What's New in mental ray® and Iray®

Jennifer O'Connor, Blue Marble 3D
George Matos, Blue Marble 3D

AV3038

This class gives you a challenging and fast-paced look at recent advances in NVIDIA ® mental ray® rendering software and the NVIDIA® Iray® renderer with 2014 Autodesk® software releases.

Learning Objectives

At the end of this class, you will be able to:

- Produce renderings using Image-Based Lighting and Final Gather Skylight modes
- Use String Options in the new Render Setup UI and via MAXScript
- Describe and use the new Unified Sampling option in mental ray
- Describe the applications for, and produce renderings with, the new Architectural and Caustics modes in Iray

About the Speakers

As the Director of Digital Artists at Blue Marble 3D, Jennifer manages the production of 2D, 3D, and Virtual Reality models and images. She joined the company in February of 2012 as 3D Visualization Specialist, bringing with her over 27 years of experience in 2D and 3D CAD.

Jennifer is the author of 5-star rated book "Mastering mental ray. Rendering Techniques for 3D & CAD Professionals" (Sybex 2010), where she literally "wrote the book" on 3d rendering for architecture and design.

You can contact Jennifer at jenni@mastering-mentalray.com

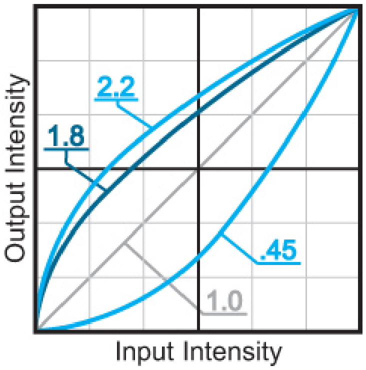
In 2013, George was appointed Principal of Blue Marble 3D, heading up the division which had already begun expanding beyond Chipman Design's architecture clients. Recognized throughout the industry, George has presented at Autodesk University Conferences in Las Vegas on rendering and animation technology as well as conducting widely-subscribed webinars for BOXX, NVIDIA and Autodesk. Blue Marble 3D places fundamental importance on supporting each client in ways to best utilize their existing technologies. Beyond creating the latest in leading-edge imagery, his team also provides a range of consultative services including technical training and development of new technology in conjunction with its strategic partners.

What's new in 3ds Max/Design 2014

Before we jump in on the specifics of mental ray and Iray in 3ds Max/Design 2014, I want to outline a couple specific things that are new for rendering and mental ray in general.

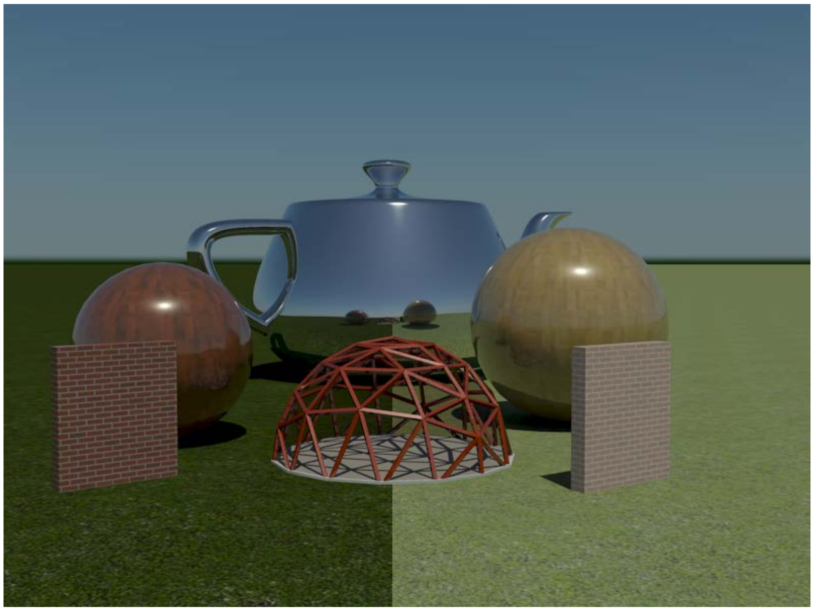
Automatic Gamma Correction

Gamma Correction adjusts the brightness of an image, based on a Gamma Curve. Gamma compensates for how human vision and computer output devices operate. Images need to be brightened in the midrange to look correct on monitors, printers, etc. Below is a representation of a Gamma Curve:



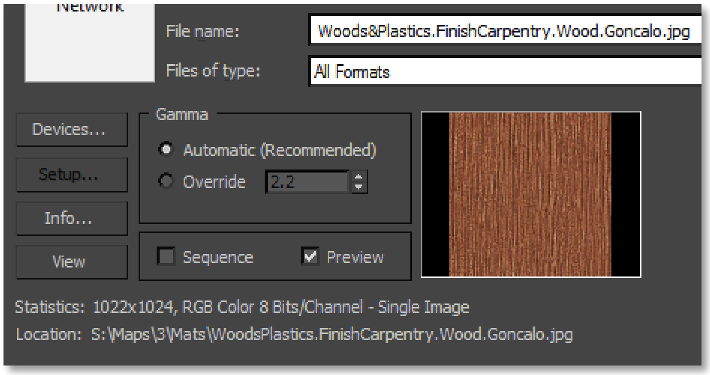
Numbers less that 1.0 de-gamma correct an image, *normalizing* it back to gamma of 1.0. Numbers greater than 1.0 add gamma correction, brightening the image.

Gamma-corrected maps typically look too bright in the renderer unless they are de-gamma corrected before rendering. In this image, bitmaps are gamma corrected on the left, and not gamma corrected on the right:



Not that gamma correction does not affect materials with color swatches (the lattice in red, above) or reflective surfaces (the teapot).

In the 3ds Max 2014 version of the *Select Bitmap Image File* dialog box you now have only two options for Gamma Correction: Automatic or Override:



File name: Woods&Plastics.FinishCarpentry.Wood.Goncalo.jpg

Files of type: All Formats

Gamma

● Automatic (Recommended)

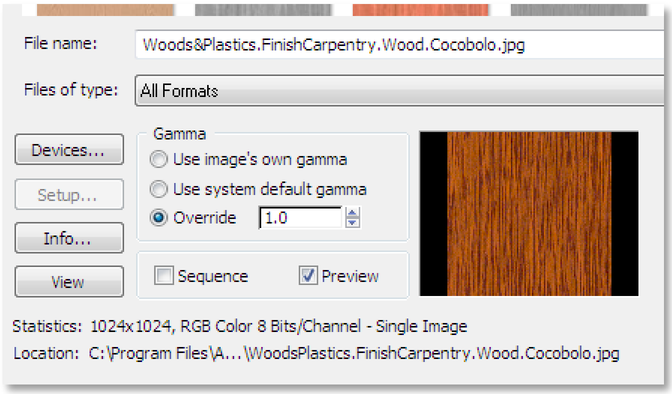
○ Override 2.2

□ Sequence ☑ Preview

Statistics: 1022x1024, RGB Color 8 Bits/Channel - Single Image

Location: S:\Maps\3\Mats\WoodsPlastics.FinishCarpentry.Wood.Goncalo.jpg

Below is the 3ds Max/Design 2013 *Select Bitmap Image File* dialog box's Gamma selections:



File name: Woods&Plastics.FinishCarpentry.Wood.Cocobolo.jpg

Files of type: All Formats

○ Use image's own gamma
○ Use system default gamma
● Override 1.0

□ Sequence ☑ Preview

Statistics: 1024x1024, RGB Color 8 Bits/Channel - Single Image
Location: C:\Program Files\A...\WoodsPlastics.FinishCarpentry.Wood.Cocobolo.jpg

With the Automatic option:

- High-Dynamic-Range Images (HDRI with floating-point color values) will use a gamma of 1.0 automatically
- 8 and 16-bit Images with embedded Gamma values (PNG, TIFF, etc.) will use the embedded value to de-gamma-correct the image
- 8 and 16-bit Images without Gamma values (JPEG, BMP, etc.) will use the default gamma value of 2.2.

Gamma Correction should be overridden if the image is used for *computation* instead of a surface color (diffuse) map, such as for Bump, Displacement, Normal Maps, Glossy Maps, etc. In this case use the Override option and set the value to 1.0, which disables Gamma correction.

Lower Thread Priority for mental ray®

Thread Priority is a way for Windows to allow high-priority applications to get more CPU time and potentially run faster than lower-priority applications. Applications you use typically all run with “Normal” thread priority.

In 3ds Max/Design 2014 mental ray will automatically use a “Below Normal” CPU thread priority in Windows when rendering, and the “Lowest” thread priority when rendering under Backburner. Foreground “Normal” priority applications will still be able to run while you render, though the render and the foreground application will be slower.

However, if you aren’t using your machine for foreground work, this will have no impact on render speed as mental ray is effectively the only program running. Be sure to disable unimportant applications and screen savers.

If you wish to change the Thread Priority options you can edit the “**mentalray_cpu.ini**” file in the **C:\Program Files\Autodesk\3ds Max Design 2014\plugcfg_ln** folder:

```
;
; - mentalray_cpu.ini
;
```

```
; Sets thread priority and thread count for mental ray renderer.
; Meaning of Thread Priority numbers are
;    0 == Normal
;   -1 == Below Normal
;   -2 == Lowest
; Meaning of Thread Count numbers are
;    0 == Use as many threads as there are cores
;    N == Use N threads
;   -N == Use as many threads as there are cores, minus N
;
[Threads]
;
; Settings for normal / interactive renders in the UI
;
;   Thread priority (defaults to -1 = "Below Normal")
RenderThreadPriority=-1

;   Thread count  (defaults to 0 = use all cores)
RenderThreadCount=0

;
; Settings for backburner / network renders
;
;   Thread priority (defaults to -2 = "Lowest")
BgRenderThreadPriority=-2

;   Thread count  (defaults to 0 = use all cores)
BgRenderThreadCount=0
```

Restart 3ds Max/Design to accept new options.

