Prototype Materials Documentation

October 2024

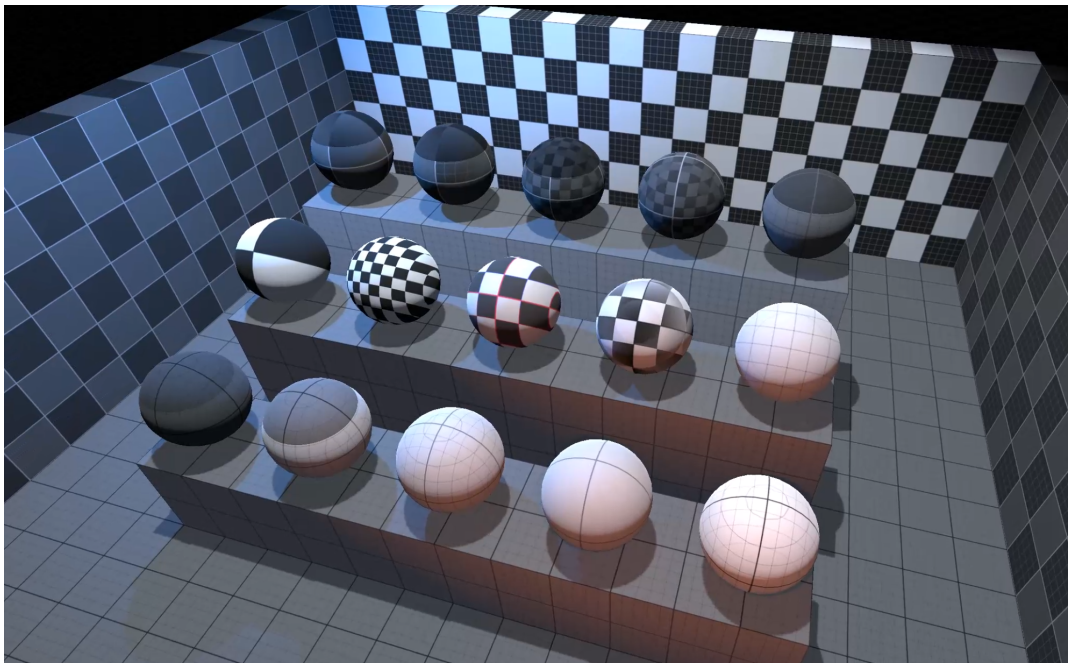
version 1.0.0

Contents

1 Greetings

Thank you for downloading Prototype Materials, a simple procedural shader with several example materials to help you prototype your scenes better. In this short documentation, we'll look into what the settings are and how to set the materials up.

If the asset was useful to you please consider dropping a good review on the Asset Store so others can find it.
If you found something missing or not quite right, do not hesitate to write an email with your feedback.



2 Project setup

2.1 Dependencies

The project uses Unity's Shader Graph for its shaders, even for the Built-in renderer.
The shader graph is added to the package definition as a dependency with the version that was used to create it, so when you download the package, Unity should automatically include the dependency as well. However if you encounter purple shaders that signify an error, please make sure you have the appropriate version of Shader Graph downloaded.
The current version of Shader Graph for the asset is **14.0.11**.

2.2 Render pipelines

The asset supports the Built-in, URP and HDRP pipelines, and for all three the materials were done in Shader Graph.

The demo scene has a simple C# script that sets the light intensities, based on which pipeline is imported into the project and which one is active.
If you encounter an error on your selected render pipeline, try updating Shader Graph to the latest stable version and see if it solves the issue.

2.3 What's in the asset?

The asset has the following structure:

```
Prototype Materials/ (root)
|
 — Materials/
|   <<contains the example materials used in the demo scene>>
|
 — Scenes/
    |
     — Demo
        <<the demo scene, one scene for all pipelines>>
|
 — Scripts/
    |
     — DemoScripts.asmdef
    |   <<needed for the light intensity setup>>
     — LightInitializer.cs
        <<needed for the light intensity setup>>
|
 — Shaders/
    |
     — Grid.shadersubgraph
    |   <<subgraph for grid creation>>
     — TopColorSwitch.shadersubgraph
    |   <<subgraph for color switch for the top of the mats>>
     — Prototype_Checkers.shadergraph
    |   <<shader for the checker materials>>
     — Protytype_Wires.shadergraph
        <<shader for the wire materials>>
|
 — Textures/
    |
     — T_Checker
        <<a basic checker texture for the shaders>>
|
 — Documentation/
    |
     — Prototype_Materials_Documentation.pdf
        <<this file>>
|
```

```
— package.json
  <<the package definition file>>
```

2.4 How to set a material to an object

Just drag and drop, or set the material from the inspector. Given that the shaders are procedural and projected onto the objects, they do not require meshes to be UV mapped to work.

