Map Editor for the Honey Hex Framework

Overview

The Map Editor for the Honey Hex Framework is a tool that adds useful map editing features to the Honey Hex Framework. With the Map Editor you can load, save, manipulate individual tiles, their rotation and blend weights, add and remove rivers, and create custom map generators.

The Map Editor is not a game framework though. The Map Editor is only for editing and saving the terrains you create for use in your game.

Installation

1. Import Honey Hex Framework 1.5
2. Add a Assets\HoneyFramework\World prefab to the Hierarchy
    a. Disable/remove the Camera Controler script on the World->WorldCamera game object
3. Import Map Editor for the Honey Hex Framework

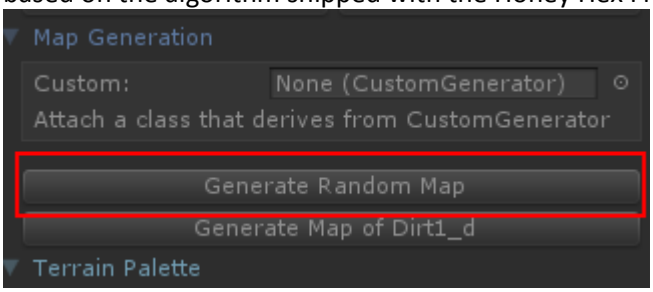
Important Notes

- Map generation and editing is done in the Run Mode only due to how the Honey Hex Framework works.
- Map dimensions are set in the Honey Hex Framework World prefab's Hex Radius and Chunk Radius settings.
- The terrain tiles are managed through the mechanism provided by the Honey Hex Framework.
- Due to the way this version Honey Hex Framework creates the terrain meshes, live update of tiles is slow. Therefor the editor has you commit changes before redrawing the map.
- Simply disable or remove the MapEditor game object once you are ready to ship your game or are done editing your maps.
- Select the MapEditor prefab to show the tools available for map editing in its Inspector window. You can use the lock functionality to "pin" the map editor inspector in Unity.

Map Generation

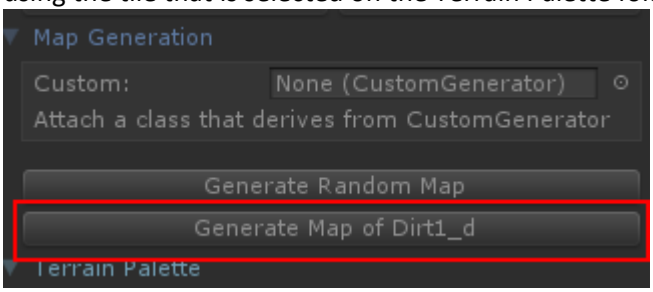
Random Map

Select the Generate Random Map button on the Map Generation foldout to generate a random map based on the algorithm shipped with the Honey Hex Framework.



Palette Map

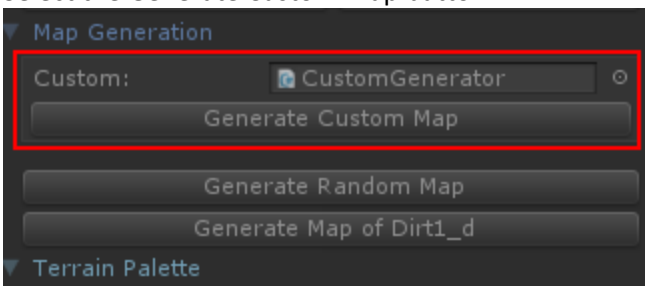
Select the Generate Map of *<selected tile>* button on the Map Generation foldout to generate a map using the tile that is selected on the Terrain Palette foldout.



