



UNIVERSITY OF MALTA
Institute of Digital Games

Creating Virtual Worlds

Lecture **7**

Terrain, Grass, and Particles

Our topics today ...



Terrain creation

Using the Unity terrain creation component



Painting terrain textures

Blending between different tiling ground textures



Painting grass billboards

Adding landscape geometry to a terrain



Particle Effects

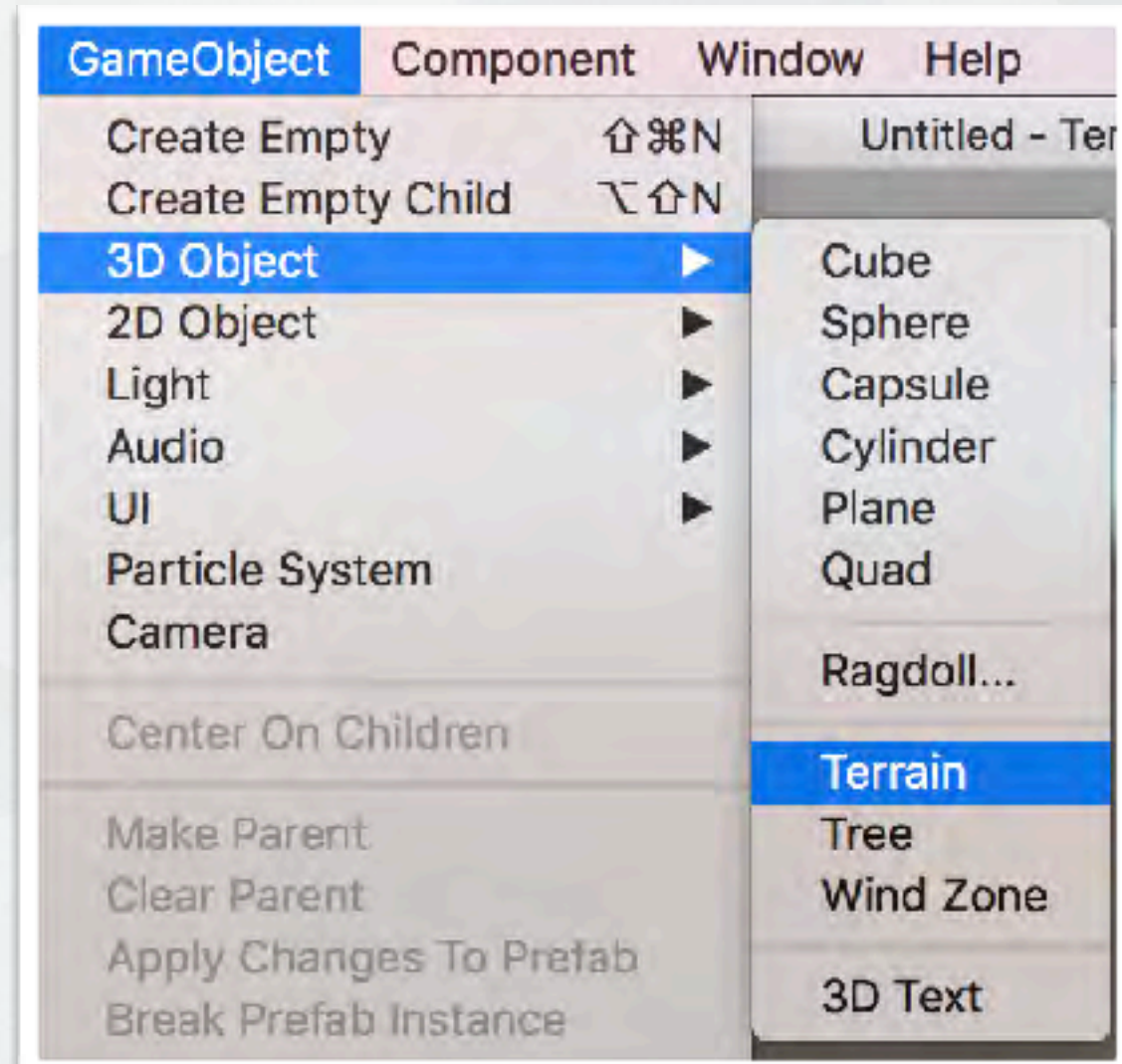
Using billboard sprites to create complex effects

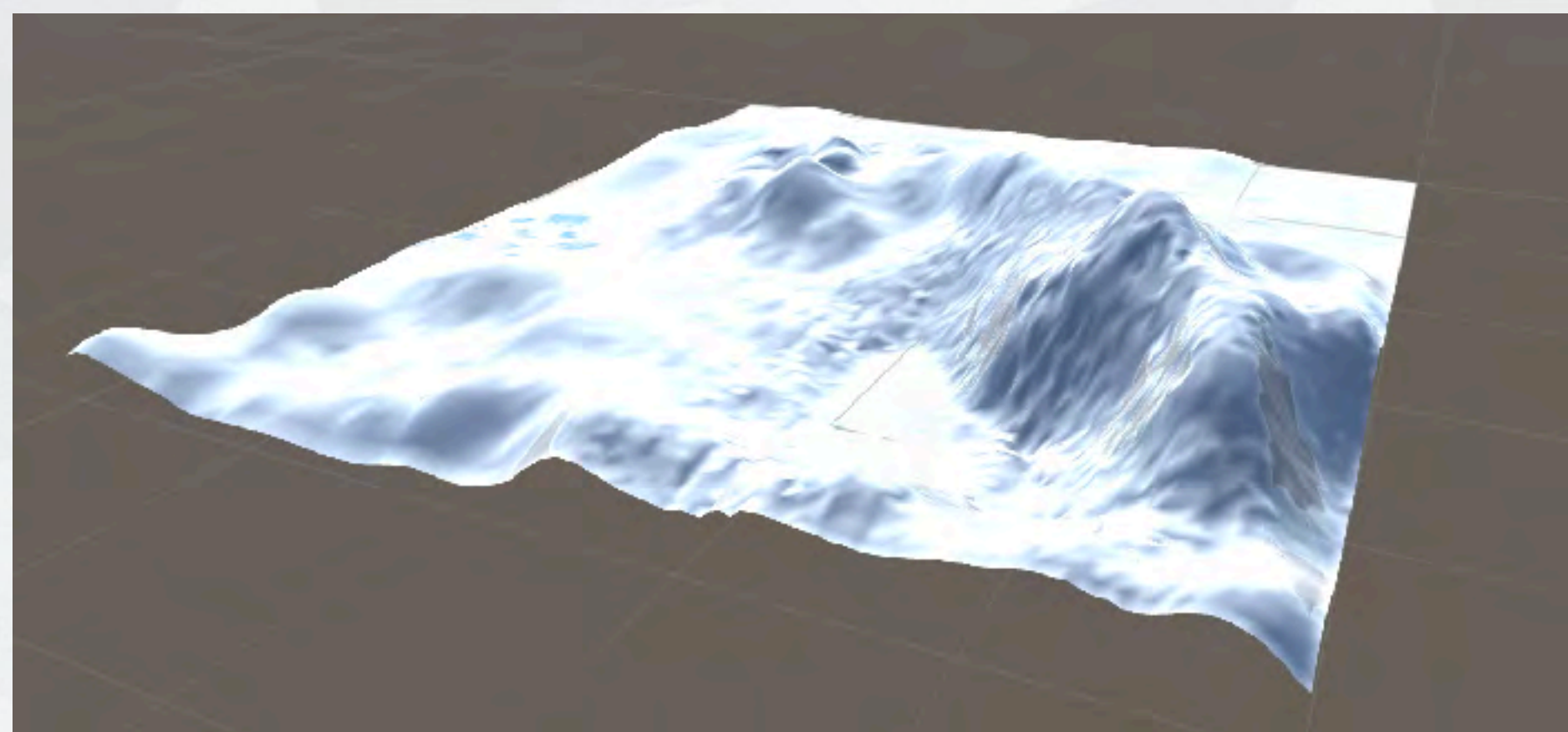
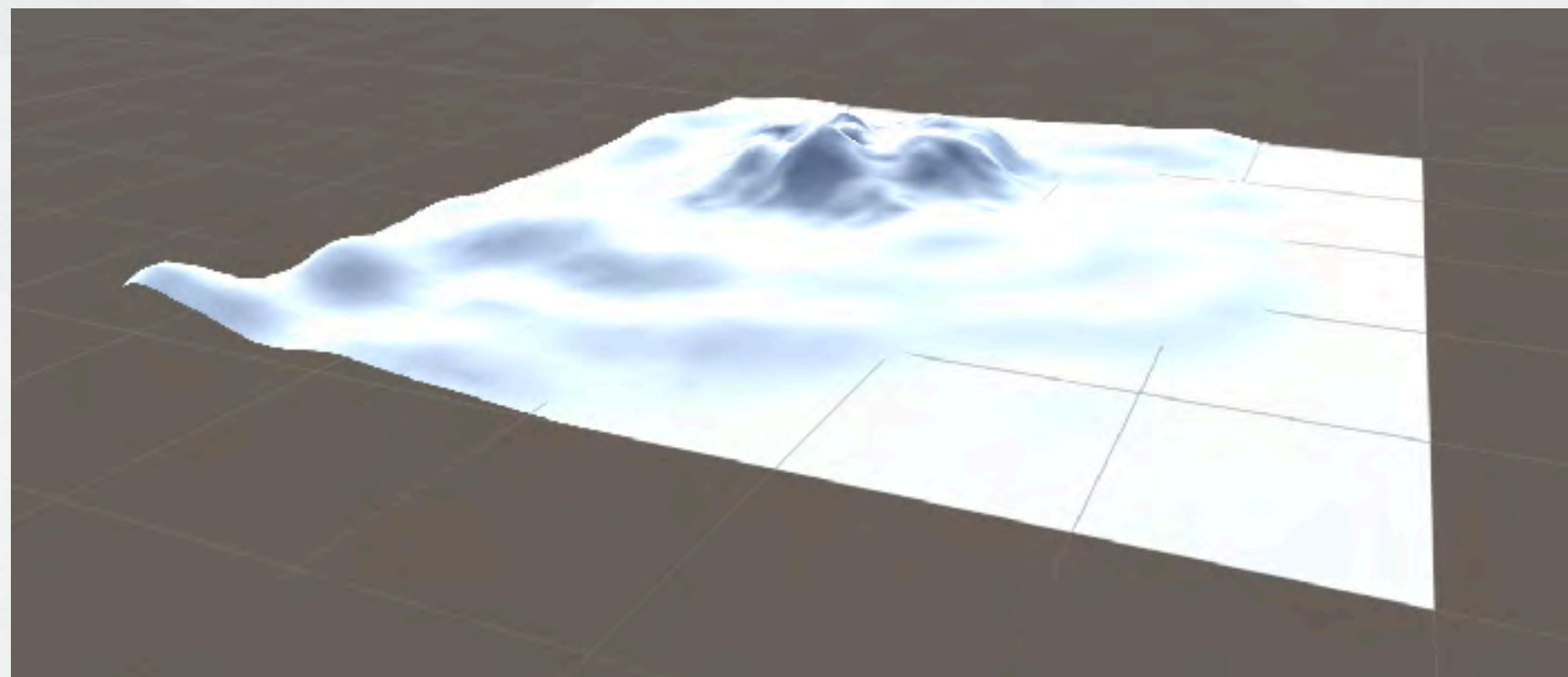
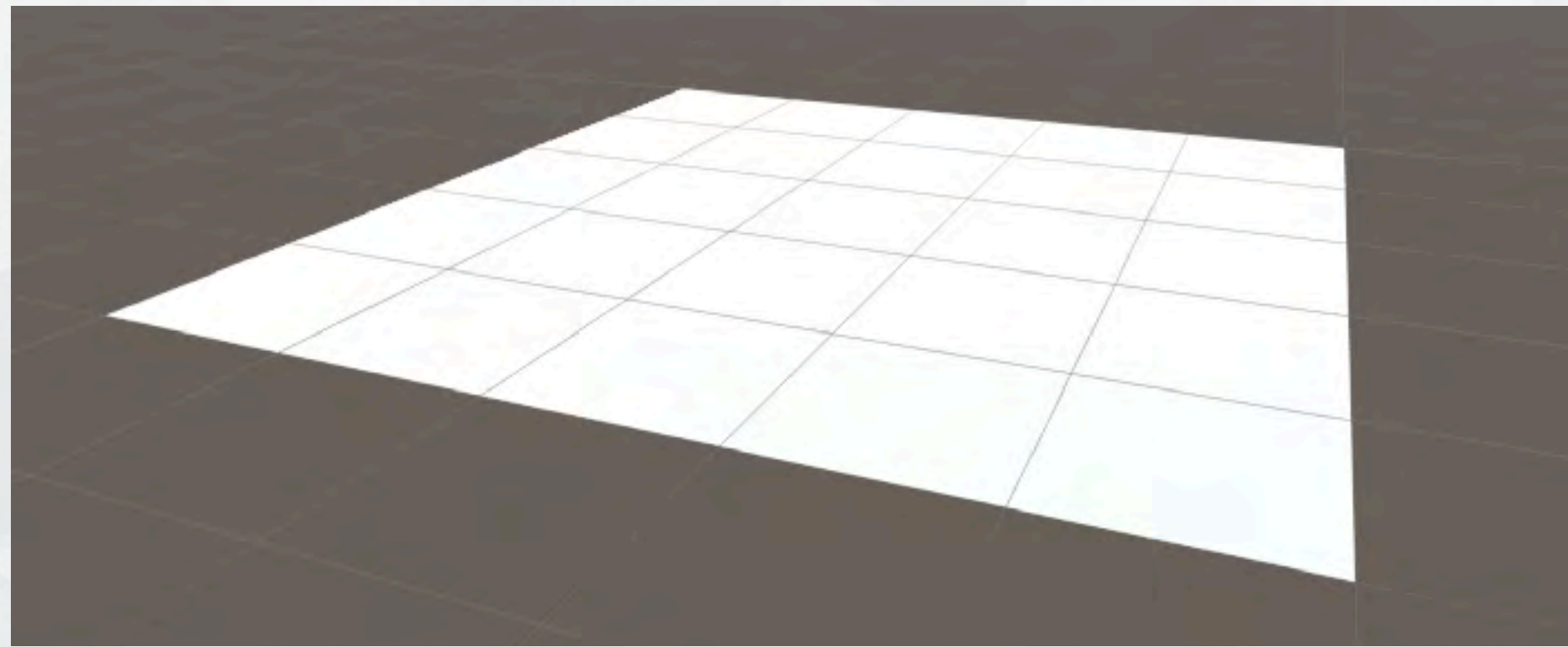
Terrains in Unity

