Material Shader Nodes Graph

Material Shader Nodes Graph adds general nodes from Blender to Unity like textures, converters. Material Shader Nodes Graph has custom node editor. And it generates subgraph.

Getting Started

Compability

You will need Unity version 2020.3 and above.
Example assets made in URP but package also works in HDRP.

Installation

Github: Clone this repository and put `Blender Nodes Graph` folder in `Assets` folder. (P.S.: Still in beta (pre-release))
Asset Store: Open `Window > Pacakge Manager` and search `Material Shader Nodes Graph` in `My Assets` (Coming Soon on Asset Store).
Then import it.

Example Assets

All Blender Nodes

To see All Material Shader Nodes Graph open `Assets > Material Nodes Graph > Example > All Nodes`. You can find all nodes grouped with colors.

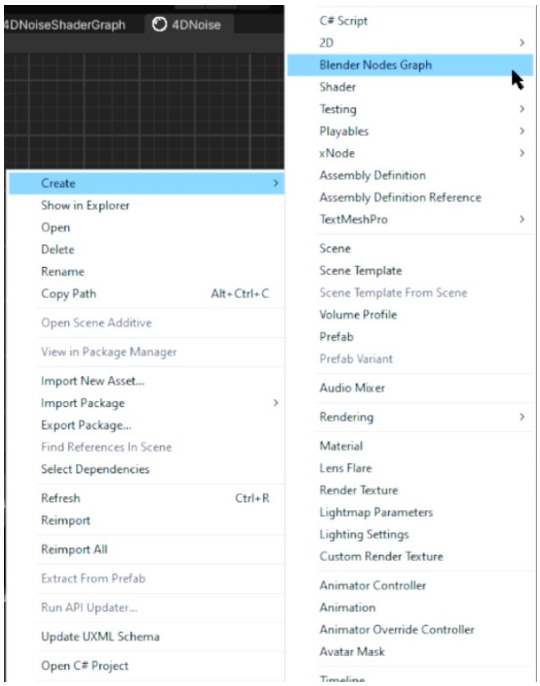
Example Scenes

For example scenes go `Assets > Material Nodes Graph > Examples`. Then you can examine graphs.

Create Material Nodes Graph

To create a graph right click in Project Window. Then `Create > Material Nodes Graph`.

Creating Material Nodes Graph:



It will creates a graph with two nodes: Material Input and Material Output.
Now create a port on Material Output, so you can use it with `Ctrl + Shift + Click` (Auto Connect) function.
Also don't use same name for Material Input and Material Output ports.

Now, you can start create and use nodes.

Using Created Graph

When you click generate button, there should be created two folders named Includes and Subgraph.
Now create a Shader Graph and a material with that Shader Graph.

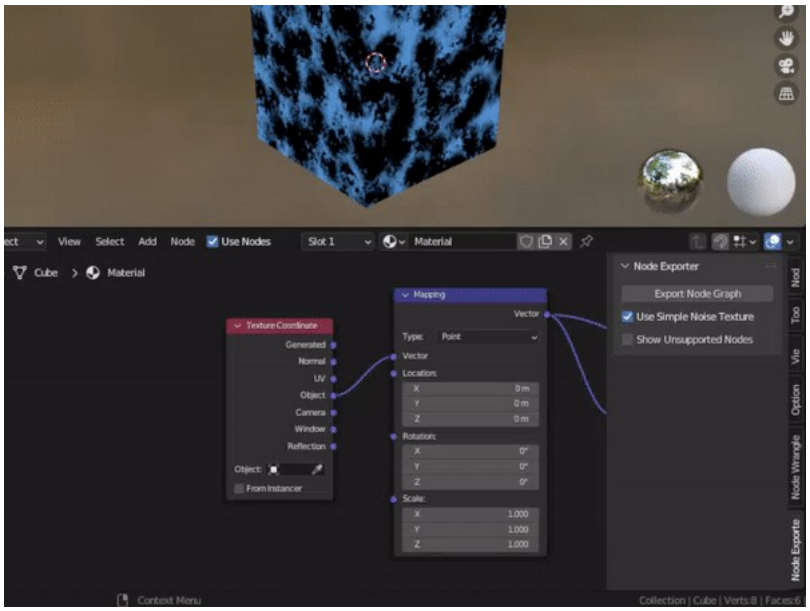
Open Shader Graph. Then, under `Create Node > Material Nodes Graph` find your graph name.
There could be underscore (_) in your graph name if your graph starts with number.

Then, connect your subgraph sockets to desired locations.

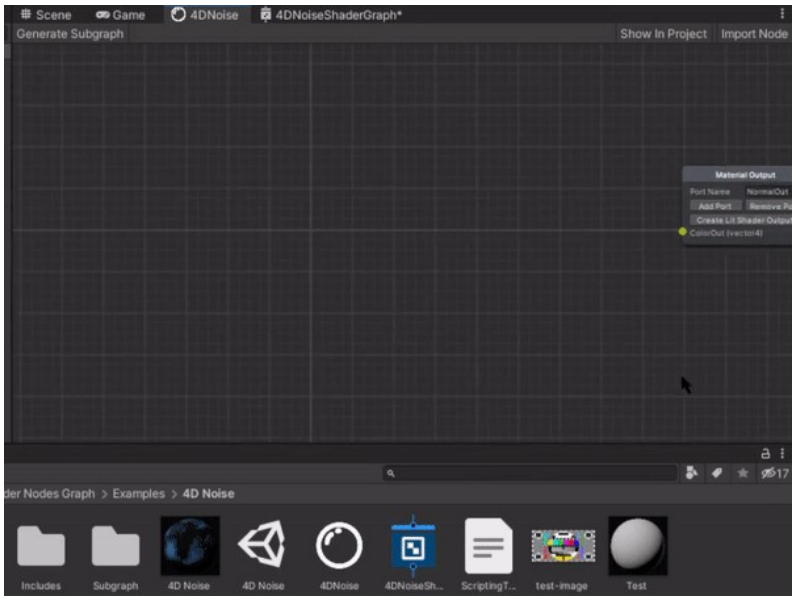
Import Blender Nodes to Unity

First install Blender Addon `Material Nodes Graph > BlenderAddon > blender_unity_exporter.py` in Blender.
Then in your Shader Editor Window click `N` to open node editor toolbar (a.k.a right menu) and click `Node Exporter`.

