialog box and select **Movable Proxy**. You can now clone the Object as instances without increasing its footprint in **GPU**[1] memory.

Motion Blur

---

[1]The GPU is responsible for displaying graphical elements on a computer display. The GPU plays a key role in the Octane rendering process as the CUDA cores are utilized during the rendering process.

You can activate **Motion Blur**[1] from the **Render Setup** window. **Motion Blur Duration** is measured by frame duration. A value of **1.0** means Motion Blur Duration is equal to the frame duration.

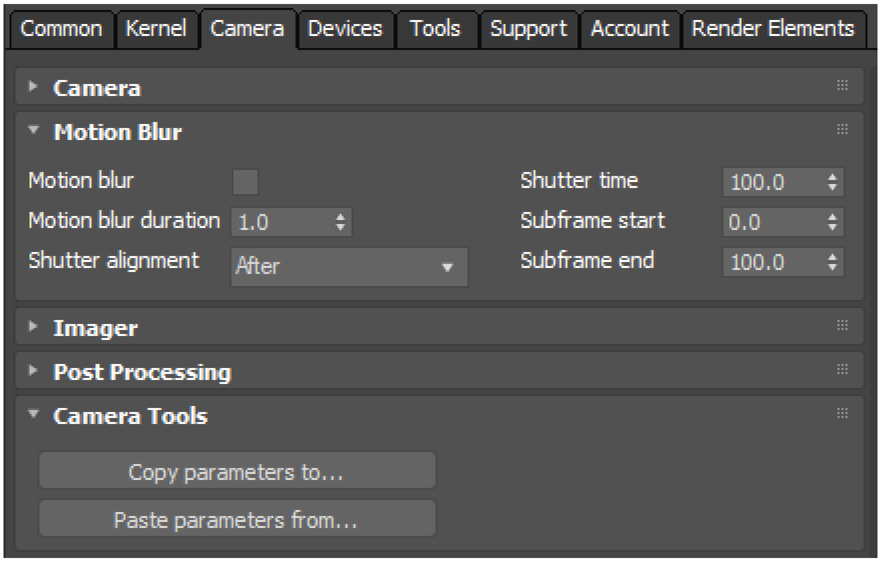


Figure 1: Accessing the Motion Blur settings in the Render Setup window

Every **Object** with motion blur should enable **Object Motion Blur** in its **Octane Properties**.

[1] An optical phenomenon that occurs when a camera's shutter opens and closes too slowly to capture movement without recording a blurring of the subject.

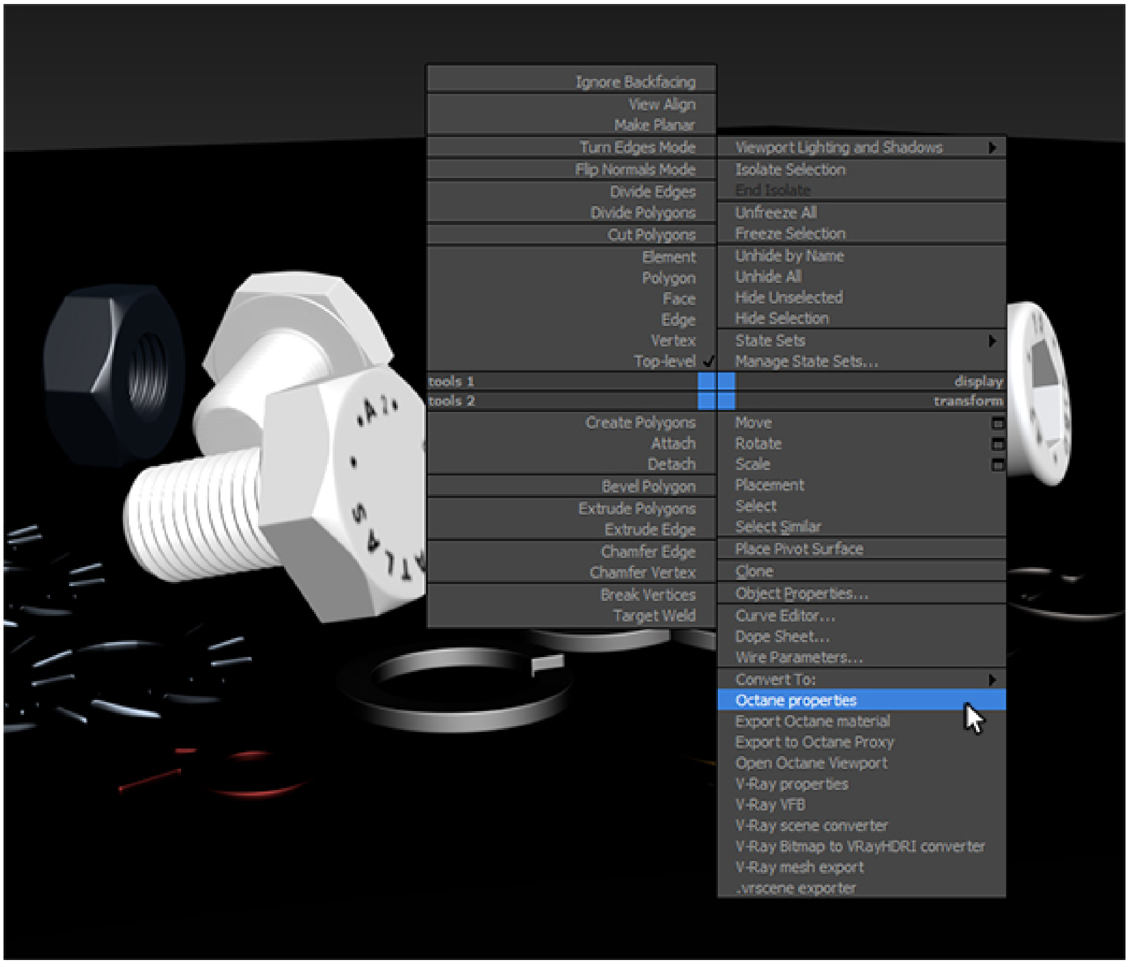


Figure 2: Accessing the Octane Object Properties by right-clicking on a scene Object

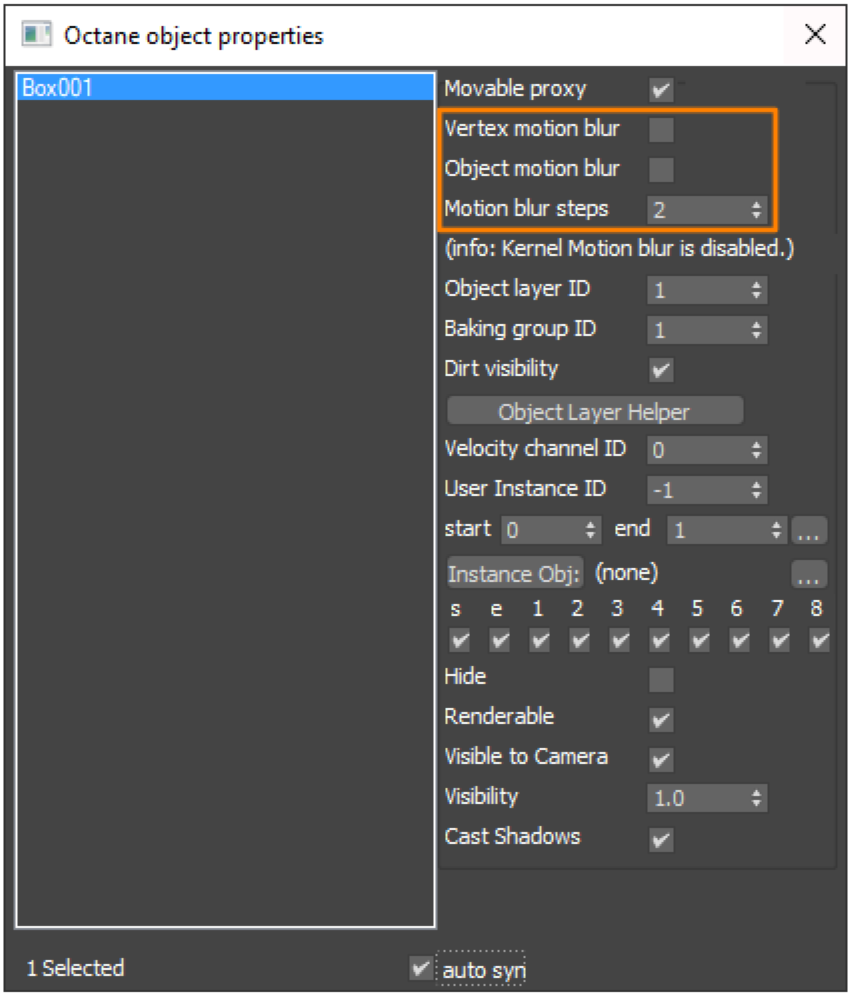


Figure 3: Activating motion blur from the Octane Object Properties window

If an Object doesn't have any deformation animation, you should enable **Movable Proxy**[1] and **Object Motion Blur**. If you have fast curved Objects moving, increase the Motion Blur steps.

---

[1]An object saved as a separate file with the purpose of being reused in larger scenes. This is used to minimize any addition to the total polygon count in the scene, especially if the scene requires the same object to appear several times. If used in conjunction with instancing, Proxies help keep very large scenes from reaching polygon limits and also keeps the relative file size of the main project file manageable.

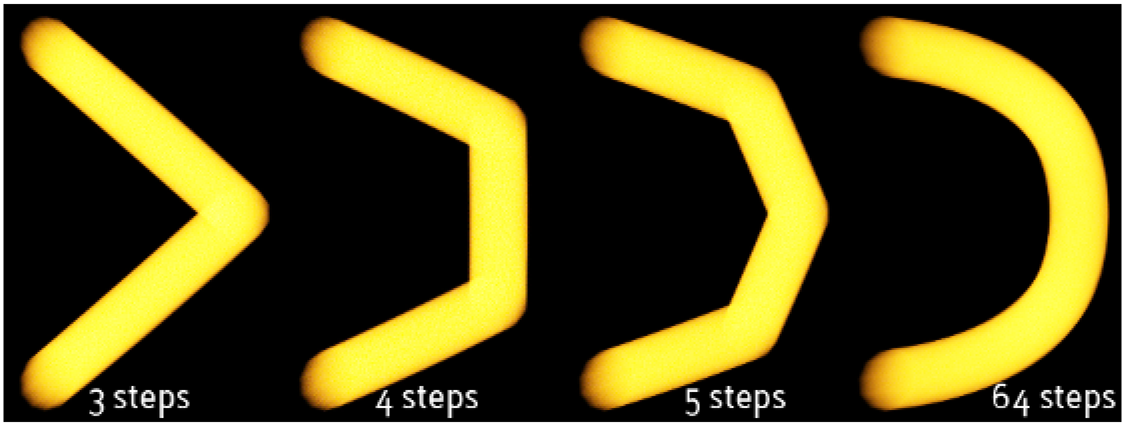


Figure 4: Motion Blur steps comparison

For Object deformation animation, you should enable **Movable Proxy** and **Vertex Motion Blur**. Vertex Motion Blur ignores Motion Blur steps, and always uses **2** steps. This is a render engine limitation.

ORBX Proxy

The **ORBX**[1] **Proxy**[2] object lets you load the ORBX® file format into a 3DS Max® scene. ORBX files can carry full scene data from other sources such as OctaneRender® Standalone, or other Digital Content Creation applications like Maya®, Houdini®, or Cinema 4D®. You can access the ORBX Proxy object from the **Octane** sub-category by clicking on the **Create** tab in the **Command** Panel. The ORBX Proxy shows up in the Viewports as an empty box, but once you open the Render Viewport, the entire ORBX scene renders.

