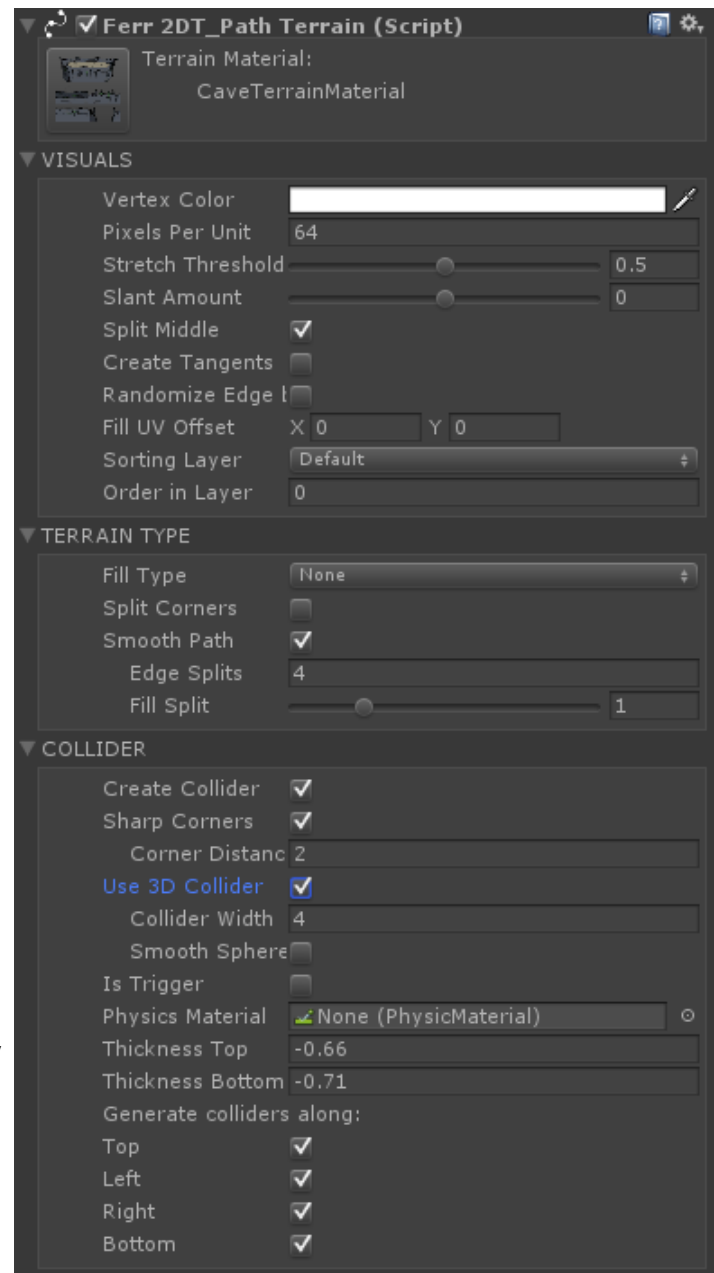


Reference

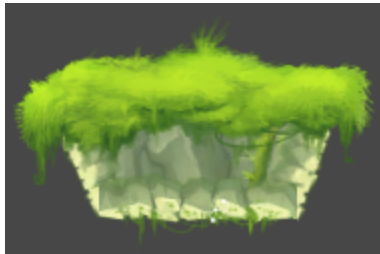
Ferr2DT_PathTerrain

This is the primary component of concern for the tool, and will probably be what you spend the most time with, so it's a good one to understand well. This component relies on data from the Ferr2D_Path component, so if you're looking for the path itself, you may wish to check there.

- **Terrain material**
 - The terrain material can always be changed at a moment's notice! This brings up the terrain material selection window that you see when creating a new terrain.
- **VISUALS**
 - **Vertex Color**
 - The vertices of the mesh are set to this color, this allows you to tweak and tint your terrain for lighting and variety while still letting Unity batch your materials for performance! Make sure you use a shader that supports vertex colors to take advantage of this.
 - **Pixels Per Unit**
 - The easiest measure for ensuring consistent pixel density. How many pixels should the tool attempt to fit into one unit of Unity space? This isn't always exact, but it's a target that the tool strives towards.
 - **Stretch Threshold**

