PBR Conversion & Texture Changer
Turn Legacy Textures into Unity5 Ready PBR

PBR Tools

VISIT WWW.INFINITYPBR.COM FOR THE LATEST UPDATES, FORUMS & MORE ASSETS.

1. INTRODUCTION
2. QUICK SET UP
3. PROCEDURAL VALUES
4. SCRIPTING
5. CHANGE LOG

* If you have any problems getting set up or using our models, please use the Unity forums to contact us and we'll get it sorted.
** If you enjoy using our models, **please write a review** on the Unity Asset Store so other developers know how cool our stuff is :D*

1. Introduction

"PBR Tools" is a collection of tools designed to make working with PBR easier. Some are runtime tools meant to be used during gameplay, while others are tools to help you convert previous textures into Unity5 ready PBR textures.

We plan on updating our assets periodically, so please check the Asset Store for available updates.

2. Quick Set Up

Combine Albedo & Opacity

This tool will output a combined AlbedoOpacity map with the opacity values in the alpha channel. Unity 5 expects this to be the case for diffuse textures, so if you have an opacity map and a diffuse, use this to combine the two for use with the Standard Shader.

1. Select the SFB_Tools_CombineAlbedoOpacity_v# material.
2. Place your Albedo (diffuse) and Opacity maps in the correct fields.
3. In the Window menu, select "Save Texture".
4. When the process is complete, you can rename/move the finished map, and delete the created material.

Combine Metallic & Roughness

This tool will output a combined MetallicRoughness map with the roughness values in the alpha channel. Unity 5 expects this to be the case for metallic textures, so if you have a roughness map and a metallic, use this to combine the two for use with the Standard Shader.

1. Select the SFB_Tools_CombineMetalRough_v# material.
2. Place your Metallic and Roughness maps in the correct fields. * *TIP! Usually an inverted gloss map is the same as a roughness map!*
3. In the Window menu, select "Save Texture".
4. When the process is complete, you can rename/move the finished map, and delete the created material.

PBR Converter

This tool makes it relatively easy to convert legacy textures to Unity5 ready PBR. It basically will create an Ambient Occlusion & Height map from the Normal *(TIP: Most of the time you should remove the height map!)* that you provide, and create a Metallic Roughness map based on the Color ID and settings you select. An image editing program is required if you need to make a Color ID map.

1. Bring your model into the scene view.
2. Replace the material on the model with the "SFB_PBRConverter" material (circle icon).
3. Load your diffuse texture into photoshop or some other image editing application.
4. Create a Color ID map using these colors: [0,0,0] [255,0,0] [0,255,0] [0,0,255] [255,255,0] Use one color for each real-world material type that your texture has. For instance, one may be a metal, another wood, the 3rd leather and the 4th skin. Perhaps the 5th is another metal, but one that is less reflective than the first. Save the map into your project.

5. Click on the "SFB_PBRConverter" material to load it in the inspect. Populate the Diffuse & Normal (optional) map, and if you already have any other maps, insert them here. Generally you won't have any other maps. [TIP: If you have a gloss map, invert it and save the file. Use that as a roughness map, and be sure to click the "Use Roughness Map" button. This may be a more accurate output for your object.]

6. For each of the colors you've chosen to use, check the box whether it is a metal or not, and then use the slider to adjust the roughness value. The more rough, the less reflective the material will be. You should see the results updated in the scene view immediately. In almost all cases, the two extreme ends of the roughness slider should be avoided.

7. When satisfied, select the "Save Texture" option from the Window menu, and choose the "EXPORT_HERE" folder when prompted.

8. Rename your material to something that works for your project, and bring that new material onto the model.

9. IMPORTANT: Currently Unity has a bug where they don't export the Albedo & Metallic maps correctly. Because of this, we've included a special "MetallicRoughness" map. Often the automated process will populate the new material with the wrong metal map. Ensure that the correct map is being used by dragging the exported "MetallicRoughness" map to the Metal input on your new material.

10. IMPORTANT: Often the height map will not look good for converted models. Its usually a good idea to simply remove the height map entirely from your material.

Texture Changer

Now you can change individual textures on your legacy models! Up to 16 different sections of your model can be isolated and adjusted, or replaced entirely with brand new materials. Works with PBR and Non-PBR materials. Base materials are not included, aside from Silver Metal. Check out our Weapons & Armor PBR Pack #1 or other models that include Base Material generators if you're looking for more materials to use.

1. Bring your model into the scene view.

2. Replace the material with a copy of the SFB_Tools_TextureChanger_v# material. [Make a copy, so you don't have to undo your changes next time you want to use this tool!]