GameObject > 3D Object > Terrain

Used to create optimised 3D geometry that serves as 'ground' within Unity.

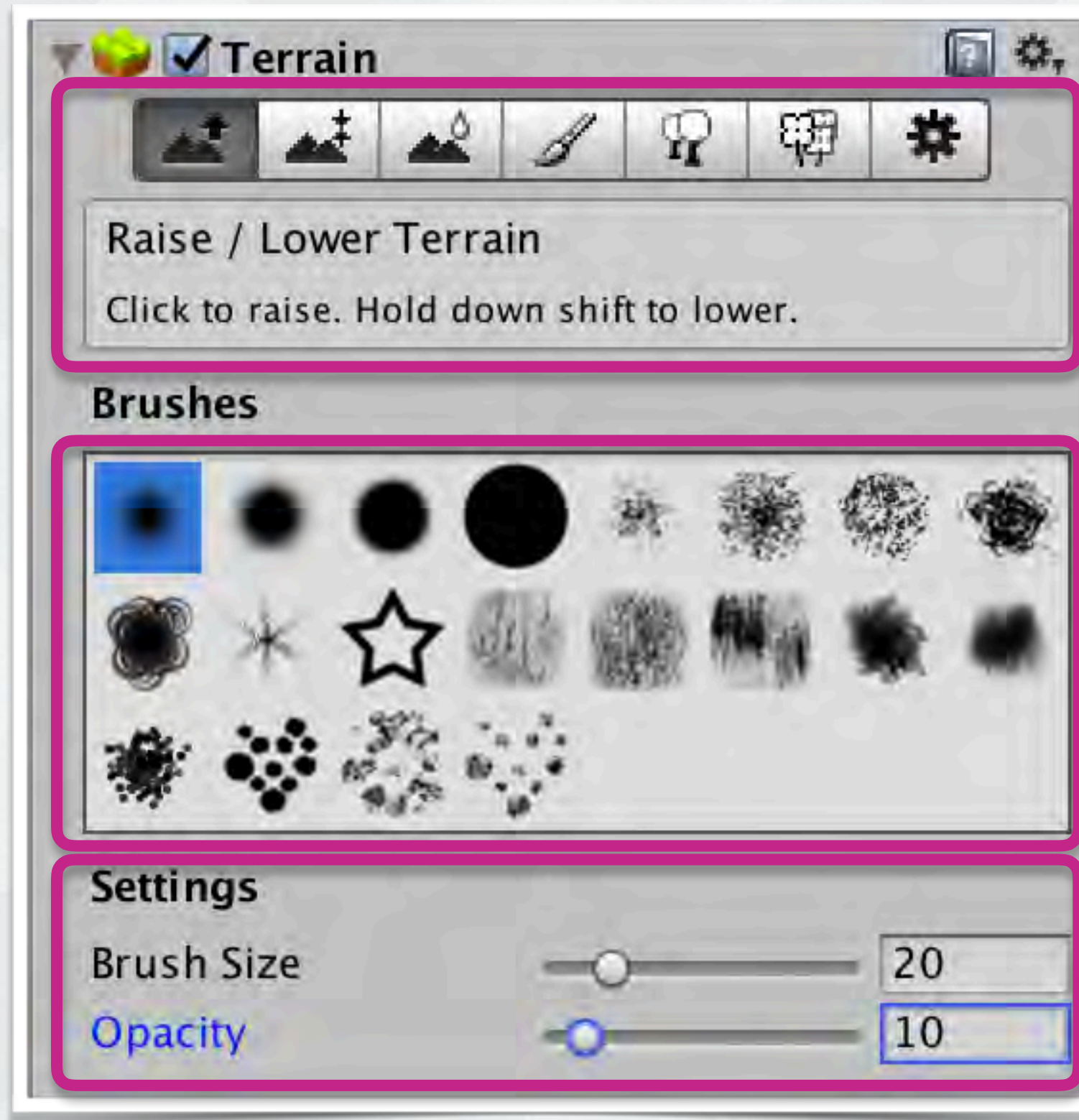
The Terrain component is also used to blend between different tiling textures – resulting in detailed ground textures without using high resolution images!





- ▶ Creates a **500 x 500 'meters'** plane in the scene (default)
- ▶ Consists of 2 components: **Terrain**, and **Terrain Collider**
- ▶ Transform node can only affect position (not rotation or scale)
- ▶ **Static** by default (using lightmaps)

Modifying the shape



Terrain Toolbar (plus tool description below)

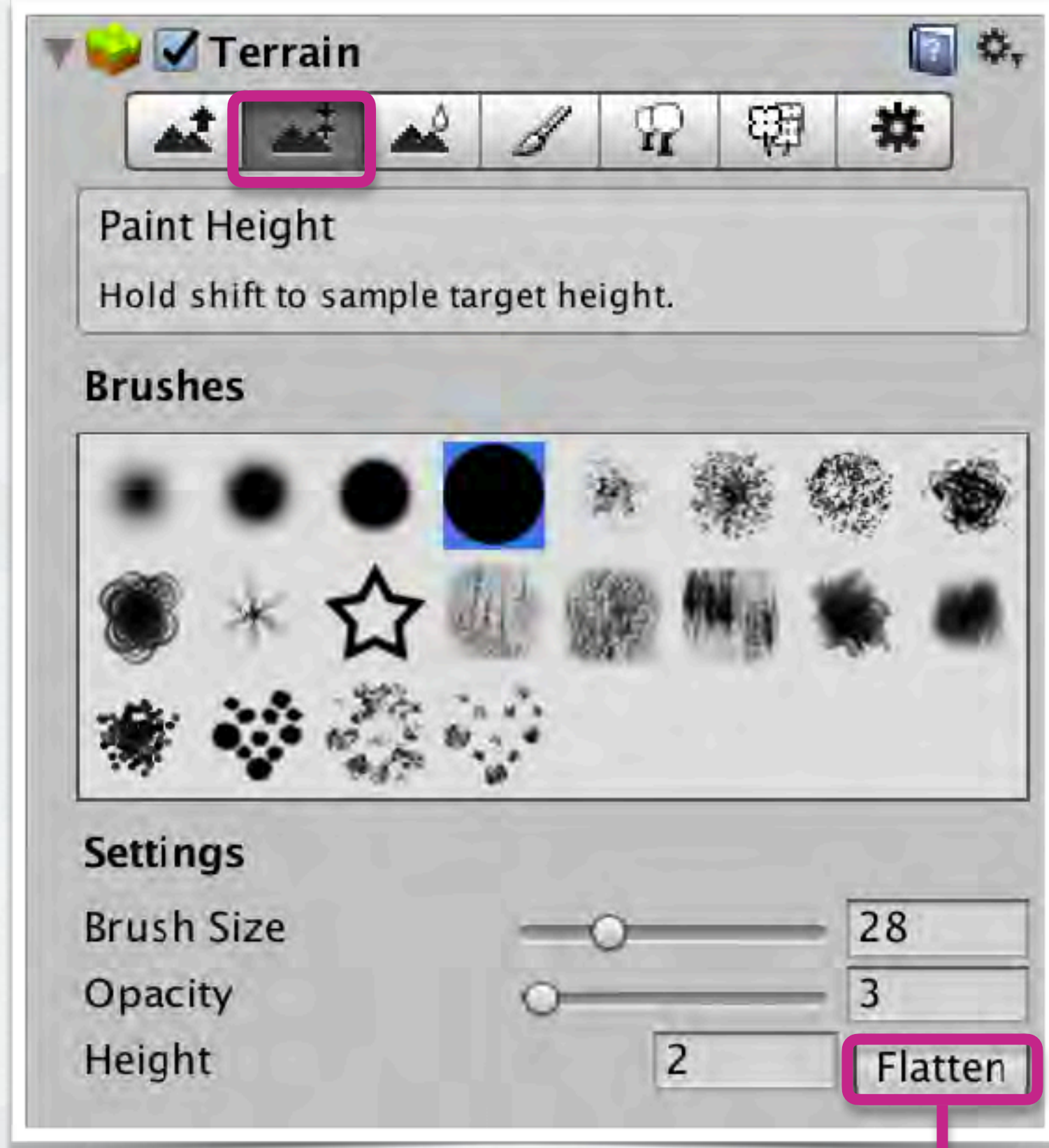
“Brush” Selection – affecting the currently selected tool

Brush size and opacity (‘strength’ of tool action)