[1] The ORBX file format is the best way to transfer scene files from 3D Authoring software programs that use the Octane Plug-in such as Octane for Maya, Octane for Cinema 4D, or OctaneRender Standalone. This format is more efficient than FBX when working with Octane specific data as it provides a flexible, application independent format. ORBX is a container format that includes all animation data, models, textures etc. that is needed to transfer an Octane scene from one application to another.

[2] An object saved as a separate file with the purpose of being reused in larger scenes. This is used to minimize any addition to the total polygon count in the scene, especially if the scene requires the same object to appear several times. If used in conjunction with instancing, Proxies help keep very large scenes from reaching polygon limits and also keeps the relative file size of the main project file manageable.

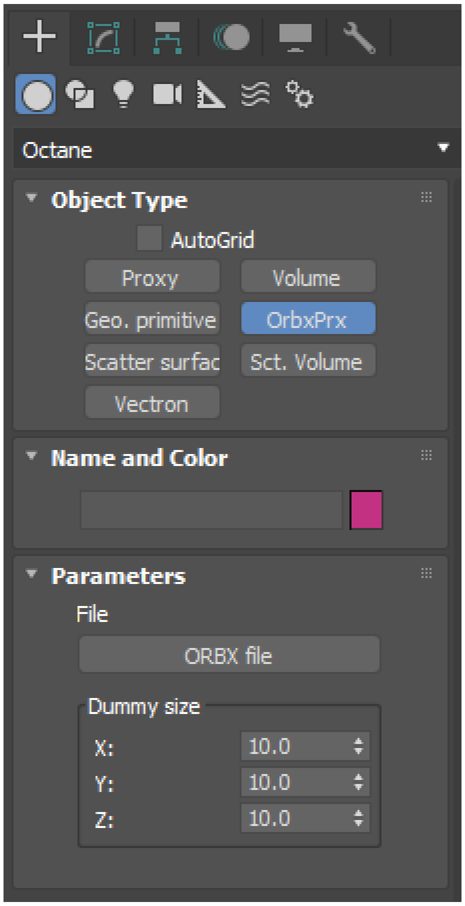


Figure 1: Accessing the ORBX Proxy from the Command Panel

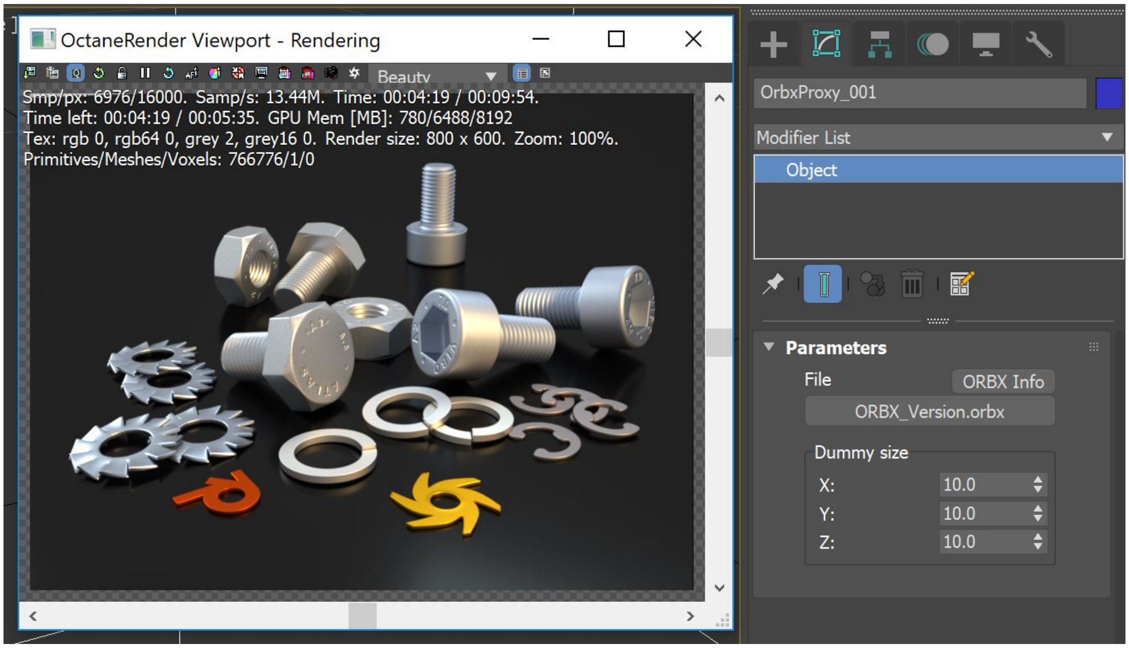


Figure 2: A rendered ORBX Proxy

Vectron

Vectron objects create geometry using the OSL language. Octane Vectron (Vector-Polygon) is a procedural primitive, providing infinite procedurally-generated scenes, **Volumes**, and geometry that bypass **Meshes** and Volumes.

Vectron has a zero-memory footprint, driving increased efficiencies when compared to Meshes or Volumes generated on CPUs. This enables Vectron to provide procedurally-generated scenes on the **GPU**[1] without using VRAM. Vectron provides tools in an artist’s workflow and helps render triangle-free geometry ny using OctaneRender's built-in OSL support and OSL texture shaders.

Procedural Primitives using **OSL** (**Open Shader Language**[2]) vector geometry nodes let you create complex shapes, surfaces, **Volumes**, warps, operators, and effects. By vectorizing **Meshes** and Volumes into **Vec-**

---

[1]The GPU is responsible for displaying graphical elements on a computer display. The GPU plays a key role in the Octane rendering process as the CUDA cores are utilized during the rendering process.
[2]A shading language developed by Sony Pictures Imageworks. There are multiple render engines that utilize OSL as it is particularly suited for physically-based renderers.

tron objects, you can manipulate Vectron nodes in revolutionary new ways. Examples include: spheres, strands, sound waves, infinite planes, liquids, clouds, oceans, flow field, and more.

Geometric Operators allow the **Procedural** OSL geometry node graphs workflow to follow the same structure as OSL texture node graphs with 4D mixing, blending, and boundary operator nodes for skinning, metaballs, and procedural resurfacing. Finally, boolean operations are also enabled in Octane Vectron.

The Vectron object is accessible from the **Create** tab in the **Command** panel under the **Octane** sub-category.

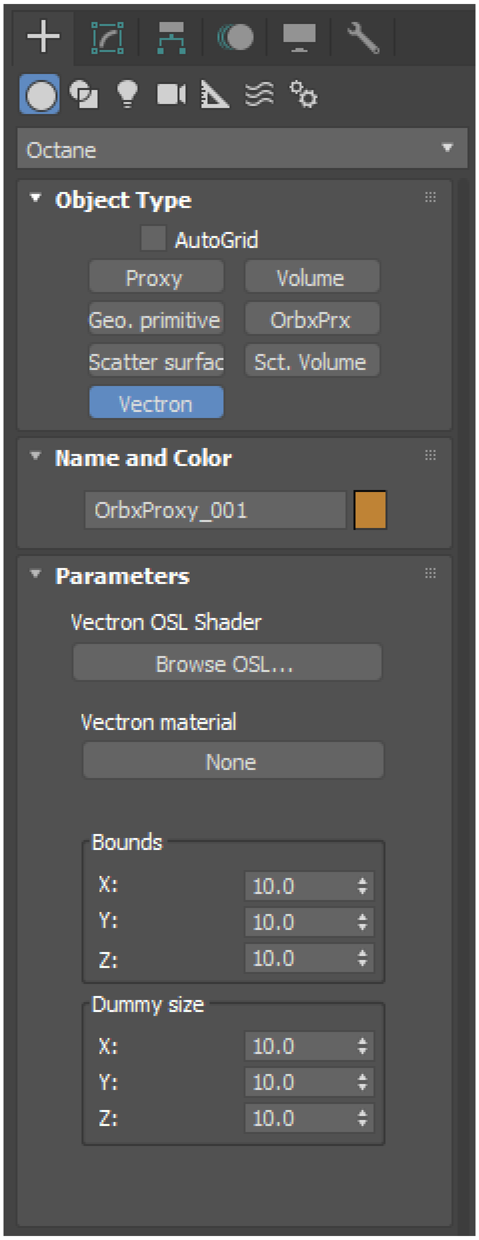


Figure 1: Accessing the Vectron object from the Command panel

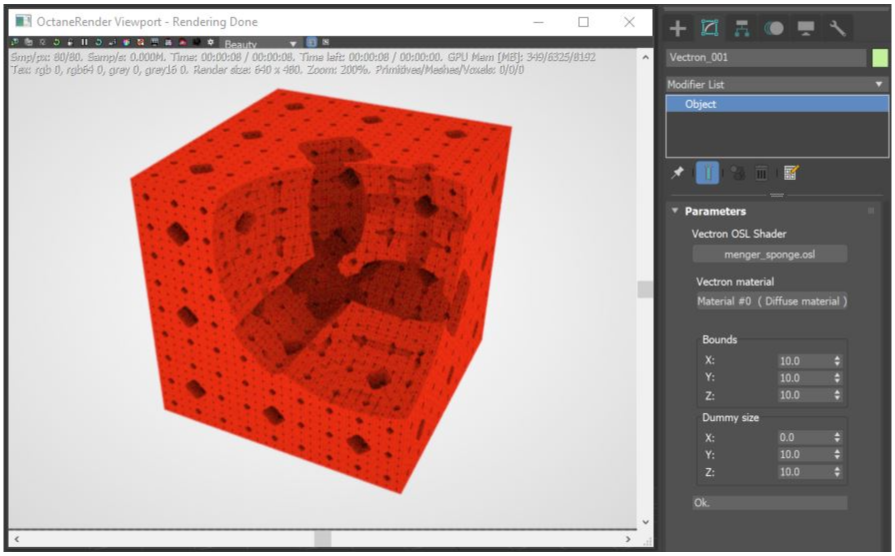


Figure 2: The Octane Vectron parameters

Vectron Parameters

Vectron OSL Shader - Reads OSL-designed geometry.

Vectron Material[1] - Accepts an OctaneRender® Material to apply to the Vectron geometry.

Bounds - Matches the Standalone parameter, and limits the geometry calculation to a certain Volume.

Dummy Size - Determines the size of the wireframe cube in the 3DS Max® Viewport.

The **Text Box** under Dummy Size shows the result of the OSL compilation. Common text feedback includes **Ok** if there's no problem, **No Shader** if no OSL is set, or an error description.

Rendering

Additional rendering tools are explained in this section to help you control settings related to:

---

[1] The representation of the surface or volume properties of an object.

- Object Visibility
- **Batch Rendering**[1]
- Rendering for **VR**[2]
- **Texture Baking**[3]
- **Deep Image**[4] Rendering
- **Render Layers**[5]

Object Visibility

OctaneRender® supports three object visibility properties for **Mesh** object visibility. You can find these by right-clicking on a scene **Object** and selecting **Object Properties** from the **Quad** menu.

Object Visibility Parameters

Visibility - Setting the value to **0** makes the Object invisible. Setting the value to **1** makes the Object visible.

Visible To Camera - When disabled, the Object casts a shadow, but the Object's surface is invisible to the camera.

Cast Shadows - When disabled, the Object won't cast a shadow, but the Object's surface is visible to the camera.

---

[1] The process of assigning sequential portions of frames to be rendered across multiple systems.

[2] Immersively engaging and experiencing depth perception in a three dimensional scene through stereo vision goggles and head-mounted displays.

[3] A process in which scene lighting is "baked" into a texture map based on an object's UV texture coordinates. The resulting texture can then be mapped back onto the surface to create realistic lighting in a real-time rendering environment. This technique is frequently used in game engines and virtual reality for creating realistic environments with minimal rendering overhead.

[4] Renders frames with multiple depth samples in addition to typical color and opacity channels.

[5] Render layers allow users to separate their scene geometry into parts, where one part is meant to be visible and the rest of the other parts “capture” the side effects of the visible geometry. The layers allow different objects to be rendered into separate images where, in turn, some normal render passes may be applied. The Render layers are meant for compositing and not to hide parts of the scene.

