



PATH PAINTER

By 3D Haven

Path Painter is a path painting system that enables the creation of gorgeous looking paths, roads, ramps and riverbeds on Unity terrains quickly and intuitively.

Version 1.2.0

Contents

About 3D Haven	3
Tutorials, Chat, Ticketed Support	4
Installation	5
Quick Start	5
What is Path Painter ?	6
Path Painter Systems and Components	7
Paint	7
Interface	7
Main Controls	8
Brush Settings - Shape	9
Textures	12
Vegetation	15
Info	17
Status Bar	18
States	19
Paint Mode	19
Edit Mode	24
Introductory Workflows	26
Opening Path Painter	26
The first path	27
Ramps, Straight Lines, Mixed Lines	35
Adding to Paths	39
Creating a mixed (straight and curved) path.	42
Connecting Elevated Paths, Riverbeds	43
Blending Embankment Textures	44
Path and Vegetation	45
MapScaler	51
Interface	51
Work Flow	51
More Tab	52
Paint API	52
Just Paint	52
Change All Settings and Paint	53

Change Any Settings and Paint.....	54
For example, the below simple script was used in the project that was used for build tests. The script uses a Pegasus path to drive the path creation:	54
Troubleshooting	55
The shape of the path is jagged, not smooth	55
The painted texture has sharp edges.....	55
Grass clearing is inaccurate and disproportionate.....	55
Can't Nicely Connect Raised Paths or Lowered Riverbeds/Paths	55
The Embankment is Not Being Painted Randomly	55
Performance Is Not Ideal When Painting Large Scale	55
Terrain Editing Reset When I Update My Path	56

About 3D Haven

Powerful, simple, beautiful. Friendly tools, gorgeous games!

3D Haven empowers artists and developers to bring their vision to life by making it very easy and quick to create. Leverage our high-quality products to take the pain out of creating amazing things and focus on creating amazing games.

Learn more at our website: <http://www.3d-haven.com/>

Tutorials, Chat, Ticketed Support

Thanks for purchasing Path Painter!

Sometime new tools can be a little overwhelming. To help you with this we have created an awesome support network for you. You can also get access to these links from the Path Painter menu in Unity.

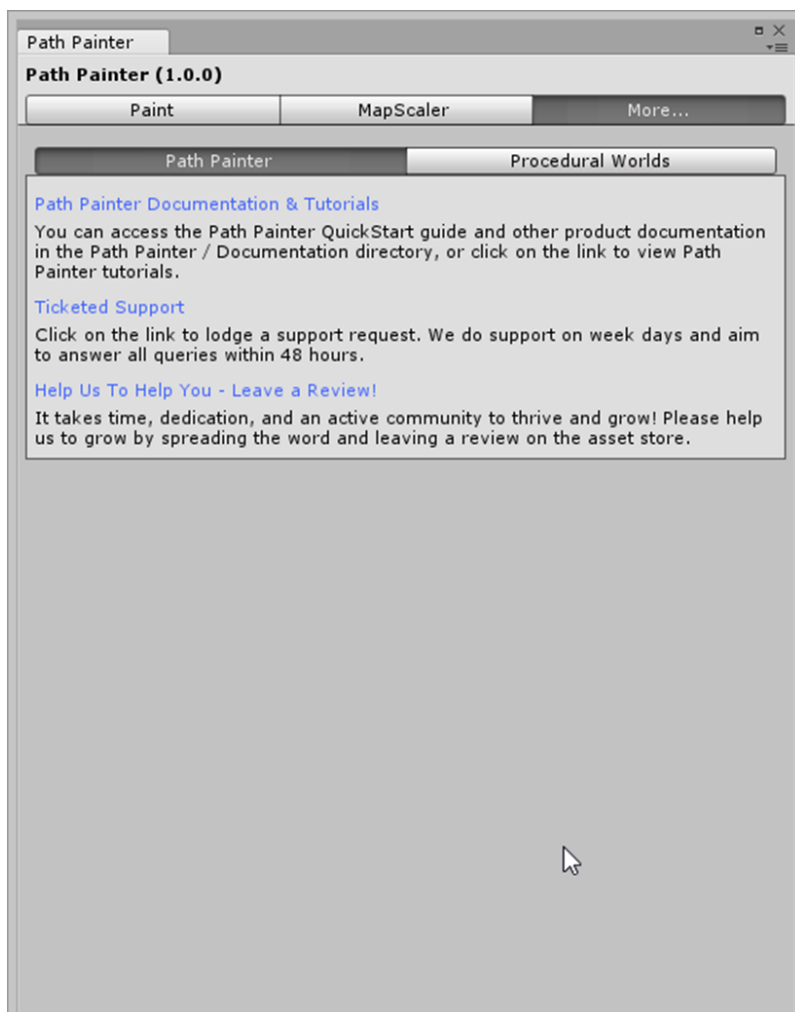
Tutorials:

<https://www.youtube.com/playlist?list=PLVm4v63CHTIHUX4jG0MsufamDfLfofROI>

Have A Chat: <https://discord.gg/btrhmvS>

Lodge a Support Request: <https://3dhaven.freshdesk.com/support/home>

Quick access can be found at More... -> Path Painter in the Path Painter Window.



Installation

Installing Path Painter will create the following folder structure:

Path Painter:

Editor:

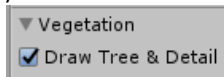
Demo: Path Painter Demo environment

Documentation: Path Painter documentation

Resources: Path Painter resources

Important Notes

- Path Painter exist only in the Editor scope. This means that none of the above will be included in your build and you don't need to remove any part of the software to optimise your builds.
- Path Painter expects all terrain tiles to be square. If this condition is not met, Path Painter will deactivate and display a message to inform the user of this behaviour.
- Path Painter operates in standard unity scenes and is subject to the usual performance constraints. Grass and trees can have huge impacts on performance. You can get around this by disabling "Draw Tree & Detail" after you find the desired grass and tree clearing for a set of paths.



Removing Path Painter

Path Painter will not be included in your builds. It has no impact in the Editor either unless it's open and active, and you don't need to remove it. To remove Path Painter you can simply delete the Path Painter folder from your Assets.

Quick Start

See the QuickStart guide for a quick overview on how to get started with Path Painter.

What is Path Painter ?