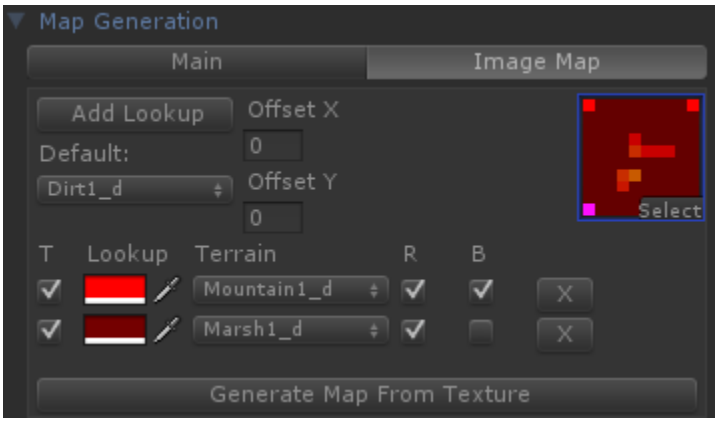
Custom Generated Map

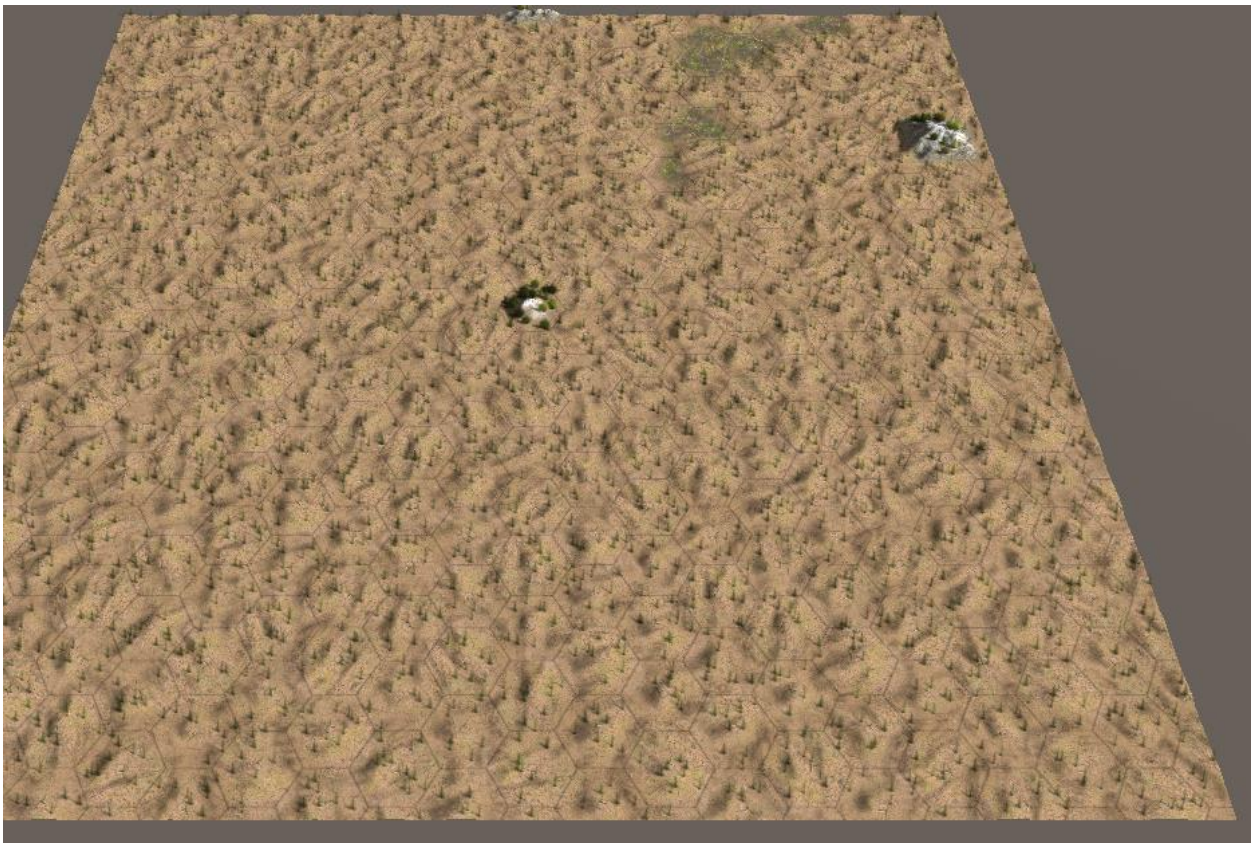
To generate a map using your own custom scripts:

1. Add a prefab containing the custom script that derives from CustomGenerator class which is located in the MapEditor\Scripts folder (see ExampleCustomGenerator.cs for an example) to your scene.
2. Drag that game object onto the Custom field in the Map Editor's Map Generation foldout.
3. Select the Generate Custom Map button.



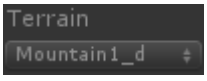
Generate Map from Image





To generate a map using an image:

1. Select an image.
2. Add lookup entries. 

3. Set the color value to associate the terrain entry with. 
4. Select the terrain the red portion of the color value corresponds to. 

5. Select the 'Generate Map From Texture' button.

Optionally:

- Default: the generator will use this terrain for hexes outside the images size.

- T: Toggles between the color picker and a numeric entry field.
- R: Allows you to use the green value of the lookup color to set the rotation instead of using random. Due to the limited value of a color component (0-255), the value will be multiplied by 2. For example, if you want 90 degrees you would enter 45 in the green color component.
- B: Allows you to use the blue value of the lookup color to set the blend instead of using random. The values range from 0 (no blend) to 255 (fully blended).
- Offset X: Offsets the image coordinates onto the map coordinates in the forward/backward direction.
- Offset Y: Offsets the image coordinates onto the map coordinates in the left/right direction.