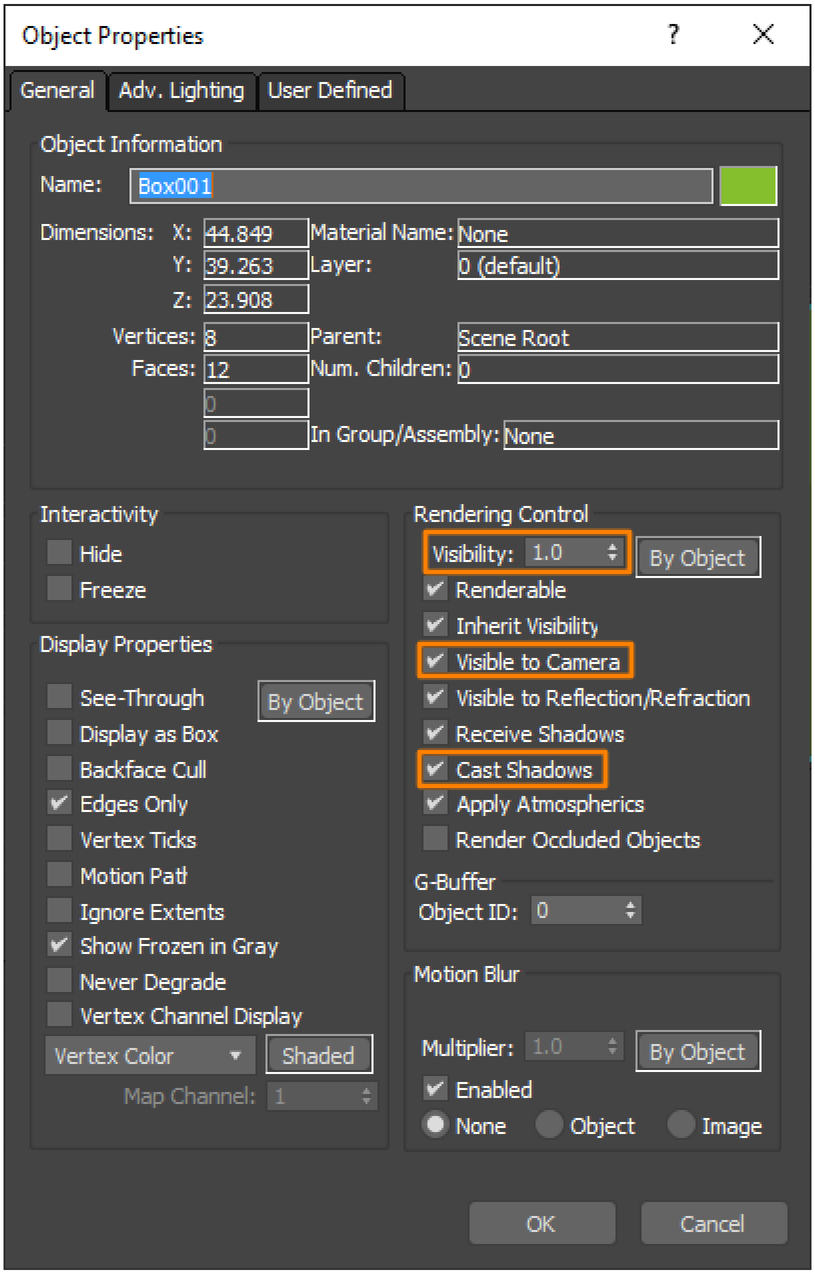


Figure 1: The three object visibility parameters

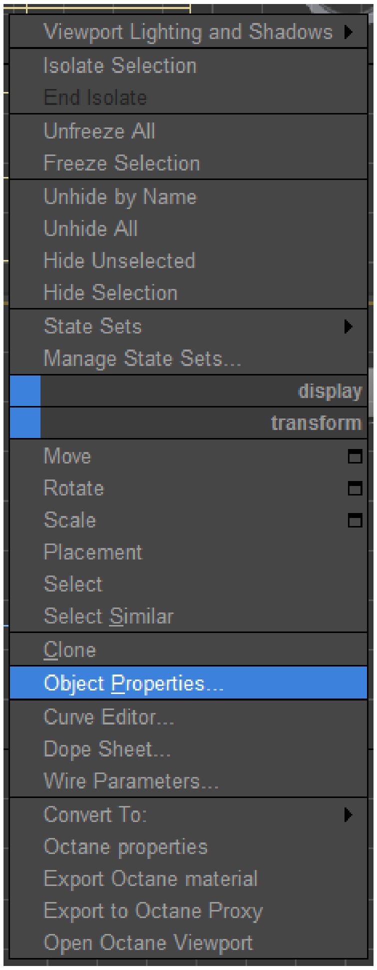


Figure 2: The Quad menu

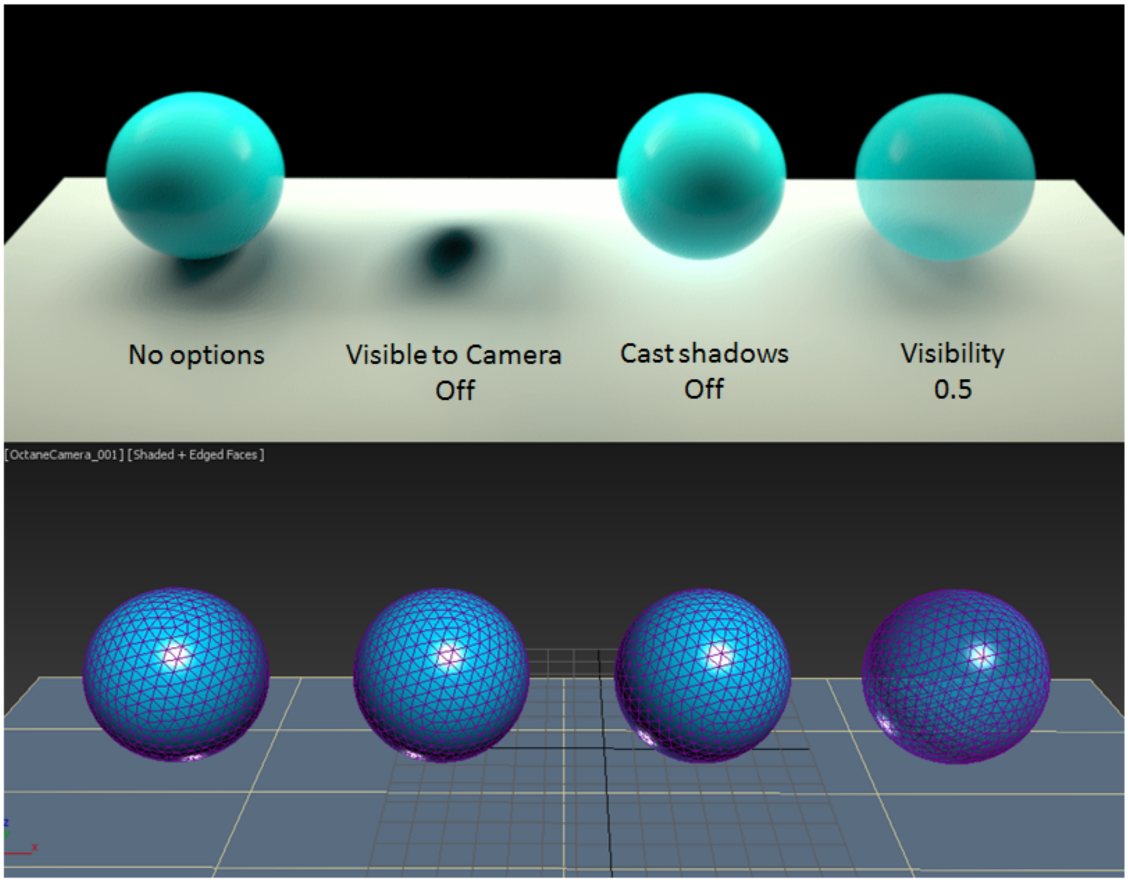


Figure 3: Comparison of object visibility options

Batch Rendering

OctaneRender® can render in batches with the 3DS Max® **Batch Render** mode. To do this, ensure that the scene in 3DS Max® is set up and ready for rendering.

Rendering Via Batch Render

To perform a batch render, go to the **3DS Max** menu and click **Rendering**, then click on **Batch Render** and add the views and their respective parameters. If you're using a scene state and a render preset for the scene, you should specify those as well. You can also override the preset frames for each batch.

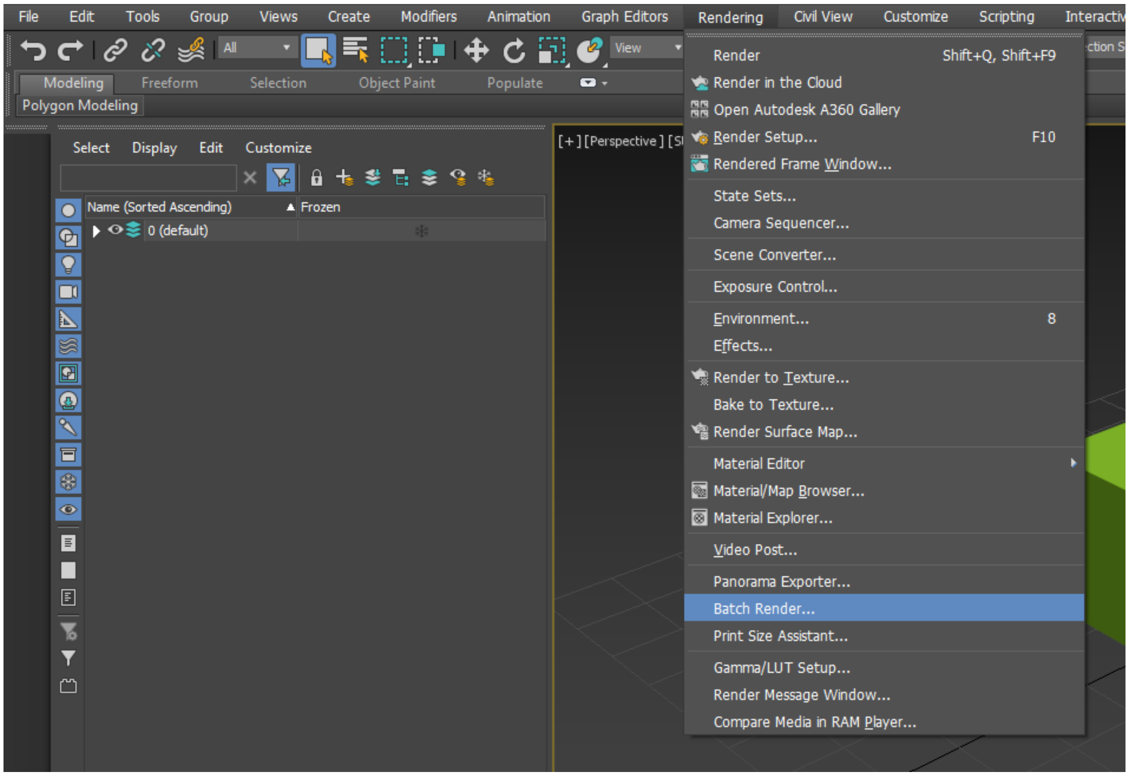


Figure 1: Selecting the Batch Render option

When batch rendering, 3DS Max® uses the sequence of batches in the dialog.