Path Painter is a system that enables rapid and precise creation of gorgeous looking paths, roads, ramps, riverbeds, and possibly mountains on Unity terrains.

With Path Painter you will:

- Paint Paths on your terrains.
- Specify the shape of the path.
- Specify the texturing of the path.
- Specify how the path affects the surrounding vegetation.
- The MapScaler utility is also included to allow scaling of the different maps of the terrain in case they don't fit your path creation needs.

Path Painter is different because:

- **It's standard** – There are no special shaders or other tricks that could cause compatibility problems (you can safely delete Path Painter after you have created your scene);
- **It's simple** – Path Painter does not have a steep learning curve and can be used out of the box by both highly experienced level designers and Unity terrain new comers;
- **It's fast** – Path Painter allows you to create all your paths for a map within minutes, with only minimal fine tuning at the end. It's paint based, so any touch up needed can be achieved by painting over areas either with Path Painter itself, or one of the built-in Unity terrain tools depending on the situation;
- **It's powerful** – Live feedback gives a lot of power to the users when they are setting up for sets of paths, or finalising a particular one;

Path Painter Systems and Components

You can control Path Painter from the Path Painter Window. Open the Path Painter Window by selecting Window -> Path Painter -> Path Painter. The Path Painter Window is made up of the following tabs:

- **Path Painter Window:**
 - [Paint](#) – the main interface;
 - [MapScaler](#) – a handy utility included in Path Painter that allows you to scale the terrain maps to higher or lower resolutions;
 - [More...](#) – Quick access to more information;

Paint

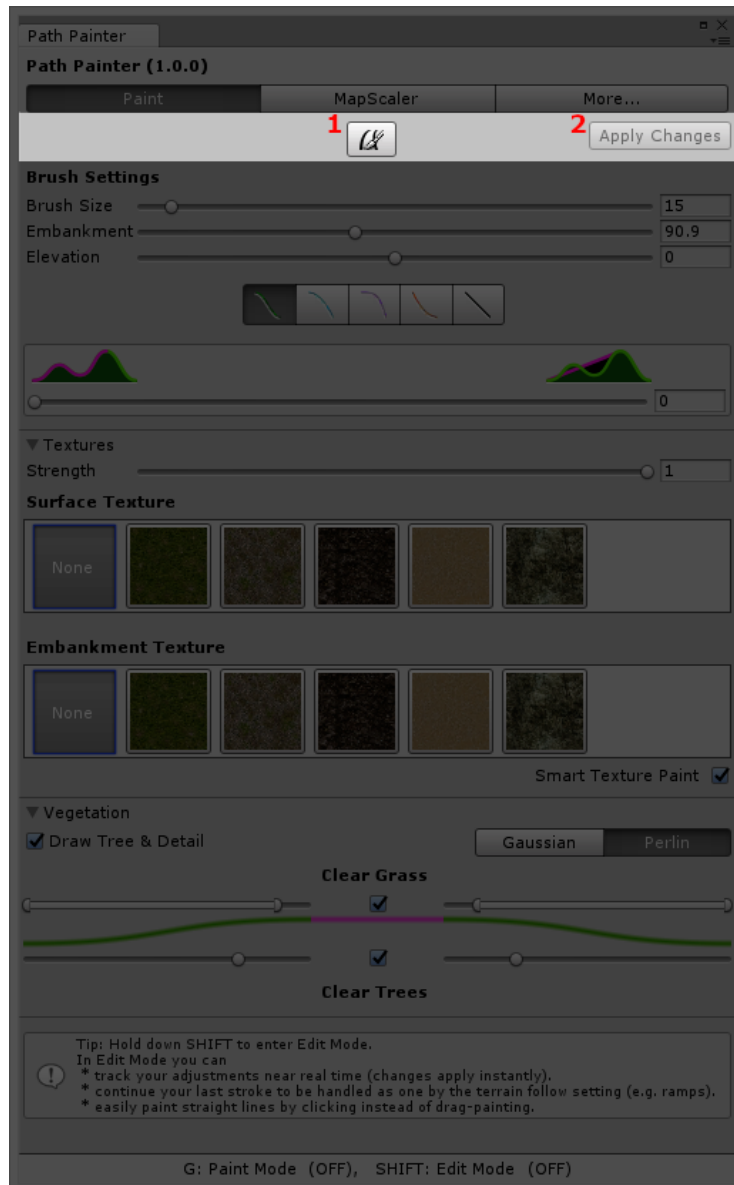
Interface

The Paint tab of the Path Painter Window is the main interface. It can be divided into 6 sections, each with their own scope. These are

- [Main Controls](#)
- [Brush Settings](#)
- [Textures](#)
- [Vegetation](#)
- [Info](#)
- [Status Bar](#)

Main Controls

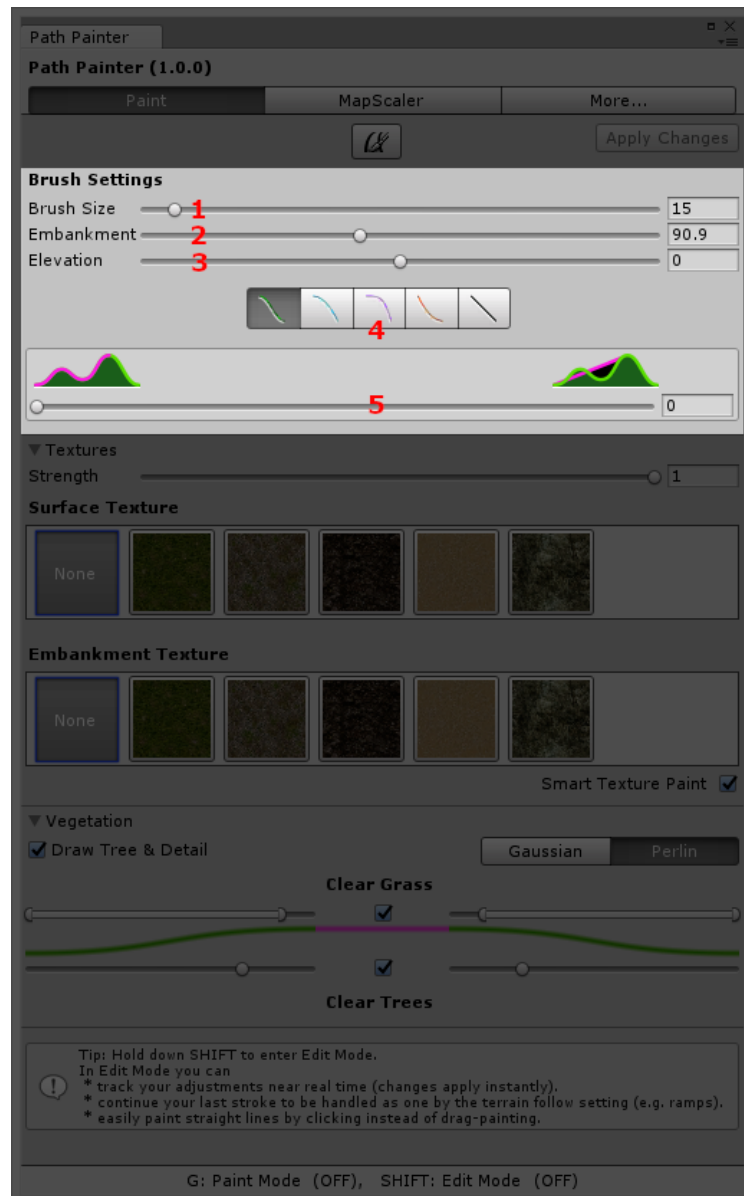
In the Main Controls section there are 2 buttons:



1. **Paint** – you can enable and disable painting with the **Paint button** (or by pressing G);
2. **Apply Changes** – is enabled when there are changes that you can apply to your active brush stroke. It is handy when you want to change a number of settings and apply them all at once to the active brush stroke;

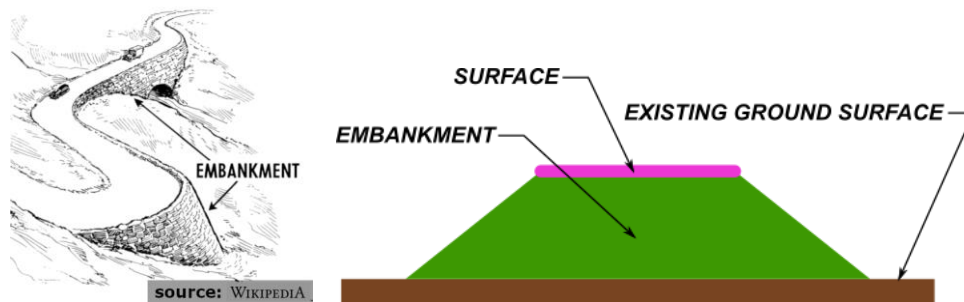
Brush Settings - Shape

Brush Settings allow the fine tuning of the shape of paths from width to terrain following.



1. **Brush Size** – is the size of the actual path (the surface) in meters (units);
2. **Embankmen (Size)** – is the overall width of the path in meters (units). The embankment is the area which can be used to blend the path into the surrounding environment. The minimum **Embankment** size is the **Brush Size** and the **Embankment** size automatically updates with the **Brush Size**. This

keeps the proportions of the brush;



3. **Elevation** – is the elevation of the path. This can be negative which is a typical setting for riverbeds. Min/max elevation is affected by the embankment size to help keep things in proportion.
4. **Embankment Curves** – have an effect on the embankment shape and texture. Changing this can have a dramatic effect on how paths blend with the environment. You can monitor the visualisation in the [Vegetation](#) section to see how the embankment changes;
5. **Terrain follow** – With this setting you can make paths
 - follow the terrain.
 - have an even slope from start to end, and
 - anywhere in between.

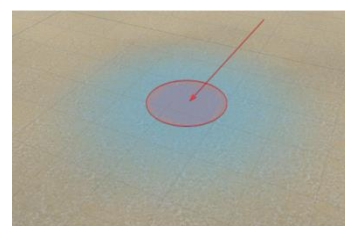
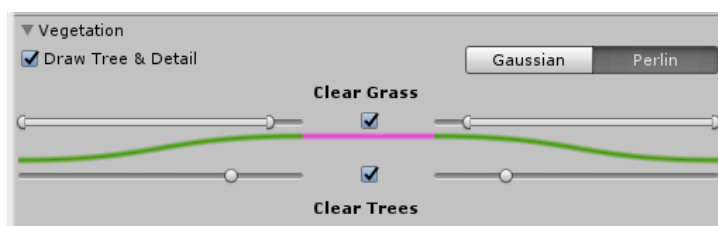
It comes very handy when making ramps or when the terrain is too rough for paths. With this you can easily create beautiful paths and fine tune how natural or artificial looking you want them to be;

The **Brush Size** and **Embankment Size** are limited by the resolutions of the terrain maps. Fine details cannot be painted on a low resolution canvas. The maximum size is also limited by the terrain's

- *Heightmap Resolution*, and
- *Control Texture Resolution* (if Texture painting is enabled).

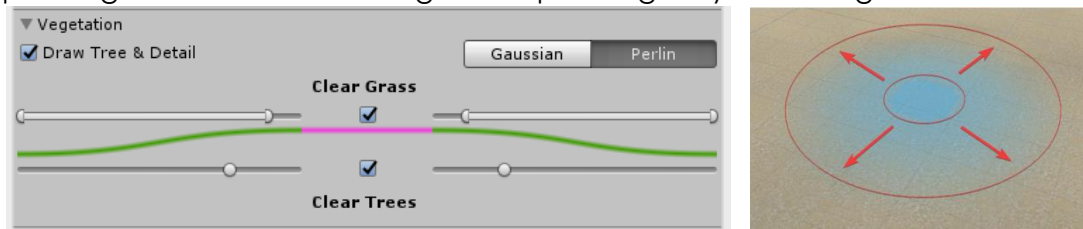
The **Brush/Surface** is

- shown purple in the visualisation found in the [Vegetation](#) section of the GUI.
- the full strength center core of the brush is visualised on the terrain below your mouse when painting is enabled.



The **Embankment** is

- shown green in the visualisation found in the [Vegetation](#) section of the GUI.
- the outer ring of the brush as visualised on the terrain below your mouse when painting is enabled. It is fading out depending on your settings.



