Physically Based Rendering and Global Illumination

# Additional resources

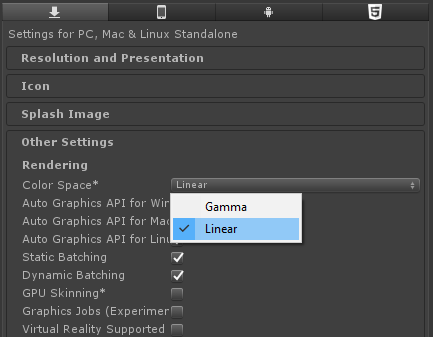
* The Comprehensive PBR Guide  
  <https://www.allegorithmic.com/pbr-guide>
* Understanding PBR  
  <https://unity3d.com/learn/tutorials/modules/intermediate/graphics/substance/01-01-understanding-pbr?playlist=17102>
* Unity Material Charts  
  <https://docs.unity3d.com/Manual/StandardShaderMaterialCharts.html>
* Working with Physically Based Rendering  
  <https://blogs.unity3d.com/2015/02/18/working-with-physically-based-shading-a-practical-approach/>
* Basic Theory of Physically Based Rendering  
  <https://www.marmoset.co/posts/basic-theory-of-physically-based-rendering>

# Setting the Project

If you want to make the most of Unity’s PBR, there are some settings that need to be tweaked.

## Linear Color Space

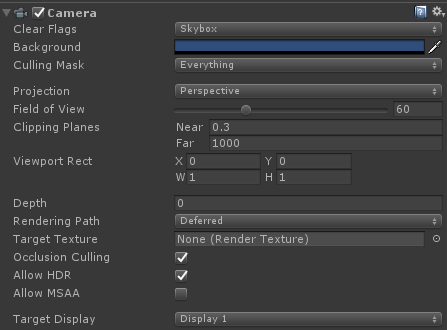
Go to **Edit > Project Settings > Player** and under **Other Settings > Rendering** change the **Color Space** from **Gamma** to **Linear**. This ensures that colours will be represented with high fidelity.



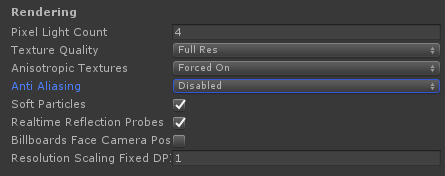
## High Dynamic Range

Certain effects, such as blooms and bright reflections “leak” onto nearby pixels. This is possible using a technique called **HDR**, which allows certain colours to be brighter than their maximum value.

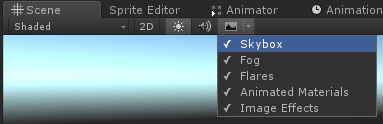
Select the Main Camera in your project, and make sure **Allow HDR** is enabled and **Allow MSAA** is disabled.



If you get a warning, is because HDR is incompatible with the standard Antialiasing technique that Unity uses. **Go to Edit > Project Settings > Quality** and disable Antialiasing.



If your scene looks very dark in the inspector, make sure that your Scene window has **Skybox** enabled.



You are now ready to use PBR materials.

# Understanding PBR

If you want to create realistic materials, they have to respect certain physical constraints.

When light hits the surface of an object, it can be **reflected**, **diffused** or **absorbed**.



The amount of light reflected, diffused and absorbed has to be equal to the amount of light that the object receives. This is known as **conservation of energy**, and is the base of physically-based rendering.

The **diffusion** and **reflectance value** of a material indicates how much light is diffused and reflected when a material is looked directly. Since the amount of light reflected strongly depends on the **smoothness** of a surface, we always give these values assuming the object is perfectly polished.

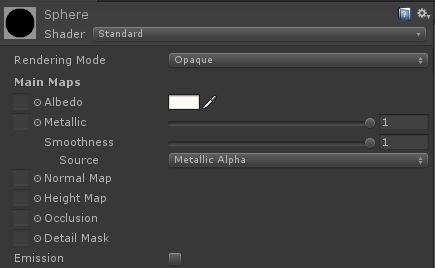
Light is an electromagnetic wave, and it interacts heavily with electrical conductors. This is why Unity differentiates between **Metals** and **Non-Metals** (also referred to as **Insulators** and **Dielectrics**). Unity treats Metals and Non-Metals differently. In particular:

|  |  |  |
| --- | --- | --- |
|  | **Diffuse Value**  **(Albedo)** | **Reflectance Value**  **(Specularity)** |
| **Non-Metals** | From Albedo texture  (RGB Range: 50-243) | Fixed at 4%  (typically 2-5%) |
| **Metals** | Black | From Albedo texture  (RGB Range: 186-255) |

All metals are black (they have a black albedo), as all the light that penetrates their surface is absorbed. The colour that metals exhibit comes from its reflection.

There are many ways (called **workflows**) to author PBR materials. Unity supports two: the **Metallic workflow** and the **Specular workflow**. The Metallic workflow is the most commonly used. While both can be tweaked to produce the same results, the Metallic workflow yield more realistic results if you know the type of material you want to use.

Every Standard (Metallic Workflow) material supports six basic textures. Usually, only two or three are used.



## Albedo

Generally speaking, this is the base colour of an object, without any highlights or shadow.

* **RGB channels**
  + For Non-Metals, this specifies their diffuse colour
  + For Metals, it specifies their reflectivity value.

The **luminosity** of the albedo is very important. If your material is a metal, it should range between 186 and 255. For non-metallic materials, it should range between 50 and 243.



For well-known material that we see every day, you should not try to eyeball your parameters. There are many **calibration charts** ([click here](https://blogs.unity3d.com/wp-content/uploads/2014/11/UnityMetallicChart.png)). you can refer to, to make sure you use physically based values. Guessing will result is non-realistic materials.



You can also download and import the [Shader Calibration Scene](https://assetstore.unity.com/packages/essentials/tutorial-projects/shader-calibration-scene-25422) from the Asset Store.

## Metallic

The Metallic map is a texture that indicates which parts of a model are conductive and which aren’t.

* **R channel**: 1 for metals, 0 for non-metals. Intermediate values are not to be used, except for transitions between metals and non-metals parts.
* **A channel**: the surface smoothness. 1 for smooth surfaces, 0 for rough ones.

Remember that:

* Painted metal is non-conductive, hence it should have a metallic value of 0.
* Scratches over painted metal can have values close to 1, as they show the metal underneath.
* Rust is non-conductive.
* Scratched metal is still conductive and should have a metallic value of 1, but a reduced surface smoothness.