For example:

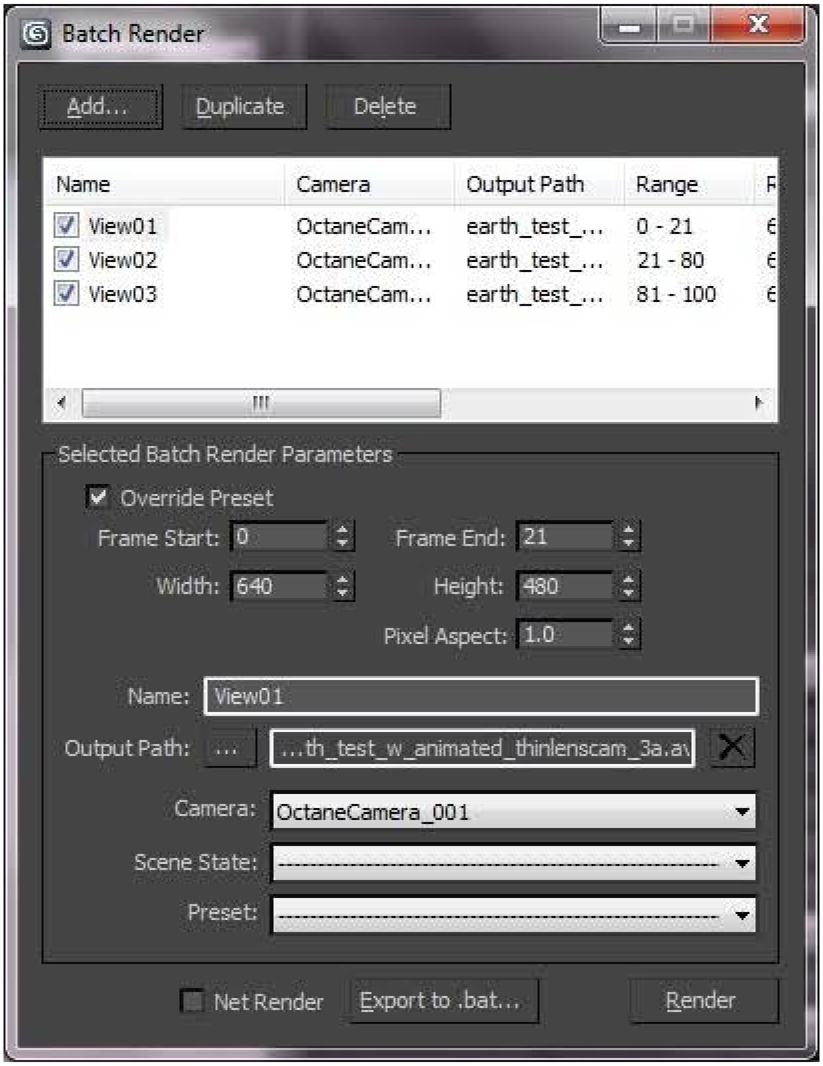


Figure 2: Selecting batches to render

The .bat resulting from the above looks like this:

```
@echo off

rem ---------------------------------------------------------------------------
-

rem -- View01

echo Rendering Batch: View01
```

```
3dsmaxcmd ^

C:Usersuser1Documents3dsMaxscenestestsEarth_test_w_animated_cam.max ^

-batchRender:View01 -outputName View01 -camera OctaneCamera_001 ^

-start 0 -end 21 -nthFrame 1 -width 640 -height 480 -pixelAspect 1


rem ---------------------------------------------------------------------------
-

rem -- View02

echo Rendering Batch: View02

3dsmaxcmd ^

C:Usersuser1Documents3dsMaxscenestestsEarth_test_w_animated_cam.max ^

-batchRender:View02 -outputName View02 -camera OctaneCamera_001 ^

-start 21 -end 80 -nthFrame 1 -width 640 -height 480 -pixelAspect 1


rem ---------------------------------------------------------------------------
-

rem -- View03

echo Rendering Batch: View03

3dsmaxcmd ^

C:Usersuser1Documents3dsMaxscenestestsEarth_test_w_animated_cam.max ^

-batchRender:View03 -outputName View03 -camera OctaneCamera_001 ^

-start 81 -end 100 -nthFrame 1 -width 640 -height 480 -pixelAspect 1
```

Canceling A Batch Render

When you run a batch (View01) from this sequence, it invokes the render engine to render that batch to completion. Batch rendering in 3DS Max® is like a shortcut for running each batch (C:\Users\user1\Documents\3dsMaxscenes…) as a script from the command line, where you are required to provide an argument for each script to specify which scene file to render, among other things. Unlike the command line, where you can press **CTRL-C** on a script while it's rendering, canceling a batch render in the 3DS Max® GUI only applies for not rendering the rest of the batches in the sequence (View02, View03). The render engine continues rendering the current batch (View01) until it's finished, then it pauses 3DS Max®. You have to wait until 3DS Max® finishes the job, then return 3DS Max® to a usable state.

After submitting a batch, 3DS Max® cannot provide any interactivity for that batch from that point, so before running a script in Batch Mode, make sure that the scene is prepared.

Rendering For VR

You can do stereo rendering in the plugins if you are familiar with the values (map type, resolution) that are compatible for **VR**[1]. The camera used for this is a stereoscopic panoramic camera with a **Cube Map** projection of equal sides (+x,-x,+y,-y,+z,-z).

The advantage of the Standalone Edition is that is has presets for VR rendering:

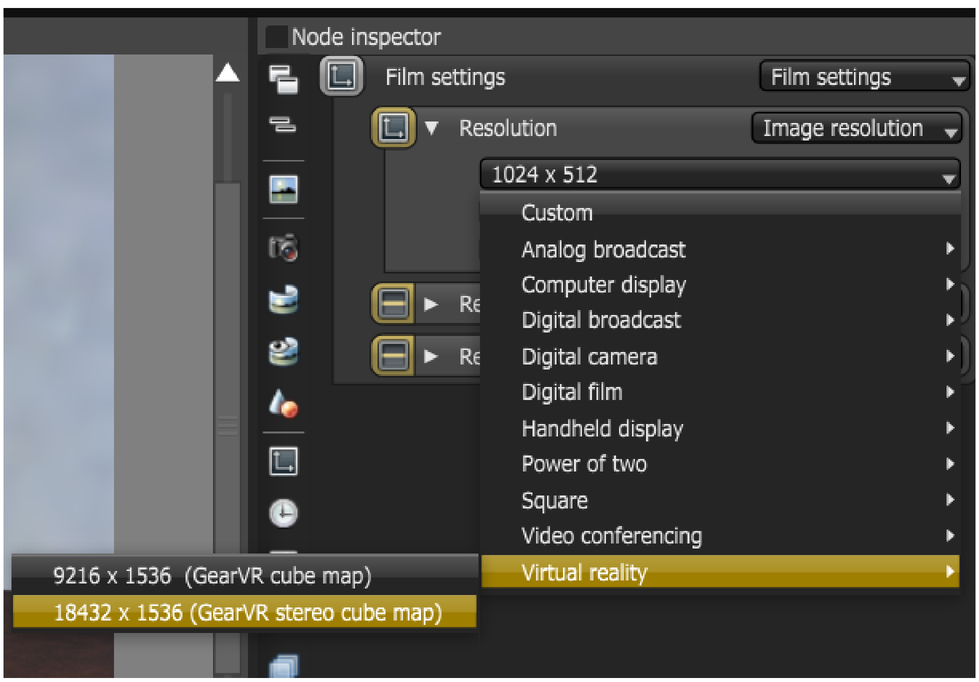


Figure 1: VR rendering presets

[1]Immersively engaging and experiencing depth perception in a three dimensional scene through stereo vision goggles and head-mounted displays.

In all cases, it is just like rendering a 2D image in OctaneRender® as a 360x180 panorama. When you are happy with how the panorama looks, it is time to set the scene up for VR.

1. Make sure that the units are set to **Meters**. It's best not to change the system unit of an existing scene, but since OctaneRender's default units are in meters, make sure these values are the values that would be equivalent to the units used in the scene. (e.g., 3DS Max® 1 cm = OctaneRender® .01 m). If possible, change the system unit value before you import or create geometry.
2. VR scenes are huge. You can temporarily set up the scene with a normal spherical panoramic camera at a low resolution (like 1024×512) so that you can preview the scene and add more textures at optimum performance.

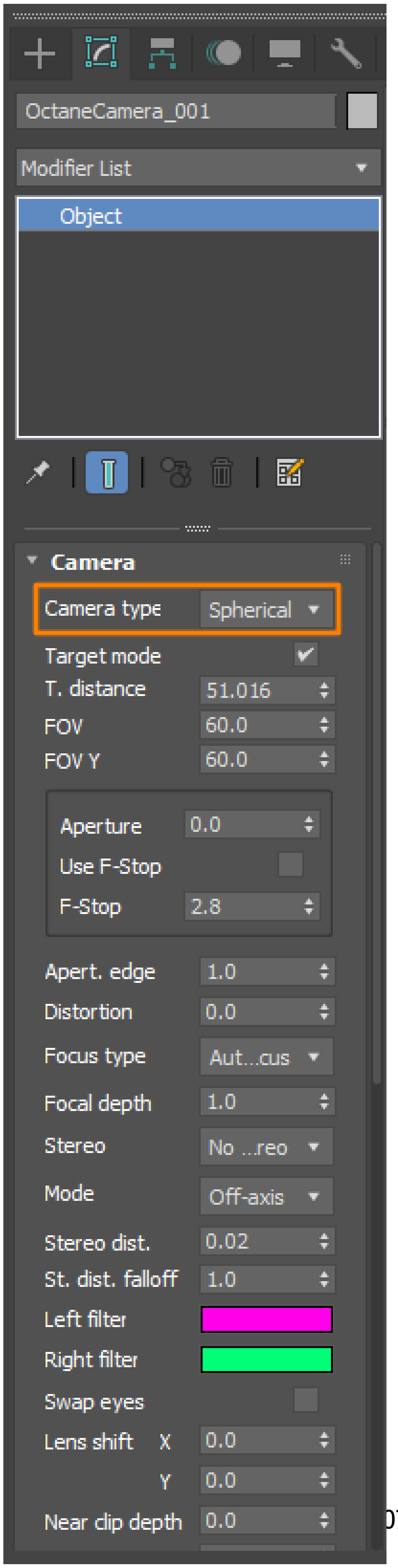

OctaneCamera_001
Modifier List
Object
Camera
Camera type
Spherical
Target mode
T. distance
51.016
FOV
60.0
FOV Y
60.0
Aperture
0.0
Use F-Stop
F-Stop
2.8
Apert. edge
1.0
Distortion
0.0
Focus type
Aut...cus
Focal depth
1.0
Stereo
No ...reo
Mode
Off-axis
Stereo dist.
0.02
St. dist. falloff
1.0
Left filter
Right filter
Swap eyes
Lens shift
X
0.0
Y
0.0
Near clip depth
0.0

Figure 2: Setting up the spherical panoramic camera

3. Turn off **Post Processing**[1] in **Render Setup**.

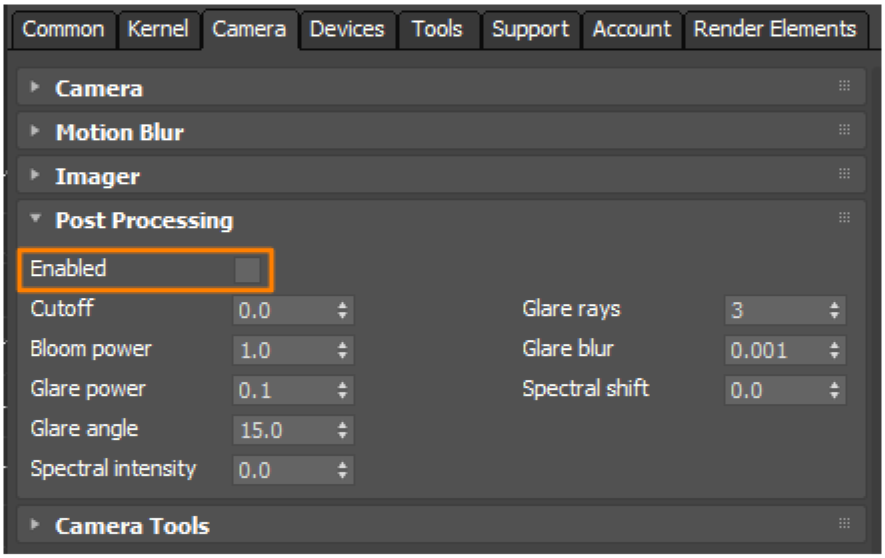


Figure 3: Disabling post-processing

4. Make sure that the **Objects** in the view are 10x the stereo offset distance.
5. Keep the **Camera** upright from the Camera **Modify** Panel.