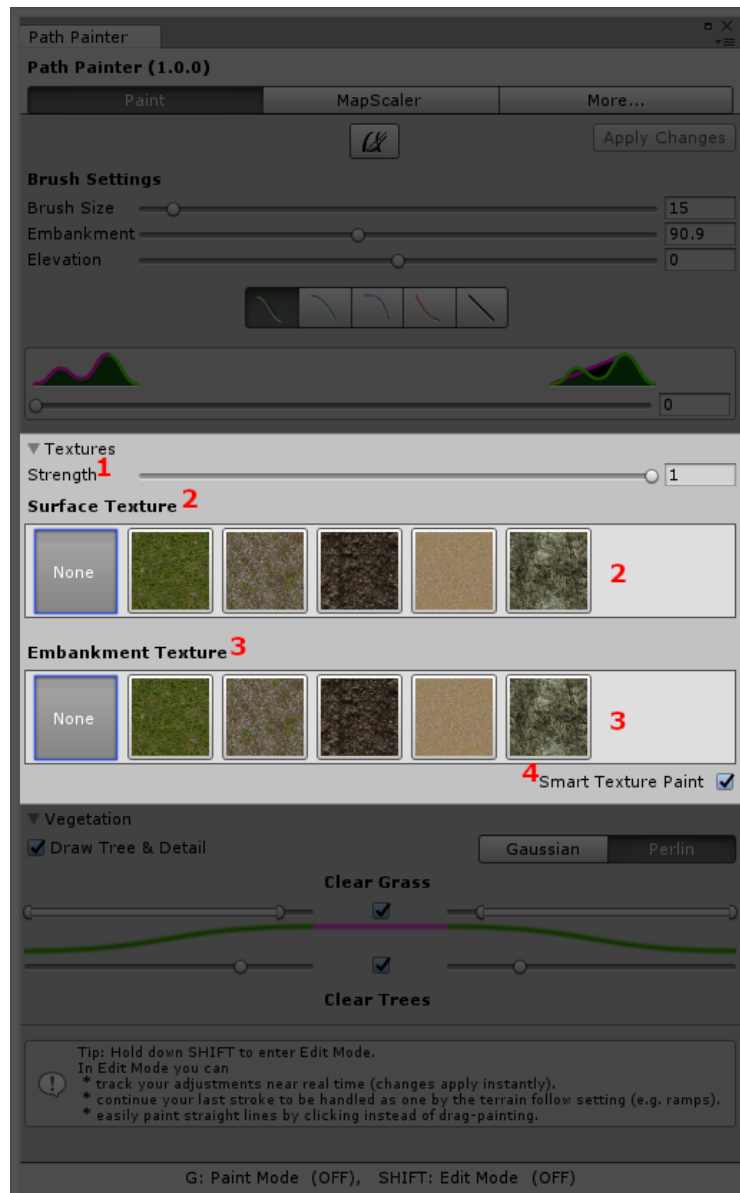
If you find yourself constrained by performance while painting something big, you can try to paint with

- Texture painting disabled (set **Surface Texture** to **None** in the [Textures](#) section)
- **Draw Tree & Detail** off (checkbox in the [Vegetation](#) section)
- smaller brush size ([Brush Settings](#))

then apply your desired settings in one go to the path.

Textures

Path Painter can apply textures to your paths. In the **Textures** section you can select which of the terrain's textures are used on the path and a number of related things.



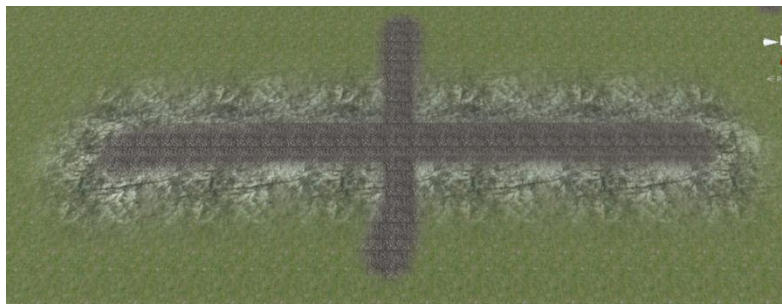
1. **Texture Strength** – is the opacity, or target strength of the path texture, depending on the scenario and the settings. It can be used for CTS height blending or if you wish to paint with a certain opacity;
2. **Surface Texture** – is the texture to be applied to the surface of the path (see **Surface** in the previous section).
3. **Embankment Texture** – is the texture to be applied to the embankment of the path (see **Embankment** in the previous section).

4. **Smart Texture Paint** – will attempt to guess how you would like to apply textures to each path. When disabled Path Painter operates in texture over paint mode that's similar to ordinary painting in that the path will cover existing textures where possible.

Most of the time you will want **Smart Texture Paint** on. In some rare circumstances it might misjudge the situation and not paint the embankment texture. That is when you want the paint to cover the existing textures. Switching **Smart Texture Paint** off will achieve that.

For example:

In this situation **Smart Texture Paint** assumes that you want to create a crossing (understandably):



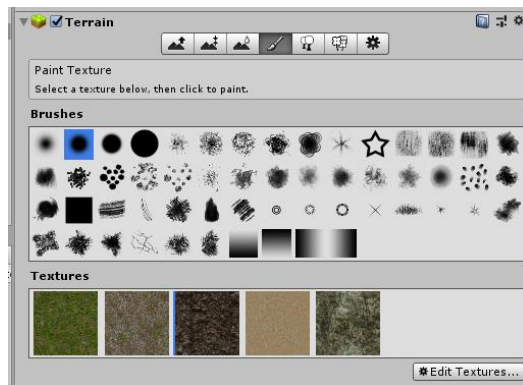
However this might not always be the case:



Turn off **Smart Texture Paint** to achieve the desired effect:

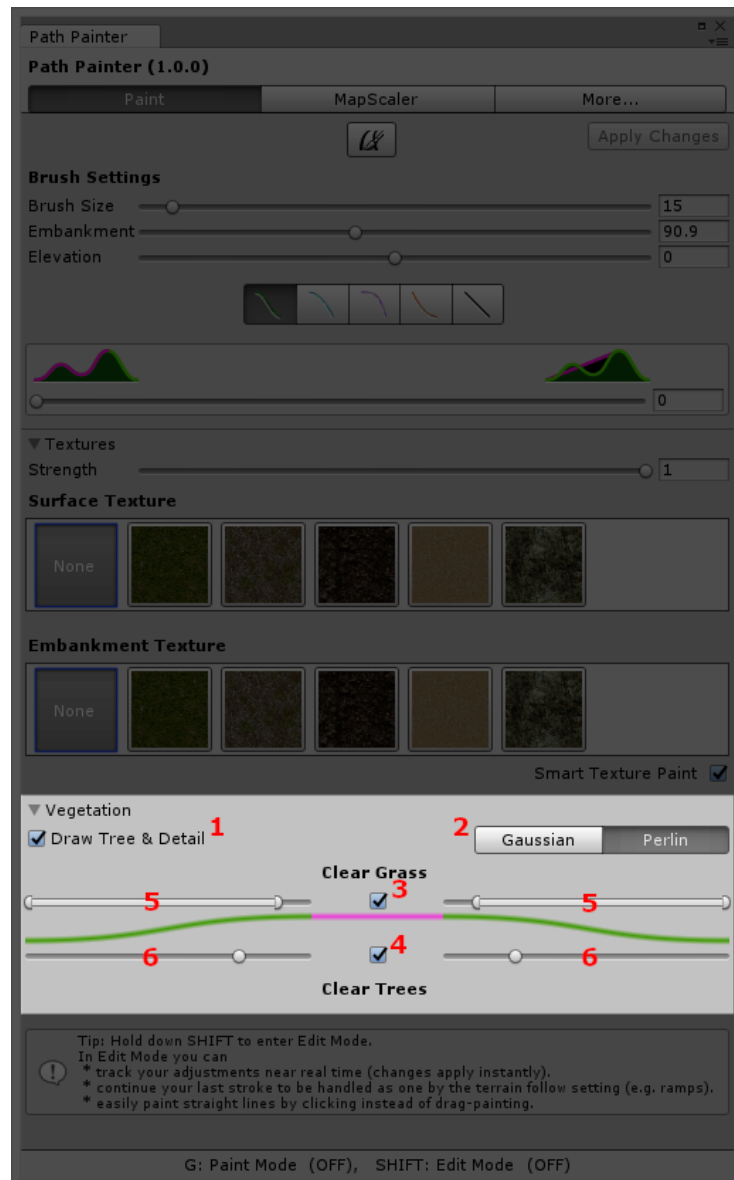


Note: If you need to edit textures of the terrain (or their settings), you can do that under the *Paint Texture* tool of the *Terrain* component as usual.

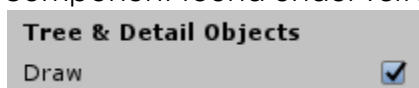


Vegetation

Path Painter makes it easy to clear vegetation in your path and achieve a natural effect.



1. **Draw Tree & Detail** – is a handy shortcut for the same switch of the *Terrain* component found under *Terrain Settings*:



While creating paths people often find themselves in need of switching this on/off;

2. **Grass clearing noise** – the chosen type of noise will be applied to grass clearing for a more natural look.
3. **Grass clearing toggle** – Enable/disable grass clearing.

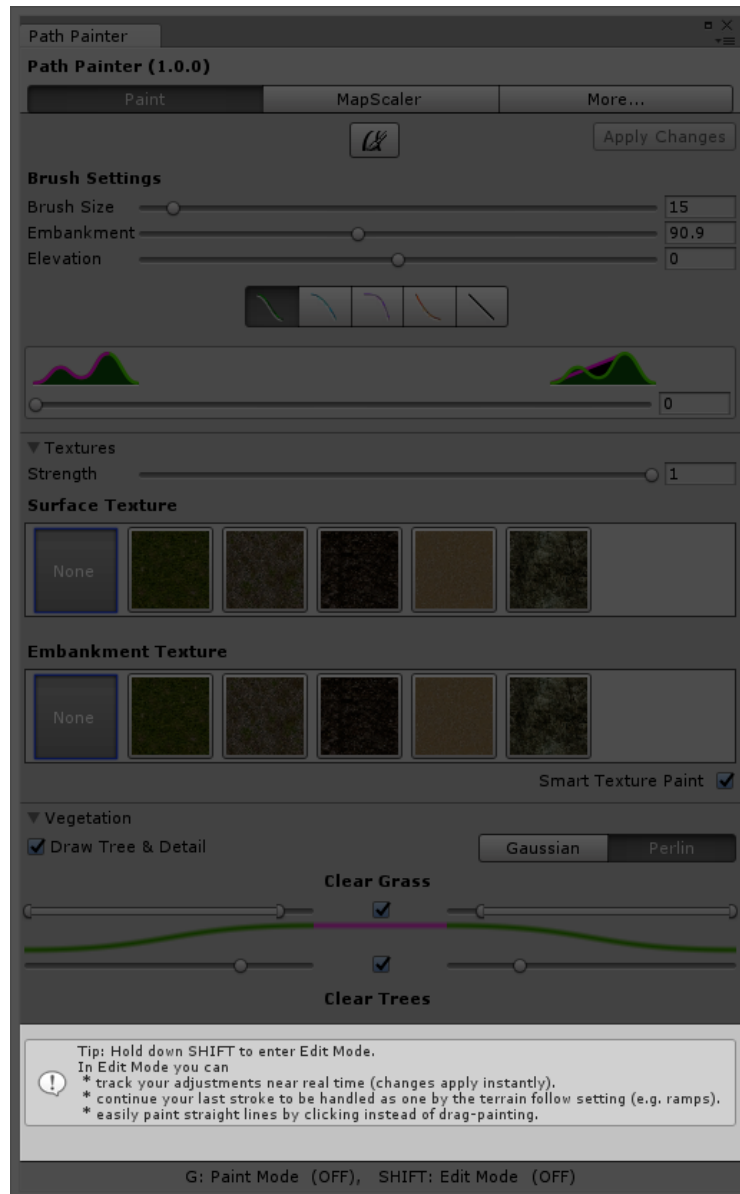
4. **Tree clearing toggle** – Enable/disable tree clearing.
5. **Grass clearing limit and thinning range** – See below the sliders the visualisaion of the cross-section of the path and set them accordingly. Full grass clearing will be applied from the center of the path to the inner knob of the range sliders. A gradual thinning with the selected noise is applied between the two knobs (in the range).
6. **Tree clearing limit** – See above the sliders the visualisaion of the cross-section of the path and set them accordingly. Full tree clearing will be applied from the center of the path to the knob of sliders.

