

## HDRI Light Control in Bryce 7.1 Pro

A HDRI can light a scene and provide a backdrop as well. What power the pixel values in the HDRI have and how the available controls behave is discussed.

### Introduction

This is a rather theoretical discussion that hopefully helps understanding how the different parameters play together to light a Bryce scene by Image Based Light (IBL). Below the *IBL* tab of the *Sky Lab* is shown with the controls that concern us here identified with red arrows.



It is assumed that an HDRI is loaded, whether it is used as backdrop as shown above does not matter, neither do shadows.

### HDRI

HDRI stands for high dynamic range image as is well known, the rest are common LDRI (low dynamic range image) and a few are DRI (dynamic range increased). Both are not of interest here.

High dynamic range is often confused with contrast, exceedingly bright parts and very dark regions in the same picture. This may be the case, but it is not necessarily so.

A picture consists of coloured pixels: red, green and blue. Each colour can have an individual brightness, e.g. dark red and light blue. Common pictures displayed on a computer screen can have 256 discrete levels of brightness: 0 is black, 255 is white.

An HDRI can have thousands of levels between a bit darker grey and a bit lighter grey; between 127 and 129, two discrete steps of mid grey in an LDRI, can have many thousand steps in an HDRI and this makes the difference. The pixel values are not integers but reals or floating point numbers.

A conventional picture can be converted to an HDRI and even the pixel values multiplied to make it appear “HDRI-ish”, but there are still only 256 different brightness levels, they are only farther apart.

Usually, we say a HDRI is a 96-bit image with 32 bit single floating point values for each of the three primary colours contrasting 24 bit with trice 8 bit for each colour. There are such HDRI files, PFM (portable float map) and TIF (tagged image file format) are two examples.

Bryce, like many other 3D computer graphics programs, uses the RGBE format identified with the file type `.hdr`. This is, in fact, not 96 bit but only 32 bit. The trick here is that only the mantissa of each colour is stored as a byte (8 bit) and the exponent of the brightest colour as the common exponent is the fourth byte; hence the name RGBE (red, green, blue, exponent).

### The HDRI used for Testing

I call this a theoretical discussion because of the HDRI I made and used here: a uniform white spherical panorama 400 pixels wide and 200 high (dynamic range 1:1). Each of the 9 HDRI has other pixel values between 0.050 and 100.000.

For such a uniform white HDRI loaded into Bryce IBL, the lowest useful pixel value is around 0.250 and the highest 10.000 if only the *HDRI Effect* control is used. With the additional controls *Apply to light source* and *Falloff* all (and more) can be used.

Doubling the pixel value in the HDRI file quadruples the light output from the HDRI (*HDRI Effect*) but only doubles the brightness of the backdrop (*Intensity*). Therefore the pixel value to light output works after the square law and pixel value to backdrop image linear.

#	HDRI Pixel Value			HDRI Effect for 255	Intensity for 192	HDRI used
	Set Decimal	Result	Hex mant.   expo.			
1	0.050	0.050	cb   7c	>999.00	76.40	W4 green
2	0.100	0.100	cc   7d	>999.00	38.10	W7
3	0.250	0.249	7f   7f	908.00	15.30	W8
4	0.500	0.497	fe   7f	227.00	7.65	W3
5	1.000	0.996	7f   81	56.75	3.83	W1
6	2.000	1.998	fe   81	14.19	1.91	W2
7	10.000	9.969	9f   84	0.58	0.38	W5
8	50.000	49.875	c7   86	>0.02	<0.08	W9
9	100.000	99.750	c7   87	<0.01	<0.04	W6



The column **Hex** shows the pixel values in the file. How the real pixel value can be calculated from those hex values is discussed in the Appendix.

It must be stressed that such a uniform white HDRI is not the normal case but the only means to test how the controls behave. A “normal” HDRI has perhaps a small very bright sun, a lot of some mid-bright colourful parts and maybe some dark shadows.

## HDRI Light Controls

*Quality* and *Saturation* are also light controls but do not change the light generated, only how many lights are created from the HDRI and used for lighting, and the saturation of the colour.