3 Material Properties

This section details what each property on the materials do. They are intuitive, so just setting one property will tell what it's for, but for reference see the documentation below.

3.1 Checker materials

- **Surface options** - these are standard lit shader options and are left as material defaults.
- **IsWorldSpace** - whether the shader should be in world or object space.
  World space means that changing the object's position or rotation has no effect on how the material is projected on it.
  Object space means that rotating or moving the object moves the 'material with it', but scale still has no effect.
- **Projection** - use TriPlanar or TopDown
  Tri planar projection projects from 3 axes, while Top down projects from one. Usually cubic objects need tri planar projection to have all their sides textured, while some spheric objects look best with top down projection.
- **Offset in local** - the shader UV offset in object mode. Useful when the material does not line up with the corners of the object.
- **Checker variant** - whether to use checkers on the main tiles or the sub tiles
- **Smoothness** - the smoothness of the shader
- **Top cell color 1** - one of the checker cell colors on the top of the material
- **Cell color 1** - one of the checker cell colors on the side of the material
- **Top cell color 2** - the other checker cell color on the top of the material
- **Cell color 2** - the other checker cell color on the side of the material
- **Top grid color** - the color of the grid on the top of the material
- **Grid color** - the color of the grid on the side of the material
- **Grid size** - the size of the main grid's cells, in Unity units. Changing the scale of the object will not affect the grid on it.