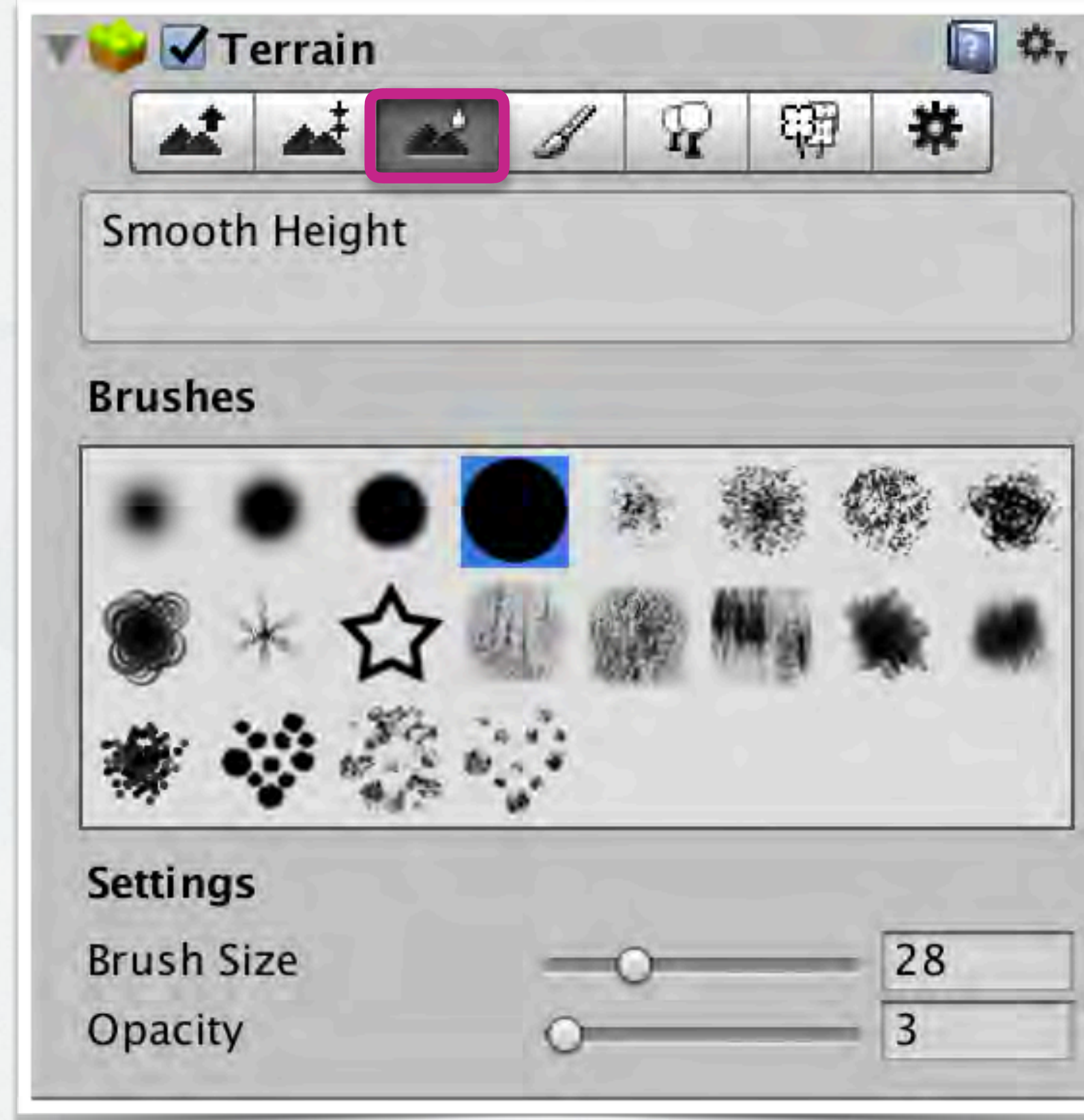
Shaded 2D Gizmos Q+All

The image shows a Unity 3D editor interface. On the left, a 3D view of a terrain with a blue circular brush applied. The right-hand side shows the Hierarchy panel with the following components and settings:

- Terrain** (checked): Static (checked), Tag: Untagged, Layer: Default.
- Transform** (expanded): Position (X: 0, Y: 0, Z: 0), Rotation (X: 0, Y: -9.588, Z: 0), Scale (X: 1, Y: 1, Z: 1).
- Terrain** (checked): Includes icons for Raise/Lower Terrain, Raise/Lower Terrain (text), Raise/Lower Terrain (instructions), Brushes (grid of brush icons), Settings (Brush Size: 28, Opacity: 3).
- Terrain Collider** (checked): Includes an "Add Component" button.



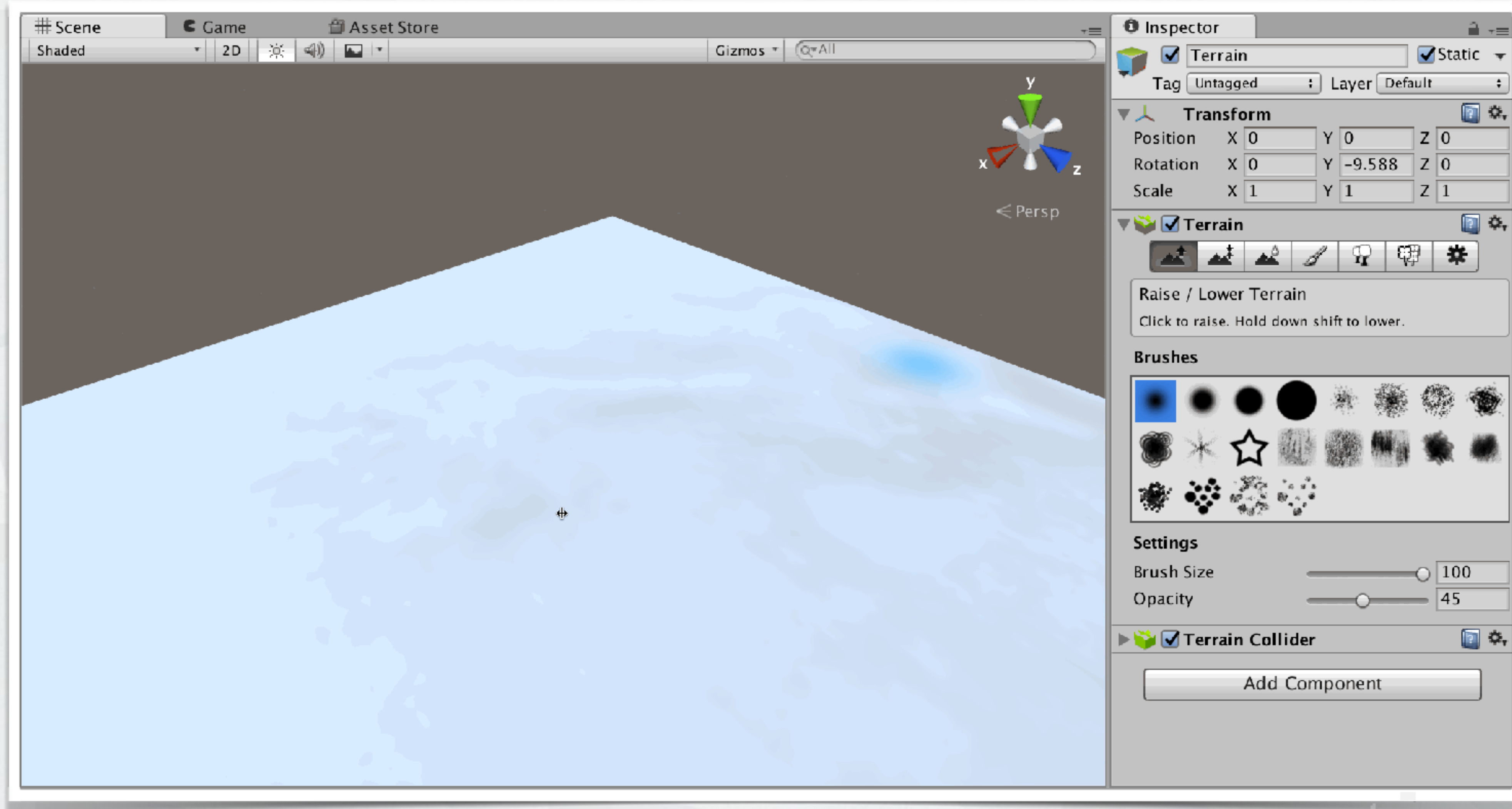
Flattens the whole terrain to specified height



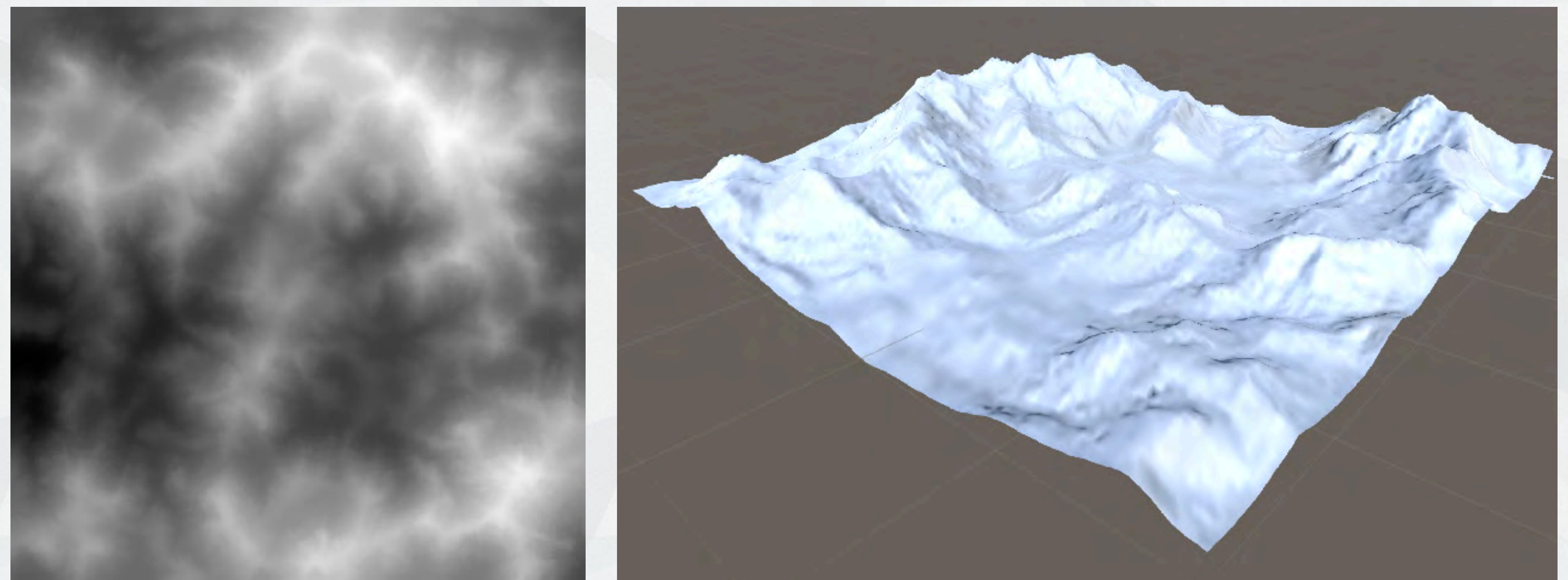
Since the terrain height cannot be negative, start by flattening it at a higher value!

Smooth brush strokes to blend between individual strokes

Video: 'Sculpting' terrain geometry



Creating a terrain based on a heightmap



A great way to import realistic terrain shapes, but RAW files can be tricky

Terrain

Terrain Settings

Base Terrain

Draw

Pixel Error

Base Map Dist.

Cast Shadows

Material Built In Standard

Reflection Probes Blend Probes

... more stuff ...

Terrain Height 40

Heightmap Resolution 1025

Detail Resolution 1500

Detail Resolution Per F 10

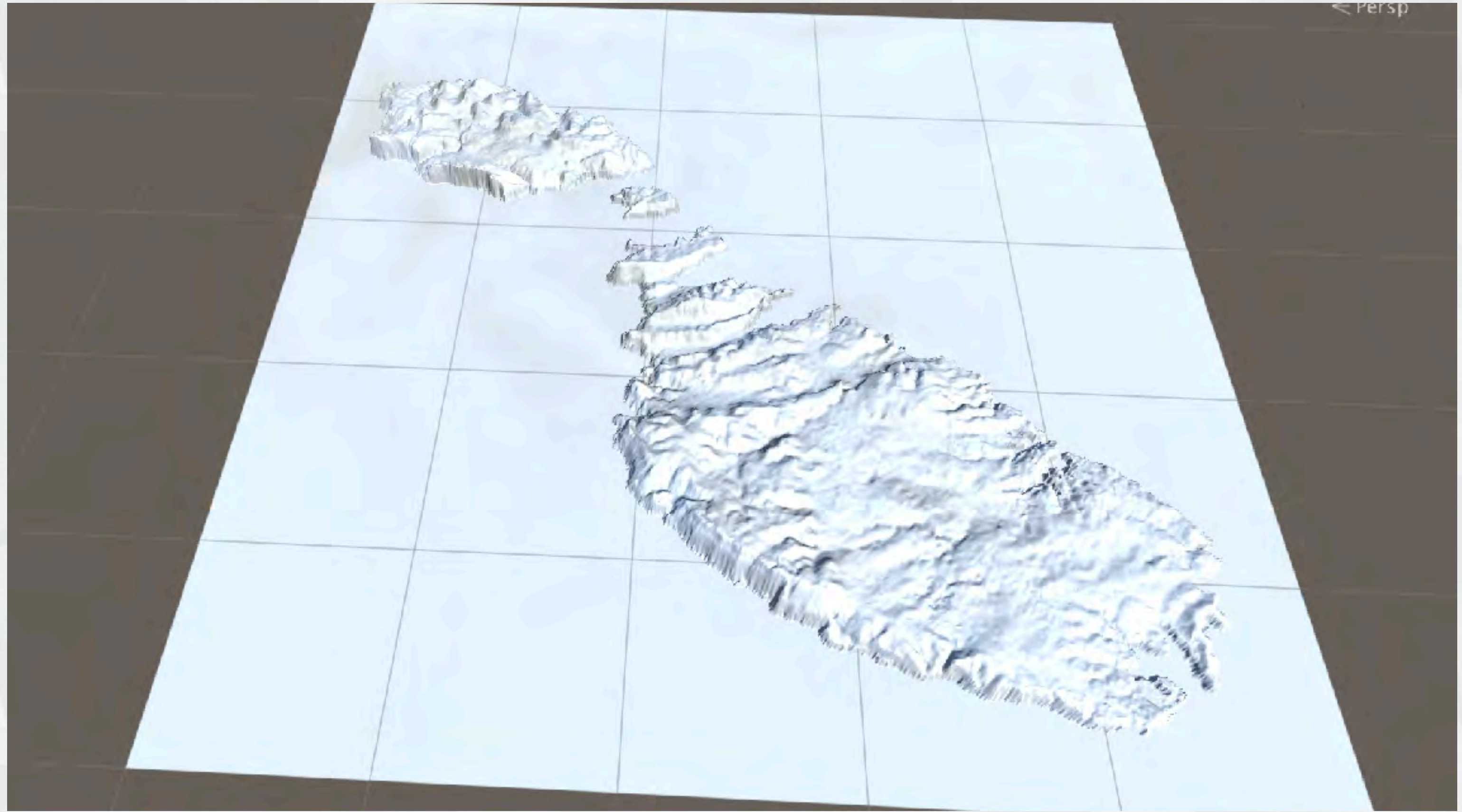
Control Texture Resol 1024

Base Texture Resoluti 1024

Please note that modifying the resolution of the heightmap, detail map and control texture will clear their contents, respectively.

