

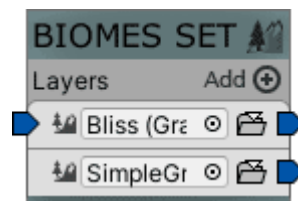
This module is devoted to the sub-graph nodes: the nodes, that that can have the other graph assigned.

Biome nodes allows mixing several graphs together, creating smooth transitions between them. This way you can make a green forest that is gradually changes with the desert at the south and with tundra at the north, and don't need to keep it all in one graph. Instead, you can use three self-sustained and independent graphs: forest, desert and tundra, and one parent graph that determines where each biome should be used.

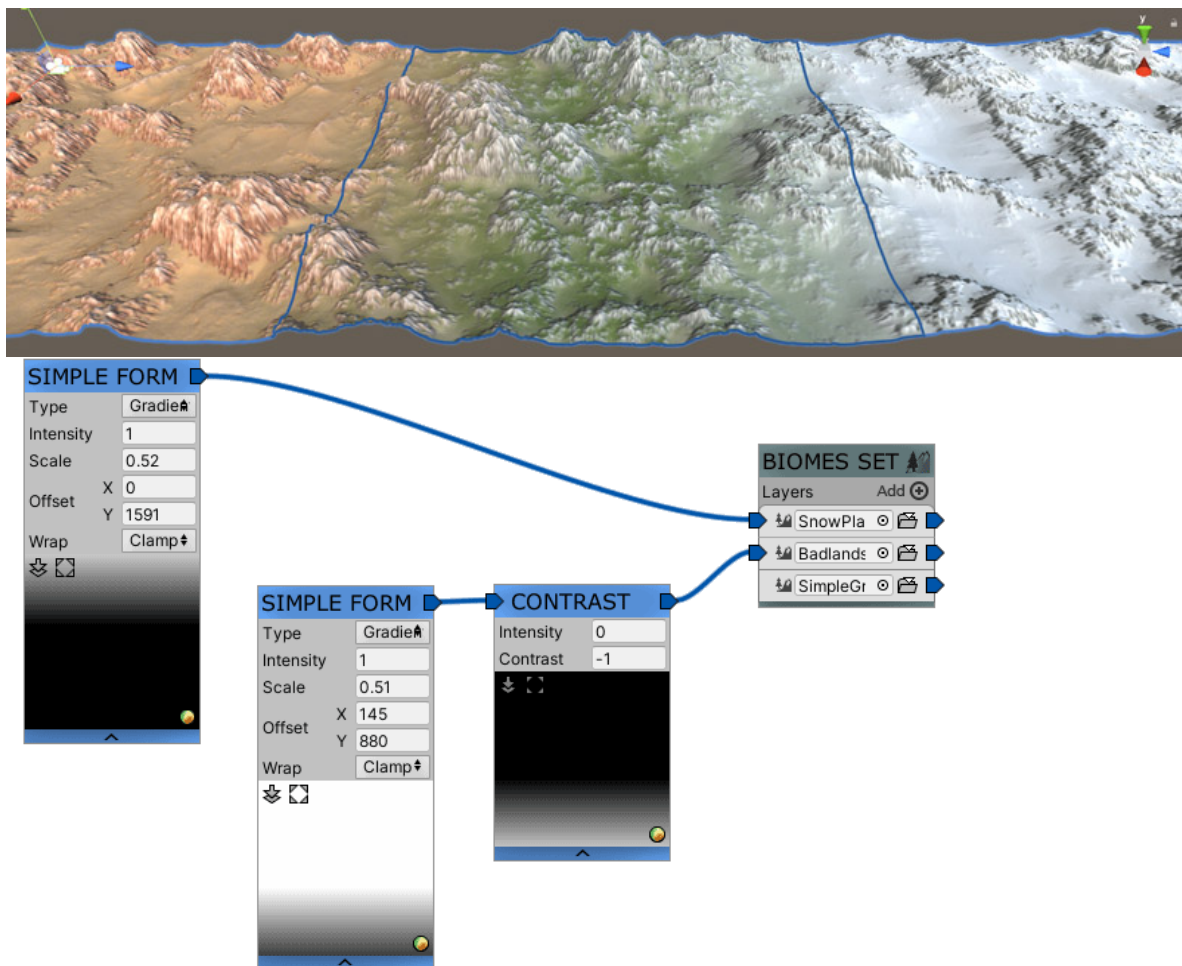
Functions are even more advanced biome nodes: they can have their inlets and outlets. Imagine using a portal right into the biome sub-graph, and the portal out - and that is what the basic function is. This way you can re-use some common elements in all of your graph, bringing them to a separate graph. Tuning and enhancing it will automatically enhance all of the graphs where the function is used.

