Protovoxel Documentation 1.0

About:

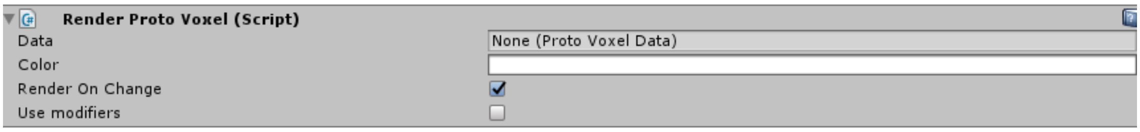
ProtoVoxel let you create quickly voxel models for prototyping, using only a 2D canvas with multiple layers.

Getting Started:

There are two scripts you usually will be dealing with ProtoVoxel:

- RenderProtoVoxel: Add to a empty GameObject to render a voxel inside your game.
- ProtoVoxelAnimate: Must be used in a GameObject with the RenderProtoVoxel script, it animate your voxel utilizing various variables.

Once you add a RenderProtoVoxel script to your GameObject, it will show like this:



Render Proto Voxel (Script)

Data: None (Proto Voxel Data)

Color

Render On Change ☑

Use modifiers □

1. Data: This will be information of a voxel to be rendered, we will see this in the next step.

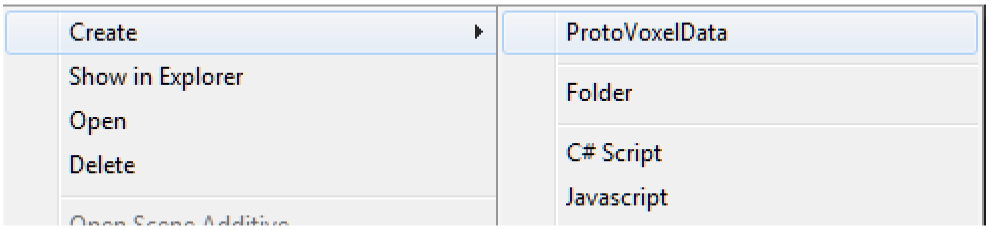
2. Color: A color to be multiplied to the voxel information, this way you can have multiple objects using the same Voxel Data with different colors.

3. Render On Change: If you apply any change to the voxel data and this option is on, you will see the changes happening in real time.