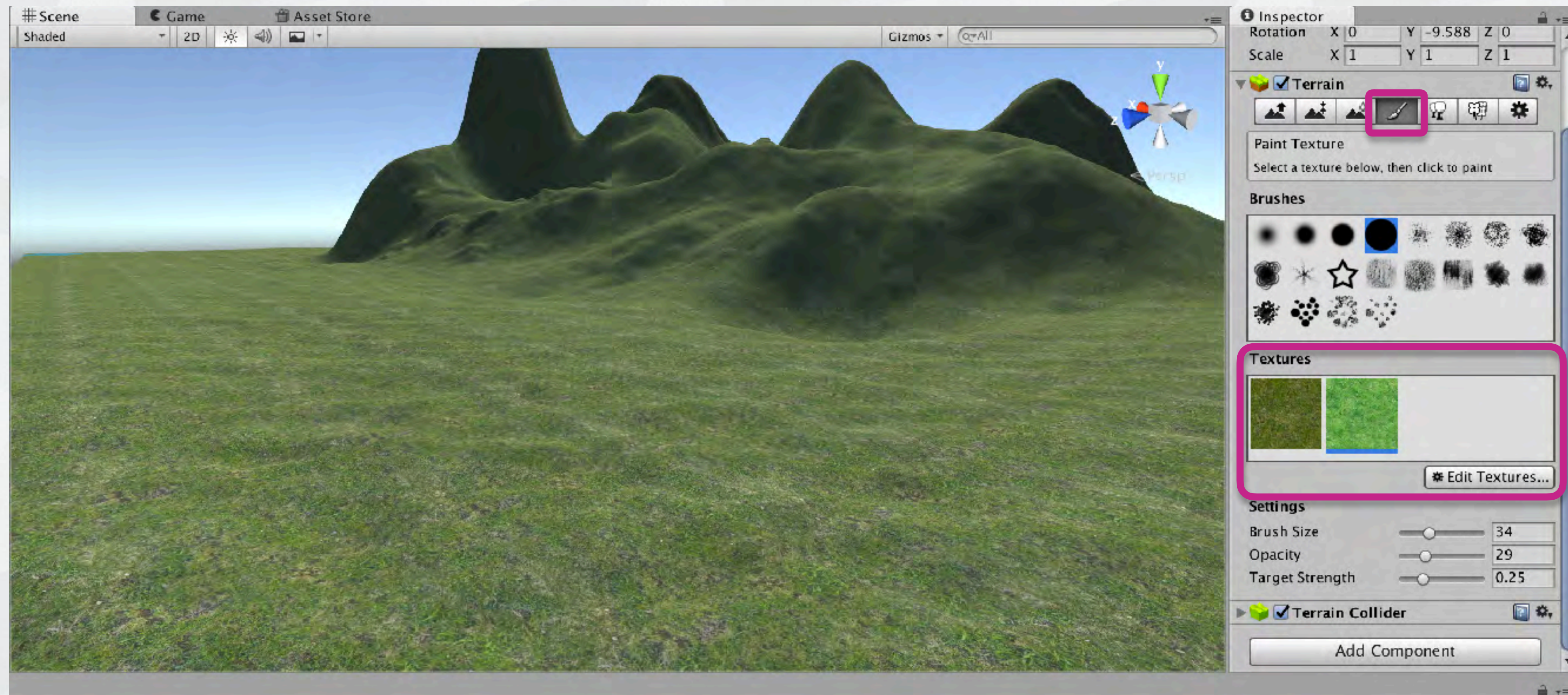
Heightmap

Import Raw... Export Raw...



Real-world heightmaps can be used to re-create real landscapes (check <http://terrain.party>)

Painting Texture Detail

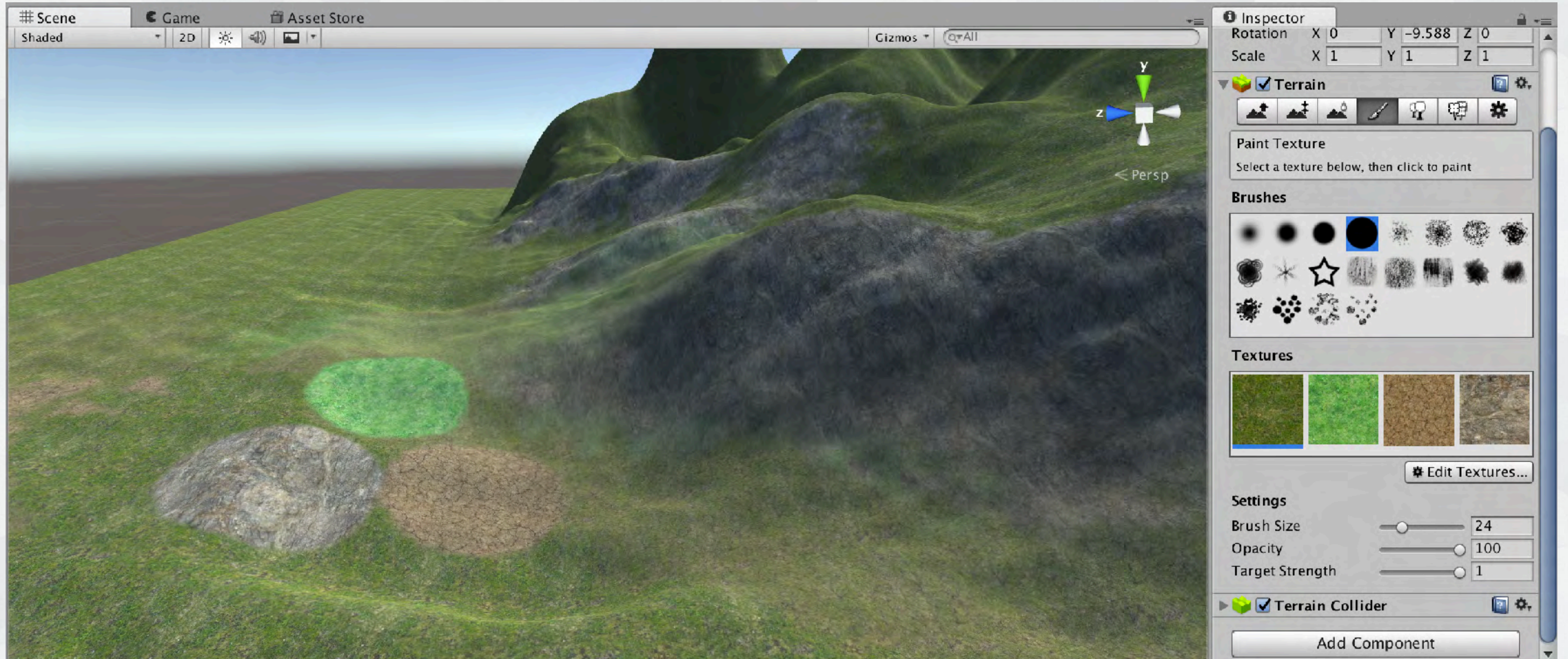


Terrains can be painted using multiple tiling textures. The first texture acts as the 'base' coat.

By blending between several tiling textures, we can have variety without requiring a gigantic image!