Figure 4: Enabling the Keep Upright checkbox

6. Make sure that lighting is as realistic as possible - try to using the **Path Tracing** or **PMC** kernels.

---

[1] Effects such as Bloom and Glare that are applied after a scene has been rendered.

7. When you are satisfied with how the scene looks, you can proceed to create the final quality render, which is the 18K cube map render. While you're in the **Camera** parameters, make sure to set the camera type to **Cube Map**, and then choose a **Stereo** mode (used side-by-side).
8. Save the resulting image as an 8-bit PNG.
9. Click **Render**.
10. The resulting image should be an 18K cube map, which you can upload into the VR viewer folder of the device that is compatible with GearVR.

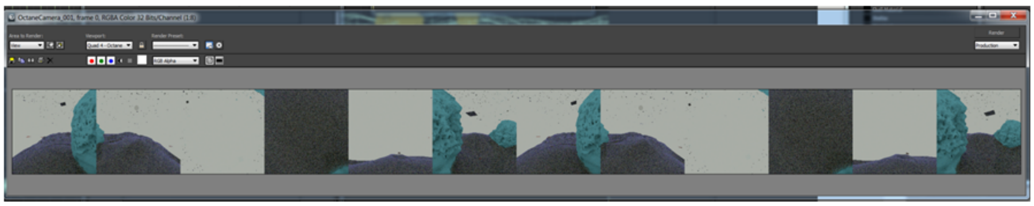


Figure 6: The final result

Texture Baking

The **Texture Baking**[1] system extracts lighting information from a mesh's surface by using its **UV** map to generate a texture that can map back to the **Mesh** later on.

In OctaneRender®, texture baking is implemented as a special type of camera that, in contrast to the **Thin Lens** and **Panoramic** cameras, has one position and direction per sample. The way these are calculated depends on the input UV geometry and the actual geometry being baked.

For each sample, the camera calculates the geometry position and normal, then it generates a ray that points towards it using the same direction as the normal, from a distance of the configured kernel's ray epsilon. Once calculated, OctaneRender® traces the ray in the same way as other camera types.

---

[1] A process in which scene lighting is "baked" into a texture map based on an object's UV texture coordinates. The resulting texture can then be mapped back onto the surface to create realistic lighting in a real-time rendering environment. This technique is frequently used in game engines and virtual reality for creating realistic environments with minimal rendering overhead.

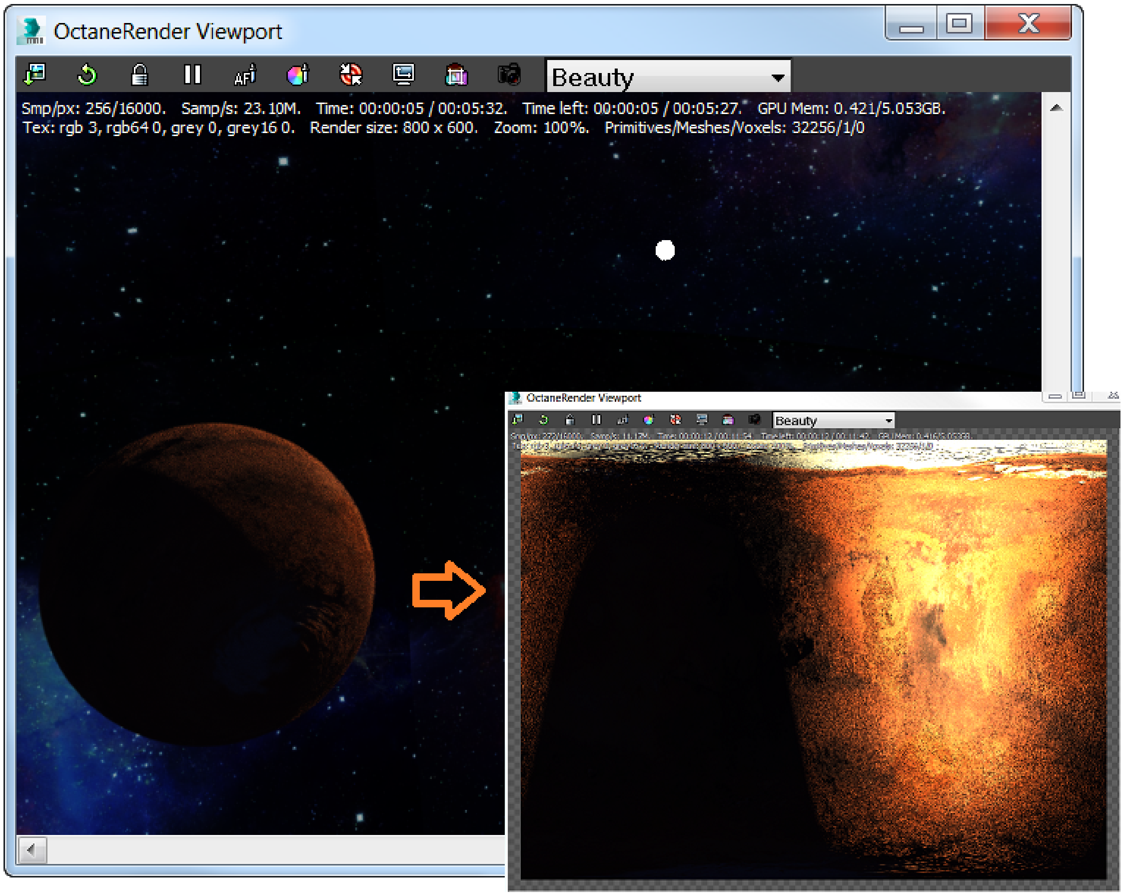


Figure 1: Texture baking sample

Mesh Pre-Requirements

In order to use a **Mesh** for texture baking, the Mesh should contain at least one UV set. In the case of **Alembic**[1], you can use up to three sets. You should not map different geometry primitives to the same UV region - otherwise, you may find artifacts due to overlapping geometry.

[1] An open format used to bake animated scenes for easy transfer between digital content creation tools.

Getting Started

Assuming you've already created a scene that contains geometry, lighting, material information, etc., the easiest way to get started is to create a copy of your render camera and replace its **Camera Type** with a **Baking** camera.

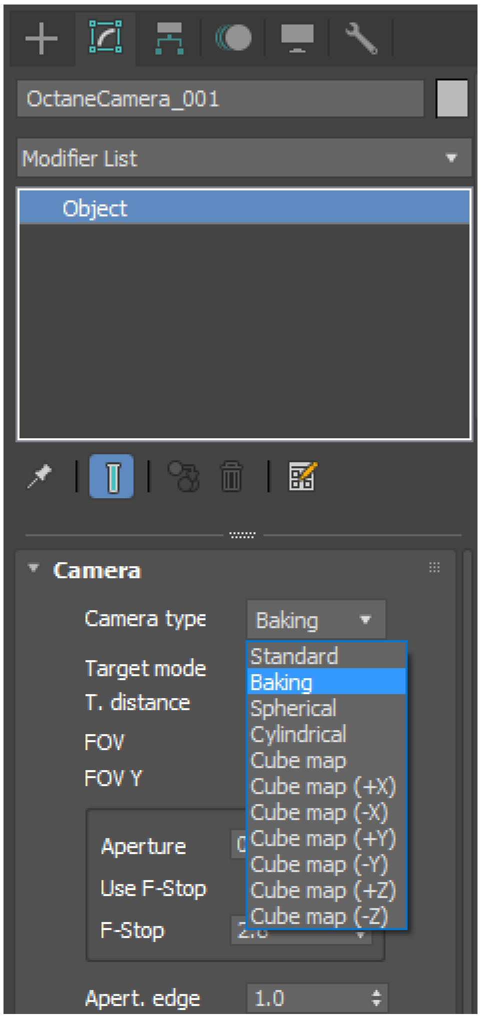


Figure 2: Selecting the Baking camera

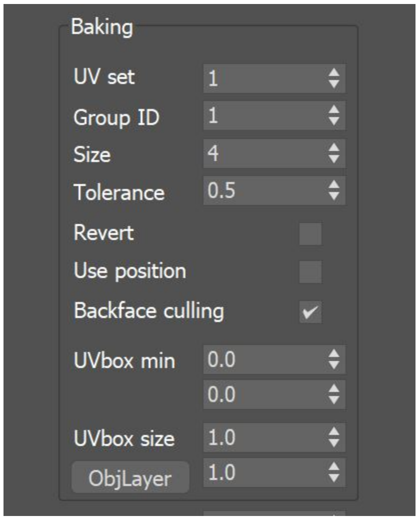


Figure 3: Baking parameters from Camera

Make sure your camera Baking Group ID is the same as the **Octane Object Properties** Baking Group ID.

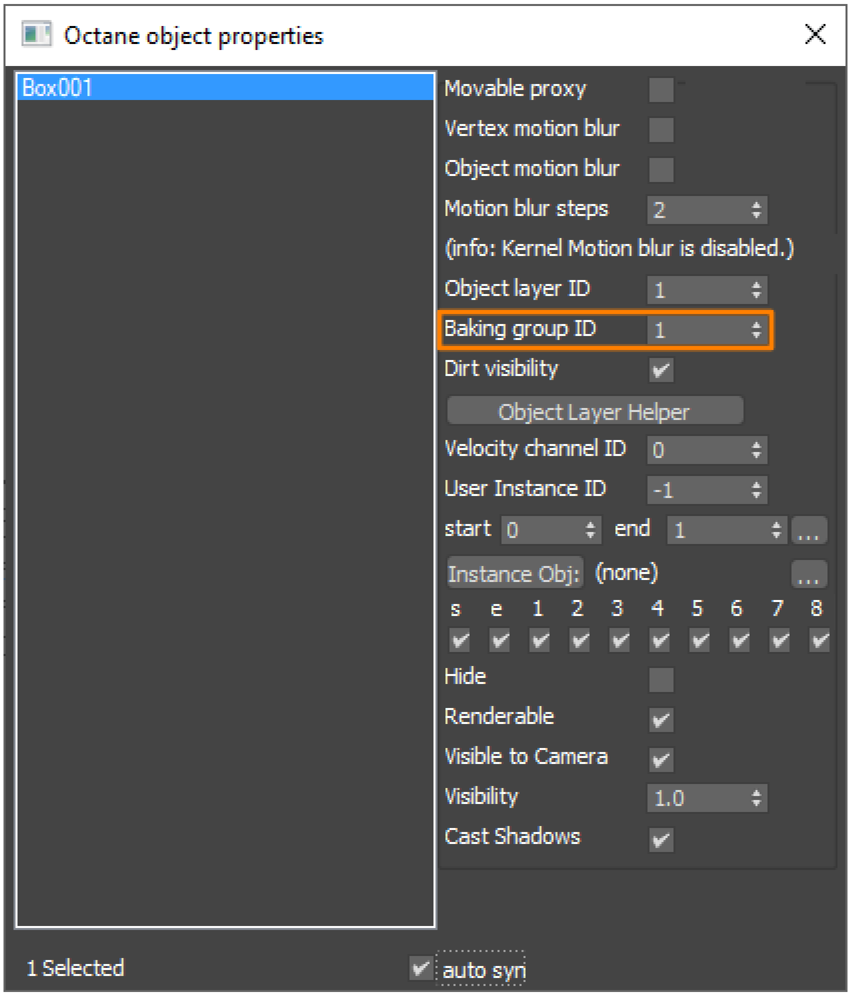


Figure 4: Octane object properties

Baking Parameters

UV Set - Determines the UV set to use for Baking.