Editing a Map

Manipulating Terrain

To change terrain tiles:

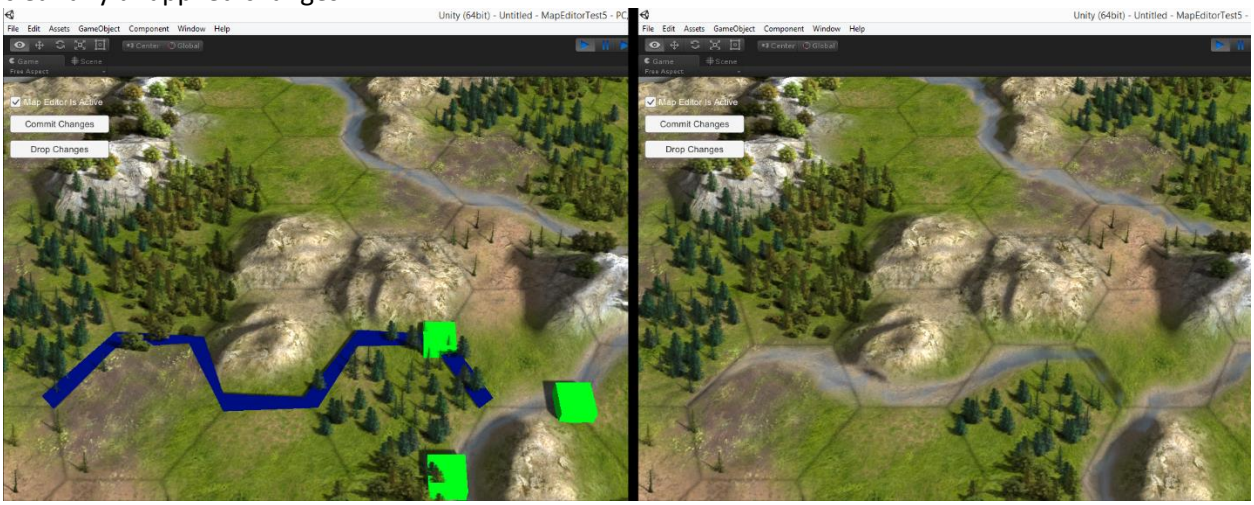
1. Run the scene in the Unity editor.
2. Select a tile on the Terrain Palette foldout; set its rotation and blend settings, if desired (random by default).
3. Left click the area you wish to paint the selected tile onto. This will add a hex tile above the current terrain with the selected tile's texture to indicate what that tile will be rendered as when changes are committed.
4. Once you have painted on the terrain tile changes, select the Commit Changes button on the game UI to have the engine redraw the map. Alternatively, you may select the Drop Changes button to clear any unapplied changes.

Adding and Removing Rivers

To add a river:

1. Run the scene in the Unity editor.
2. Right click a hex; this will bring up three blue cubes directly connected to the hex intersection where the river was created (the leftmost intersection of the selected hex by convention).
3. Right click any green cube to create a node on the river.

4. Once you have drawn the river nodes, select the Commit Changes button on the game UI to have the engine redraw the map. Alternatively, you may select the Drop Changes button to clear any unapplied changes.



To delete a river select the Delete button on the River Management foldout for the river you wish to remove or select the Delete All Rivers to remove all rivers from the map.

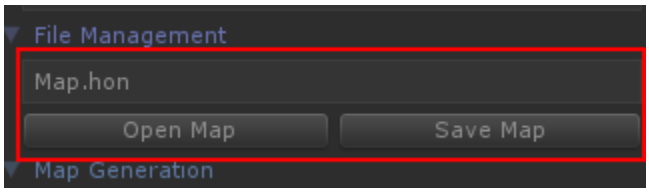
File Management

To save a map:

1. Select the Save Map button on the File Management foldout to launch the Save Map dialog.
2. Navigate to the location you wish to save the map data to.
3. Give the map a name.
4. Click the Save button on the Save Map dialog. The file will be saved with a 'hon' file extension.

To load a saved map:

1. Select the Load Map button on the File Management foldout to launch the Load Map dialog.
2. Navigate to the location where your map data is saved to.
3. Select the map.
4. Click the Save button on the Load Map dialog to load and render the map. The map name will be displayed above the Load and Save buttons.



Miscellaneous

Uncheck the Map Editor Is Active checkbox in the game window to disable drawing when the mouse buttons are used.