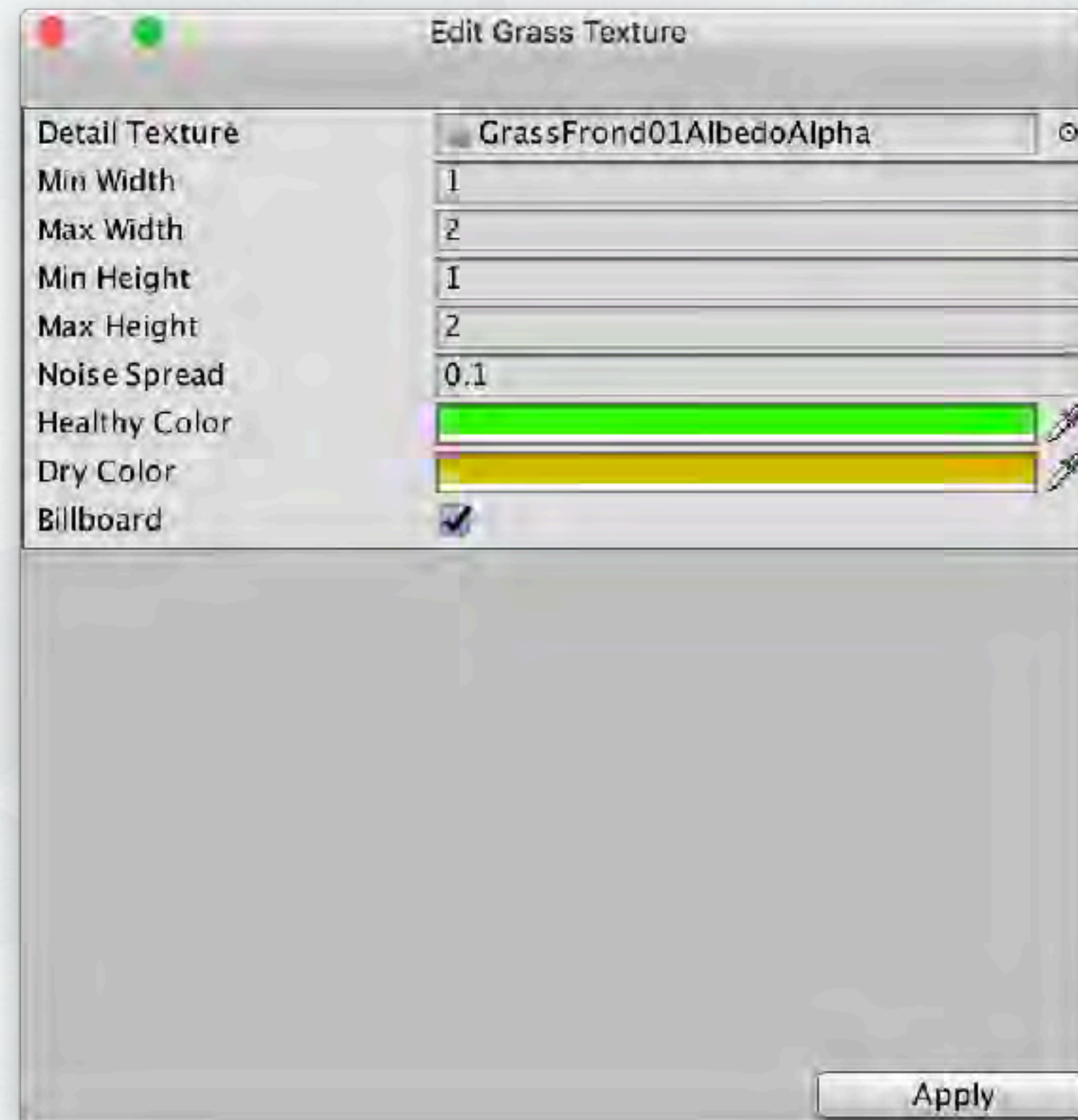
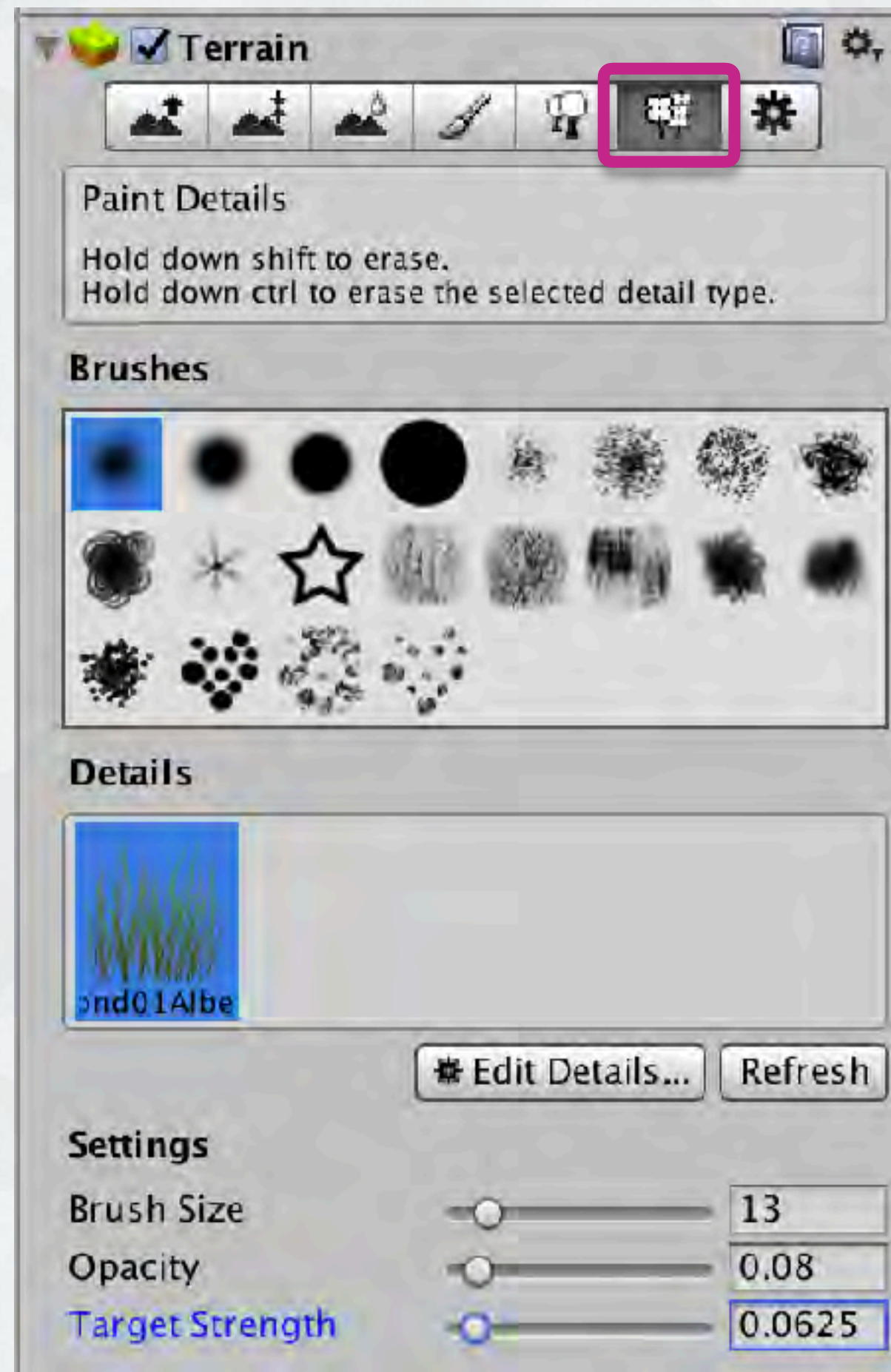
**Example of blending between two textures
(base texture + dry ground texture)**



**Example of painted textures with a hard border.
Also note the use of the rock texture on the right half.**

Grass

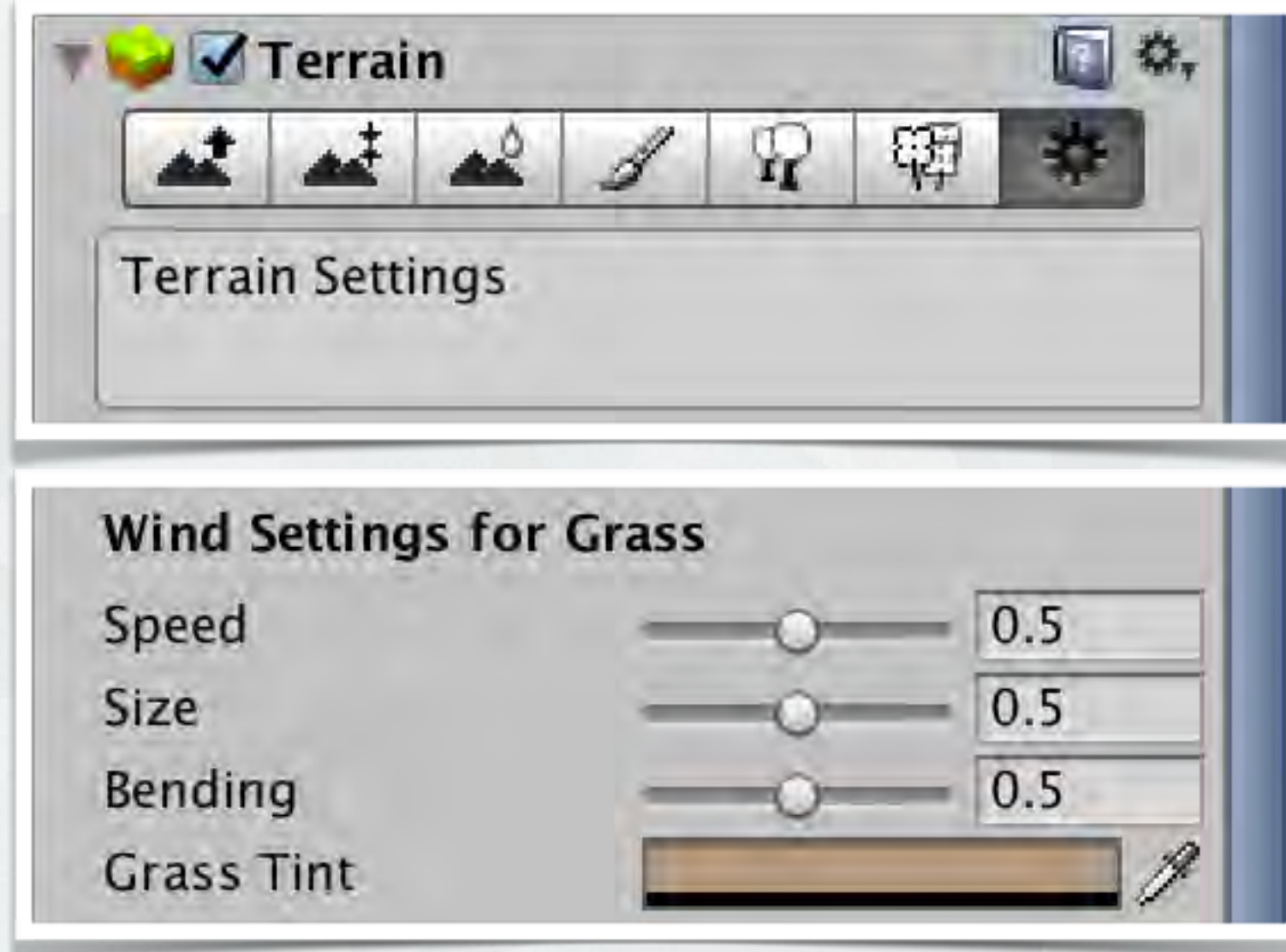
Apart of texture, we can also 'paint' geometry, such as grass billboards.



Billboards are sprites that are always angled towards the camera!