Group ID - Specifies the baking group to bake. By default, all objects belong to baking group number **1.** You can arrange new baking groups by making use of Object layers or **Object Layer** maps, similar to the way render layers work.

Size - Specifies the number of pixels added to the **UV** map edges.

Tolerance - Specifies the tolerance for keeping or discarding edge noise.

Revert - Flips the camera directions when enabled.

Use Position - If you use a baking position, OctaneRender® traces camera rays from the specified coordinates in world space instead of using the Mesh surface as reference. This is useful when you bake within position-dependent artifacts such as the ones produced by **Glossy** or **Specular** materials.

In order to tell the Baking camera what geometry to bake, select the proper Baking Group ID in the Baking camera.

You can use render layers, passes, imager settings, etc. the same way as with other types of cameras, which allow you to extract lighting and material information.

Backface Culling - Determines if back-facing geometry will be included in the baking.

UVbox Min/Size - Specifies the area that the Baking camera takes into account. You can pan and zoom in and out of the camera in case your UV geometry is not within the **0,0** - **1,1** region.

Obj. Layer - Creates or edits an **Object Layer Helper** object in the scene. When using a Baking camera, this lets you edit the Baking transform matrix.

Baking Tips

Set your Kernel's **Filter** size to **1.0**. Since baking doesn't need much anti-aliasing, this bakes more precise data.

Set the **Imager** response to **Linear/Off** to disable specific camera response curves.

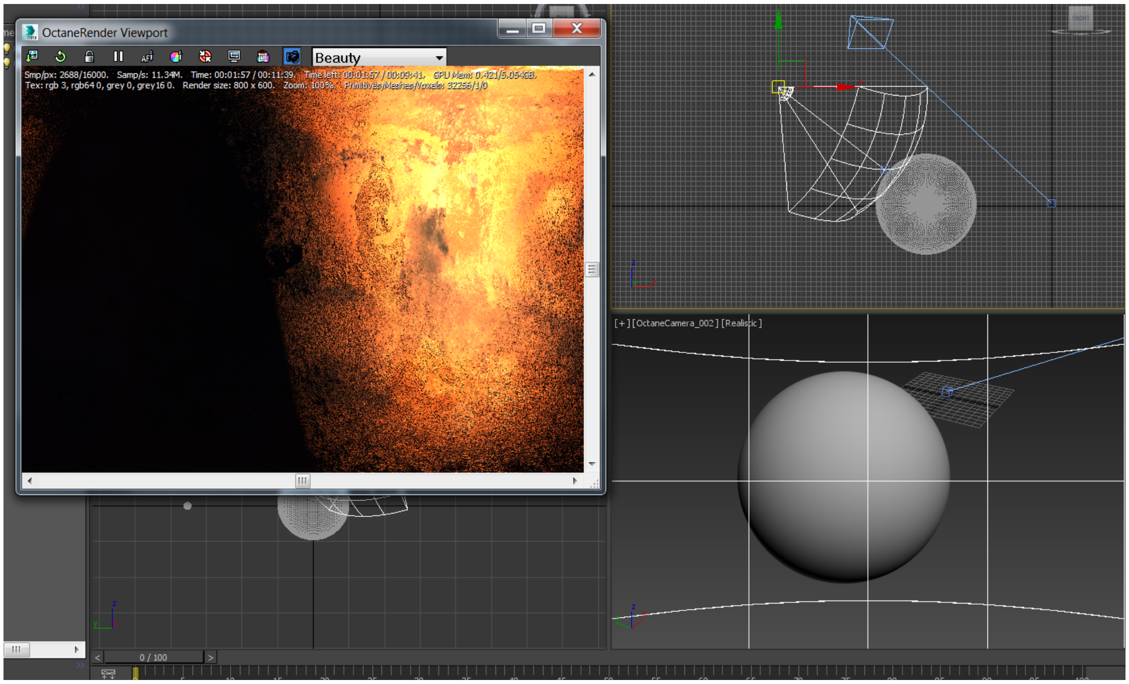


Figure 5: A texture baking sample in the OctaneRender Viewport

Deep Image Rendering

Deep Image[1] rendering improves the compositing workflow by storing Z-depth with samples. It works best in scenarios where traditional compositing fails, like masking out overlapping **Objects**, working with **Images** that have depth-of-field or motion blur, or compositing footage in rendered **Volumes**.

Most major compositing applications now support Deep Image rendering. The disadvantage of Deep Image rendering is the large amounts of memory it requires to render and store deep images.

The standard output format is OpenEXR.

Enabling Deep Image Rendering

You can enable Deep Image rendering by going to the **Render Setup** window, clicking on the **Kernel** tab, then selecting the **Deep Image** checkbox . Deep Image rendering works with the **Path Tracing** and **Direct**

[1] Renders frames with multiple depth samples in addition to typical color and opacity channels.

Lighting kernels.

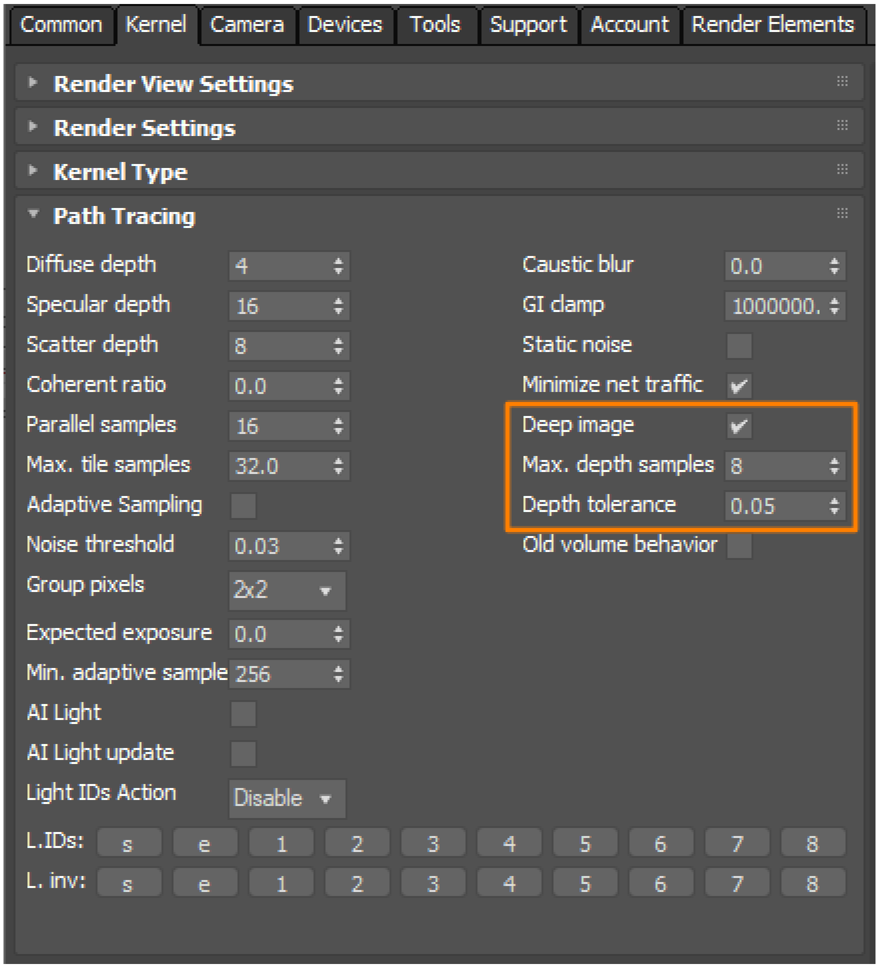


Figure 1: Deep Image parameters in the Path Tracing kernel tab

Deep Image Parameters

Deep Image - Enables deep image rendering. The Path Tracing and Direct Lighting kernels support deep image rendering.

Max. Depth Samples - Specifies an upper limit for the number of deep samples stored per pixel.

Depth Tolerance - Specifies a merge tolerance - i.e., when two samples have a relative depth difference within the depth tolerance, they merge.

For a typical scene, the **GPU**[1] renders thousands of samples per pixel. However, VRAM is limited, so it's necessary to manage the number of samples stored with the Deep **Render Passes**[2] and Max. Depth Samples parameters.

Render Layers

Render layers separate scene geometry into parts, where one part is meant to be visible and the other parts capture the side effects of the visible geometry. The layers allow different **Objects** to render into separate images where you can apply render passes. The render layers are meant for compositing, not hiding parts of the scene.

To set up a scene for rendering in layers, assign an **Octane Layer ID** for the scene's Objects, and set the Object geometry type to be **Movable Proxy**[3] (Figure 1). You can find this option by right-clicking on the Object in the **Viewport** and selecting **Octane Properties** from the **Quad** menu (Figure 2).

[1] The GPU is responsible for displaying graphical elements on a computer display. The GPU plays a key role in the Octane rendering process as the CUDA cores are utilized during the rendering process.
[2] Render passes allow a rendered frame to be further broken down beyond the capabilities of Render Layers. Render Passes vary among render engines but typically they allow an image to be separated into its fundamental visual components such as diffuse, ambient, specular, etc..
[3] An object saved as a separate file with the purpose of being reused in larger scenes. This is used to minimize any addition to the total polygon count in the scene, especially if the scene requires the same object to appear several times. If used in conjunction with instancing, Proxies help keep very large scenes from reaching polygon limits and also keeps the relative file size of the main project file manageable.

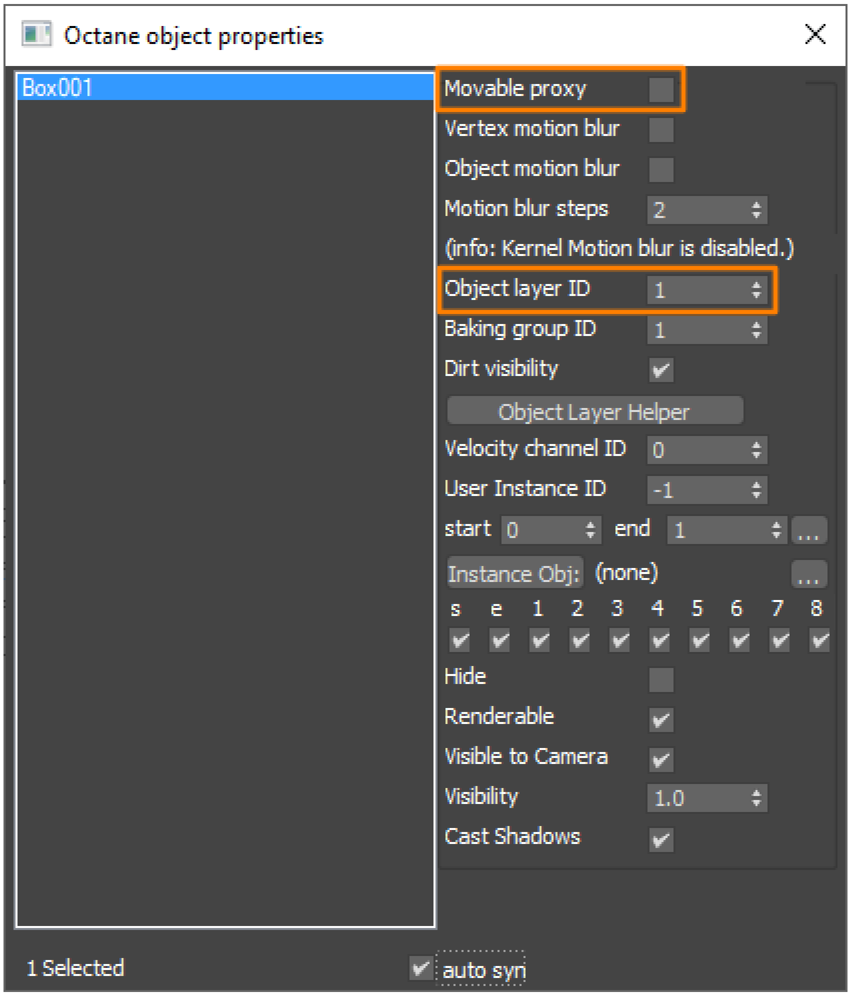


Figure 1: Assigning Object Layer IDs and Movable Proxy in the Octane Object Properties window