And more than that, functions can have internal parameters exposed. Consider it as as a parameter passed to the function. You can expose a field of the internal function graph, and set this field value from the parent biome. And the most exciting thing that the exposed field values are independent from the function graph - you can use two or more same functions with exposed different values.

Biomes Set



Blends multiple biome layers the same way texture layers are blended together with the Textures Output. It creates a smooth transitions between the biomes.



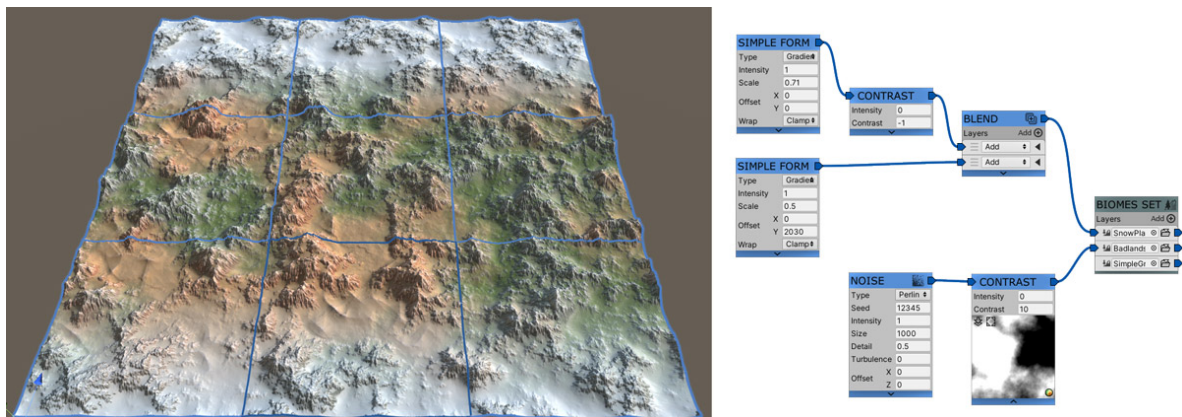
Biomes Set has a [normalizer](#) built-in. It blends layers and normalizes them so that their sum is always 1. This way the layers order matters - each upper layer will overlap the bottom ones. Consider layer masks as Photoshop layers masks.

The Background layer does not require an input since it is regarded as completely filled (a constant of 1).

Note that for the best blending effect biomes should have the relief matching as close as possible - if some biome has a mountain or a plane in some area then all of the other biomes should have the mountain or a plane here. This could be achieved by the main Noise generator the same Seed, Size and Offset values for all the biomes. See the included demo biomes Badlands and Snow Plains as an example.


Use Cases




The typical use scenario assumes assigning large-scale maps as inputs. They could be either multi-tile gradients (Simple Form), or noise with the Size value of 1000 or more, or any other map with a pattern that spans across multiple tiles.



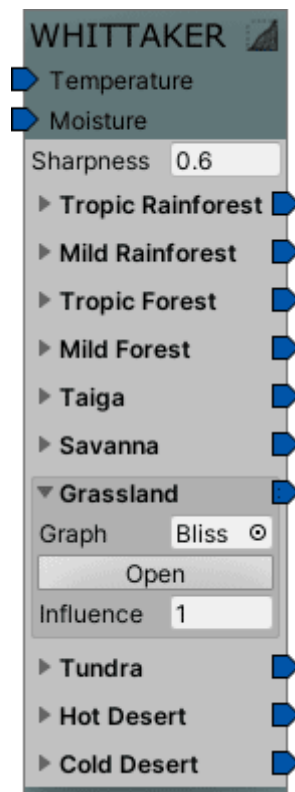
Mixing two biomes (green and yellow) with Noise, and adding snow biome from north and south.

Properties

Biomes use the standard [Layers](#) interface. Each layer contains the graph slot, where the biome graph should be assigned. The layers could be moved by clicking and dragging the biome icon .

The graph could be opened as a biome by clicking the Open button . In contrast to opening the graph by double-clicking it in Project window, accessing it with the Open button will open it as a biome. You can return to the parent biome with the Folder Up  button in the toolbar, and access other biomes with the Folder Tree  (also in a toolbar).