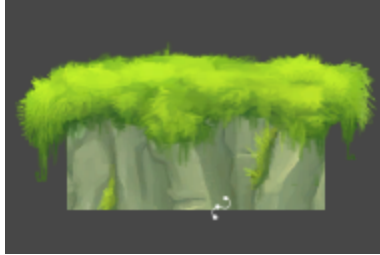


- This changes where Ferr2D chooses to add new segment pieces, stretch or squish essentially. You probably don't need this value, but in the event you do, it's there!
- **Slant Amount**
 - This affects the z-values of the terrain edge, adding a slant to it! It can be handy for adding a subtle parallax effect to the edge.
- **Split Middle**
 - This adds an extra row of verts through the center of the terrain edge. It reduces texture warping slightly, especially around curves!
- **Create Tangents**
 - If you're using a material with normal maps, you'll need Ferr2D to generate tangents! Generating tangents is a little expensive in the editor though, so it's not enabled by default.
- **Randomize Edge by World Coordinates**
 - If you're procedurally updating a mesh, this can be pretty useful. It uses the world coordinate as the random seed, so edges in the same physical location will always generate the same edge. Without it, you may experience flickering edge types on procedural update in some cases.
- **Fill UV Offset**
 - The fill UVs are based on world coordinates, but if you need a little more control over the position of the fill texture, this is it!
- **Sorting Layer and Order in Layer**
 - These are the same as the Unity Sprite's attributes, and helps you to integrate with those sprites better!
- **TERRAIN TYPE**
 - **Fill Type**
 - **Closed**
 - The interior of the path is filled, and edges go the entire way around the path, including the closing segment.



■ Skirt

- The interior of the path is filled, and extra 'skirt' points are dropped down in order to close the terrain shape.



■ None

- The interior will not be filled, only edge materials will be applied. The closing segment will only be present if the accompanying Ferr2D_Path component is marked as 'closed'.



■ Inverted Closed

- The exterior of the path is filled, and edge directions are flipped to match an inward facing direction. This is excellent for creating interiors.



- **Fill Z Offset** (Fill type != None)
 - In order to avoid Z-fighting issues, it is often best to offset the fill mesh by a small amount. You can tweak this amount here, but the default should normally be acceptable.
- **Skirt Y Value** (Fill type = Skirt)
 - When fill type is set to Skirt, this is the Y value that the skirt will drop to. This value is in local space, not global.
- **Split corners**
 - When true, this will break any corners into new segments. This is primarily used to apply different materials to segments facing separate

directions. If you are using a terrain material with only one edge defined, this should be turned off.

- **Smooth path**
 - This adds a curve to the terrain that approximately follows the path you set. It can add a lot of extra verts to the mesh and collision mesh, so it can degrade performance if there are too many!
- **Edge Splits**
 - The number of times the edge body segment gets split for the curve!
- **Fill**
 - The number of times the collider and fill mesh get split relative to the 'Edge Splits' attribute.
- **COLLIDER**
 - **Create Collider**
 - When checked, the tool will create a MeshCollider on Start. It will also display a green line in the Scene window, indicating the shape of the collider. For a video on working with colliders, check here: <http://youtu.be/vLpzLyONFGY>
 - **Sharp Corners**
 - Ferr2D will attempt to make the corner of the collider a sharp corner rather than a beveled corner!
 - **Corner Distance**
 - How far out will Ferr2D extend the collider edge in order to get a sharp corner? This is to avoid infinity corner spikes, or bits that just jut out too far =D
 - **Use 3D Collider**
 - By default, Ferr2D creates a 2D collider, but it can create 3D ones as well if you use 3D physics for your game!
 - **Collider Width**
 - This tells the extent of the collision mesh along the Z axis in Unity units, if you're using a 3D collider.
 - **Smooth Sphere Collisions**
 - Rigidbody components have a Smooth Sphere Collisions attribute, this sets it when the colliders are generated!
 - **Is Trigger**
 - Collider components have an Is Trigger attribute, this sets it when the colliders are generated!
 - **Physics Material**
 - Rigidbody components have a Physics Material attribute, this sets it when the colliders are generated!
 - **Thickness Top** (Fill type = None)

- An offset from the path for the top side of the collider.
- **Thickness Bottom** (Fill type = None)
 - An offset from the path for the bottom side of the collider.
- **Offset Top** (Fill type != None)
 - An offset from top edge segments for the collider.
- **Offset Left** (Fill type != None)
 - An offset from left edge segments for the collider.
- **Offset Right** (Fill type != None)
 - An offset from right edge segments for the collider.
- **Offset Bottom** (Fill type != None)
 - An offset from bottom edge segments for the collider.
- **Generate Colliders Along** (top, left, right, bottom)
 - In some cases, you only want the tops of the terrain objects to be solid, or something similar. This allows you to disable certain edges from the colliders.