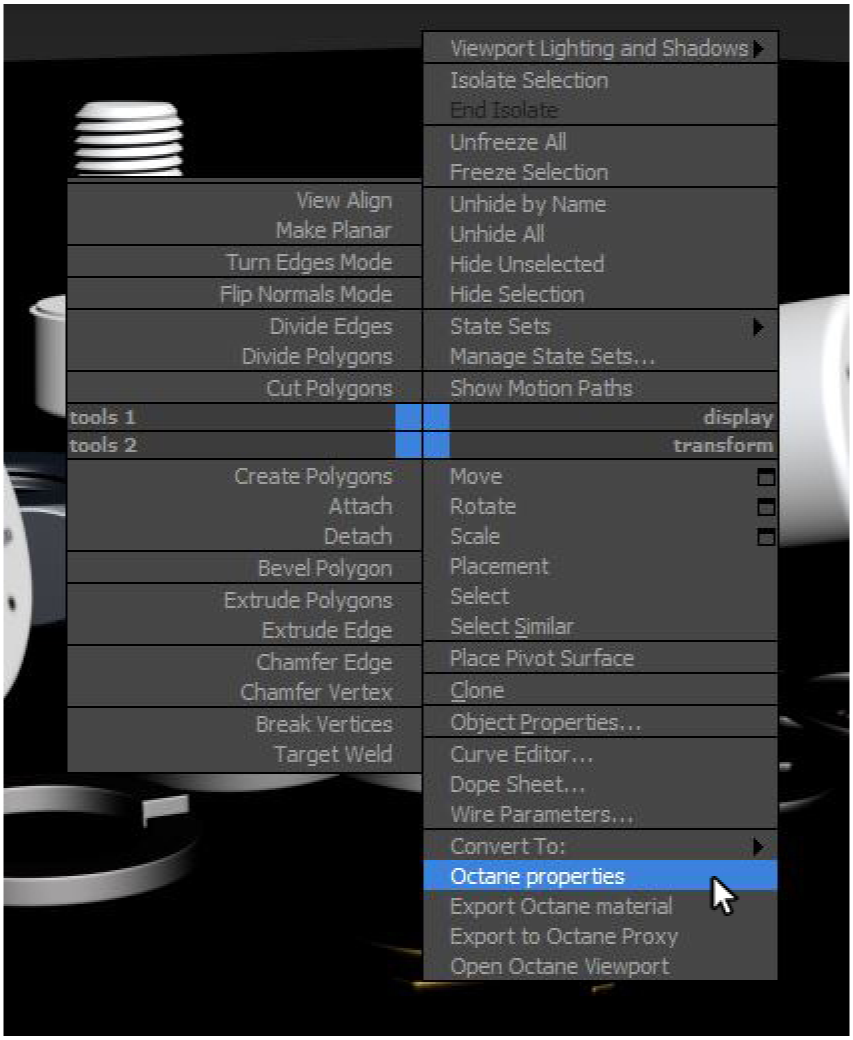


Figure 2: Accessing the Octane Properties by right-clicking on the Object in the Viewport

Finally, enable the **Render Layers**[1] feature in **Render Setup** from the **Kernel** tab. If the Object Layer IDs are set up for each object, OctaneRender® can render separate passes for those objects that share an ID.

---

[1] Render layers allow users to separate their scene geometry into parts, where one part is meant to be visible and the rest of the other parts "capture" the side effects of the visible geometry. The layers allow different objects to be rendered into separate images where, in turn, some normal render passes may be applied. The Render layers are meant for compositing and not to hide parts of the scene.

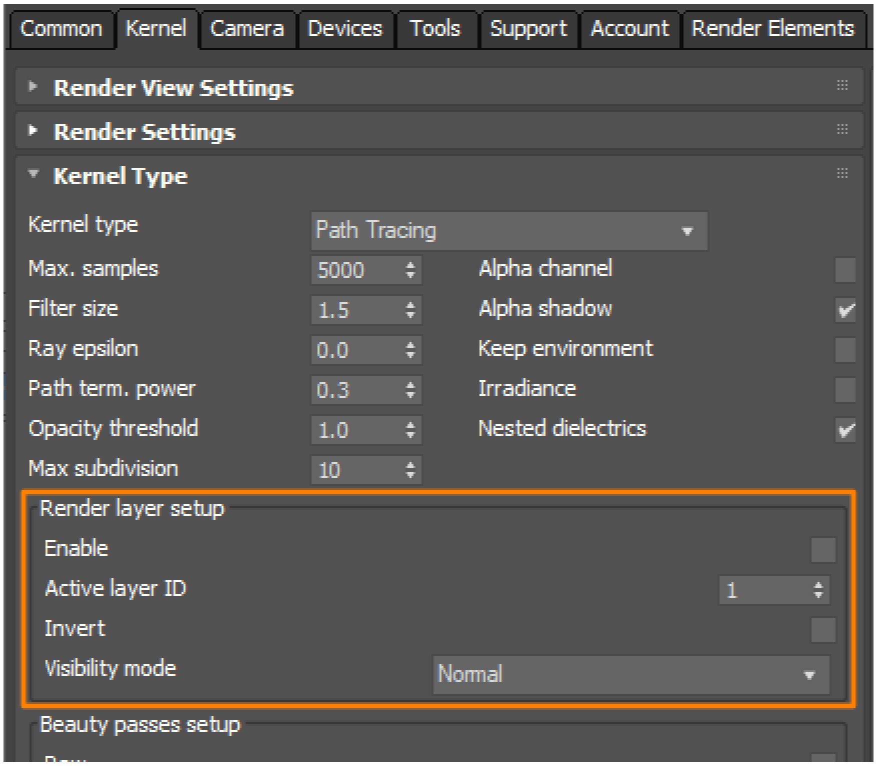


Figure 3: Enabling the Render Layers feature from the Render Setup window

The main beauty pass renders the Object IDs that correspond to the **Active Layer ID**, and cuts out everything else. You can also choose to invert everything by toggling **Invert** in the **Render Layer** rollout.

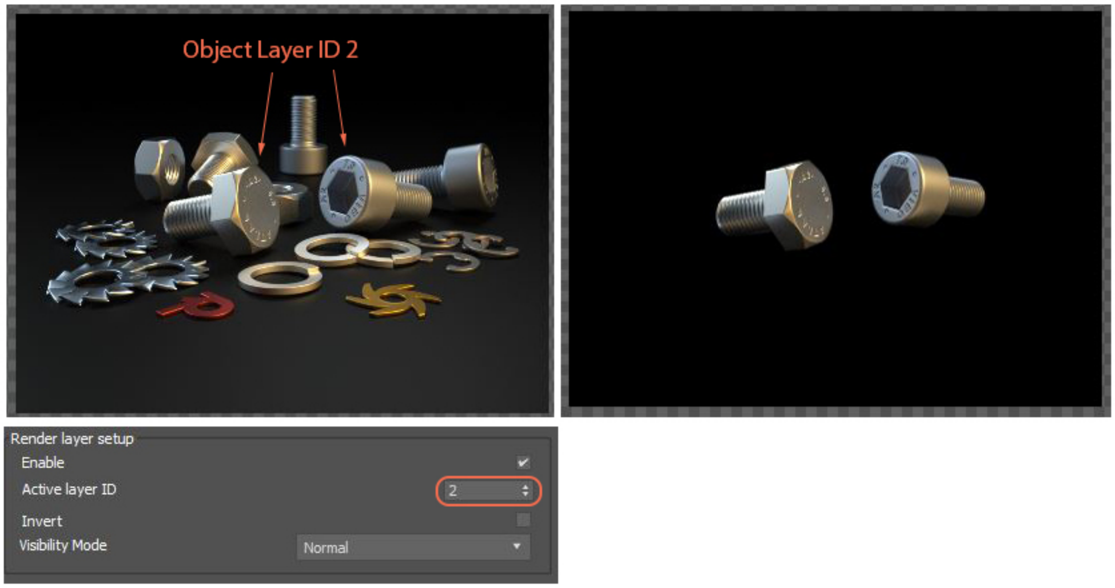


Figure 4: Rendering objects with an Object Layer ID of 2

Visibility Mode has four options:

- **Normal** - Objects on the specified Active Layer ID are rendered, and objects with other Object Layer IDs rendered as cutouts.
- **Hide Inactive Layers** - Make Objects on the inactive layers invisible in the rendering, without rendering cutouts.
- **Only Side Effects** - Similar to Normal, except the active layer is invisible to the camera, but its side effects are still captured without obstructing the active layer.
- **Hide From Camera** - Renders inactive layer shadow contributions, but keeps the Objects invisible in the rendering.

Glossary

A

Absorption

Defines how fast light is absorbed while passing through a medium.

Adaptive Sampling

A method of sampling that determines if areas of a rendering require more sampling than other areas instead of sampling the entire rendering equally.

Alembic

An open format used to bake animated scenes for easy transfer between digital content creation tools.

Alpha Channel

A greyscale image used to determine which areas of a texture map are opaque and which areas are transparent.

Anti-Ghosting

The automatic or manual correction involved in the merging a stack of images during the creation of a High Dynamic Range image. The process aims to correct the strange effect when objects that change position in the image set is partially visible (like a ghost) in the final HDR image.

Aperture

Determines how much light enters a camera lens. A large aperture produces a narrow depth of field and a small aperture produces a wide depth of field.

AR

Viewing a conceptual three dimensional scene in context to see how it might look in the real world.

Augmented Reality

Viewing a conceptual three dimensional scene in context to see how it might look in the real world.

B

Batch Rendering

The process of assigning sequential portions of frames to be rendered across multiple systems.

Black Body

An opaque object that emits thermal radiation. In Octane, this is used to designate illumination properties for mesh emitters.

D

Deep Image

Renders frames with multiple depth samples in addition to typical color and opacity channels.

Depth Buffer

A measure of object distances from the camera typically represented as a grayscale image.

Depth of Field

The distance between the nearest and farthest objects in a scene that appear acceptably sharp in an image. Although a lens can precisely focus at only one distance at a time, the decrease in sharpness is gradual on each side of the focused distance, so that within the DOF, the unsharpness is imperceptible under normal viewing conditions. source: wikipedia (https://en.wikipedia.org/wiki/Depth_of_field)

Diffuse

Amount of diffusion, or the reflection of light photons at different angles from an uneven or granular surface. Used for dull, non-reflecting materials or mesh emitters.

Diffuse material

Used for dull, non-reflecting materials or mesh emitters.

Displacement

The process of utilizing a 2D texture map to generate 3D surface relief. As opposed to bump and normal mapping, Displacement mapping does not only provide the illusion of depth but it effectively displaces the actual geometric position of points over the textured surface.

DoF

The distance between the nearest and farthest objects in a scene that appear acceptably sharp in an image. Although a lens can precisely focus at only one distance at a time, the decrease in sharpness is gradual on each side of the focused distance, so that within the DOF, the unsharpness is imperceptible under normal viewing conditions. source: wikipedia (https://en.wikipedia.org/wiki/Depth_of_field)

Drivers

Files that allow hardware devices to communicate with an operating system. In the case of Octane, the latest Nvidia drivers should be used.

E

Effective Focus Range

The distance between the nearest and farthest objects in a scene that appear acceptably sharp in an image. Although a lens can precisely focus at only one distance at a time, the decrease in sharpness is gradual on each side of the focused distance, so that within the DOF, the unsharpness is imperceptible under normal viewing conditions. source: wikipedia (https://en.wikipedia.org/wiki/Depth_of_field)

Emissions

The process by which a Black body or Texture is used to emit light from a surface.