### HDRI Effect

*HDRI Effect* should be named Diffuse like for the sun or any conventional light source because it works the same way. The range is -999 to +999 with a resolution of 2 decimal places. Positive values (blue control)  cast light onto the scene; negative values (yellow control)  “suck” light out of the scene. The middle value is 0 and no light gets out to the scene.

This control works linear; this means doubling the *HDRI Effect* value also doubles the light output and the objects lit by it get double as bright. It may be that the pixel values in an HDRI are too weak to light a scene even at 999; or too strong and it gets difficult to adjust *HDRI Effect* in 0.01 increments. To cope with such cases, there are additional controls.

### Falloff and Radius

There are three *Falloff* options: *None*, *Linear* and *Squared*. These are confusing labels because conventional light sources also have the *Falloff* options *None*, *Linear* and *Squared*, additionally *Ranged*. The labels make sense for the conventional lights (radial, spot) since the distance of such a light source to an object can be varied. With linear falloff, only half the light from a radial reaches the object when its distance from the light source is doubled.

The HDRI light sources are at infinite distance and it does not matter in what distance an object is from infinity. Like in the real world and ignoring the effects of air and haze: it does not matter whether you are at the sea or on the highest mountain, the brightness of the sun is the same, even though the light falls off by the square law; the difference in distance from the sun can be neglected.

A better name for *Falloff* in the IBL tab would be Attenuation Type. The *Falloff* option must be seen together with *Radius*, which should be labelled Attenuation. *Radius* is the real attenuator and the *Falloff* type selected (*Linear* or *Squared*) determines the behaviour of *Radius*.

HDRI Effect	Radius		Brightness R = G = B
	Linear	Squared	
14.2	None	None	255
28.4	10	7.07	255
56.8	20	10	255
113.6	40	14.14	255
227.2	80	20	255
454.4	160	28.28	255
908.8	320	40	255
908.8	640	80	127

The range of *Radius* is from 10 to 1000 in 0.01 increments. *Linear Falloff* works linear: if the *Radius* is doubled, only half of the light gets into the scene. *Squared Falloff* falls off by the square law: if *Radius* is doubled, only a fourth of the light reaches the scene.

The table above shows how *HDRI Effect* was used to compensate for the increase of *Radius* for both options. Because the lowest *Radius* setting is 10, there is no *Squared* measurement at 7.07, it was calculated.

### Intensity with *Apply to light source* Enabled

*Intensity* controls the backdrop brightness if *Use as backdrop* is enabled; if not, it does nothing. *HDRI Effect* works as Diffuse providing the light to the objects in the scene and *Intensity* controls the brightness of the HDRI as backdrop.

The above is true if *Apply to light source* is not enabled and it is not enabled by default. If it is enabled and another HDRI is loaded, it is disabled again. Click on the arrow right of the *Intensity* value to access the *Apply to light source* option.

If the *Apply to light source* option is enabled, *Intensity* additionally works as multiplier for the HDRI light. It can boost the light (well known) — and also dim the light (less known fact) like *Falloff* and *Radius* do.

The *Intensity* control runs from 0 to 100 with a resolution of 2 decimal places. How can it work as dimmer and amplifier? The neutral setting is 5.00. There is no difference in the light from the HDRI on the scene whether *Apply to light source* is enabled or not.

Below 5, from 0.01 to 4.99 the light from the HDRI is attenuated by the square. For any given *HDRI Effect* value and light on the scene, if *Intensity* is halved, only a fourth of the light reaches the scene. In this range, it works like *Falloff Squared*.

Above 5, from 5.01 to 100 the light from the HDRI is multiplied by the square. For any given *HDRI Effect* value and light on the scene, if *Intensity* is doubled, four times as much light reaches the scene. In this range, it is the opposite of *Falloff Squared*. How the values can be calculated is shown in the Appendix.