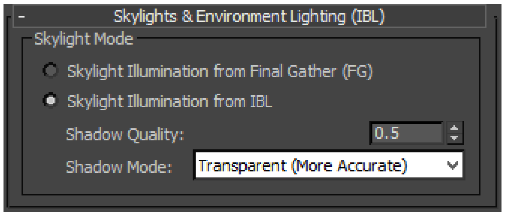
You can also control the number of cores mental ray will use. Remember that any changes will remain until you alter them, so if you limit the number of cores, all subsequent renders will be slower until you change it back.

And now for the new features of mental ray and Iray.

Rendering using Image-Based Lighting (IBL)

New in 3ds Max/Design 2014 is the ability to light a scene using an environment map, a technique known as Image Based Lighting, or IBL. The environment map is used with a Skylight object and a new render option called *Skylight Illumination from IBL* to light the scene. The steps to create IBL in your scenes are on pages 7 and 8.

In the Render Setup dialog box and the Global Illumination tab is the new *Skylight Illumination from IBL* option:


- Skylights & Environment Lighting (IBL)

Skylight Mode

○ Skylight Illumination from Final Gather (FG)

● Skylight Illumination from IBL

Shadow Quality: 0.5

Shadow Mode: Transparent (More Accurate)

This is the default for new scenes. (The “Global Illumination” tab was formerly labeled “Indirect Illumination”. “Photon Mapping” now more accurately describes the old Global Illumination feature.) The option “Skylight Illumination from Final Gather (FG)” is the classic option that requires physical lights to produce skylight illumination.

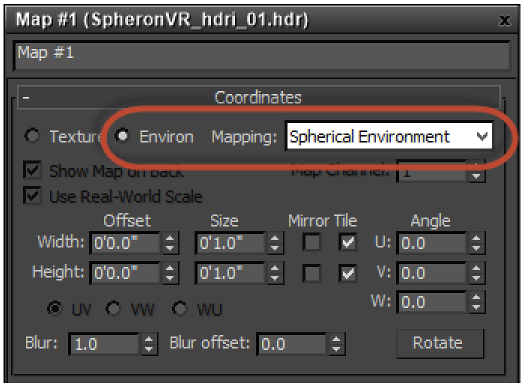
As the name of the tab suggests, this is a Global Illumination effect, and is not generally used for creating detailed shadows. Higher-resolution images generally produce sharper lighting effects when detailed lights are provided in the image, however these maps tend to get exceptionally large and can be more costly. Additional practical lights may be required if detailed shadows are required in an image.

That said, using the new **Unified Sampling** option (covered later in this document) results in improved IBL shadow samples. Since each sampling option (Classic or Unified) produces slightly different results, it is a good idea to compare them in your specific scenes.

The main advantage of producing illumination with an image is that it is much easier to light a scene to match a specific location, or to quickly create a particular look-and-feel. You can turn off Final Gather (FG) completely and still light your model, although we generally use FG for our scenes.

HDR images are available from outlets like Dosch Design, A&G Tool Co, and HDR Source, among others. You can create your own spherical HDR images with equipment from companies like SpheronVR and software from HDR Light Studio, HDR Shop, and Unified Color. Applications like HDR Light Studio allow you to create your own light maps, including “soft box” lights, to easily generate lighting for product or still-life renders.

New in 3ds Max/Design 2014: selecting a bitmap from the Environment and Effects dialog box and the Environment Map button, or dragging a bitmap from Windows Explorer into a viewport, now defaults the Bitmap map to a Spherical environment setting rather than as Texture or Screen mapping:



Map #1 (SpheronVR_hdri_01.hdr)

Map #1

Coordinates

○ Texture ● Environ Mapping: Spherical Environment

☑ Show Map on Back Map Channel: 1

☑ Use Real-World Scale

	Offset	Size	Mirror	Tile		Angle
Width:	0'0.0"	0'1.0"	□	☑	U:	0.0
Height:	0'0.0"	0'1.0"	□	☑	V:	0.0
					W:	0.0

● UV ○ VW ○ WU

Blur: 1.0 Blur offset: 0.0

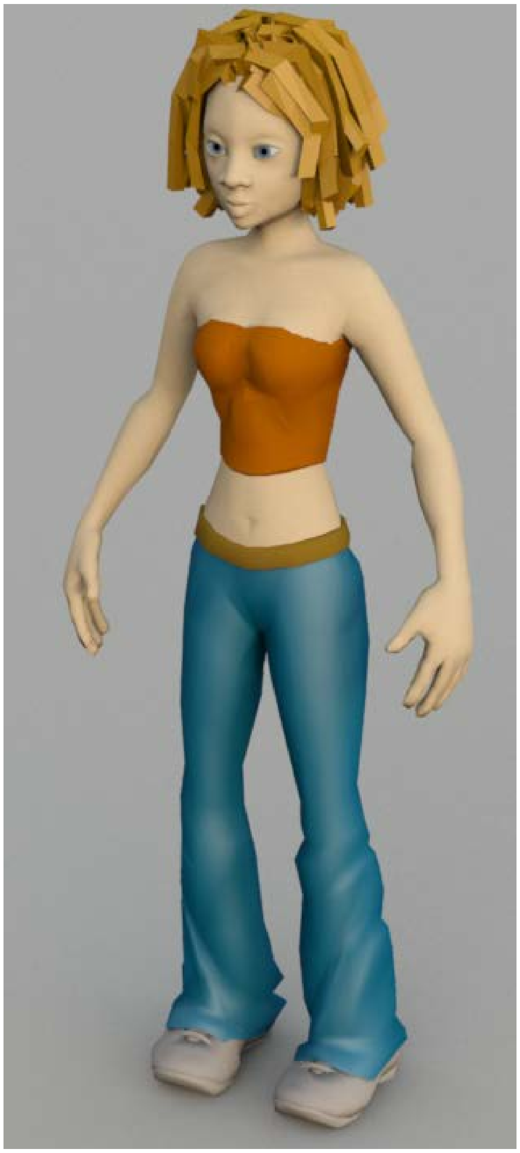
3ds Max files and images used in this document are available at: http://mastering-mentalray.com/AUniversity/2013/

To use IBL in your 3ds Max/Design scene:

1. Open or create a scene for use with IBL. The “IBL_start.max” file is available on the AU site for this class or at the link above.
2. Change to Unified Sampling and set Quality to 2.0.
3. In the Global Illumination panel, turn off the “Enable Final Gather” checkbox.
4. Set the “Skylights & Environment Lighting (IBL)” radio button to the “Skylight Illumination from IBL” option.
5. In your scene, create a Skylight object, found in the Create tab -> Lights category, and Standard drop down.
6. In the Modify panel, change the Skylight’s “Sky Color” option to the “Use Scene Environment” option.
7. Download the “SpheronVR_hdri_01.hdr” file from the AU page for this class and save it to your .\sceneassets folder.
8. Press “8” to open the “Environment and Effects” dialog box.
9. Set the Exposure Control to “mr Photographic Exposure Control” and set EV to 3.0.
10. Click on the “None” button in the “Background” section and choose “Gamma & Gain”.
11. Open the Slate material editor and left-click drag the new Gamma & Gain shader from the Environment and Effects dialog box to a Slate view. Choose “Instance”.
12. From the Material/Map Browser drag-drop a Bitmap into the view. In the Select Bitmap Image File dialog box choose the “SpheronVR_hdri_01.hdr” file.
13. Double-click the bitmap to open its parameters. Change the Coordinates to Environ and Spherical Environment.
14. In the HDRI Load Settings dialog box, choose “Real Pixels” and “Def. Exposure”, then press OK.
15. Wire the output of the new bitmap to the input of the Gamma & Gain shader.
16. Right-click the Gamma & Gain shader and choose “Show All Additional Params”, then press the “+” symbol next to “Additional Params”.
17. Change the Gamma setting to 1.0, the Gain to 20, and turn off the checkbox for Reverse Gamma Correction.
18. Render the camera view.

Unlike Iray where a single environment map produces illumination, mental ray allows you to have multiple Skylight objects with different maps. This could, for instance, allow you to create Skylights containing individual HDRI lights which you can adjust and rotate (via the U Offset in the bitmap) as needed.

Below is a character rendering using only Image Based Lighting and no Final Gather:



Using String Options in the new Render Setup UI and via MAXScript

mental ray String Options allow users to access advanced and unsupported render features within a DCC application. In the past you needed to know how to use MAXScript to enable and control these options; 3ds Max/Design now allows you to use options directly in the UI.

A typical MAXScript program may look like this:

```
m = mental_ray_string_options
m.AddOption "progressive" on
```

The first line creates a pointer to the mental ray string options, and makes for less typing in additional steps. The second line sets the string option, which will stay with the current session of 3ds Max/Design.

When multiple string option values are sent to mental ray they accumulate, and are not replaced. For instance, if you send an ‘on’ and later and ‘off’ for Progressive Rendering, both options are stored by mental ray and the *last* option sent is ultimately used.

A best practice is to remove options before writing new options, or after rendering, using this code:

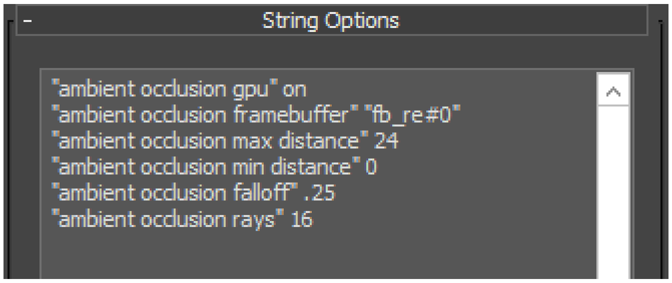
```
m = mental_ray_string_options
for i = m.numOptions to 1 by -1 do
        m.removeOption i
```

For more sophisticated programs you may want to only remove specific options with this code:

```
m = mental_ray_string_options
for i = m.numOptions to 1 by -1 where matchPattern (m.getOptionString i)
        pattern:"progressive*" do
                m.removeOption i
```

..and replace the “progressive” text with the String Option you with to remove. Since the last string option sent to mr will always be used, and it is not required to remove obsolete options.