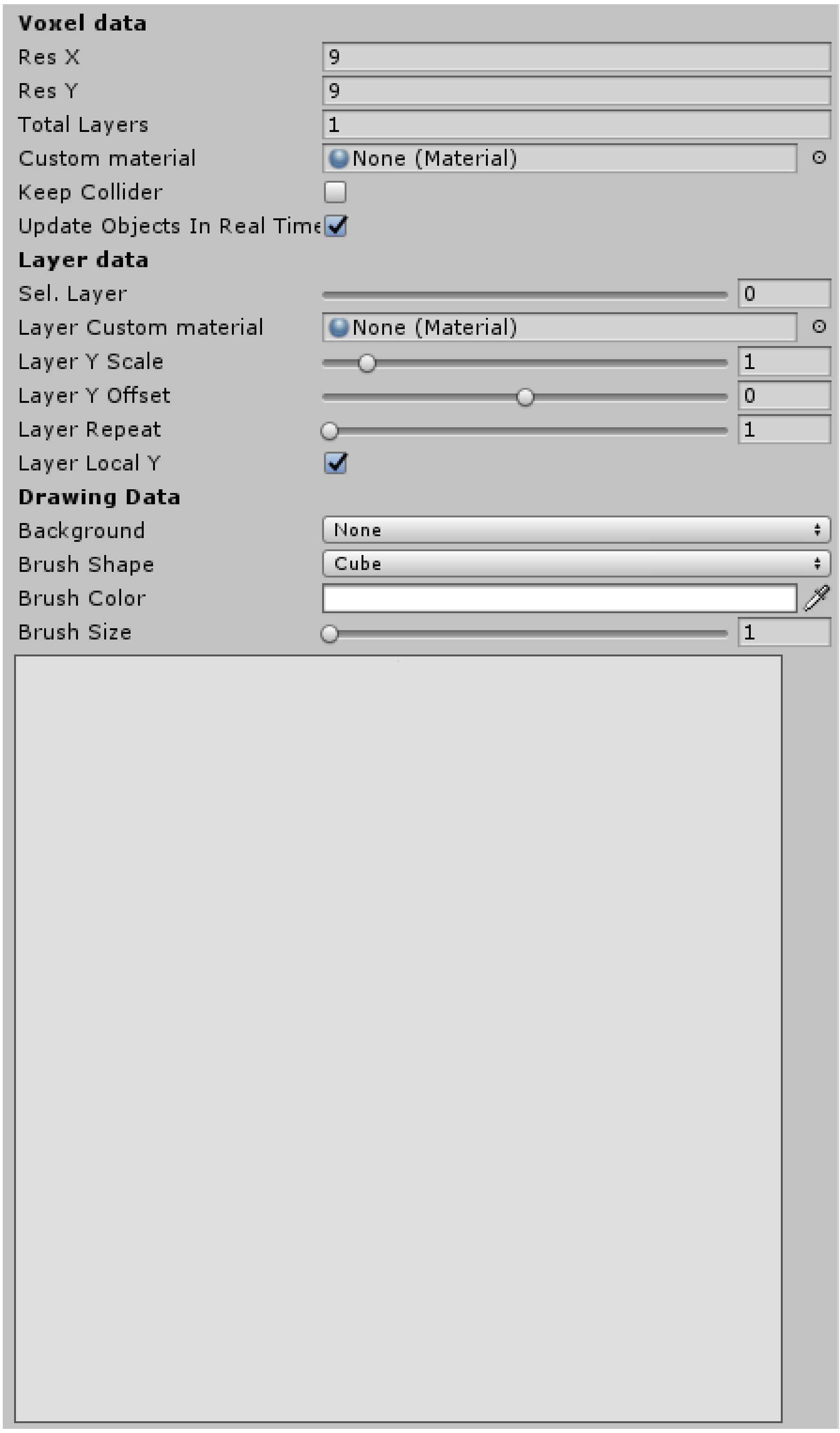
4. Use Modifiers: Modifity this voxel only for this GameObject, we will see this later on.

So, first we need to create a Voxel Data so our GameObject can render it.

Create a folder in the Assets, and with the right mouse button, create a ProtoVoxelData:



Once you create this object, you will see in the inspector this information:



Voxel data

Res X: 9

Res Y: 9

Total Layers: 1

Custom material: None (Material)

Keep Collider: □

Update Objects In Real Time: ☑

Layer data

Sel. Layer: 0

Layer Custom material: None (Material)

Layer Y Scale: 1

Layer Y Offset: 0

Layer Repeat: 1

Layer Local Y: ☑

Drawing Data

Background: None

Brush Shape: Cube

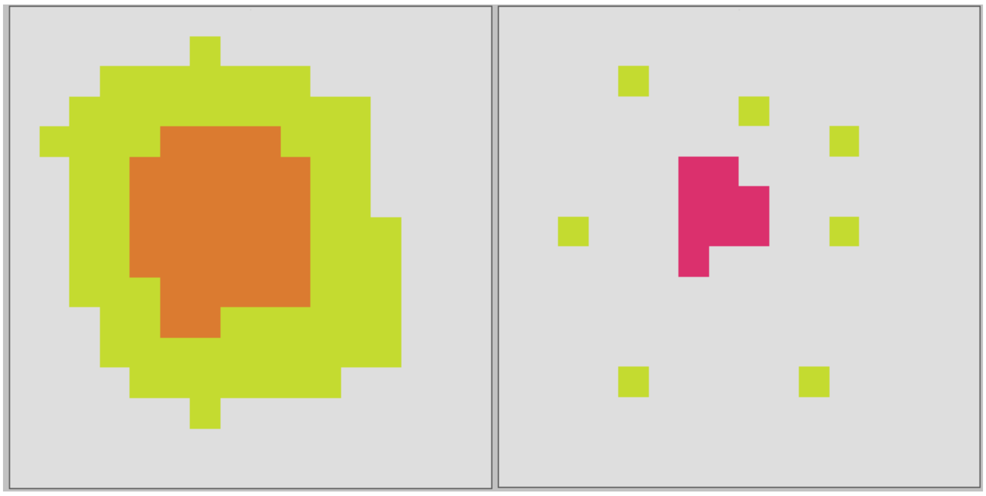
Brush Color:

Brush Size: 1

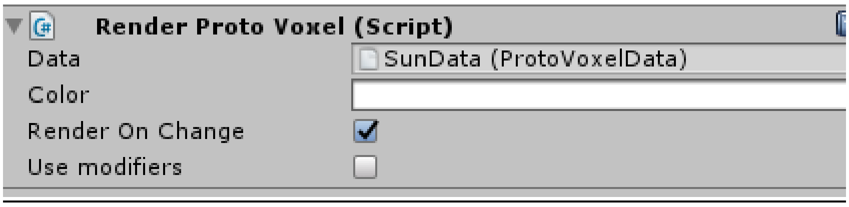
This is where the magic happens, you create a voxel by drawing a image in a 2D canvas. So lets go by all elements:

1. Voxel Data:
    a. Res X/Y: How much pixels you want your canvas in the X/Y axis, this create more detail for you to drawn.
    b. Total Layers: A voxel is created from multiple 2D layers, each layer stack on top of the last one, you can set how much layers your final voxel will have with this.
    c. Custom Material: The voxel will use the standard material, or you can set a new material here that will be used by this voxel.
    d. Keep Colliders: Will add colliders to the voxel if on.
    e. Update Objects In Real Time: If on, any object in the scene that have this Voxel will update in real time while you drawn in the canvas.
2. Layer Data:
    a. Sel Layer: The inspector will show the information and canvas of the selected layer.
    b. Layer Custom Material: If you want a material only for this layer, it will override the default material.
    c. Layer Y Scale: The Y scale can be changed to create a taller layer.
    d. Layer Y Offset: Make this layer hover above or overlay with the last later.
    e. Layer Repeat: Create multiple copies of this layer on top of each other, good for towers and alike.
    f. Layer Local Y: Make the position of this layer relative to the last layer. If turned off the layer will appear on Y = 0 position of the voxel.
3. Drawing Data:
    a. Background: While drawing in the canvas, you can render the lower Layer or upper layer to get a reference, making it easier to draw.
    b. Brush Shape: You can drawn squares, that are the most common in voxels, or you can select another prefabs, even a custom GameObject to use as a brush.
    c. Brush Color: Set the color that will be used in the canvas:
    d. Brush Size: Control the size of the brush

So, once you got everything set up, just start to draw, for me, I made little sun with two layers (Layer 1, Layer 2):



So, lets add this information to our RenderProtoVoxel:



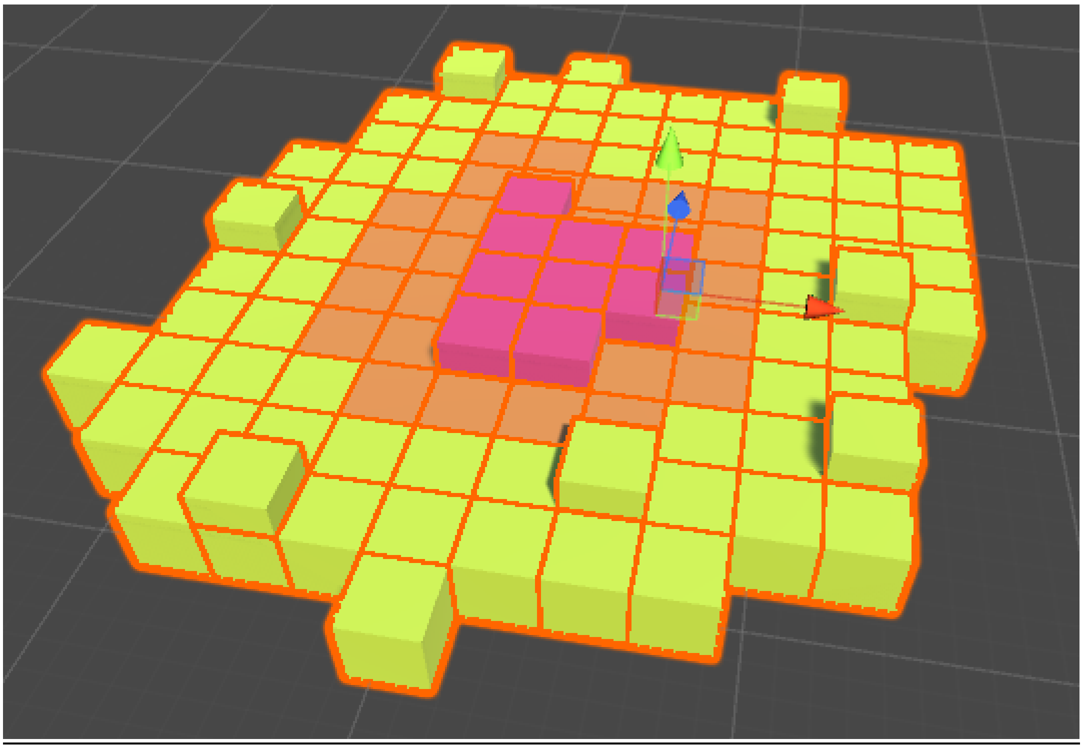
Render Proto Voxel (Script)