Video: Painting grass



**We can define
a fake wind
effect for
painted grass**

Video: Wind effect for grass

Particle Effects

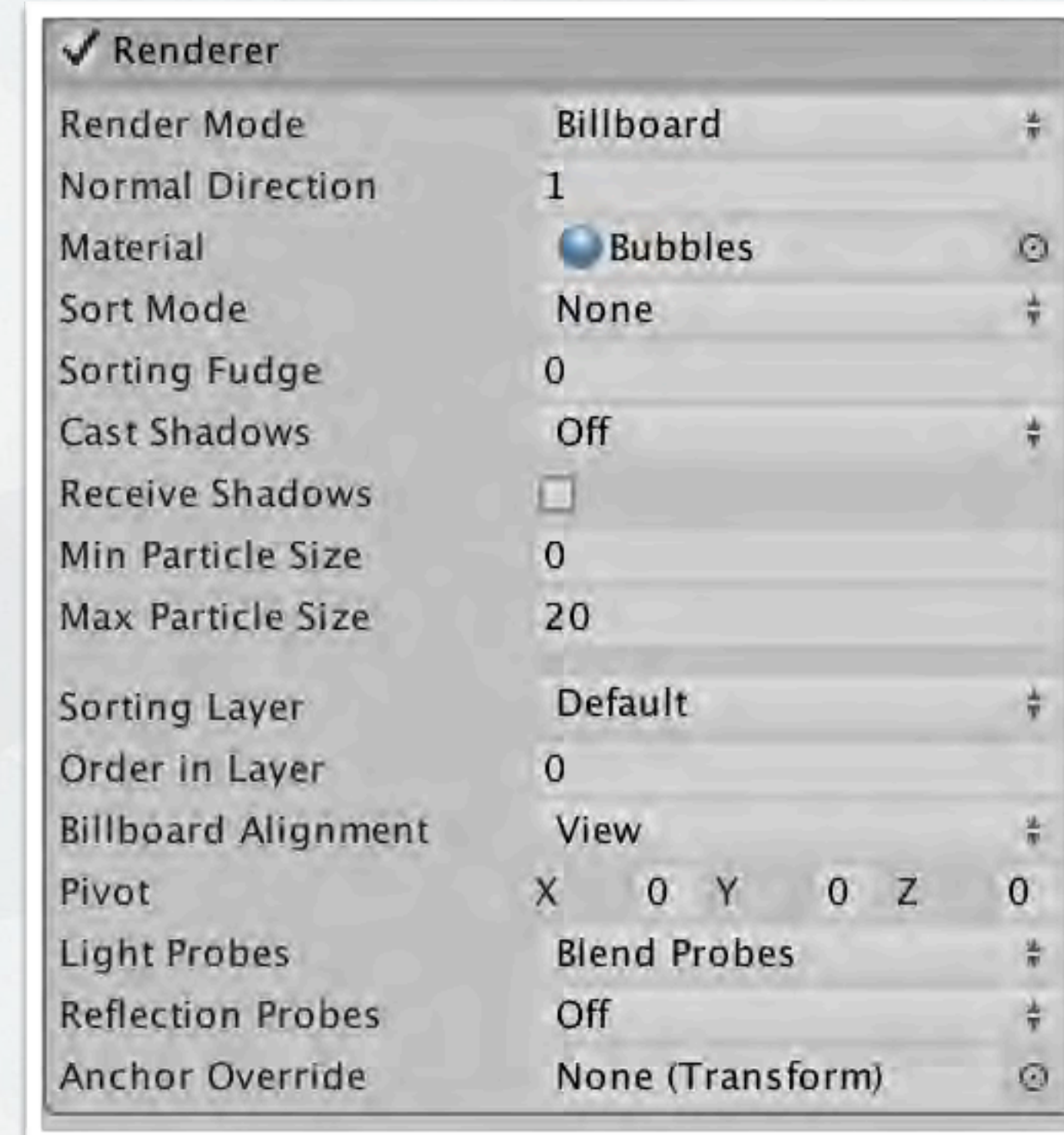
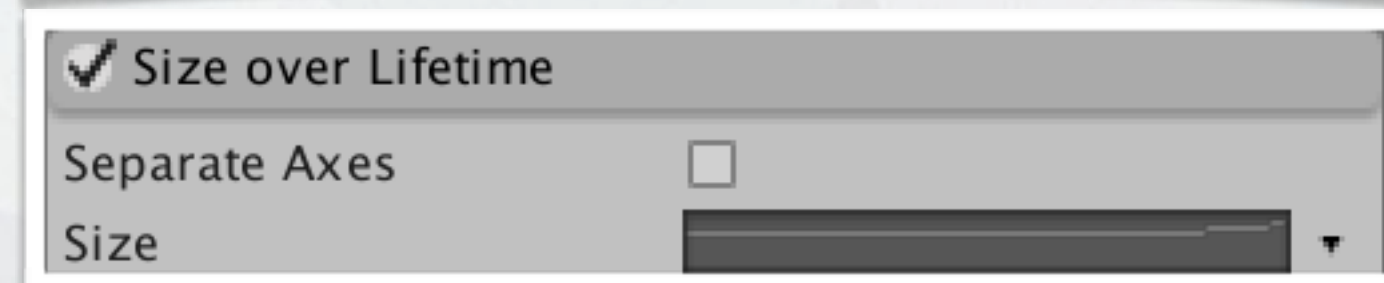
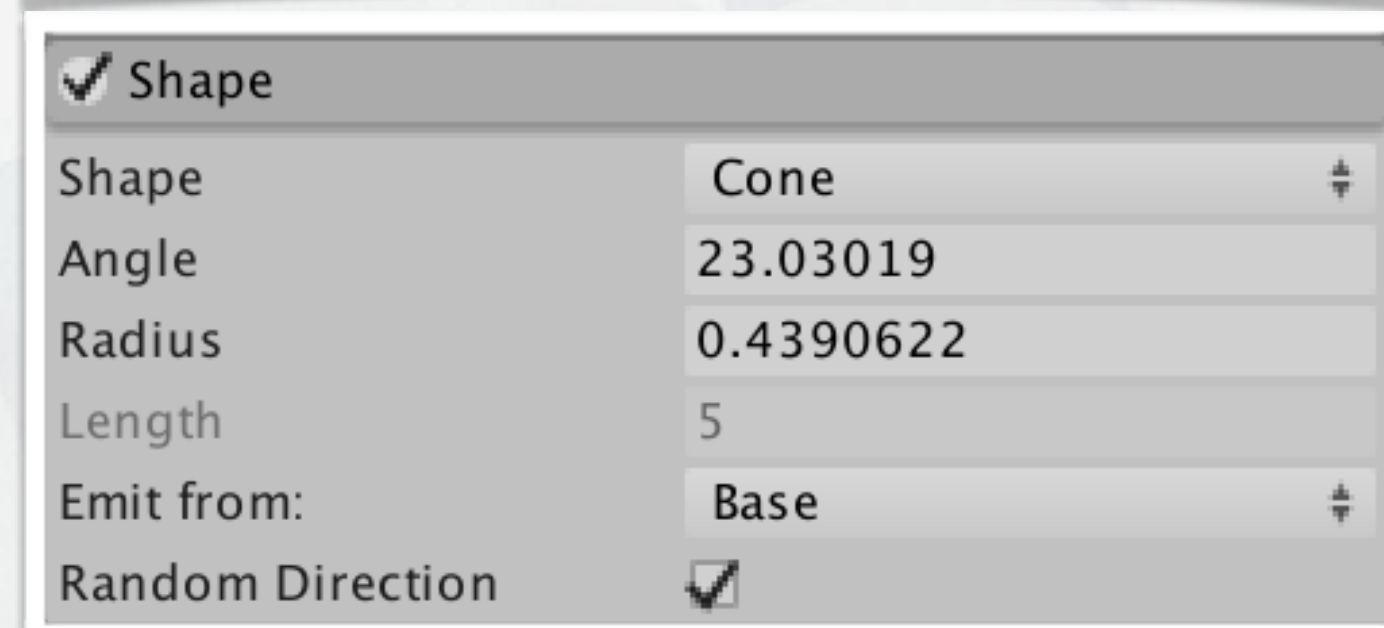
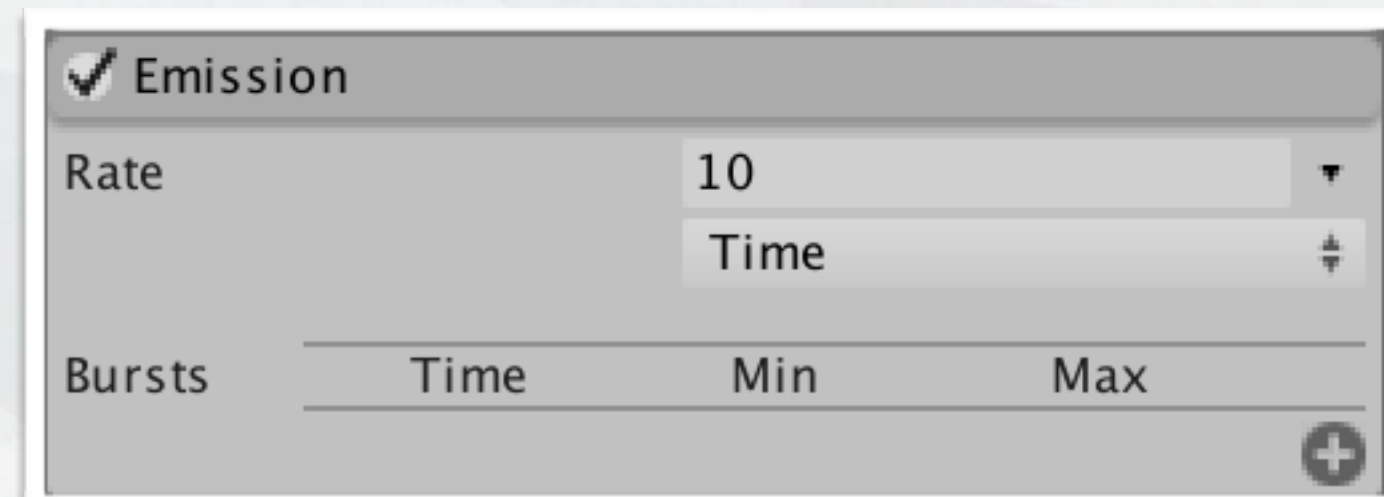
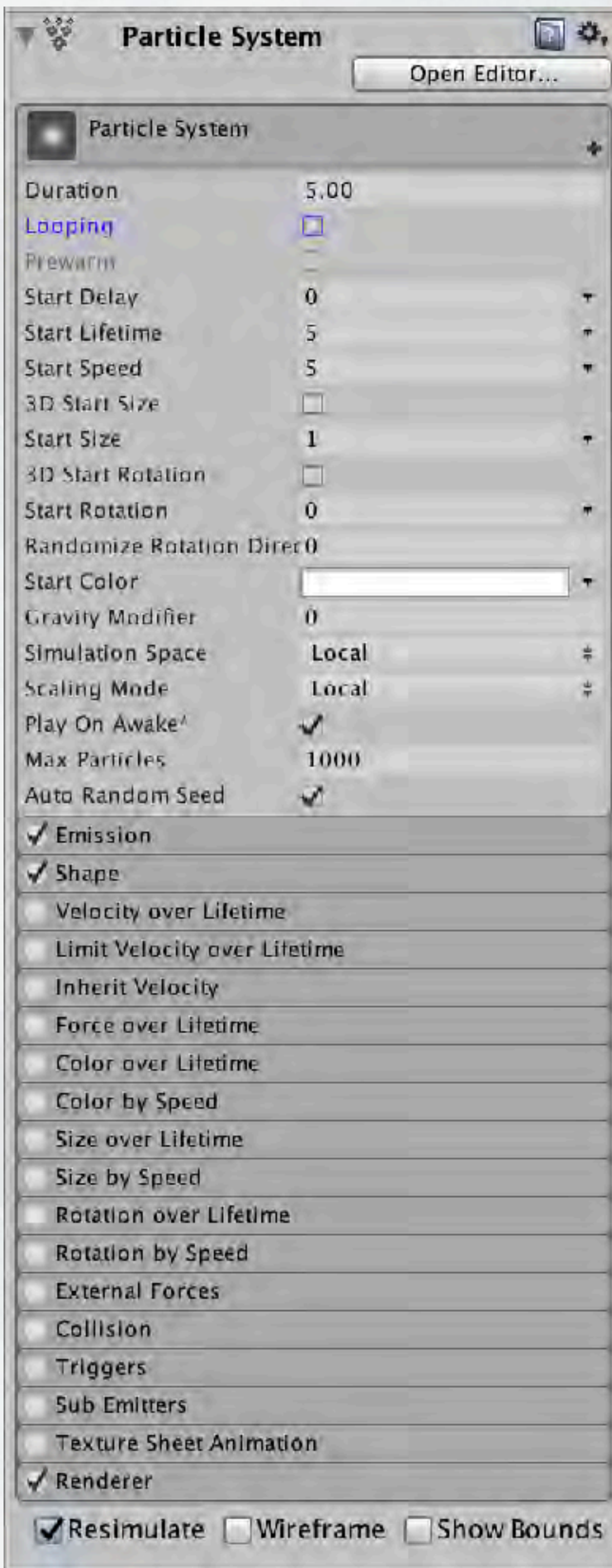
GameObject > Particle System

Particle effects generally consist of two elements: a **particle**, and a **particle emitter**.

A particle is typically a billboard sprite with a certain lifespan and behaviour over its lifespan. Once a particle has 'lived' for a predefined amount of time, it is destroyed.

The particle emitter determines the shape in which particles are 'thrown' into existence, often affected by external forces, such as gravity.

The particle system component has an extensive range of options – with several sub-components that can be activated as needed.



✓ Emission

Rate 10

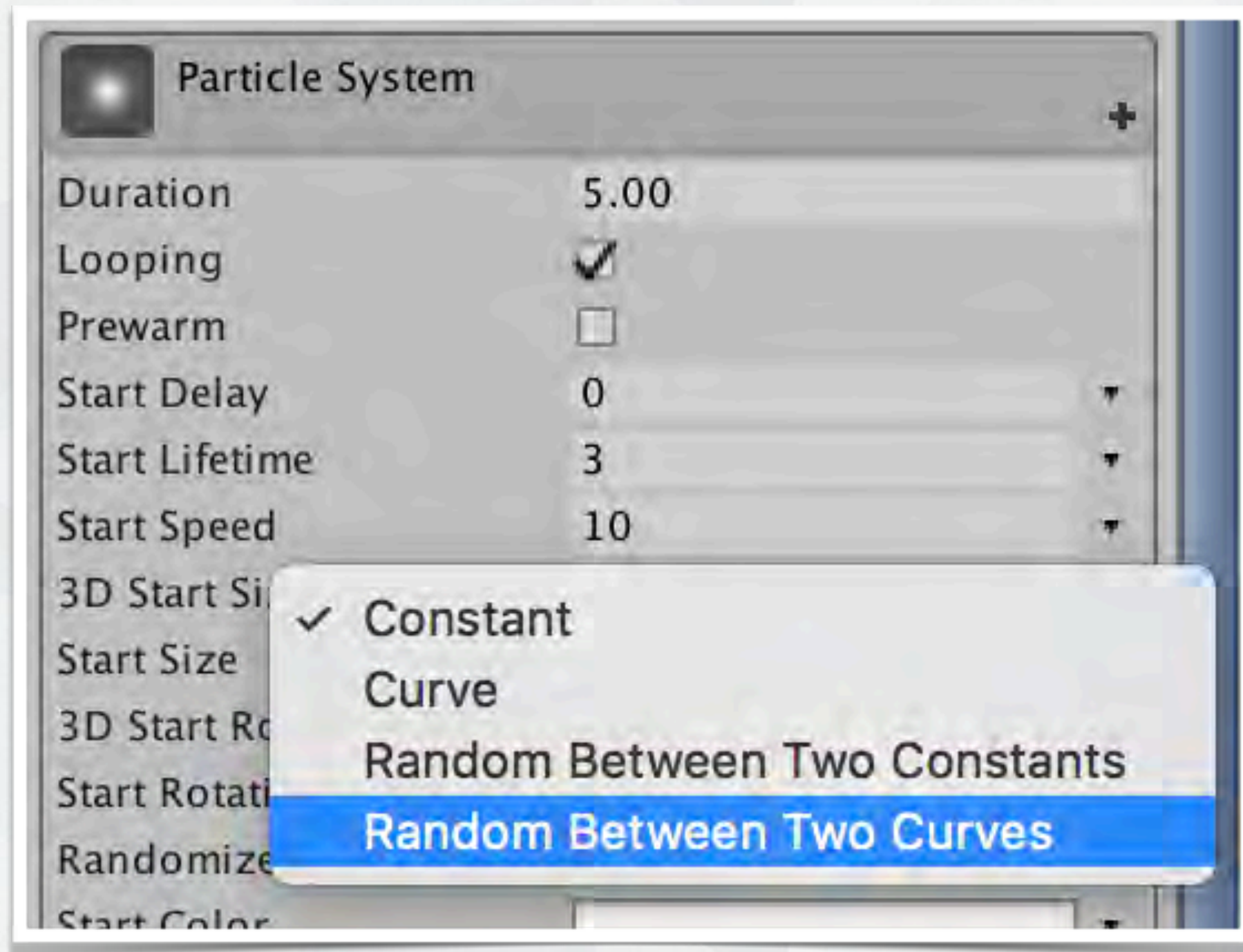
Time

Bursts	Time	Min	Max
	1.00	50	100

We can set how many particles are created, as well as initiate 'bursts' at specific points in time.