New for 3ds Max/Design 2014 is the ability to place String Options into the Render Setup dialog box, and not have to worry about adding or removing options via MAXScript. In the Renderer tab, at the bottom, is the String Options roll-down:



```
"ambient occlusion gpu" on
"ambient occlusion framebuffer" "fb_re#0"
"ambient occlusion max distance" 24
"ambient occlusion min distance" 0
"ambient occlusion falloff" .25
"ambient occlusion rays" 16
```

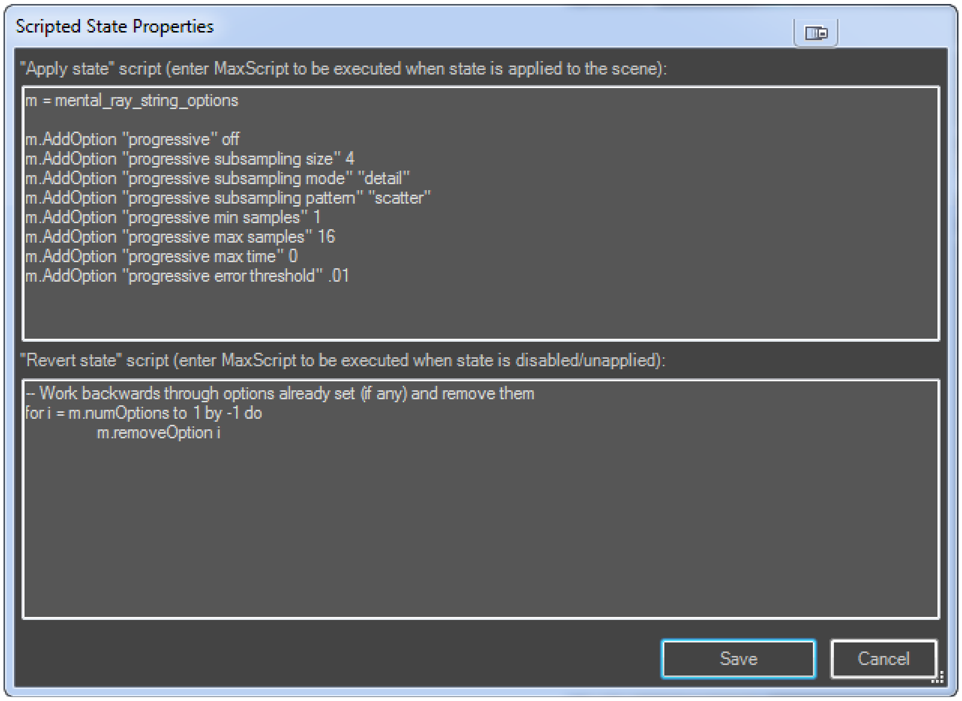
The options are automatically added and removed with each render. To add Progressive Rendering options to mental ray, for instance, copy/paste this code:

```
"progressive" on
"progressive subsampling size" 4
"progressive subsampling mode" "detail"
"progressive subsampling pattern" "scatter"
"progressive min samples" 1
"progressive max samples" 16
"progressive max time" 0
"progressive error threshold" .01
```

These options are documented in the mental ray documentation, available at http://docs.autodesk.com/MENTALRAY/2014/ENU/mental-ray-help//files/manual/index.html

Although the addition of the String Options in the Render Setup dialog box is exceptionally useful, understanding how to add String Options into a MAXScript, or into a Scripted State within State Sets, is essential if you may need different options for Production vs Draft rendering, for instance. The Progressive options above are not generally useful for production and cannot be used for texture baking, but are very useful during the iterative rendering process.

Using String Options are covered in the MAX Script documentation http://docs.autodesk.com/3DSMAX/14/ENU/MAXScript%20Help%202012/index.html?url=files/GUID-F64266AA-19D7-4F81-A3A7-90152F50F23-2402.htm,topicNumber=d28e770512



"Apply state" script (enter MaxScript to be executed when state is applied to the scene):

```
m = mental_ray_string_options

m.AddOption "progressive" off
m.AddOption "progressive subsampling size" 4
m.AddOption "progressive subsampling mode" "detail"
m.AddOption "progressive subsampling pattern" "scatter"
m.AddOption "progressive min samples" 1
m.AddOption "progressive max samples" 16
m.AddOption "progressive max time" 0
m.AddOption "progressive error threshold" .01
```

"Revert state" script (enter MaxScript to be executed when state is disabled/unapplied):

```
-- Work backwards through options already set (if any) and remove them
for i = m.numOptions to 1 by -1 do
        m.removeOption i
```

Ideally, use only one method of setting String Options to avoid confusion. Changes to String Options in the Render setup dialog box are not recorded in State Sets.

You can list what options are enabled in your scene with this code, which you can run in the MAXScript Editor window and view the output in the MAXScript Listener window:

```
m = mental_ray_string_options
for i = 1 to m.numOptions do
        format "Option %: % = %\n" i (m.GetOptionString i) (m.GetOptionValue i)
```

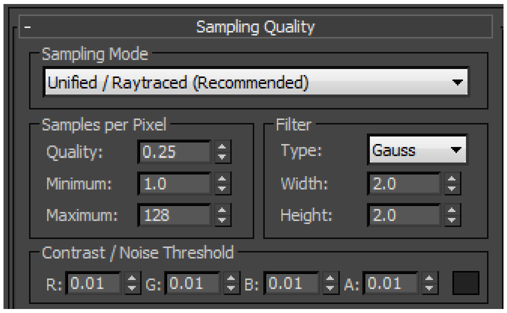
String Options are not permanent and are cleared once 3ds Max and you scene is reloaded, therefore, placing the options in the Render Setup dialog box, in State Sets, or as a Pre-Render Script, ensures that they are set properly with each render.

Next I cover the new Unified Sampling option in mental ray.

Using Unified Sampling in mental ray

Unified Sampling is the new easier and faster algorithm for rendering scenes in mental ray. With Unified Sampling you have a simple Quality setting that controls image sampling within a large range. Unified Sampling works not only on the pixels of your image but also works in time, which allows it to render effects such as Motion Blur very quickly.

Below are the new Render Settings dialog roll-down for Sampling Quality showing Unified / Raytraced settings:



Sampling Quality

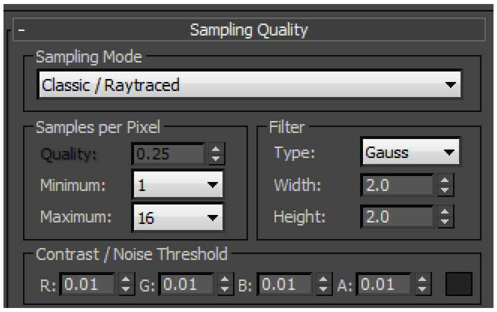
Sampling Mode: Unified / Raytraced (Recommended)

Samples per Pixel
Quality: 0.25
Minimum: 1.0
Maximum: 128

Filter
Type: Gauss
Width: 2.0
Height: 2.0

Contrast / Noise Threshold
R: 0.01 G: 0.01 B: 0.01 A: 0.01

Below are the settings for the old Classic / Raytraced settings:



Sampling Quality

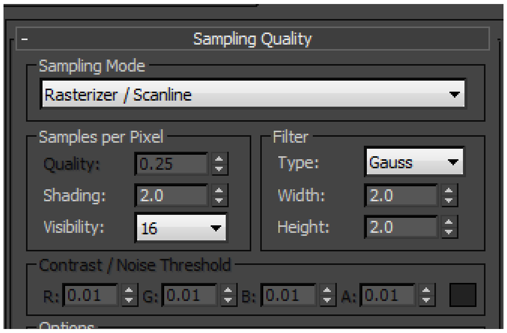
Sampling Mode: Classic / Raytraced

Samples per Pixel
Quality: 0.25
Minimum: 1
Maximum: 16

Filter
Type: Gauss
Width: 2.0
Height: 2.0

Contrast / Noise Threshold
R: 0.01 G: 0.01 B: 0.01 A: 0.01

A fast Rasterize / Scanline option is also available:



Sampling Quality

Sampling Mode: Rasterizer / Scanline

Samples per Pixel
Quality: 0.25
Shading: 2.0
Visibility: 16

Filter
Type: Gauss
Width: 2.0
Height: 2.0

Contrast / Noise Threshold
R: 0.01 G: 0.01 B: 0.01 A: 0.01

Options

The benefits of Unified Sampling are:

- Accelerated ray-traced Motion Blur.
- Better able to render small details.
- Improved IBL shadow samples

- Generally faster rendering of all scenes, including images with Depth of Field.

Exploring Unified Sampling Settings

Quality:

This is a noise level threshold, and higher values deliver more refined images. The Quality value range is **0.1** to **20.0**, and the default is **0.25**. This works with the Min/Max and Contrast settings. Areas with more noise adaptively get more attention. Using large **Quality** settings may require the **Max** settings to be correspondingly larger.

Min / Max:

This is the sampling range the Quality setting work between; the actual sampling (number of rays cast per-pixel) is determined by the Quality setting, with Min and Max as an upper and lower sampling boundary.

The **Min** range is **0.1** to **64.0** (decimal), with the default of **1.0**. Numbers less than 1.0 start grouping pixels together. A Min of **1.0** (1 ray per pixel) is recommended for most purposes. My experience is that very small values (**0.1**) will lose details in your image; if it first looks at too large of an area it may not see the contrast change it needs to resolve the image.

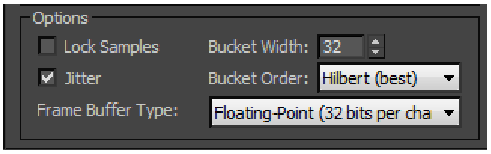
The **Max** range is **1** to **100000** (integer). Default is **128**. A Max of 128 is useful for most design visualization purposes; however large values will be required for both Motion Blur, and to resolve fine details.