Data SunData (ProtoVoxelData)

Color

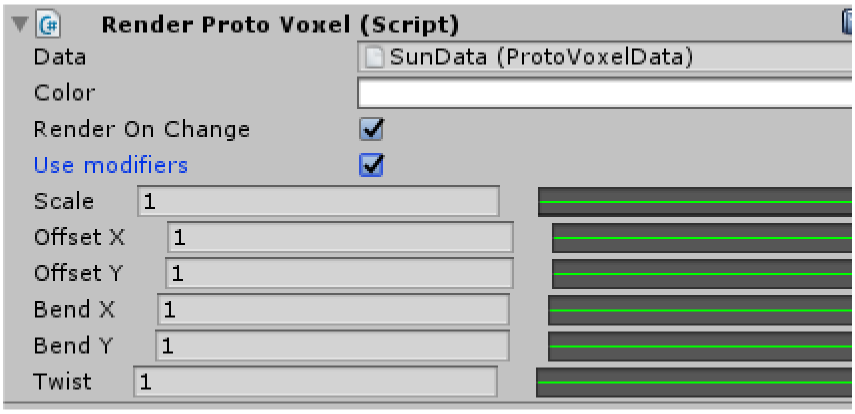
Render On Change ☑

Use modifiers ☐



TADA! Instantly we have a sun in our scene (if render on change is turned on)

So, let's see those modifiers parameters inside RenderProtoVoxel:



Render Proto Voxel (Script)

Data: SunData (ProtoVoxelData)