Whittaker



Uses the [Whittaker diagram](#) to automatically place different biomes based on the temperature and precipitation factors.

Inlets

- **Temperature:** the heat map required for the diagram to work. It could be a gradient (Simple Form) towards the south/north, or basic elevation (Import, Noise), or both (mixed with Blend). 0 stands here for 0 ??C of annual temperature, while 1 for 30 ??C of annual temperature.
- **Moisture:** precipitation map. Could be a distance from shoreline, or some random map (Noise). 0 here is no rainfall, 1 is 400 cm of annual precipitation.

Outlets




Each of the biomes has an outlet with the generated mask it's using. Just in case it will be needed for the further modification. This map is generated regardless of sub-graph is assigned or not.

Properties

Sharpness: determines the smoothness of the transition between biomes.

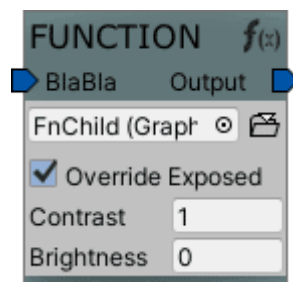
Each of biomes:

- **Graph Slot:** the sub-graph with the biome.

The graph could be opened as a biome by clicking the Open button . In contrast to opening the graph by double-clicking it in Project window, accessing it with the Open button will open it as a biome. You can return to the parent biome with the Folder Up  button in the toolbar, and access other biomes with the Folder Tree  (also in a toolbar).

- **Influence:** extends or reduces the spread of this biome. In case it turns out that some of the biomes area is too big or too small it could be adjusted with the Influence factor.

Function



Functions are biome nodes that have their inputs and outputs. Imagine using a portal right into the biome sub-graph, and the portal out - and that is what the basic function is. This way you can re-use some common elements in all of your graph, bringing them to a separate graph. Tuning and enhancing it will automatically enhance all of the graphs where the function is used.

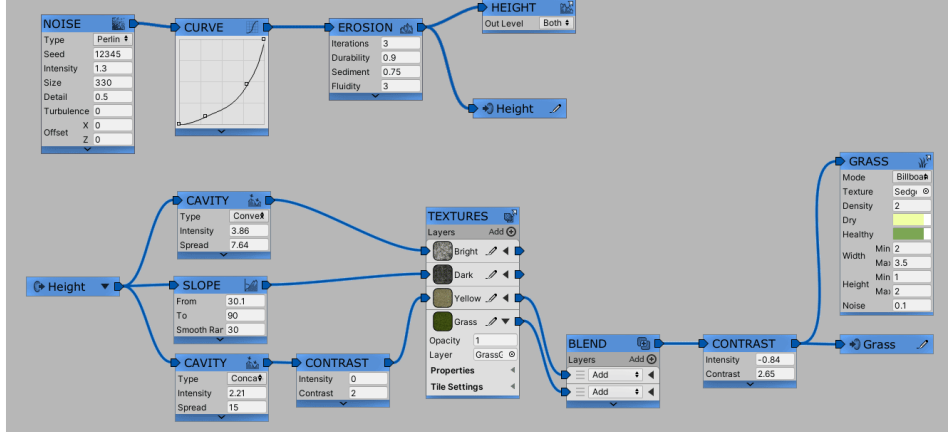
Use case

As an example can take the demo graph SimpleGraph from the Tutorial scene (make a copy to apply changes).

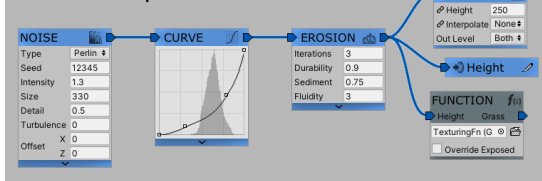
We can bring all of the texturing group into a function. To do this we create one more graph with all the nodes from the texture group imported. Instead of portals we use Function Input and Function Outputs.

In a parent (SimpleGraph Copy) graph we remove them, and replace with a function.