Unlike Classic Min/Max sampling, these can be any number in its respective ranges, and a large range may not necessarily generate long renders as the Quality setting is still in control. For instance, setting Max to **1024** and Quality to **0.75** is likely just as fast as a Max of **64**; the Quality value limits the sampling automatically. Conversely, large Quality settings may require larger Max settings to be effective. Setting Quality to **20** if Max is **64** likely generates the same results if Quality is set to **4**.

Contrast / Noise Threshold:

The same as classic Spatial Contrast, and this uses RGB Values from **0.0** to **1.0**. Default value is **0.01**. This is essentially an error threshold, where when one sample (a pixel or sub-pixel) is different from another sample by a color value greater than the Contrast color, additional sampling (subdivision) occurs. To further optimize render speed, these values are weighted internally based on how human vision works, i.e. the color blue is less sensitive in the human eye therefore it is a waste of time to subdivide based on blue.

The other Sampling options work as before, however the algorithm for Unified Sampling jitters in normal operation.



☐ Lock Samples — Bucket Width: 32

☑ Jitter — Bucket Order: Hilbert (best)

Frame Buffer Type: Floating-Point (32 bits per cha

Analyzing Unified Sampling Speed

Unified Sampling can share rays for glossy, shadows, motion blur, etc, for speed improvement up to 10x for effects like Motion Blur.

Generally, expect about a 25% increase in speed for static images for the same level of image quality. Rendering of effects such as Motion Blur may be as much as 1000% faster than Classic sampling, and usually is just slightly slower than without the effect.

Practical Unified Sampling Settings

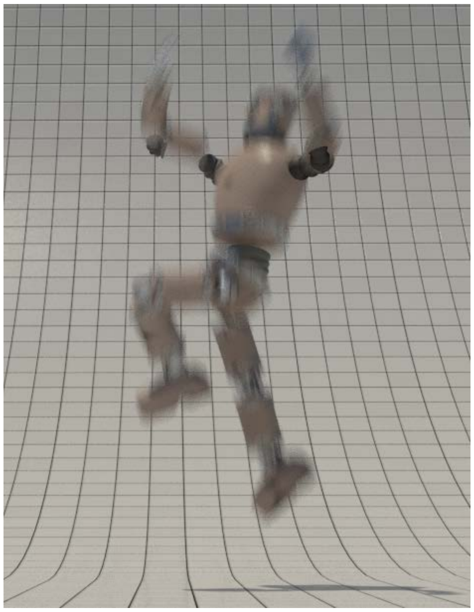
- Quality setting for most scenes range from 0.05 (draft) to 1.5 or 2.0 if more detail is needed.
- A Quality setting of 0.5 works for many scenes.
- Final still-image renders are usually 0.75 to 1.0.

Use Quality greater than 2.5 with caution. This is only really needed for high-res product renders and, if need, for more refinement in Motion Blur or IBL shadows.

Motion Blur:

Unified Sampling happens in both time and space, allowing for greater render efficiency for Motion Blur. With Unified Sampling, Motion Blur can be as fast as raster post-process methods of generating motion blur.

For a 130-frame motion-blurred robot animation benchmark scene of a Tornado Kick (robot courtesy of Zap Andersson):



HDR Motion Blur camera shader vs Motion Blur, (130 frame average):

Classic w/HDR Motion Blur:	**0239s**/frame
Classic w/mr Motion Blur:	**1206s**/frame
Unified w/HDR Motion Blur:	**0178s**/frame
Unified w/mr Motion Blur:	**0179s**/frame

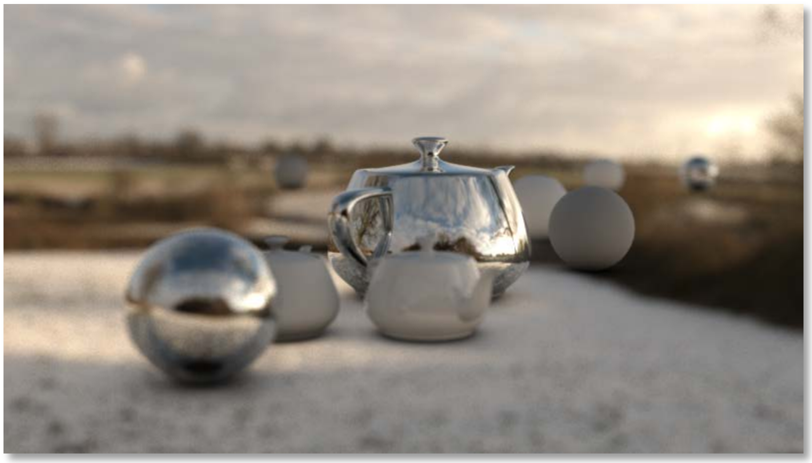
Classic Sampling = **1** / **16**
Unified Sampling Q=**0.75**

This animation can be downloaded from http://mastering-mentalray.com/AUniversity/2013/MotionBlur/

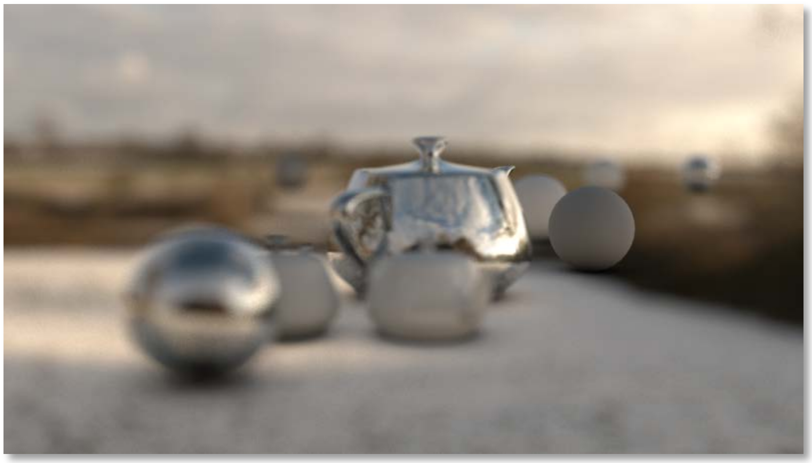
Comparing DOF Camera Shader vs Multi-Pass Effect:

The speed of Motion Blur and Unified Sampling gets the most attention, however our experience is that Depth-of-Field effects also are highly accelerated when using Unified Sampling.

Rendered using the DOF camera shader:



Rendered using the mental ray Multi-Pass effect in the camera:



Benchmarking using the DOF Camera Shader:

Classic Sampling 1/16 No DOF:	**032s**
Classic Sampling 1/16 w/DOF:	**181s**
Unified Sampling Q=1 No DOF:	**029s**
Unified Sampling Q=1 w/DOF:	**107s**

Benchmarking using the mr Multi-Pass Effect:

Classic No DOF:	**032s**	(1/16)
Classic Sampling w/ DOF:	**151s**	
Unified Sampling No DOF:	**029s**	(Q=1 Min=1 Max=128)
Unified Sampling Q=1:	**042s**	
Unified Sampling Q=3:	**153s**	
Unified Sampling Q=8:	**151s**	(no change >8 w/ Max=128)
Unified Sampling Q=10:	**215s**	(Min=4 Max=256)

Above, with Q > 8 there was no change in render times until the Max value was increased to 256. For still-image renders, a high Max value and a Q of 5 to 10 is reasonable.

Using Unified Sampling in older versions of 3ds Max/Design

The Unified Sampling technology was added in mental ray 3.9 and the 2012 versions of Autodesk products. You can access these newer features via String Options; the MAXScript may look like this:

```
m = mental_ray_string_options
for i = m.numOptions to 1 by -1 do
m.removeOption i

m.AddOption "unified sampling" on
m.AddOption "samples min" 1.0
m.AddOption "samples max" 128
m.AddOption "samples quality" 0.75
m.AddOption "samples error cutoff" [0.01, 0.01, 0.01, 0.01]
```

Next I cover Iray in detail, including the new Architectural and Caustics rendering modes and the Iray material.

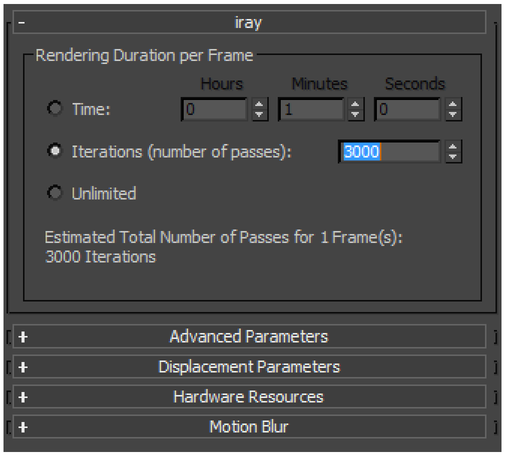
Introducing the new Architectural and Caustics modes in Iray®

First, I cover Iray in general before getting to the new Architectural and Caustics features and Iray material.

What is Iray?

Iray is a point-and-click photo-realistic rendering mode for mental ray, which appears in 3ds Max/Design as a separate renderer. It first appeared in 3ds Max 2011, and has evolved quickly since then.

Iray has simple controls for image quality based on Iterations or Time. It is primarily a GPU-based renderer and works fasted on high-end game and workstation-class GPUs, however it also supports efficient CPU-only rendering.