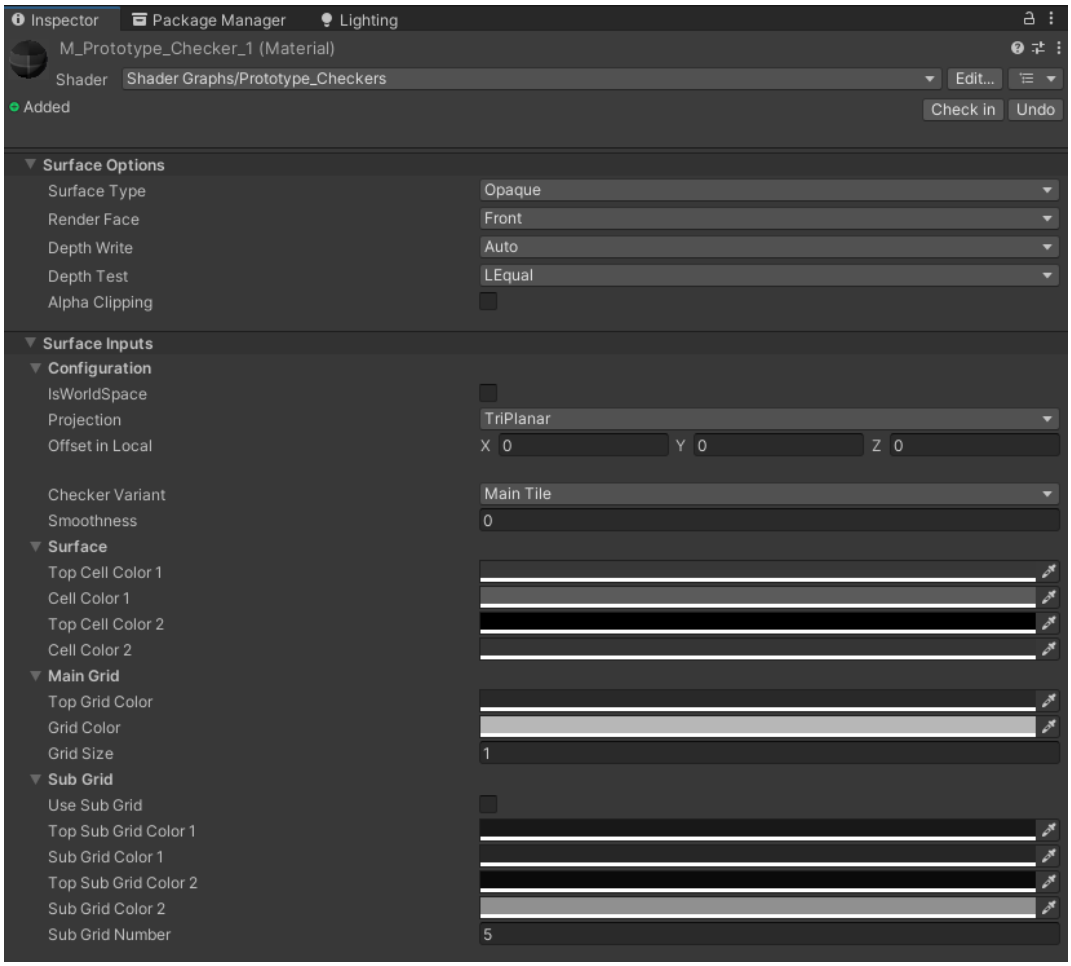


Figure 1: Checker material settings

- **Use sub grid** - whether to use a sub grid alongside a main grid.
- **Top sub grid color 1** - the color of the sub grid on top of the object, for one of the cells.
- **Sub grid color 1** - the color of the sub grid on the side of the object, for one of the cells.
- **Top sub grid color 2** - the color of the sub grid on top of the object, for the other type of cells. In tri planar projection mode using a sub grid layout sometimes the sub grid colors are projected on top of each other, making the second color not show up.
- **Sub grid number** - how many subgrids should be within a main grid

3.2 Wire materials

- **Surface options** - these are standard lit shader options and are left as material defaults.
- **IsWorldSpace** - whether the shader should be in world or object space.
  World space means that changing the object's position or rotation has no effect on how the material

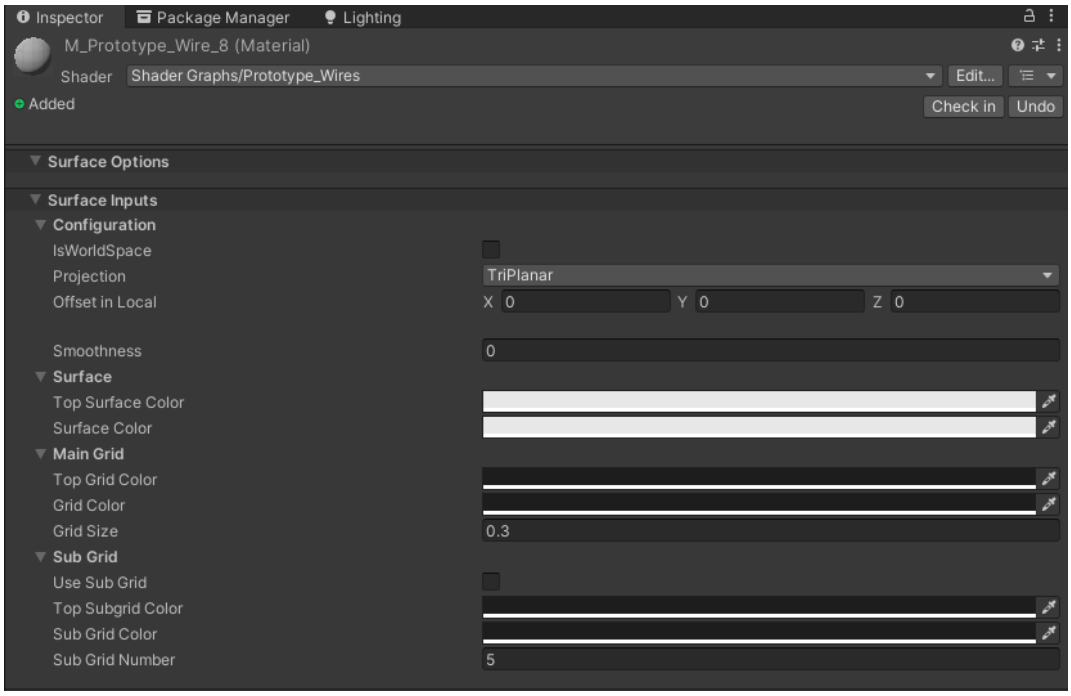


Figure 2: Wire material settings

is projected on it.
Object space means that rotating or moving the object moves the 'material with it', but scale still has no effect.

- **Projection** - use TriPlanar or TopDown
  Tri planar projection projects from 3 axes, while Top down projects from one. Usually cubic objects need tri planar projection to have all their sides textured, while some spheric objects look best with top down projection.
- **Offset in local** - the shader UV offset in object mode. Useful when the material does not line up with the corners of the object.
- **Smoothness** - the smoothness of the shader
- **Top surface color** - the color of the surface of the object on its top
- **Surface color** - the color of the surface on the side of the object
- **Top grid color** - the color of the grid on the top of the object
- **Grid color** - the color of the grid on the side of the object
- **Grid size** - the size of the grid, in Unity units. Independent of the object's scale as the material does not use the object's UVs.
- **Use sub grid** - whether to draw a sub grid on top of the main grid

- **Top subgrid color** - the color of the sub grid on the top of the object
- **Sub grid color** - the color of the sub grid on the side of the object
- **Sub grid number** - how many subgrids should be in a main grid