Path Painter operates in standard unity scenes and is subject to the usual performance constraints. Grass and trees can have huge impacts on performance. You can get around this by disabling **Draw Tree & Detail**. For example if you are experiencing delays at the end of each painting action and when doing Undo, you can do the following:

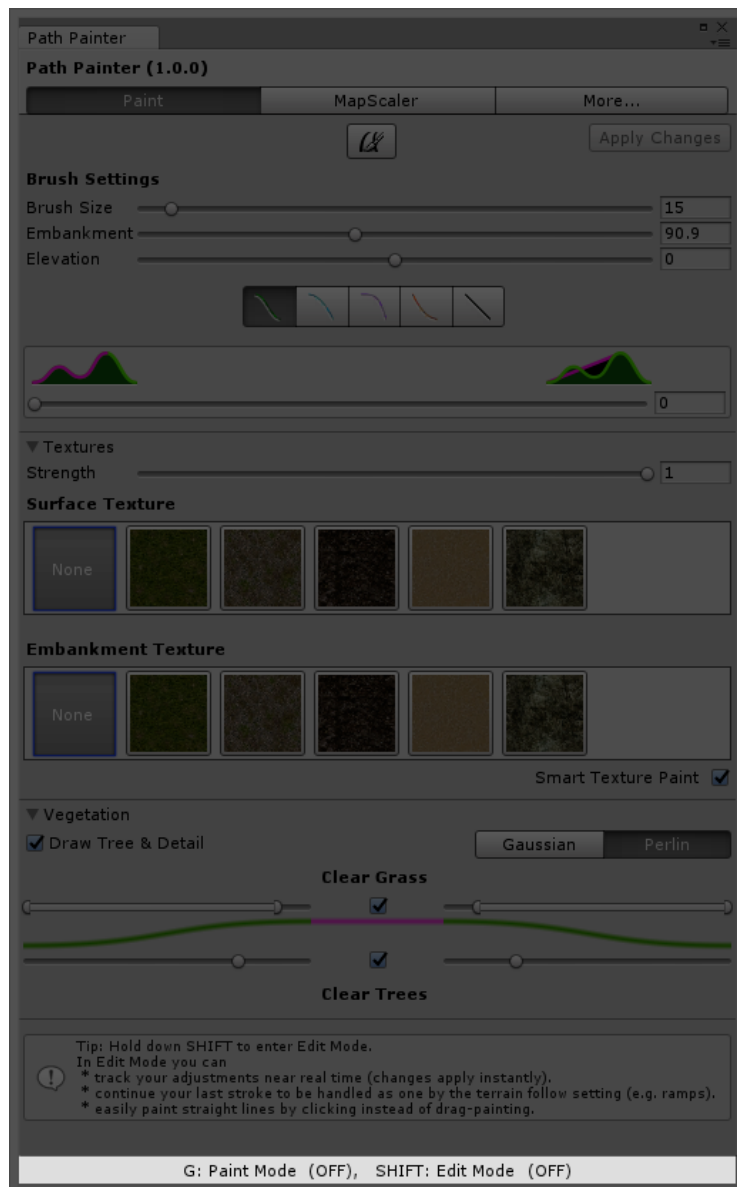
1. When you starting on a set of paths that will have the same vegetation settings, paint one path.
2. Setup everything to your liking.
3. Turn off **Draw Tree & Detail**.
4. Paint all the paths you need with these settings.
5. At the end you will only experience the delay one when you turn **Draw Tree & Detail** back on.

Info

This area will display information and tips.



Status Bar

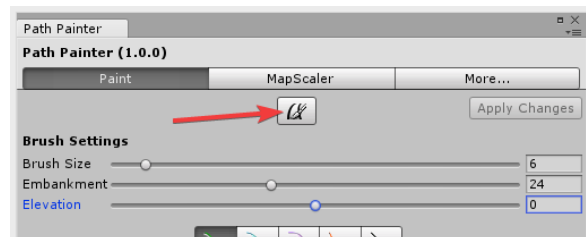


The **Status Bar** displays hotkeys and status information. Here it's visible if *Paint Mode* or *Edit Mode* is active and what functions are available. It is most beneficial while getting familiar with Path Painter.

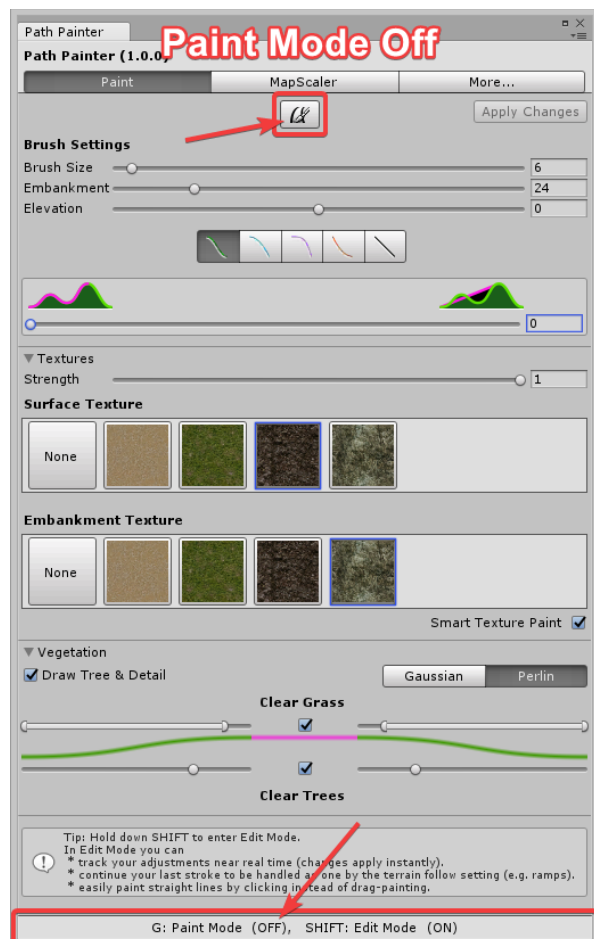
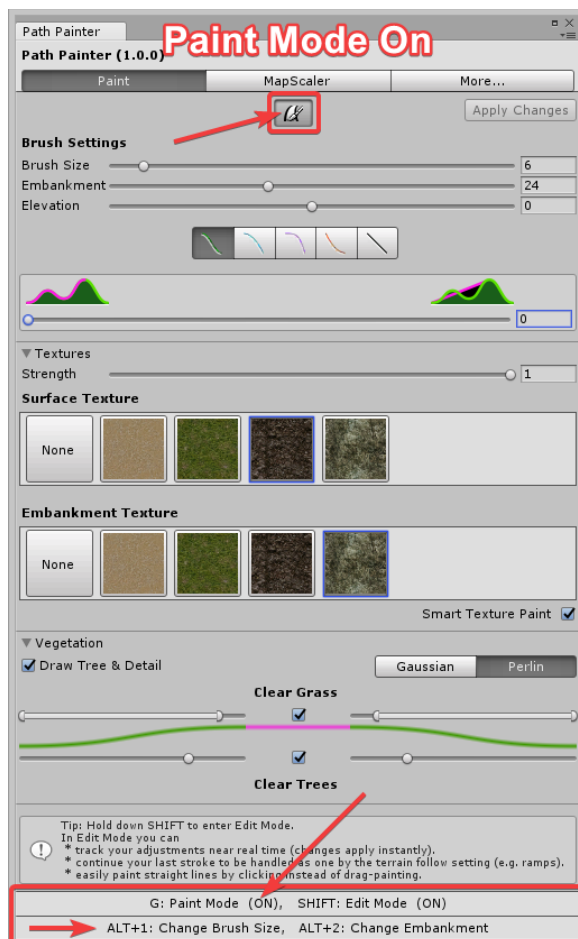
States