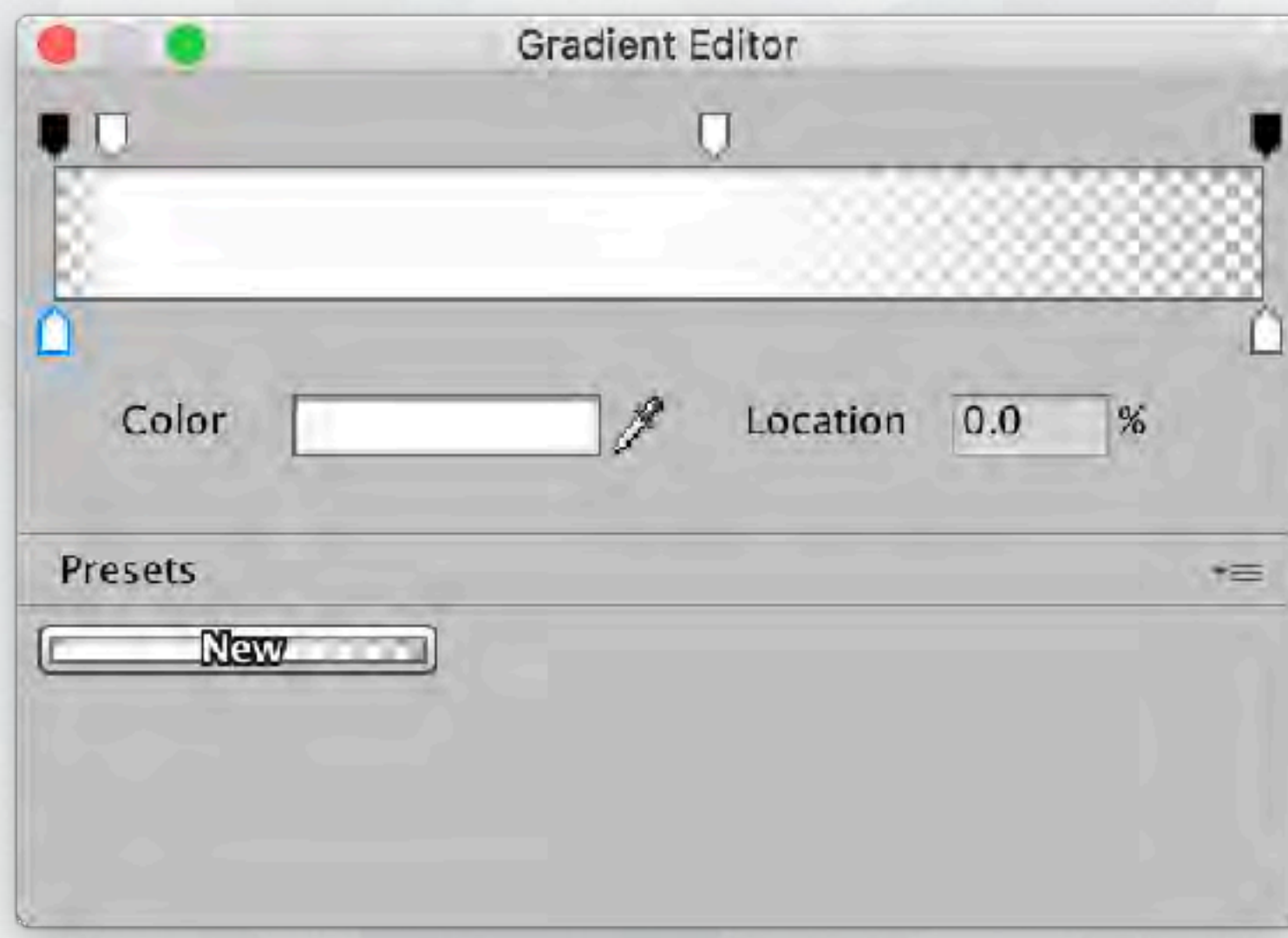
Video: Particle bursts



Many parameters can be set to random values (within a given range). This way we can define a less 'mechanic' behaviour.



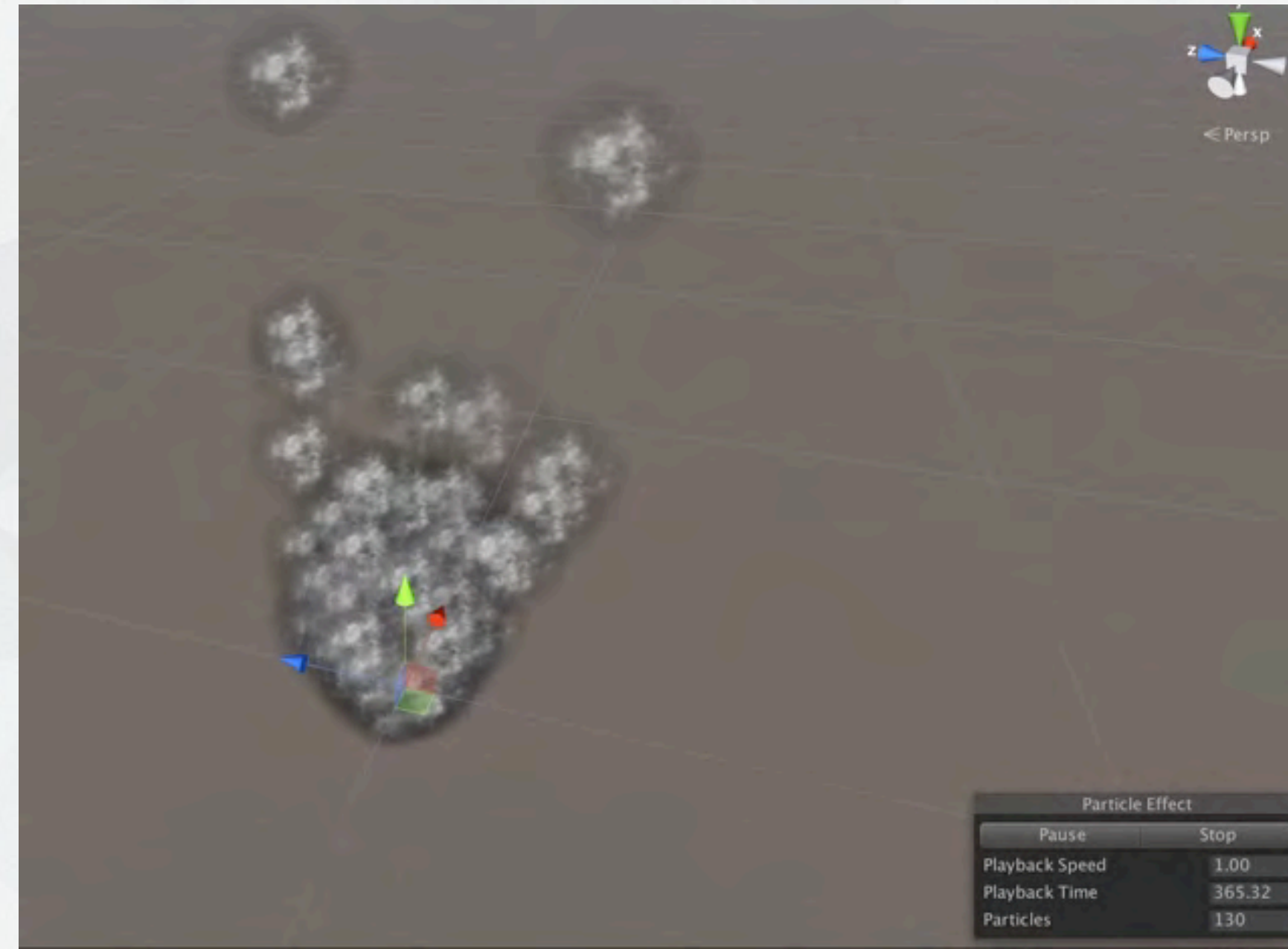
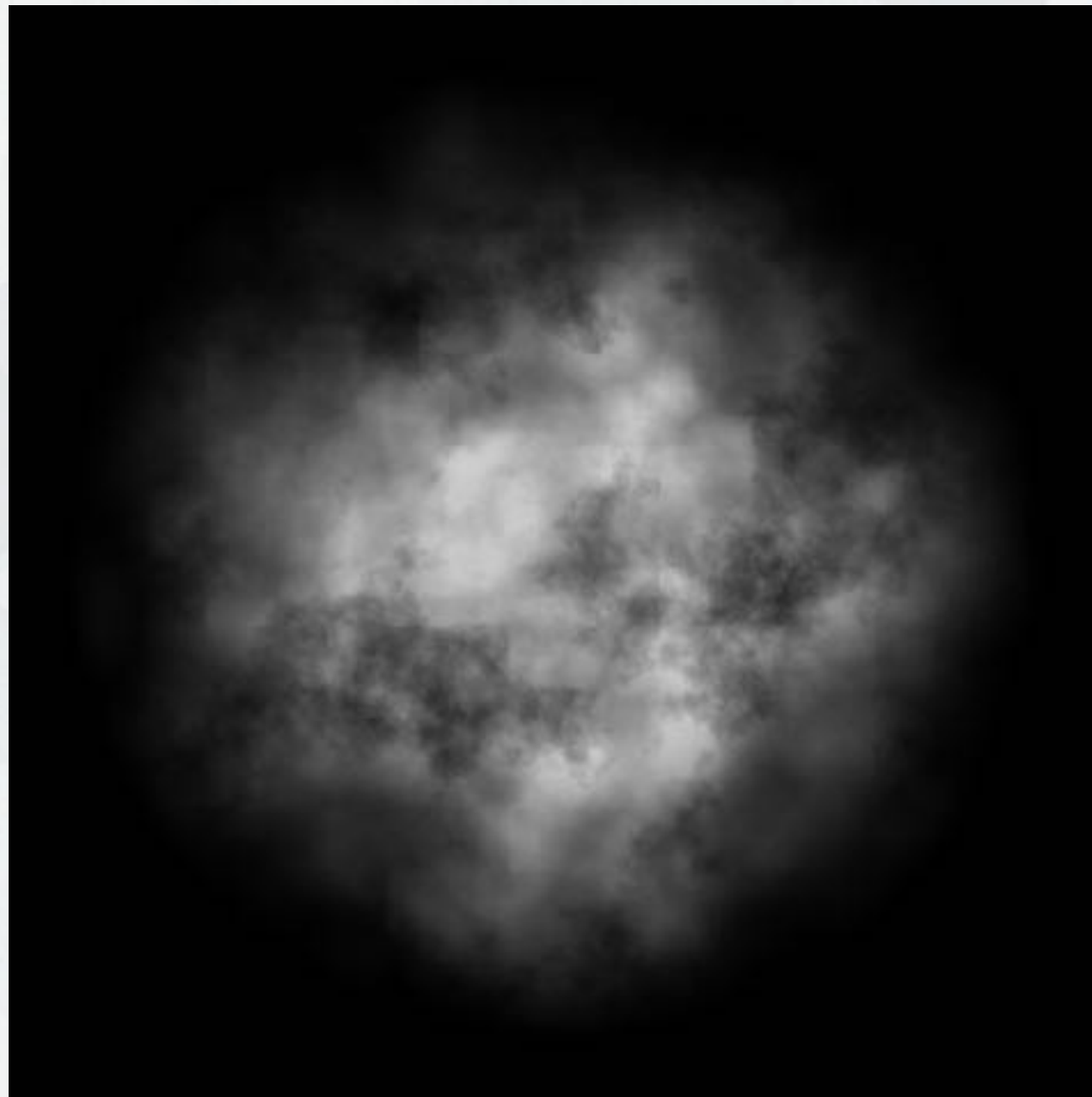
Video: Particle burst with random speed



By changing the color over the lifetime of a particle, particles no longer 'pop in' and 'out' of existence.