Color

Render On Change ☑

Use modifiers ☑

Scale 1

Offset X 1

Offset Y 1

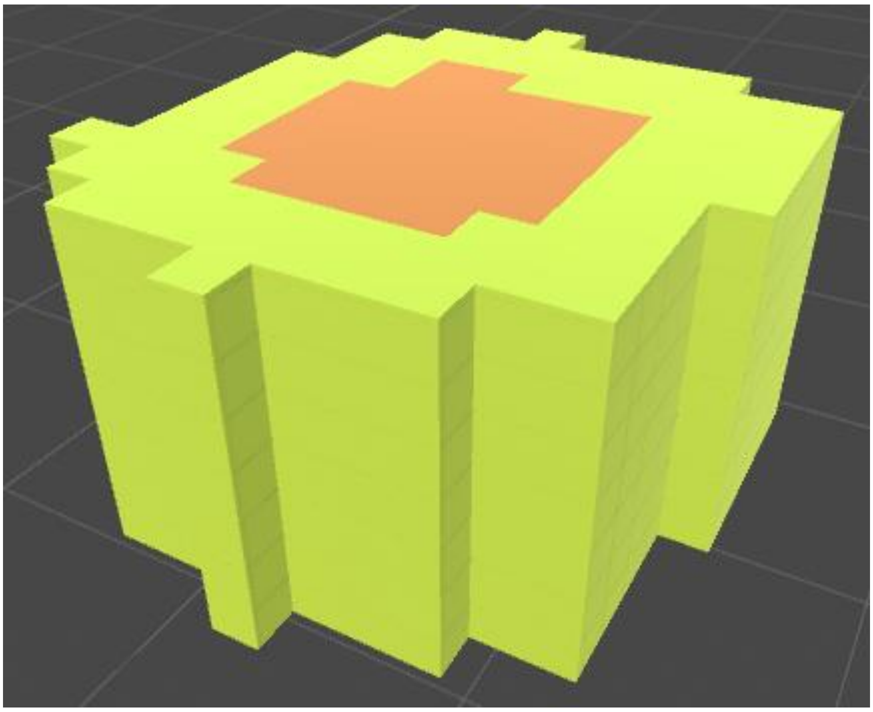
Bend X 1

Bend Y 1

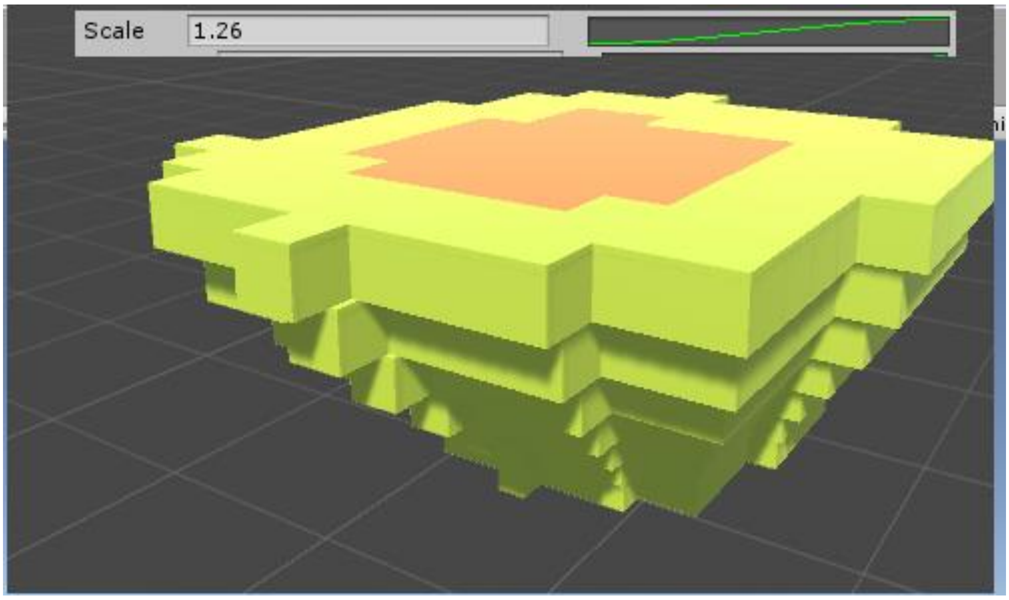
Twist 1

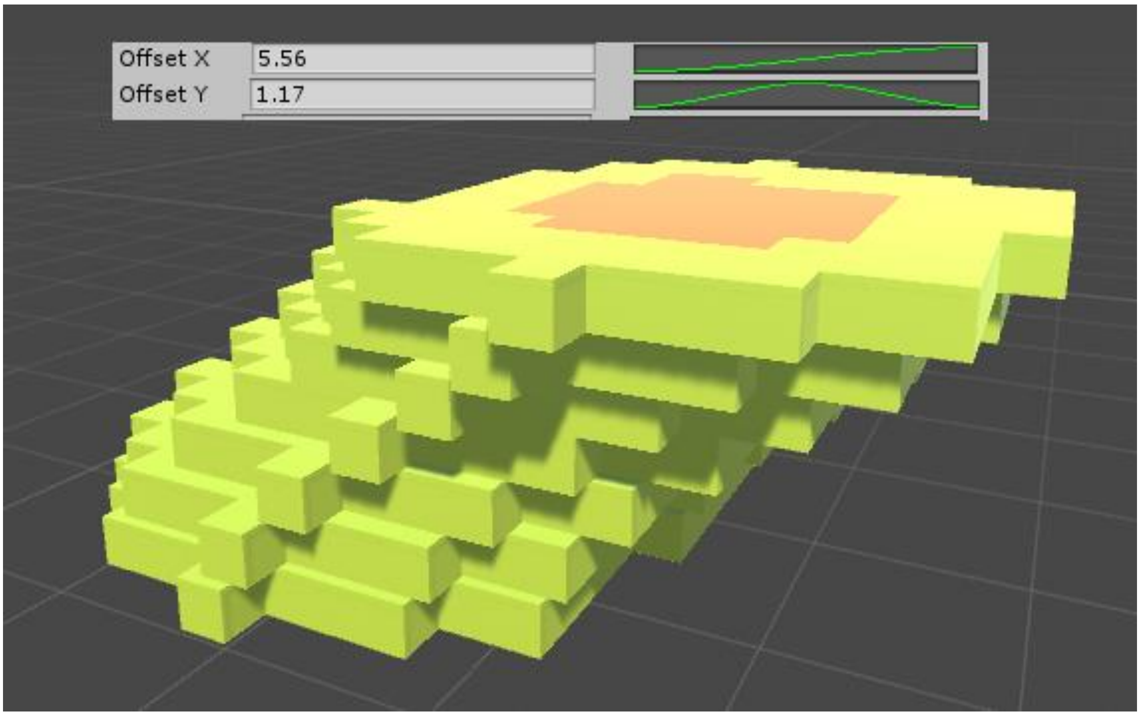
1. Scale: Control the scale of the voxel, the animation curve control the scale from -Y to +Y using the float value.