3. In the "Main Inputs" tab, bring in the texture maps you already have for your model. Only the Albedo (diffuse) is required. You'll need to include a Color ID map. Refer to the tabs with numbers and color values to know which colors to use. You can use up to 16 colors with this tool.

4. Under the "Main Inputs" tab, check the boxes that you'd like to use, such as "Use Normal?" if you'd like to use your default normal.

5. Next, for each of your materials (IE, each color in your Color ID map), choose if you're going to use a Custom Material under the "Main Customs" tab.

6. For each number (Color ID color) that will be a custom material, bring in the proper texture maps under the "# [###,###,###]" tab for that number. This tool expects PBR materials, but you can use the blank Normal (Normal Map), pure black (Metal Roughness) and pure white (AO) maps we include if you do not have a PBR material.

7. Next, for each numbered tab (without the color values), make any adjustments to the color & roughness values. All of these changes should be visible in the scene view quickly.
8. When you are satisfied, use the Save Textures tool to export your maps and create a new Standard Shader material with them.
9. Replace the materials on your model with the new material, and you're all done!

Normal to Height & AO

Use this tool to produce a Height map and an AO map based on your normal map.

1. Open the tool in the inspector & assign your normal map to the appropriate field.
2. Select the format of your normal map. If you don't know, try using the material on your object, and you may be able to tell quite easily which format is correct.
3. Adjust the sliders to taste. The default is often a very good mix.
4. Use our tool to export the maps. Window->Save Texture.
5. Your maps will be imported, and you can remove the created material from your project.

3. Procedural Values

The included Run Time procedural materials are designed to be used in game. They start with "SFB_RT_". You'll find some basic scripting guidelines here, but we encourage you to use the Forums on the Unity website if you have more questions, as we aren't the best coders.

SFB_RT_Tools_ColorBlender: Use this tool during gameplay to blend a color into your texture. There are options for blend type & color. The opacity of the color can help limit the maximum blend achieved. Use a greyscale mask to limit the effect.

SFB_RT_Tools_ColorModifier: Use this tool during gameplay to modify the Hue, Saturation, Lightness, and Contrast of your texture. Use a greyscale mask to limit the effect

SFB_RT_Tools_ColorBlender

Category	Name	ID | Type Min,Max	Description
	Blend Amount	`blendAmount` `float (0.0,1.0)`	The amount of blending. 0 = off.
Blend	Blend Color	`blendColor` `Color RGBA`	The color value of the blend.
	Blend Type	`blendType` `int (1,4)`	Which type of blending to use? [1 = Copy] [2 = Multiply] [3 = Additive] [4 = Overlay]

SFB_RT_Tools_ColorModifier

Category	Name	ID | Type Min,Max	Description
	Hue	`colorHue` `float (0.0,1.0)`	Hue shift
	Saturation	`colorSaturation` `float (0.0,1.0)`	Saturation shift. 0.5 = no change
Color	Lightness	`colorLightness` `float (0.0,1.0)`	Lightness shift. 0.5 = no change
	Contrast	`colorContrast` `float (-1.0,1.0)`	Contrast shift. 0 = no change

4. Scripting

It's possible to change values during run time. We include a few versions of the material, some of which are optimized for common run-time options. In those cases, you'll likely want to bake maps for the base materials you plan on using (which do not change at run time), and use the optimized versions. This will speed up the changes in game.

Please Note: We are not the best coders. There may be more ways of doing what we're doing, perhaps better ways. Please use the forums on our site and the Unity forums if you'd like to discuss or ask the community about various ways of doing this. **We are also using Unity Script because, simply, it's what we currently understand.** *Check out our demo scripts for more extensive examples.*

```
var substance                     : ProceduralMaterial;

// Set an Int or a Float value
substance.SetProceduralFloat("Grunge2Volume", 0.5);

// Set a Color value
substance.SetProceduralColor("Grunge2Color", Color(1,1,1,1));

// Get a Vector2 value
var currentOffset   : Vector2 = substance.GetProceduralVector("Grunge2Offset");

// Set a Vector2 value
substance.SetProceduralVector("Grunge2Offset", Vector2(currentOffset[0],currentOffset[1]));
```

LOD	Tris
RockMonster	11,686
RockMonster_LOD1	5,842
RockMonster_LOD2	2,920
RockMonster_LOD3	1,460
RockMonster_LOD4	730
RockMonster_LOD5	364

5. CHANGE LOG

v2.0	Added Normal->Height, AO Baker, and Texture Changer
v1.0	Initial Version