Click `Export Node Graph` and import generated .json file to Unity.



Finally drag and drop your .json file to Blender Nodes Graph Editor and connect desired sockets to Material Output.

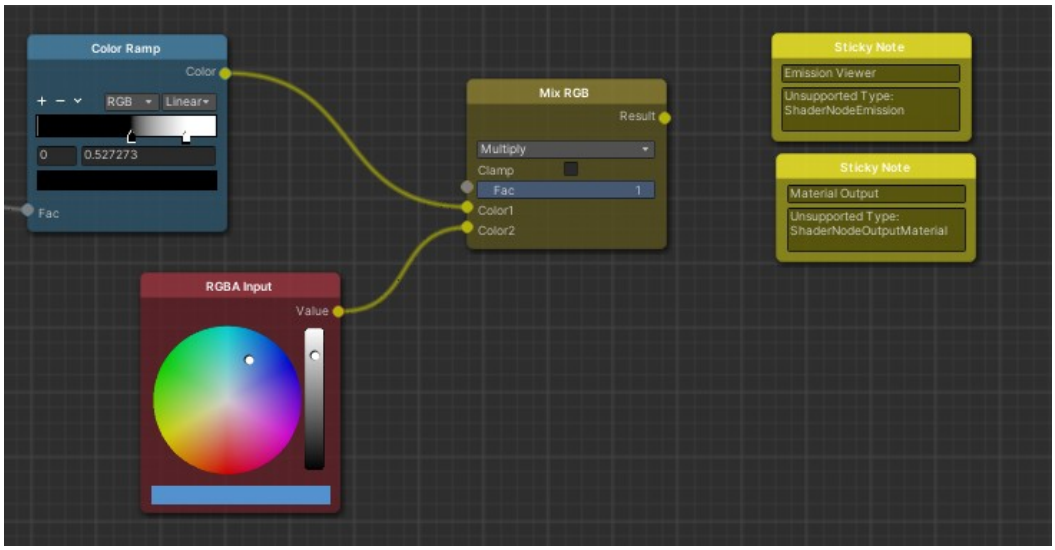


Use Simple Noise Texture

Blender’s default Noise Texture code is heavy. As far as I know Noise Texture uses Perlin Noise. But Simple Noise Texture uses Simplex Noise.

Show Unsupported Nodes

If you enable it, it shows unsupported nodes as sticky note in Unity. It can be useful in debug purposes.



Texture Nodes

All Textures

All textures workflow same as like in Blender. Just connect sockets desired locations.

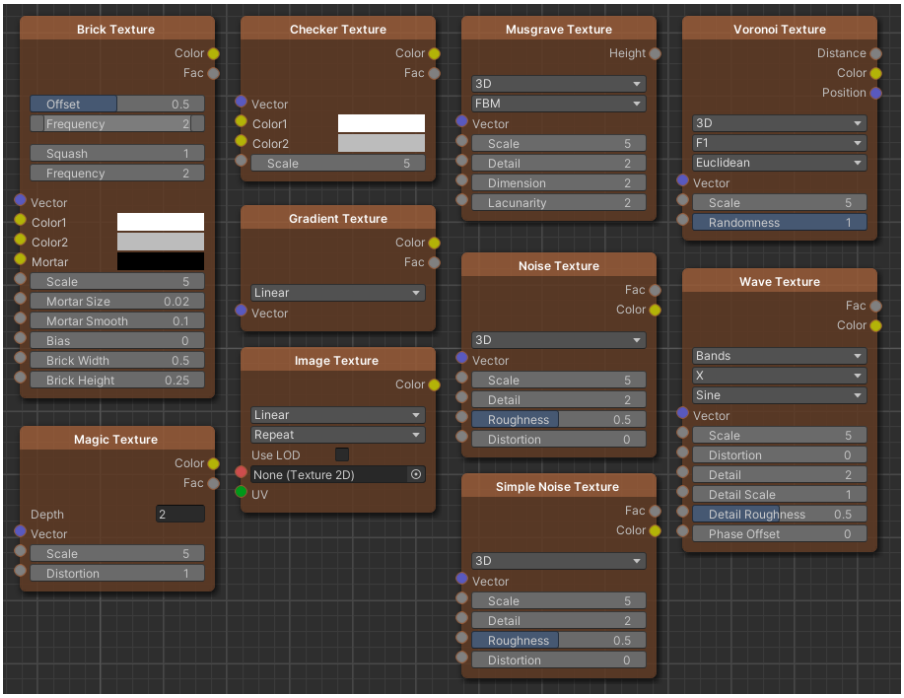


Image Texture

To create Image Texture Node, just drag your image to Material Shader Nodes Graph Editor or create new Image Texture Node with `Shift + A` then drag your image to texture socket.

Limitations

High calculation needed textures:
* Noise Texture
* Voronoi Texture
* Musgrave Texture

Those textures are procedurally generated textures. That's why using them too much and in high settings will result so much calculations in gpu side. So be careful while using those textures.