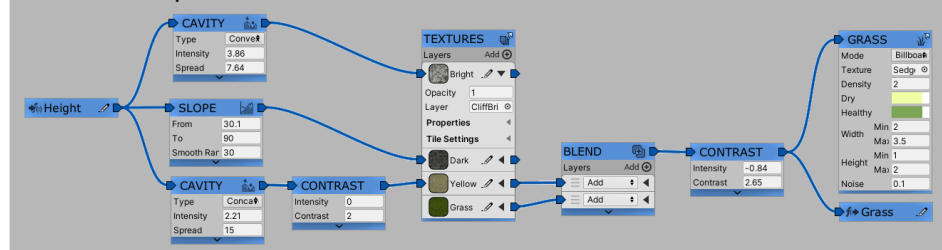
Demo Graph



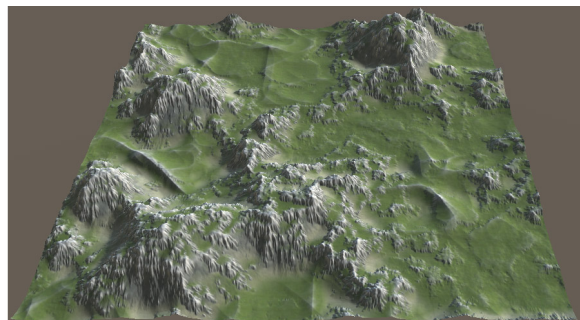
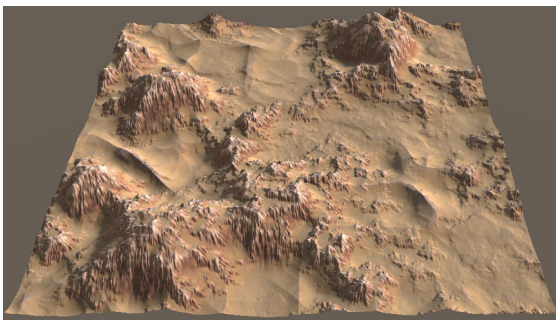
Parent Graph



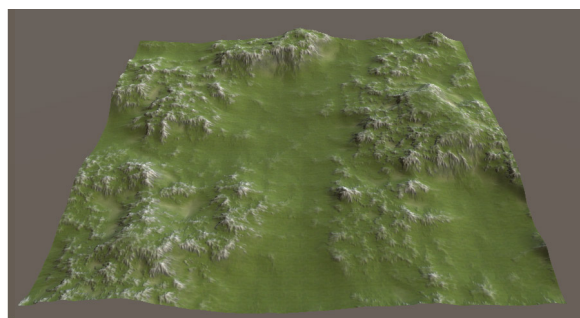
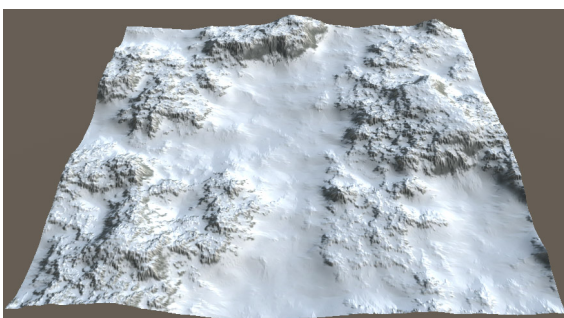
Function Graph



This won't affect the visual representation of the terrain, since, except some technical details, the generate process is quite the same. However, we can insert this function in other graphs in the same way, and it will automatically textures any terrain in the same way it did in Tutorial scene. Don't forget to make copies and remove texturing nodes.




Tutorial texturing applied to Badlands graph.



And the same texturing applied to the Snow Plains.

Value Exposing

Right-clicking on a value field in any graph and selecting Value -> Expose  will make this value to be exposed by modifying externally. If you turn on Value Override in your function node you can change it without opening child graph.




Inlets and Outlets

Function Inlet and Outlet connectors will appear after the functions graph is assigned in slot. For each of the Function Input generators in function graph one Inlet will be created. As well as for each Function Output in function graph one Outlet will be added.

Inlet or Outlet named the same way Function Input/Output is named in function graph.

Properties

- **Graph Slot:** the sub-graph with the function.

The graph could be opened as a biome by clicking the Open button . In contrast to opening the graph by double-clicking it in Project window, accessing it with the Open button will open it as a biome. You can return to the parent biome with the Folder Up  button in the toolbar, and access other biomes with the Folder Tree  (also in a toolbar).

- **Override Exposed:** enables the modifying of the function graph fields that are marked as Exposed. These fields will be listed below. Modifying any of the values in function will not change the value in graph, nor it will affect other functions - this change is active only in this function scope.

Note that overriding values creates some overhead and slows down a graph generate a bit, so try not enable this feature if it's unnecessary.

- List of overridden values (if Override Exposed enabled and function graph has values exposed).