2. Offset X/Y: Offset each layer by a value, use the animation curve to control the offset in the X/Y axis.
3. Bend X/Y: Bend each layer, creating a curve effect in the voxel.
4. Twist: Rotate each layer.

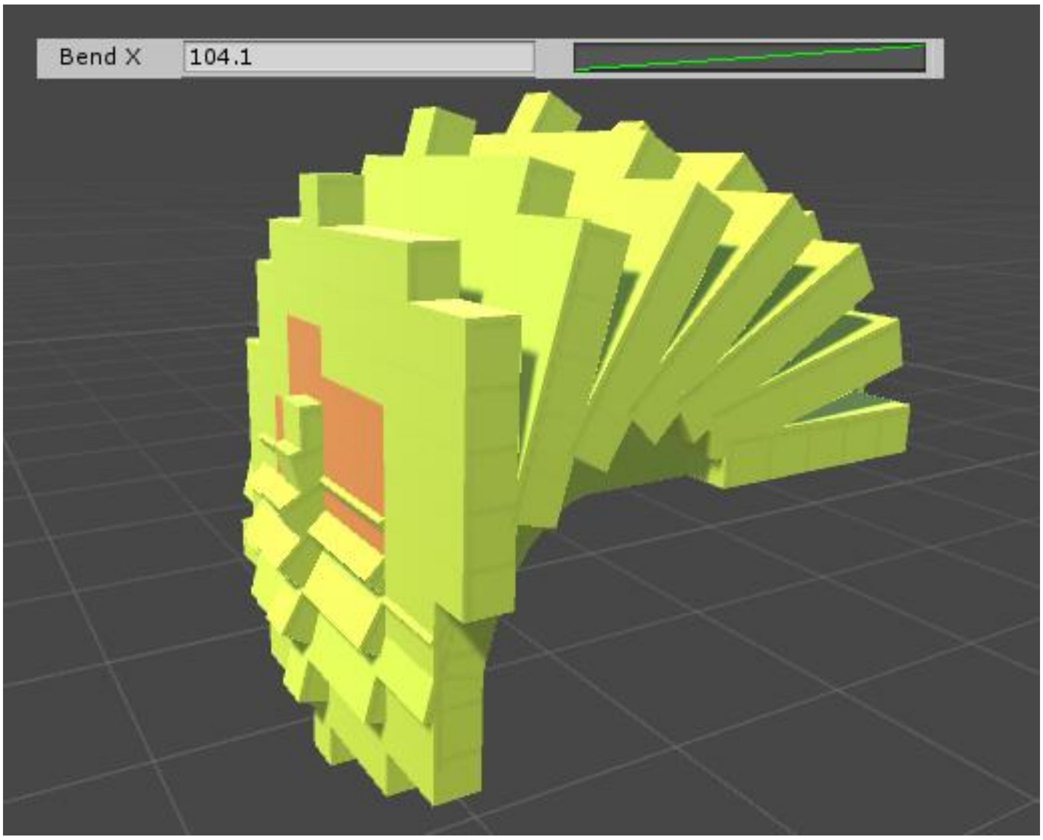
So, using the sun example, let's use the "Layer Repeat" to create a small tower with our sun:



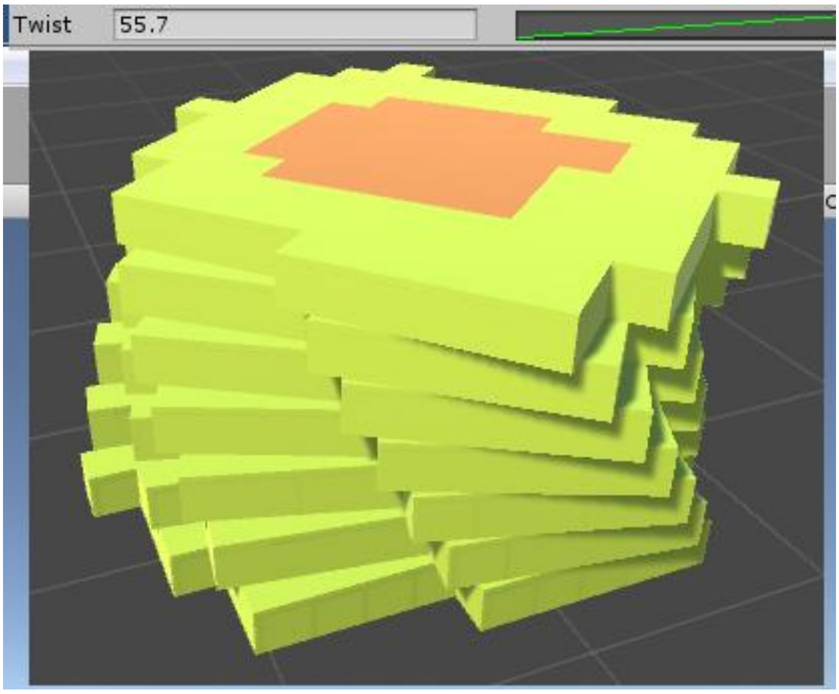
Set see each modifier:


Scale 1.26


Offset X 5.56
Offset Y 1.17

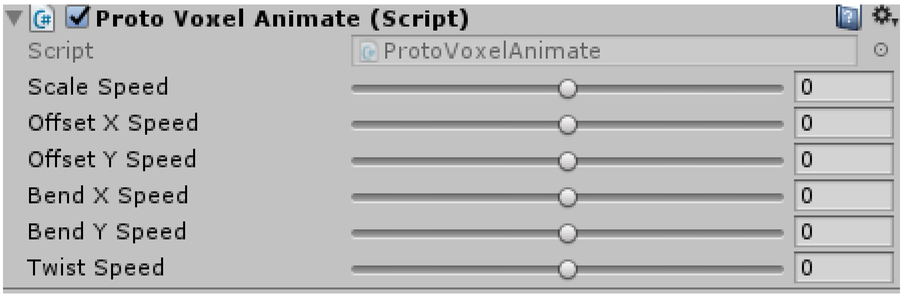


Bend X 104.1



Twist 55.7

Now we can animate each of these modifiers if we want, just add the ProtoVoxelAnimate script:



☑ Proto Voxel Animate (Script)

Script	ProtoVoxelAnimate
Scale Speed	0
Offset X Speed	0
Offset Y Speed	0
Bend X Speed	0
Bend Y Speed	0
Twist Speed	0

Just set the speed you want each value to be updated and hit play, and this is all!