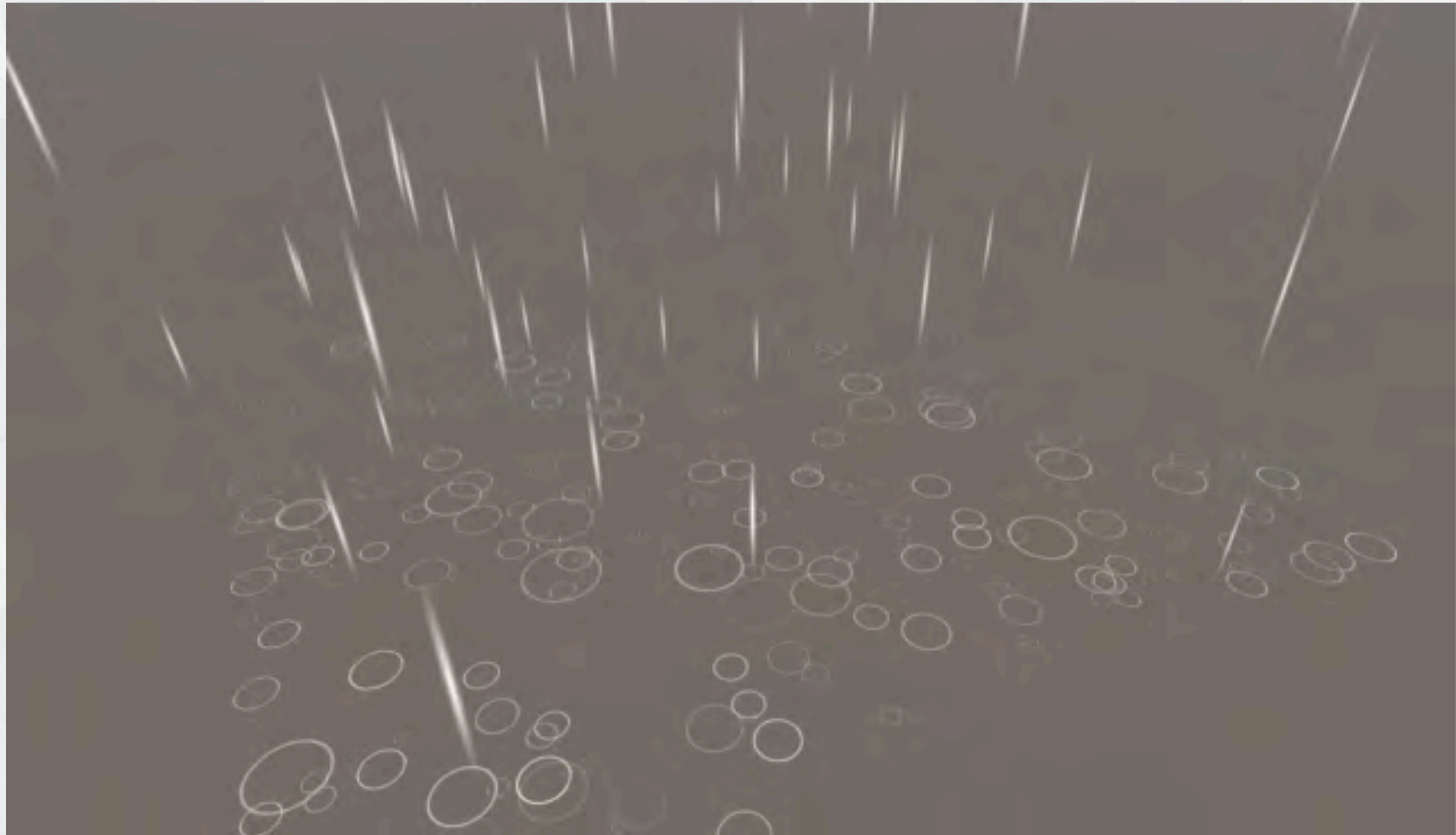


Video: Fading particles in and out

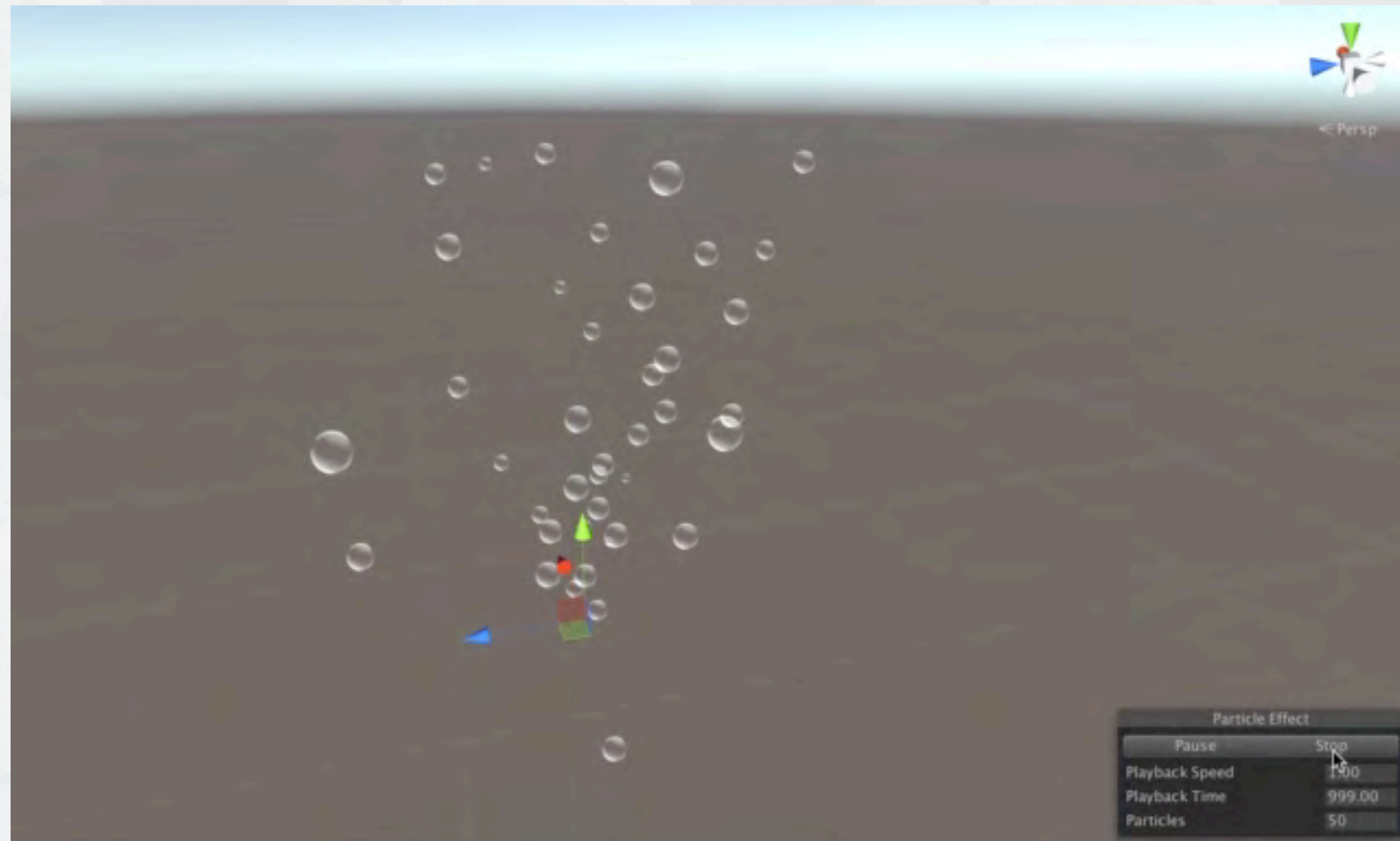
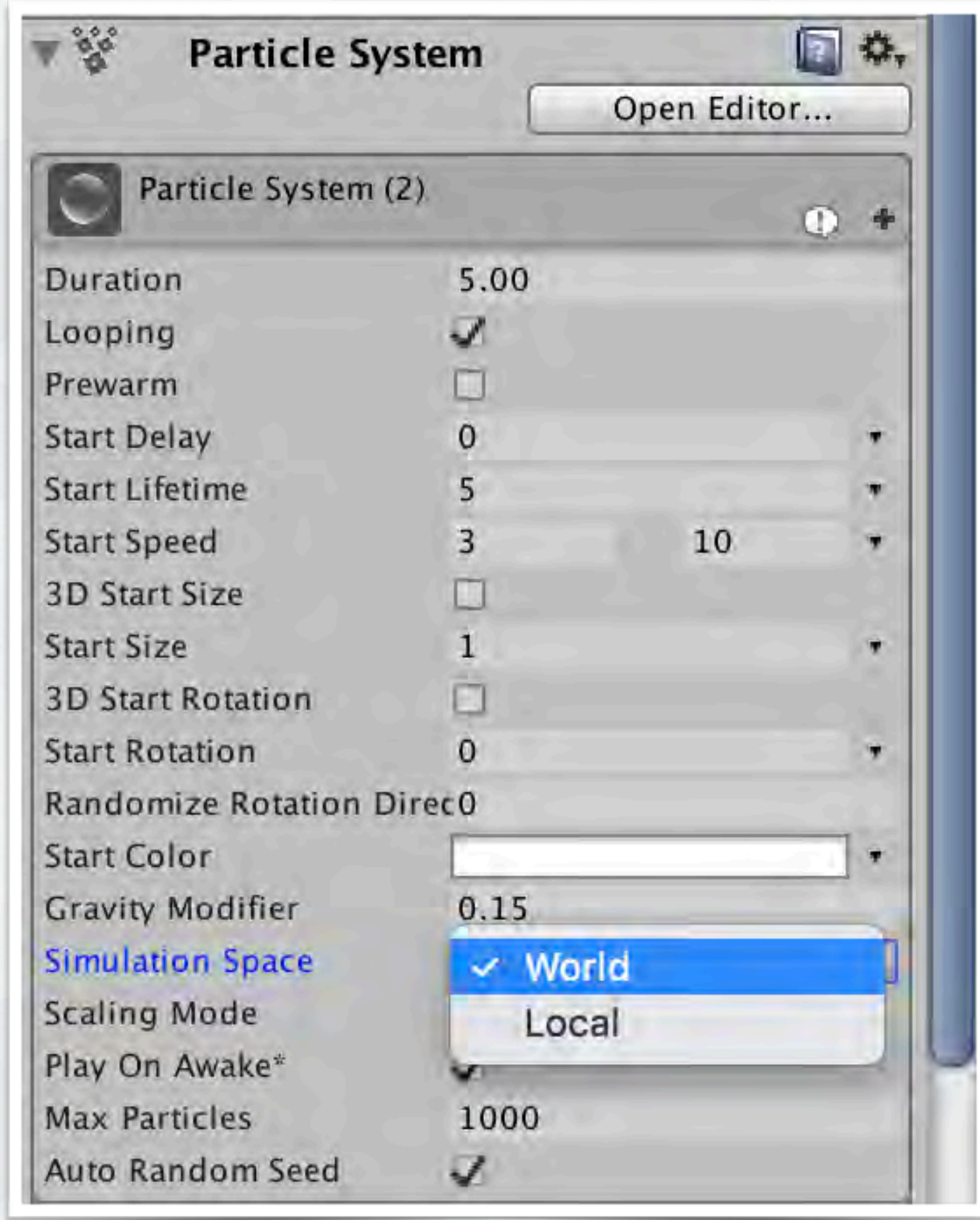


Video: Particle system before and after parameter tweak

A well-designed particle sprite can be used to create quite neat effects — especially by tweaking some parameters, such as rotation of particles or scaling of particles over time.



Video: Combination of two particle systems



Video: Moving particle emitter with world or local simulation space



Break Time!

But before, please give us some feedback:

<http://tiny.cc/degplus2>

After the break:

Unity Practice – Creating particle effects from scratch