### Intensity if HDRI is used as a backdrop

It is quite fine that the light the HDRI casts onto the scene can be controlled by *Intensity* and together with *Falloff* Bryce can accommodate almost every HDRI no matter how the pixel values in it are. If the HDRI is also used as a backdrop, we may be faced with a too dark or too bright backdrop because *Intensity* is set to adjust the light but not the backdrop.

If *Intensity* values below 5 must be used to get the light right, it is better to disable *Apply to light source* and adjust the brightness with *Falloff* and *Radius* instead. *Intensity* can then be used for the backdrop brightness only.

If the light from the HDRI is too weak, the only way to get more light out of it is by enabling *Apply to light source* and increasing *Intensity* — but then the backdrop may become too bright, even completely white.

There are two means to adjust an overly bright backdrop: (1) use *Transparency* in the IBL tab or (2) create a sphere and set it to a neutral density filter in the *Materials Lab*.

**Transparency:** in the main GUI set the sky to *Custom Sky* and set *Sky Color* — the centre colour swatch — to fully black (see first picture below, inset at lower right). In the IBL tab click on the white arrow right of *Use as backdrop* and enable *Blend with sky*. Now move the *Transparency* control from 0 towards 100 until the backdrop has the desired brightness. This mixes the bright backdrop with the black sky and thus it gets darker. In the picture below, the HDRI preview is burned out but the backdrop looks good at *Transparency* 90.

**Sphere:** create a sphere; make it big enough to cover the whole scene including the camera. Best (but not mandatory) set it to maximum *Size*  $X = Y = Z = 102,400$  and *Position* it at the world centre  $X = Y = Z = 0$ . In the *Materials Lab* (see second picture below) click on the button to get the default material. Open the *Material Options* and disable all shadow options (*Cast*, *Receive* and *Self*). Then adjust *Transparency* until the brightness of the HDRI backdrop is as desired. The sphere works as a neutral density filter.



Above: Transparency method; below: Sphere method.



## Specularity

*Specularity* is the quality of gloss. In order to see a glossy spot, there must be one or several very bright lights in the HDRI. Obviously, the white HDRI used here does not have such a light source and hence there is no gloss. However, if a HDRI has a strong visible sun, gloss may appear on the object. If the *Intensity* option *Apply to light source* is enabled, the multiplication is not only applied to *HDRI Effect* but to *Specularity* as well; and *Falloff* reduces *Specularity*.

## Conclusion

Bryce has the means to handle almost any HDRI, no matter how high or low the pixel values in the file are. Remember that changing an HDRI disables *Apply to light source* (among other settings, see <https://horo.ch/docs/mine/pdf/XchgHDRI.pdf>).

## Appendix

### Calculating the pixel value stored in the HDRI file (page 2)

The **Hex** column in the table on page 2 shows how the mantissa bytes (only one shown since red = green = blue) and the common exponent byte appear in the file. The floating point value represented by them can be easily calculated:

$$((m + 0.5) / 256) \times 2^{(e - 128)}$$

where *m* is the mantissa byte in decimal and *e* the exponent byte.

Example row 5: mantissa 7f = 127, exponent 81 = 129.

$$((127 + 0.5) / 256) \times (2^{129 - 128}) = 0.4980 \times 2^1 = 0.9961.$$

### Calculating values from Intensity settings (page 4)

We define:

pv0	original pixel value	[1:255]
pv1	resulting pixel value	[1:255]
i	<i>Intensity</i> value	[0.01:1000]
he0	original <i>HDRI Effect</i> value	[0.01:999]
he1	resulting <i>HDRI Effect</i> value	[0.01:999]

What is the new pixel value on an object at a given *Intensity* value?

$$pv1 = (i / 5)^2 \times pv0$$

How must *HDRI Effect* change to compensate *Intensity* to keep the pixel value?

$$he1 = (5 / i)^2 \times he0$$

If we define additionally

i0	initial <i>Intensity</i> value
i1	new <i>Intensity</i> value

we can calculate the relative change.

$$pv1 = (i1 / i0)^2 \times pv0$$

and for  $pv1 = pv0$ :

$$he1 = (i0 / i1)^2 \times he0$$