EXR

Also known as OpenEXR. This image file format was developed by Industrial Light & Magic and provides a High Dynamic Range image capable of storing deep image data on a frame-by-frame basis.

F

FBX

.fbx (Filmbox) is a proprietary file format developed by Kaydara and owned by Autodesk since 2006. It is used to provide interoperability between digital content creation applications. As of Octane 3.07, a scene node will also be available as an FBX file, allowing for quick and easy transport of assets from industry standard DCC applications

Field of View

The area that is visible to a camera lens usually measured in millimeters. A wide angle lens provides a larger field of view and a telephoto lens provides a narrow field of view.

Focus Range

The distance between the nearest and farthest objects in a scene that appear acceptably sharp in an image. Although a lens can precisely focus at only one distance at a time, the decrease in sharpness is gradual on each side of the focused distance, so that within the DOF, the unsharpness is imperceptible under normal viewing conditions. source: wikipedia (https://en.wikipedia.org/wiki/Depth_of_field)

FoV

The area that is visible to a camera lens usually measured in millimeters. A wide angle lens provides a larger field of view and a telephoto lens provides a narrow field of view.

G

Gamma

The function or attribute used to code or decode luminance for common displays. The computer graphics industry has set a standard gamma setting of 2.2 making it the most common default for 3D modelling and rendering applications.

Glossy

The measure of how well light is reflected from a surface in the specular direction, the amount and way in which the light is spread around the specular direction, and the change in specular reflection as the specular angle changes. Used for shiny materials such as plastics or metals.

Glossy material

Used for shiny materials such as plastics or metals.

GPU

The GPU is responsible for displaying graphical elements on a computer display. The GPU plays a key role in the Octane rendering process as the CUDA cores are utilized during the rendering process.

Graphics Card

The GPU is responsible for displaying graphical elements on a computer display. The GPU plays a key role in the Octane rendering process as the CUDA cores are utilized during the rendering process.

H

Hardware

Any physical device present in a computer system. A Nvidia GPU is a required hardware device for using the Octane Render engine.

HDRI

An image which presents more than 8 bit per color channel unlike most common image formats.

High Dynamic Range Image

An image which presents more than 8 bit per color channel unlike most common image formats.

I

IES

An IES light is the lighting information representing the real-world lighting values for specific light fixtures. For more information, visit http://www.ies.org/lighting/.

IFL

(Image File List) file is an ASCII file that constructs an animation by listing single-frame bitmap files to be used for each rendered frame. When you assign an IFL file as a bitmap, rendering steps through each specified frame, resulting in an animated map. (reference: https://knowledge.autodesk.com/support/3DS-max/learn-explore/caas/CloudHelp/cloudhelp/2017/ENU/3DSMax/files/GUID-CA63616D-9E87-42FC-8E84-D67E1990EE71-htm.html)

Imaging

Refers to periodically creating an image from the renderer for display (which may or may not include tone mapping as one step).

Independent Software Vendor

An individual or business that builds, develops and sells consumer or enterprise software. Although ISV-provided software is consumed by end users, it remains the property of the vendor. An ISV is also known as a software publisher.

Instancing

Instancing an object means taking a single imported mesh object, such as an OBJ or an FBX and making multiple copies, each of which can be placed in different parts of the scene. This saves an enormous amount of computational resources because only a single object is loaded into the scene.

Interactive Photorealistic Rendering

Provides artists a quick preview of the image prior to the final render, and efficiently allows for adjusting some elements in the scene such as lights, shaders and textures interactively. An IPR image contains shading and lighting data including some for visibility, in addition to the software render.

IPR

Provides artists a quick preview of the image prior to the final render, and efficiently allows for adjusting some elements in the scene such as lights, shaders and textures interactively. An IPR image contains shading and lighting data including some for visibility, in addition to the software render.

ISV

An individual or business that builds, develops and sells consumer or enterprise software. Although ISV-provided software is consumed by end users, it remains the property of the vendor. An ISV is also known as a software publisher.

K

Kernels

By definition, this is the central or most important part of something. In Octane, the Kernels are the heart of the render engine.

L

LDR

Image formats that have 8 bits per color channel such as the common image formats JPEG, PNG, GIF among others.

Low Dynamic Range

Image formats that have 8 bits per color channel such as the common image formats JPEG, PNG, GIF among others.

Lua

A scripting language that supports procedural, object-oriented, functional, and data-driven programming. It can be used to extend Octane's functionality.A scripting language that supports procedural, object-oriented, functional, and data-driven programming. It can be used to extend Octane's functionality.

Lua Scripting

A scripting language that supports procedural, object-oriented, functional, and data-driven programming. It can be used to extend Octane's functionality.A scripting language that supports procedural, object-oriented, functional, and data-driven programming. It can be used to extend Octane's functionality.

M

Material

The representation of the surface or volume properties of an object.

Materials

A set of attributes or parameters that describe surface characteristics.

Mediums

The behavior of light inside a surface volume described by scatter, absorption, and transmission characteristics.

Mesh Emitters

The ability for a surface to emit illumination usually described by a Black Body or Texture emission type.

Mix material

Used to mix any two material types.

Mixed

The ratio of diffuse and specular reflection.

Motion Blur

An optical phenomenon that occurs when a camera's shutter opens and closes too slowly to capture movement without recording a blurring of the subject.

N

Network Rendering

The utilization of multiple CPUs or GPUs over a network to complete the rendering process.

NGE

Node Graph Editor

O

Open Shader Language

A shading language developed by Sony Pictures Imageworks. There are multiple render engines that utilize OSL as it is particularly suited for physically-based renderers.

Open SubDiv Surfaces

A set of open source libraries that implement high performance subdivision surface (subdiv) evaluation on massively parallel CPU and GPU architectures. This code path is optimized for drawing deforming surfaces with static topology at interactive framerates. Source: Pixar (http://graphics.pixar.com/opensubdiv/docs/intro.html).

OpenVDB

Dreamworks' open-source C++ library housing the data structures and tools implementation for storing and manipulating volume data, like smoke and other amorphous materials. The purpose of OpenVDB is mostly to have an efficient way to store volumetric data in memory and on disk. It has evolved into a more general toolkit that also lets you accomplish other things, such as fracturing volumes, converting meshes to volumes and vice versa. However, it does not include a computational fluid dynamics solver, and therefore it cannot procedurally generate smoke or fire.

OpenVDB is fully integrated as a library in OctaneRender. For more information about OpenVDB, check at http://www.openvdb.org/.

ORBX

The ORBX file format is the best way to transfer scene files from 3D Authoring software programs that use the Octane Plug-in such as Octane for Maya, Octane for Cinema 4D, or OctaneRender Standalone. This format is more efficient than FBX when working with Octane specific data as it provides a flexible, application independent format. ORBX is a container format that includes all animation data, models, textures etc. that is needed to transfer an Octane scene from one application to another.

Out-of-Core

When scene assets become too large to load completely onto the system's GPU, Out-of-Core technology allows the render engine to utilize the CPU to assist in the rendering process.

P

PBR

A contemporary shading and rendering process that seeks to simplify shading characteristics while providing a more accurate representation of lighting in the real world.

Portal

A technique that assists the render kernel with exterior light sources that illuminate interiors. In interior renderings with windows, it is difficult for the path tracer to find light from the outside environment and optimally render the scene. Portals are planes that are added to the scene with the Portal material applied to them.

Post Processing

Effects such as Bloom and Glare that are applied after a scene has been rendered.

Projections

Methods for orienting 2D texture maps onto 3D surfaces.

Proxy

An object saved as a separate file with the purpose of being reused in larger scenes. This is used to minimize any addition to the total polygon count in the scene, especially if the scene requires the same object to appear several times. If used in conjunction with instancing, Proxies help keep very large scenes from reaching polygon limits and also keeps the relative file size of the main project file manageable.

Proxy Server

A Proxy Server, also known as an application-level gateway, is an intermediary server between the local network and the external servers from which a client is requesting a service. The external servers will only see the network proxy server's IP address thus providing some degree of security and privacy. There are various kinds of proxies, the most common are Web Proxies.

R

RAW

In HDR imaging, this refers to minimally processed HDR image formats. Raw files can have 12 or 14 bits per color channel, although the available dynamic range might be cut down due to noise.

Render Layers

Render layers allow users to separate their scene geometry into parts, where one part is meant to be visible and the rest of the other parts “capture” the side effects of the visible geometry. The layers allow different objects to be rendered into separate images where, in turn, some normal render passes may be applied. The Render layers are meant for compositing and not to hide parts of the scene.

Render Passes

Render passes allow a rendered frame to be further broken down beyond the capabilities of Render Layers. Render Passes vary among render engines but typically they allow an image to be separated into its fundamental visual components such as diffuse, ambient, specular, etc..

S

Scattering

Defines how fast light gets scattered when traveling through the medium.

Shadow Catcher

The Shadow Catcher can be used to create shadows cast by objects onto the surrounding background imagery. The shadows cast are not limited to simply a ground plane but can be cast onto other surfaces of varying shapes.

Spectral Light Transport

A technique in which a scene's light transport is modeled with real wavelengths. Spectral rendering can also simulate light sources and objects more effectively, as the light's emission spectrum can be used to release photons at a particular wavelength in proportion to the spectrum. Source: Wikipedia (https://en.wikipedia.org/wiki/Spectral_rendering).

Specular

Amount of specular reflection, or the mirror-like reflection of light photons at the same angle. Used for transparent materials such as glass and water.

Specular material

Used for transparent materials such as glass and water.

T

Texture Baking

A process in which scene lighting is "baked" into a texture map based on an object's UV texture coordinates. The resulting texture can then be mapped back onto the surface to create realistic lighting in a real-time rendering environment. This technique is frequently used in game engines and virtual reality for creating realistic environments with minimal rendering overhead.

Textures

Textures are used to add details to a surface. Textures can be procedural or imported raster files.

TMO

Maps HDR images to standard displays which have a limited dynamic range. The more prominent TMOs are Mantiuk'06, Reinhard'02, Drago, and Durand.

Tone Mapping

Refers to applying a curve to an image to reduce dynamic range

Tone Mapping Operator

Maps HDR images to standard displays which have a limited dynamic range. The more prominent TMOs are Mantiuk'06, Reinhard'02, Drago, and Durand.

Transformations

Tools used to rotate and position 2D and 3D texture maps onto 3D surfaces.

Transmission

A surface characteristic that determines if light may pass through a surface volume.

U

Unbiased Rendering

Unbiased rendering does not introduce any "errors" or shortcuts into the rendering process. It will calculate all scene data using real-world calculations. This type of rendering is known for producing exceptional render quality.

V

VDB

Dreamworks' open-source C++ library housing the data structures and tools implementation for storing and manipulating volume data, like smoke and other amorphous materials. The purpose of OpenVDB is mostly to have an efficient way to store volumetric data in memory and on disk. It has evolved into a more general toolkit that also lets you accomplish other things, such as fracturing volumes, converting meshes to volumes and vice versa. However, it does not include a computational fluid dynamics solver, and therefore it cannot procedurally generate smoke or fire. OpenVDB is fully integrated as a library in OctaneRender. For more information about OpenVDB, please see http://www.openvdb.org/.

Virtual Reality

Immersively engaging and experiencing depth perception in a three dimensional scene through stereo vision goggles and head-mounted displays.

Volume Medium

A shading system designed to render volumes such as smoke and fog.

VR

Immersively engaging and experiencing depth perception in a three dimensional scene through stereo vision goggles and head-mounted displays.

Z

Z-Buffer

A measure of object distances from the camera typically represented as a grayscale image.

Z-Depth

A measure of object distances from the camera typically represented as a grayscale image.

Index

I

K

L

M

N

O