Paint Mode

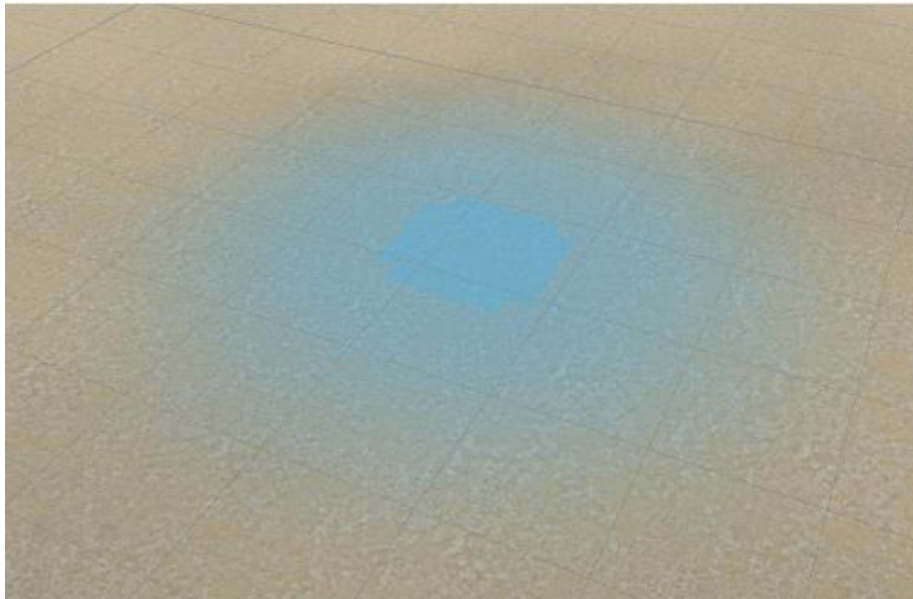
Painting is enabled in **Paint Mode**. You can enter paint mode by clicking the **Paint Button** or by pressing G.



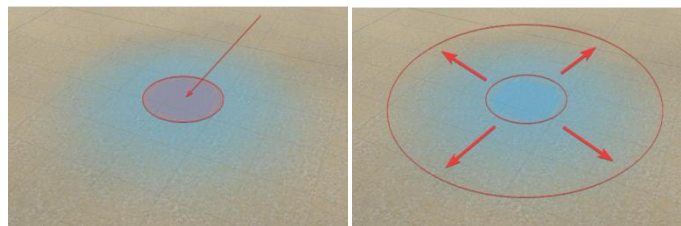
Both the button and the [Status Bar](#) shows when **Paint Mode** is active



and a brush visualisation will be visible on the terrain.

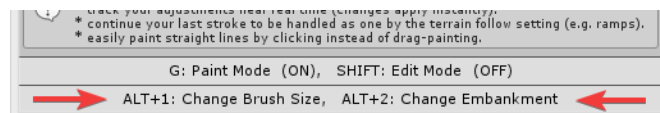


The visualisation displays both the surface and the embankment according to the settings.



The **Brush Size**, **Embankment Size**, and **Embankment Curve** are all visible on the visualisation.

In **Paint Mode** you can use hotkeys to [Change the Brush Size](#) and [Change the Embankment Size](#) in the Scene View. The [Status Bar](#) also displays this

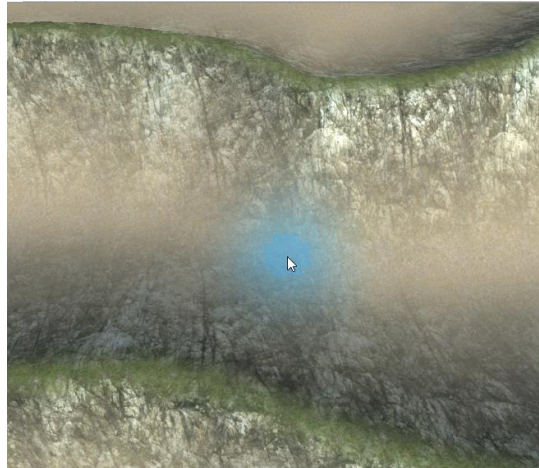


Changing Brush Size - in Scene View

Changing **Brush Size** in the Scene View can often come handy. Note: the same limits apply as in [Brush Settings](#) of the GUI.