- iray

Rendering Duration per Frame

○ Time: Hours 0, Minutes 1, Seconds 0

● Iterations (number of passes): 3000

○ Unlimited

Estimated Total Number of Passes for 1 Frame(s):
3000 Iterations

+ Advanced Parameters
+ Displacement Parameters
+ Hardware Resources
+ Motion Blur

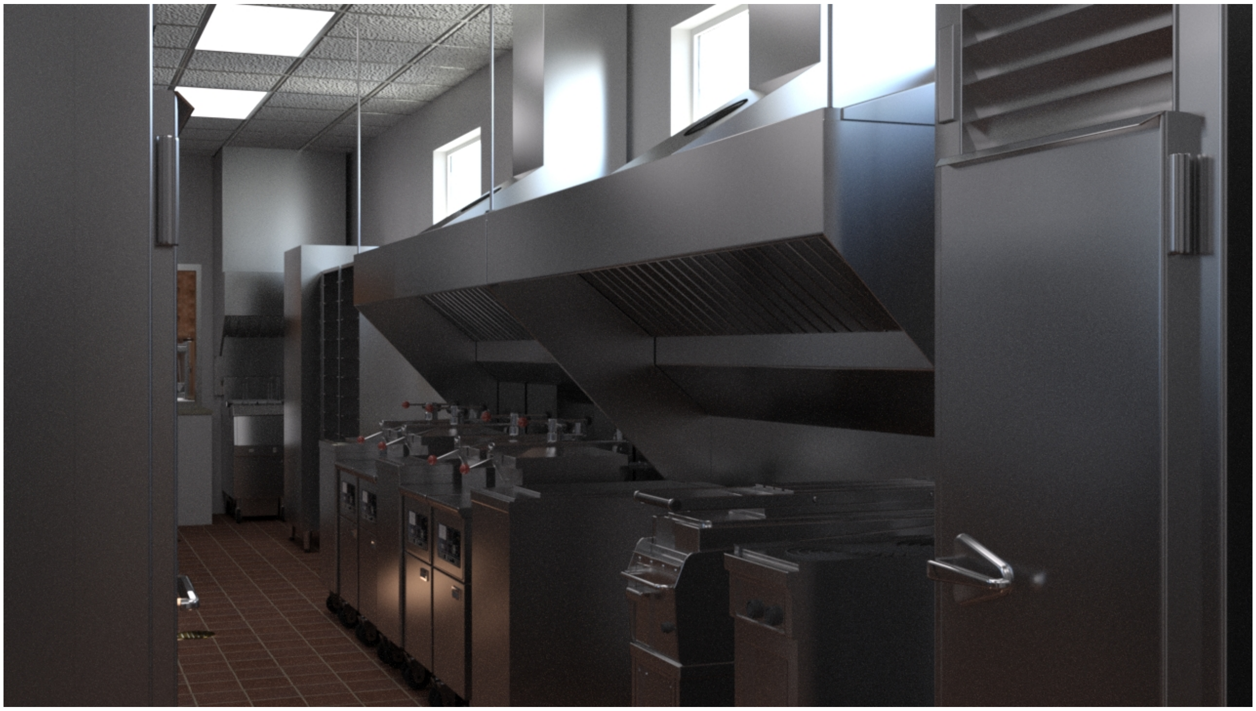
Why Use Iray?

- Easy to use, point-and-click rendering.
- Progressive render for quick previews
- Uses all Autodesk and Arch & Design (mental ray) Materials.
- Supports modern, advanced Iray-specific materials (Iray material is covered at the end of this document)
- Fast rendering of numerous photometric area light sources
- Automatic Image-Based Lighting
- Fast Motion Blur and DOF effects.
- Geometry lighting through self-illuminated materials
- Matte/Shadow/Reflection support in Iray 3
- mr Sky Portal support for small windows

Why Not Use Iray?

- You typically need fast NVIDIA GPUs with large memory for practical render times in most scenes. That said, if there are a lot of lights or reflective surfaces, Iray may still beat mental ray using CPU-only.
- No Render Elements in 3ds Max.
- Some maps, materials, and plugins may not be supported, although more support is added with each new version.
- Global Illumination is not interpolated across a surface, which may lead to sharp shadow artifacts on low-poly objects.

In the image below, mental ray had a difficult time generating smooth reflections in the stainless surfaces, and producing a smooth lighting simulation. It was exceptionally spotty and blotchy. This was the first project we did with Iray, and the results were what we needed, albeit 17 hours of render time on my quad-core laptop.



The client loved the results, and we are still doing primarily Iray rendering for them three years later. In benchmarks, current versions of Iray running on a fast 6-core i7 machine are about as fast using CPU-only rendering as it is running on a Quadro 5000.

Considerations when using Iray®

Iterations

Generally, the Iterations setting is your image quality control setting. We have a variety of hardware and using Iterations is the only way to endure that the same amount of work is being done on each image. Iray progressively refines an image until it reaches the time or iterations limits. For most interior scenes, 3000 to 5000 iterations are needed. For exterior day-lit scenes, 1500 to 3000 iterations are the norm.

Image Size and Noise

For most scenes we will render an image double the size needed, but this depends a bit on what the image will be viewed on, or if it is going to print. If we are sending the image to a client to be viewed in 2k then a final-quality mage will be rendered at 4k resolution, 4x the number of pixels, then reduced in Photoshop. This eliminates any visual artifacts and allows us to use less iterations than if we tried to get a noise-free image at 2k resolution. It does take more time and is generally for final-quality images.

If an image is going to print we will likely not render larger than what is needed, and will produce an image that is from 100 to 150DPI for up to a 24x36 print; larger images may be 50 to 150 DPI. The process of printing will smooth and blends pixels, and makes high DPI and extreme render sizes unnecessary. For larger images there are plenty of tools that will resize images into the size a client requires without extreme render sizes. Large prints are typically viewed at greater distances, and are typically much lower resolution that you might imagine.

Memory

Your scene polygons and bitmaps must fit into the physical memory of the GPU cards in your machine. If you have multiple GPUs, it does not combine memory (two 3Gb cards does not equal 6Gb with SLI), and each card must have enough memory. If any one card is insufficient, then that one card is disabled and those with enough memory will continue. If none have enough, Iray uses CPU-only rendering.

It is important to optimize your geometry wherever possible and hide anything that isn't being rendered, or remove it from the scene. We use a lot of XRef scenes to control what is loaded and also allow us to divide production among multiple artists.

For Revit scenes, eliminating structure hidden in walls and ceilings is usually the first task. Pierre-Felix Breton at Autodesk has a MAXScript which will organize objects into layers based on Category and Type. It is available at: http://www.pfbreton.com/2011/11/set-layers-by-revit-category-and-family/ and on the www.mastering-mentalray.com site.

Textures are another area where using something that is, for instance, 512x512 rather than 1024x1024, can have a big impact on memory use. For users that use large satellite images that cover a wide area, for instance, Iray may not be practical without newer 12Gb GPUs.

Iray, unlike mental ray, cannot load geometry on-demand and must hold all objects and textures in GPU memory. Memory economies that you might realize with instancing and mr Proxy objects do not help with Iray as each object is fully realized in the GPU.

Most of the architectural scenes we do fit within 1.5Gb and 3Gb cards. We had one particularly large scene this year with a lot of 3ds solid models in a very large Revit building that took over 4gb of memory. At this point we only buy GPUs with 6Gb of memory or more (Titans, for instance).

Map and Materials Support

Architectural, Composite, Ink'n'Paint, Raytrace, Shellac, Top/Bottom, and Car Paint materials will render gray. Any mental ray camera, output or effect shaders can't be used. Standard materials are now supported.

The current list of supported maps are: Checker, Color Correction, Dent, Gradient Ramp, Marble, mr Ocean, Perlin Marble, Speckle, Substance, Tile, Waves, and Wood. This covers most that you would need for design visualization.

Plugin Support

Certain plugins, such as RPC, are an issue. It is impossible to list every issue that might come up for you, so it is important to test your plugins before committing to Iray. Because we rely heavily on Iray we have shifted over content that we need to compatible products. For plants and trees we go with XFrog and Marlin, among others, and edit materials as necessary. iToo's ForstPack plants work, but he automated proxy tools currently have issues.

GPU Hardware: GTX, Quadro and Tesla

A Graphics Processing Unit, or GPU, is at the heart of the speed you will experience with Iray. The CPU certainly contributes, and older versions of Iray suffered a penalty when using CPU. Now you can use all the resources in your machine.

When Iray was first introduced there was a lot of discussion about the use of game-rated GTX graphics cards vs workstation-class Quadro and Tesla class cards. The concern was that the GTX cards would burn-out, as they were not built for 24/7 rendering. Some people have had issues, and that may be attributed to the cards or the amount of air circulation, it is hard to tell.

Our experience has been that the GTX cards hold-up perfectly well in systems designed for cooling high-performance hardware. The use of CPU water-cooling to keep internal temperatures to a minimum, and using computer cases with a large amount of incoming cool air, have resulted in no failures in GTX hardware in our systems.

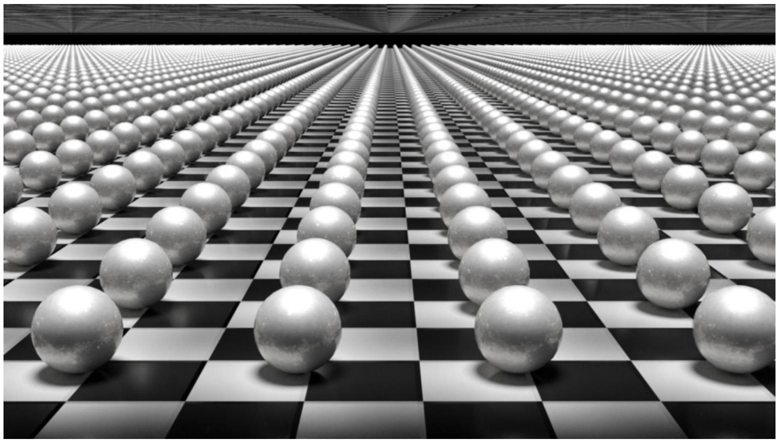
The folks at migenius have benchmarked GPU hardware and came up with some numbers, and you can find those here from mi genius: http://www.migenius.com/products/nvidia-iray/iray-benchmarks

The best bang for the buck is the GTX 580 (up to 3Gb memory) and the Titan card (6Gb memory). If you don't need the memory – and most of our scenes fit into 3Gb – then certainly use the GTX 580. We have a number of GTX 580 and Titan cards and they work great.