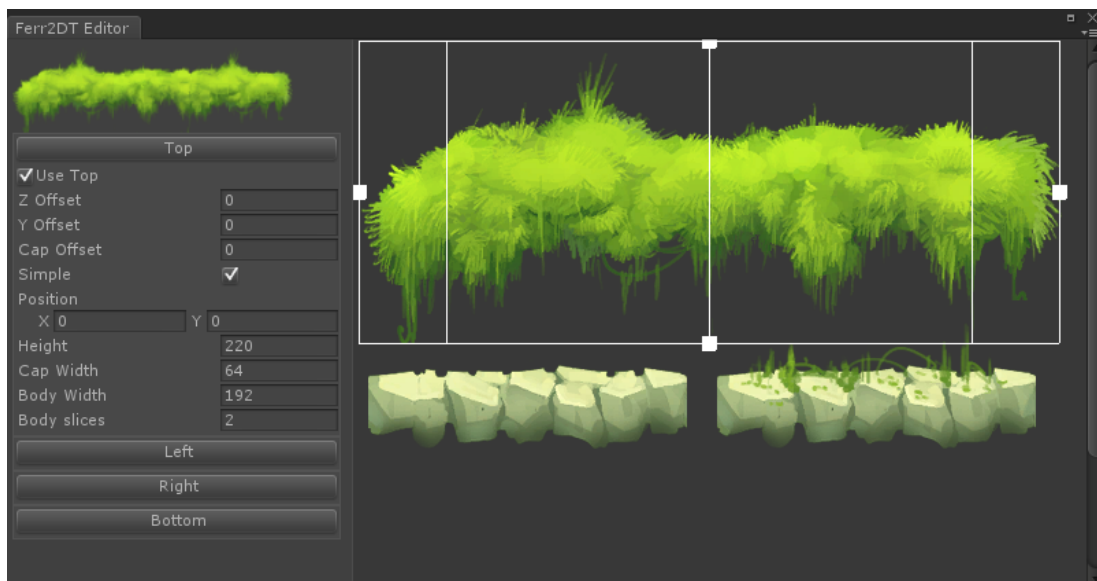
Ferr2D Terrain Material Editor

The Terrain Material describes how materials are applied to the Path Terrain. You can define a single edge using the “top” edge in order to achieve things like pipes and branches, or you can define all 4 to create a regular terrain material. Check out some of the example materials to see what sort of things you can create with it!

For video tutorials of this, check out these:

http://youtu.be/_V9v2RXkUog

<http://youtu.be/TYQSuhsyCDE>



- **Edge Material**
 - This is the material that will be used by the tool for all edge pieces of the terrain. The included shader, “Ferr/Unlit Textured Transparent Vertex Color” was designed for this role, but it is not essential for the terrain to work correctly. For most edge materials, a shader that supports transparency and vertex color should take advantage of all the tool’s features.
- **Fill Material**
 - This is the material that will be used by the fill mesh. The included shader, “Ferr/Unlit Textured Vertex Color” was designed for this role, but it is not essential for the terrain to work correctly. We recommend against using transparent materials for this, as transparent materials can be quite expensive, especially on mobile.
- **(Top, Left, Right, Bottom)**
 - **Z Offset**
 - Z fighting will occur often when creating terrain materials edges, so in order to avoid it, it is best to offset the edge mesh by a small amount.
 - **Y Offset**
 - This will offset the edge mesh from the path by a certain amount. This can be extremely helpful for lining up edge materials better.
 - **Cap Offset**
 - This will allow you to adjust where the cap meshes are at the end of their segments. This can also help with lining up edge materials.
 - **Body**
 - The “body” of this edge is the segment of the Edge Material that will be repeated along the bulk of the path.
 - X,Y,W, and H are measured in pixels from the top left corner of the texture. These are stored as UV coordinates internally, so if you resize the texture, they will resize with the texture.
 - By increasing the “Total” variable, you may have multiple body segments for the tool to randomly choose from when generating the mesh.
 - Use the < and > arrows to pick between the different body segments available to the tool.
 - **Left Cap, Right Cap**
 - This is a small material segment that is placed at the end of each edge to cap it off. These are optional.