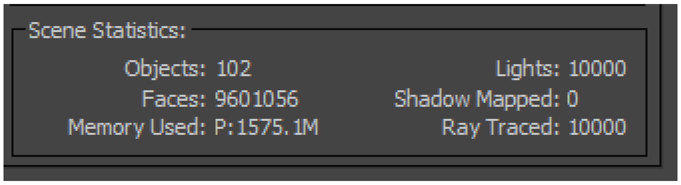
The Titan cards are built to the same specifications as Quadro/Tesla, and under the watchful eye of NVIDIA, and should be rugged enough to take continuous rendering. We are currently buying only Titan cards.

Using Unlimited Lights

One significant advantage to Iray over other rendering technologies is that Iray an extraordinary number of lights you have in your scene without a hit on render time. In the full-HD image below there are 10,000 photometric raytraced area lights, rendering 10,000 car-paint spheres on a checkboard pattern, rendered in full HD on 6 GPUs in 55 minutes:



By adding an HDR environment map and reducing the iterations, this image can be done in under 10 minutes with reasonable aliasing of the checkerboard pattern. These are the scene statistics for the image (note that 100 spheres are 1 object):

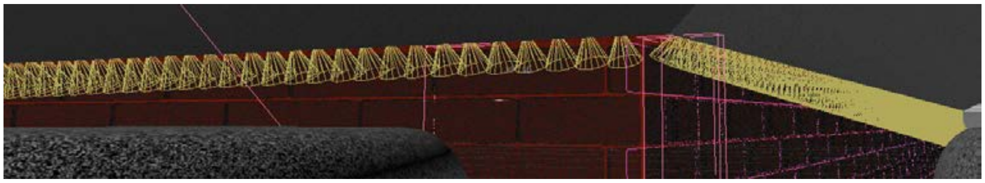


Scene Statistics:

Objects: 102
Faces: 9601056
Memory Used: P:1575.1M
Lights: 10000
Shadow Mapped: 0
Ray Traced: 10000

A vast majority of our scenes require accurate simulation of illumination. We will model or place every LED in an LED fixture or string of lights to provide accurate illumination and distribution. In some scenes we will model the reflector or provide additional geometry to control the distribution of each LED, as shown in this image:

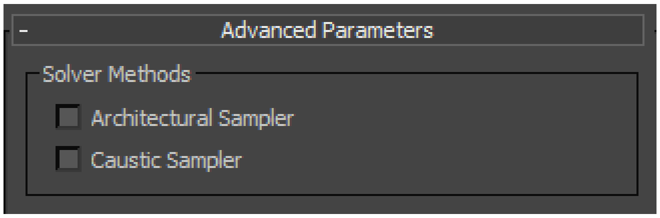
Rendering a scene with every LED in extensive exterior signage, or under-counter illumination along bars and counters, would be very lengthy in mental ray. With Iray we can come closer to an accurate lighting simulation for our clients and not have to worry about the quantity of lights. Below is the under-bar lighting to the interior scenes shown earlier:



In many scenes we will use self-illuminated materials and surfaces if the lighting does not need to be directed or exacting, as with the spot lights above. Unlike mental ray in 3ds Max and the A&D/Autodesk materials, this is true object lighting.

Using Iray® Architectural and Caustics Modes

New in 3ds Max/Design 2014 is the Architectural and Caustics rendering options:



- Advanced Parameters

Solver Methods

□ Architectural Sampler

□ Caustic Sampler

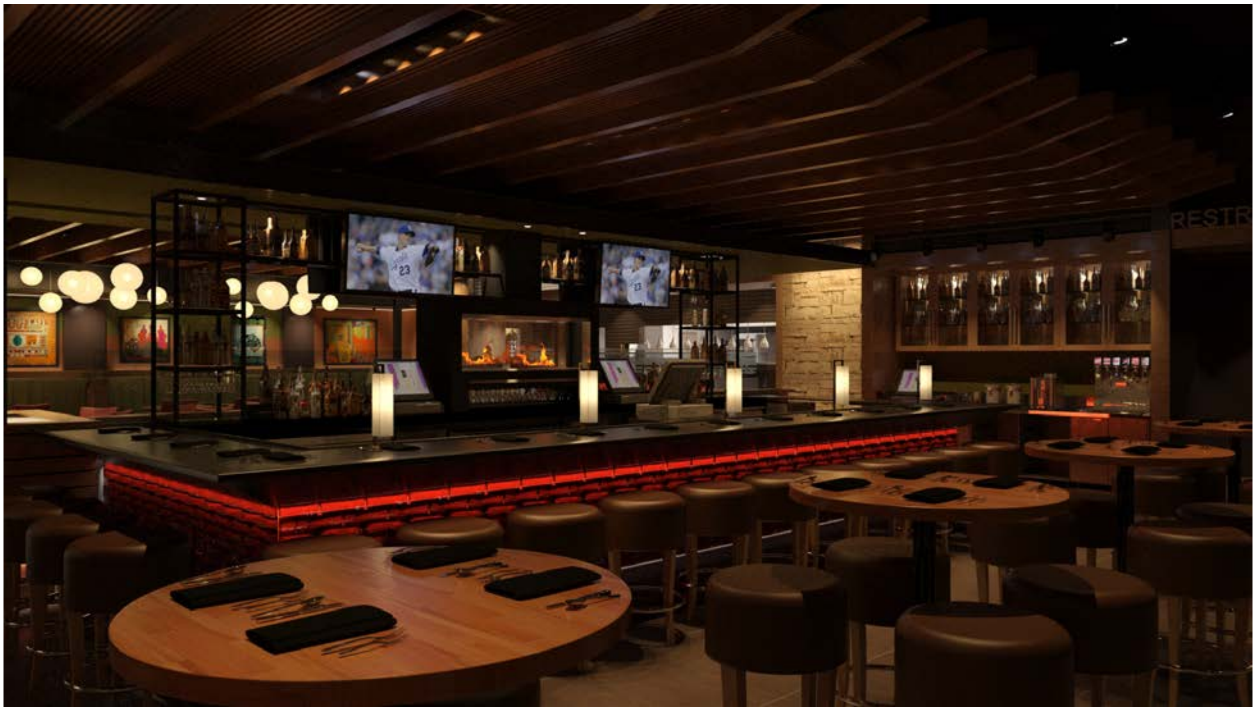
The Architectural option is intended for use in complex interiors, and is generally unnecessary for exterior day-lit scenes. It generates superior images, but requires 25% to 50% more iterations than without this option in order to produce the same quality level. Also note that much of the brightness in an Architectural Sampler rendering will come later in the process, so if you are testing this option, be sure to let it run for a while before adjusting exposure, etc.

The benefits of Architectural mode are:

- Brighter images with better Indirect Illumination
- Better glossy reflections
- Better interior illumination through small portals
- Better for translucent surfaces
- Better for Sub-Surface Scattering materials

Architectural is our preferred option for interiors, and Caustics when the effect is noticeable.

Below is an image rendered without the Architectural and Caustic options:

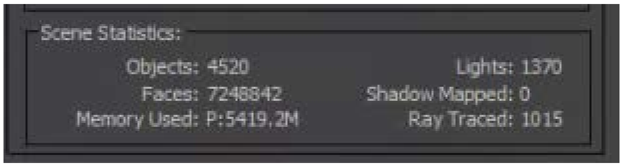


The image below has both the Architectural and Caustic effects.

It is difficult to compare these images by looking at them on the page, and I invite you to download and compare them in the RAM Player of 3ds Max. You can download the images from here: http://mastering-mentalray.com/AUniversity/2013/Architectural/

I've also included images and comparisons online for just Architectural and just Caustics.

These are rendertime Scene Statistics for these images (the non-Raytraced lights are hidden/unused Revit lights):



Scene Statistics:

Objects: 4520	Lights: 1370
Faces: 7248842	Shadow Mapped: 0
Memory Used: P:5419.2M	Ray Traced: 1015

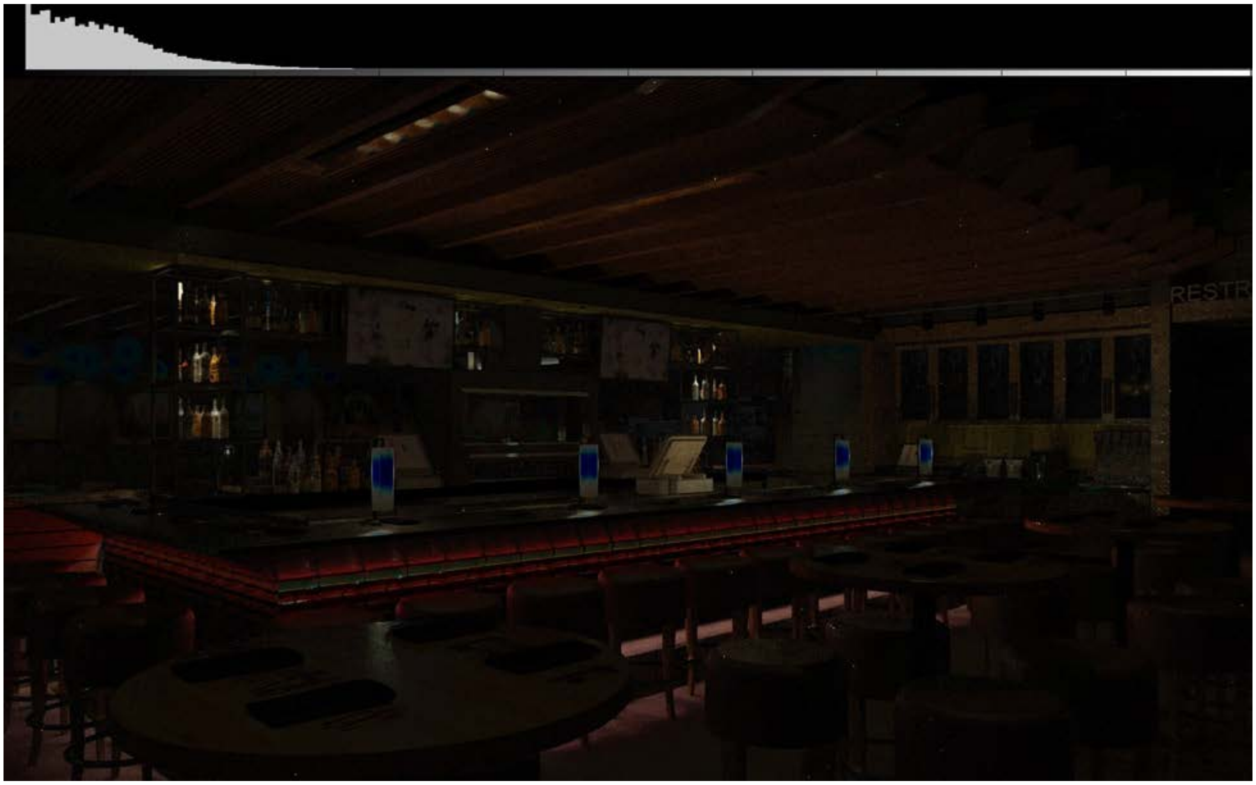
The render times at 1920x1080 and 5000 iterations, with 6 Titan GPUs were:

• Plain (no Architectural or Caustics)	047.5 minutes
• Architectural	078.3 minutes
• Caustics	075.25 minutes
• Architectural + Caustics:	135.25 minutes

NVIDIA provides an application in the 3ds Max/Design installation folder, NVIDIA sub-folder, called "imf_diff.exe". This program will perform various comparisons and image processing operations on images; type "imf_diff /?" in the NVIDIA folder to see a complete list. For this document it is most useful as it will create a new image that is the difference between two source images. Open up a command window and enter the following to compare the architectural and caustics image with one without those two features:

```
"C:\Program Files\Autodesk\3ds Max Design 2014\NVIDIA\imf_diff.exe"
Interior_Bar_view01_ArchiCaustic.jpg Interior_Bar_view01_Plain.jpg
PlainArchiCausticDiff.jpg
```

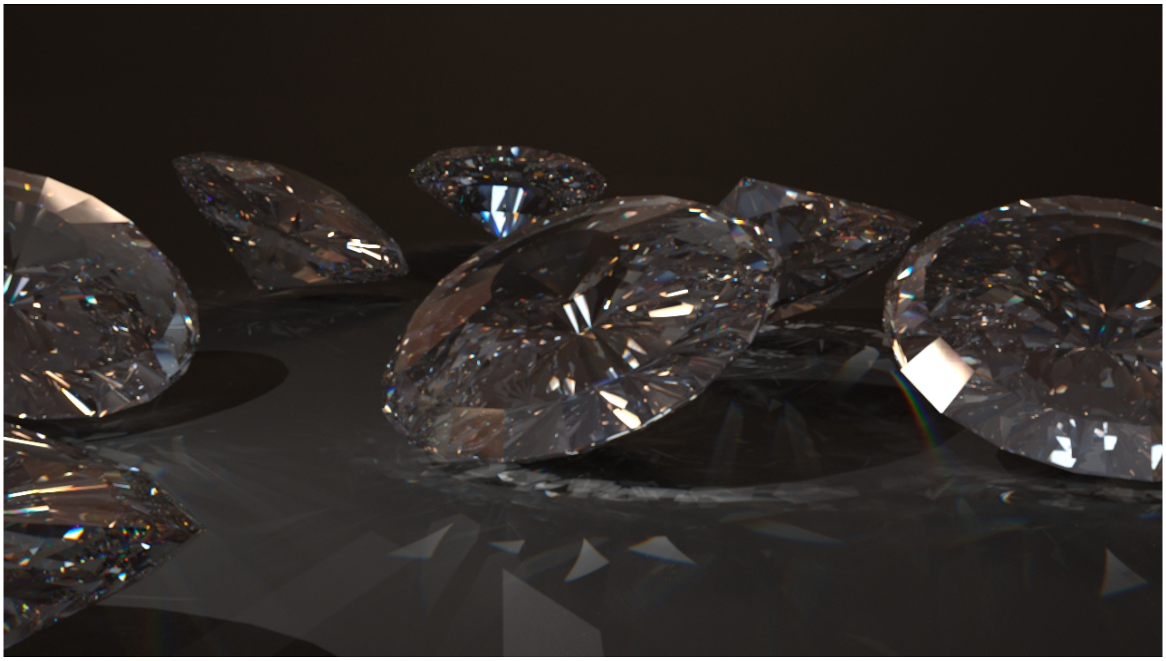
Enter this as all one line. The quotation marks are required as there are spaces in the command line. A batch file is on the link above, as are the processed files. This is the output of the command above:



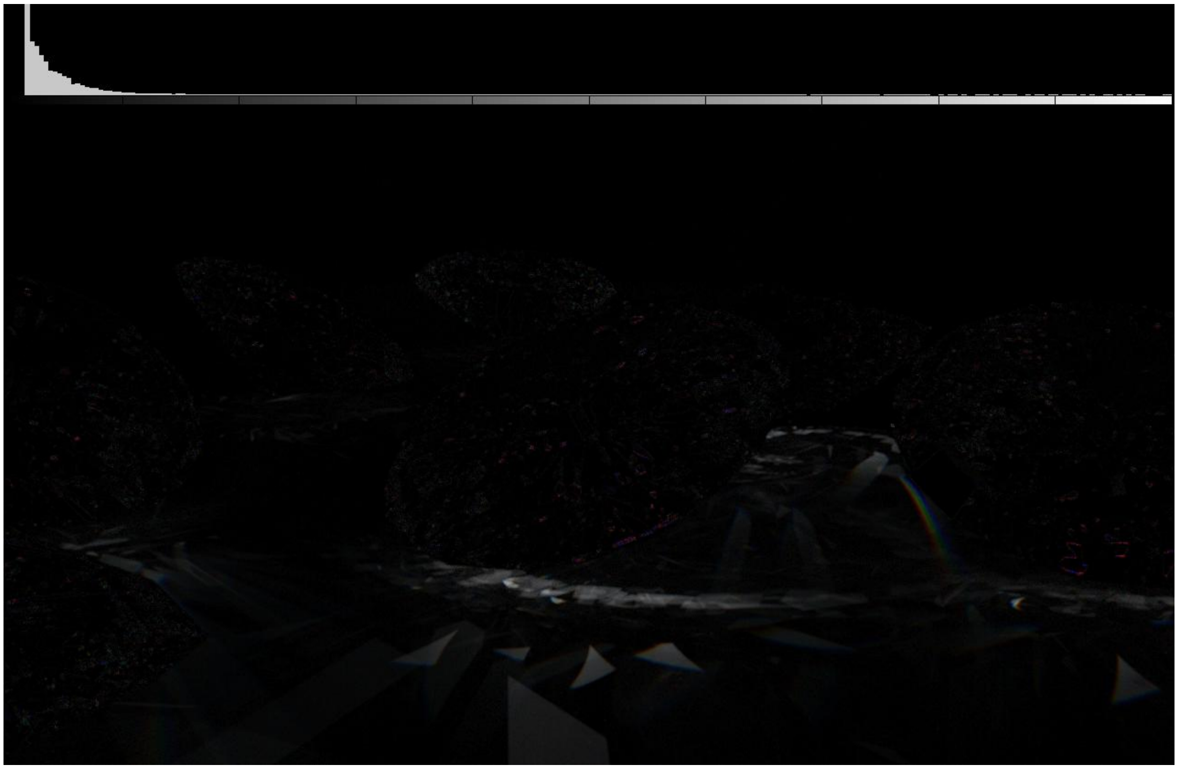
The image includes a histogram along the top. The imf_diff program is documented in the mental ray online documents: http://docs.autodesk.com/MENTALRAY/2014/ENU/mental-ray-help//files/manual/imf_diff.html.

Using Caustics

With scenes where caustics are obvious, the new Caustics option produces spectacular results whereas previous there would be no caustic effects:



In previous iterations of Iray, and in the current version without Caustics enabled, this would be the imf_diff comparison with the above and a no-caustics image:

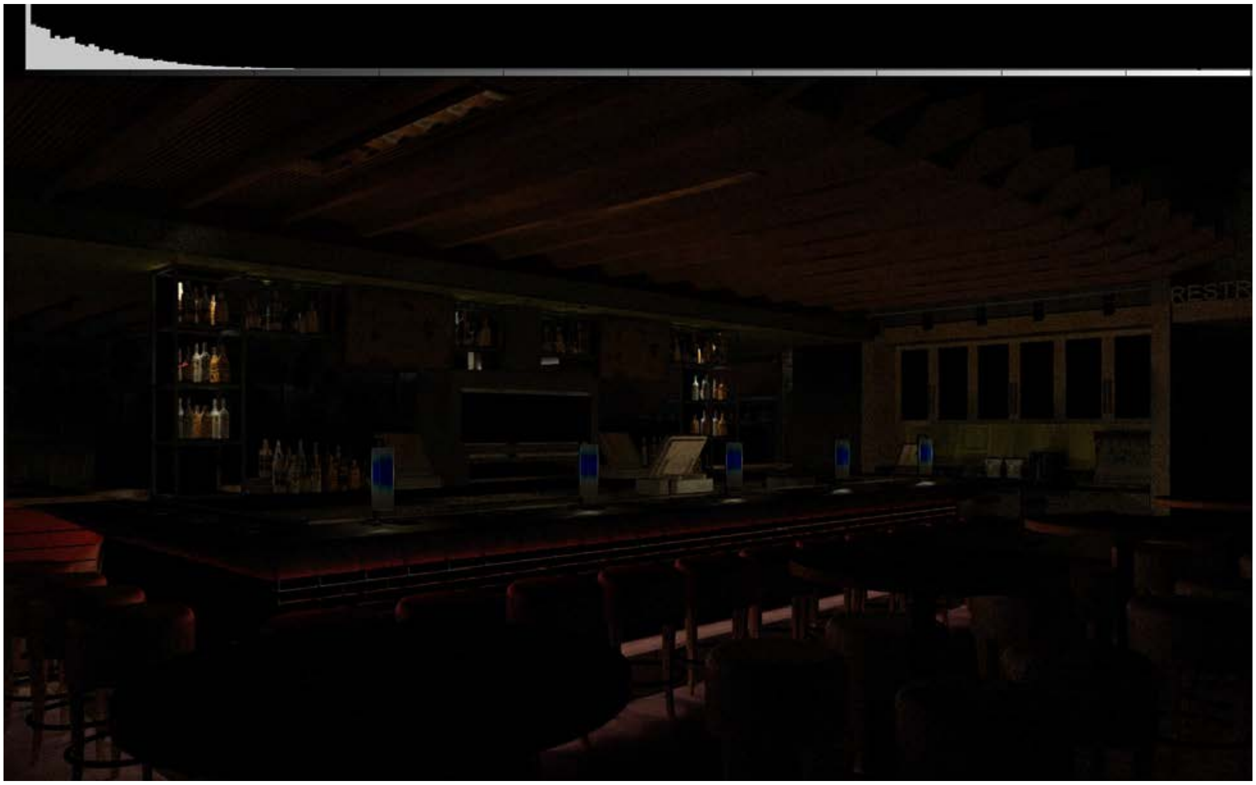


An image without caustics shows just a hint of caustics in a few areas, and it does not look like caustics. Caustics and non-caustics images and a mesmerizing animation are at: http://mastering-mentalray.com/AUniversity/2013/Caustics/

The Caustics image was 16 minutes to render vs. the non-Caustics image at 8 minutes, exactly double the render time for caustics. This scene uses the Iray material and the Diamond preset, along with an HDRI for reflection and global illumination, and a single spot light for focused light and shadows. The diamond scene was provided free from the Turbosquid web site and the user eruultmax. http://www.turbosquid.com/3d-models/free-diamonds-3d-model/602353

Generally, both the Architectural and Caustics options should be utilized only when the effect is needed, as there is a definite time penalty.

For the architectural interior scene rendered earlier, the difference between no caustic and just the caustics option are this:



If you download and compare the images yourself you'll see that this scene was brighter with the caustics option. The front of the bar has red glass brick and the caustic option allowed significantly more light to propagate and reach the step and floor. The glass liquor displays also showed a lot more brightness, and additional light is reflected onto the ceiling.

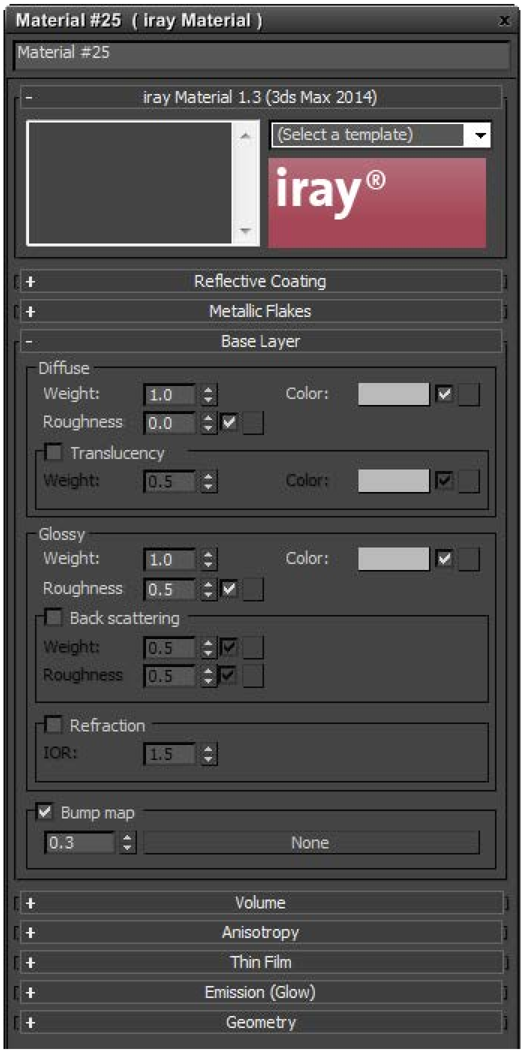
Using Iray® Materials

Although this doesn’t come stock with 3ds Max/Design, the developers at NVIDIA ARC have provided a special material that takes advantage of features and optimizations not available through the Arch & Design material. The material plugin can be obtained from here: http://blog.irayrender.com/post/52945163035/iray-material-plugin-for-3ds-max

Once you download the Zip file for your version of 3ds Max you’ll find a PDF that explains the Iray material in detail. I will not rehash that well-written material, and instead I’ll give you an overview, cover our impressions on the material, and detail practical applications. At right is the dialog box for the Iray material.

Layered and BSDF Materials

The Iray material is a new *Layered* material type, which can efficiently use rays to generate each layer, providing some measure of efficiency. It is also a Bidirectional Scattering Distribution Function (BSDF) material. A BSDF material combines classic Bidirectional Reflectance Distribution Function (BRDF) and Bidirectional Transmittance Distribution Function (BTDF).



Material #25 (iray Material)

Material #25

iray Material 1.3 (3ds Max 2014)

(Select a template)

Diffuse
Weight: 1.0 Color: ☑
Roughness 0.0 ☑
□ Translucency
Weight: 0.5 Color: ☑

Glossy
Weight: 1.0 Color: ☑
Roughness 0.5 ☑
□ Back scattering
Weight: 0.5 ☑
Roughness 0.5 ☑
□ Refraction
IOR: 1.5

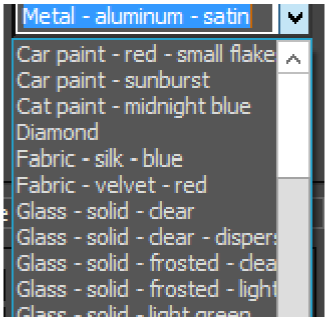
☑ Bump map
0.3 None

Each roll-down menu area represents a layer or an effect. The components of this material are:

- 3 Weighted Layers:
    - Top: Reflective Coating, including:
        - Roughness
        - Fresnel or Custom Curve
        - Bump
    - Middle: Metallic Flakes
    - Bottom: Base Layer, including:
        - Diffuse
            - Roughness
            - Translucency
        - Glossy
            - Backscattering (fabrics like velvet, silk)
        - Refraction (transparency)
        - Bump
- Volume Effects
    - Absorption (colored glass)
    - Sub-surface scattering (jade, wax)
    - Spectral Dispersion (diamonds, etc.)
- Thin film coating (in nanometers) on each layer
- Anisotropy for:
    - Reflective Coating
    - Base Layer
- Emission (true self illumination)
- Geometry Effects:
    - Displacement map
    - Round corners
    - Cutout

The more weighted a layer is, the more it occludes the lower layer.

Like the Arch & Design material, the Iray material includes a number of presets which are great out-of-the-box, or as a starting point to build your own materials:


Metal - aluminum - satin
Car paint - red - small flake
Car paint - sunburst
Cat paint - midnight blue
Diamond
Fabric - silk - blue
Fabric - velvet - red
Glass - solid - clear
Glass - solid - clear - disper
Glass - solid - frosted - clea
Glass - solid - frosted - light
Glass - solid - light green

Reference images of the materials are at: http://mastering-mentalray.com/AUniversity/2013/IrayMtl/Samples/

You do not have to use the Iray material with Iray; Arch & Design and the Autodesk materials work perfectly well, although some options and ‘cheats’ are not used in Iray. If you may be using mental ray at some point, for Lighting Analysis for instance, you may want to stay with A&D for compatibility. That said, we find the Iray material to be fast, flexible, with a deep feature set which generates excellent results.

Our own testing of the Iray material vs the A&D material shows that the Iray material can be up to 60% faster than A&D for materials. In scenes with a lot of frosted glass, using the A&D Frosted Glass preset will render half as fast as the Iray material’s Frosted Glass Clear preset.

In general, non-transparent materials are slightly faster with the Iray material, as are simple transparent materials like Solid Glass. Translucent, volumetric and back-scattering materials are much faster with the Iray material.

The developers at NVIDIA have done a great job documenting their material in a file you can download from here: http://blog.irayrender.com/post/52945163035/iray-material-plugin-for-3ds-max - the Zip files include the material plugin for 3ds Max and a PDF that documents the material.

Thank You!

Thank You for attending the class or watching online!

Many thanks to the kind folks at BOXX Tech for the use of their hardware for testing and presentation. Thanks to Zap Andersson for the wealth of information he provides the world on mental ray, and the work he and the folks at Autodesk and NVIDIA ARC do to make mr and Iray a little better every day. Thanks to Cassie and the team at NVIDIA ARC for the use of mental ray Stand Alone for some of the testing.

Useful Links:

Files for this presentation: http://mastering-mentalray.com/AUniversity/2013/

Zap's mental ray tips: http://mentalraytips.blogspot.com/

Iray Developer's Blog:

NVIDIA ARC Forums: http://forum.nvidia-arc.com

NVIDIA ARC mental ray: http://www.nvidia-arc.com/mentalray.html

NVIDIA ARC Iray: http://www.nvidia-arc.com/iray.html

Elemental Ray: http://elementalray.wordpress.com

Joep's Material Converter: http://3dstudio.nl/producten/maxscripts/autodesk-material-converter

Online mental ray Documentation: http://docs.autodesk.com/MENTALRAY/2014/ENU/mental-ray-help//files/manual/index.html

Online MAXScript Documentation:
http://docs.autodesk.com/3DSMAX/14/ENU/MAXScript%20Help%202012/index.html