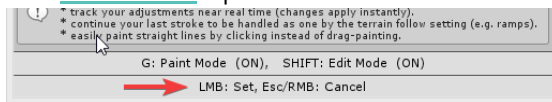
Press the appropriate hotkey (visible in the [Status Bar](#)) while

- the **Paint Mode** is active
- the Scene View is focused and
- the brush is visible on the terrain

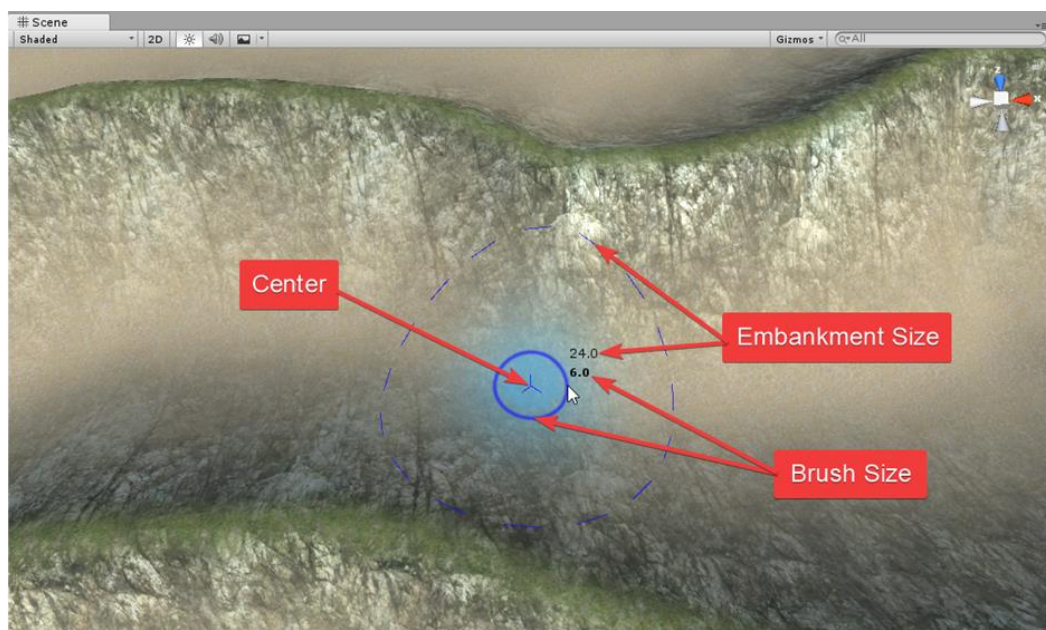


When you hit the hot key the following will happen:

- The [Status Bar](#) updates.



- Outlines of the brush appear.
- The mouse cursor snaps into place for the change.
- The current values will be displayed next to the mouse (**Brush Size** in bold).



As you move the mouse, **Brush Size** will update. You can now

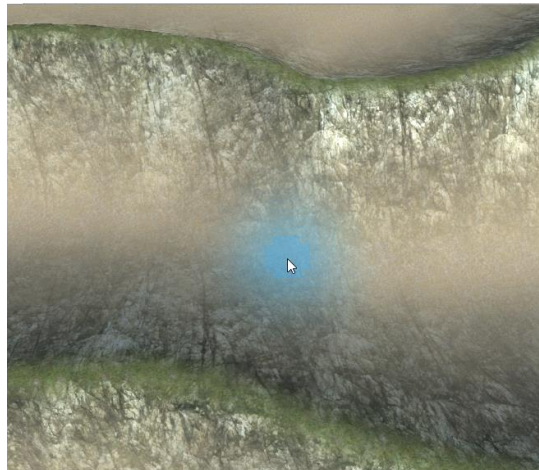
- set the **Brush Size** with the Left Mouse button once you are happy with it, or
- cancel with the Right Mouse button or by pressing Esc.

Changing Embankment Size - in Scene View

Changing **Embankment Size** in the Scene View can often come handy. Note: the same limits apply as in [Brush Settings](#) of the GUI.

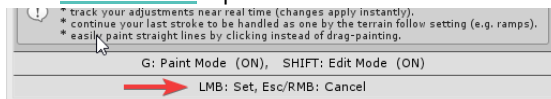
Press the appropriate hotkey (visible in the [Status Bar](#)) while

- the **Paint Mode** is active
- the Scene View is focused and
- the brush is visible on the terrain

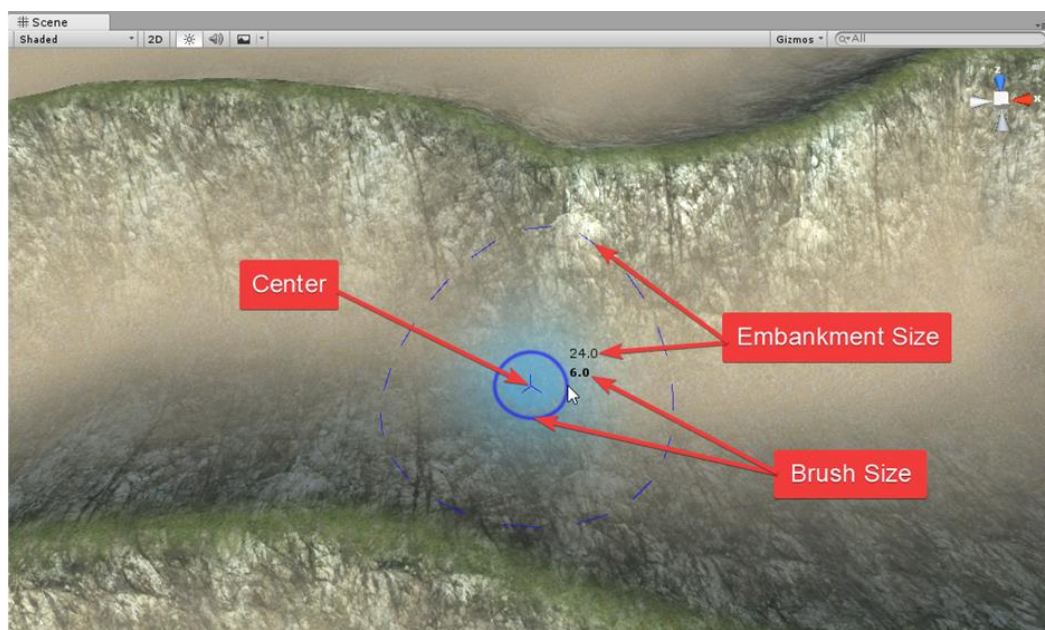


When you hit the hot key the following will happen:

- The [Status Bar](#) updates.



- Outlines of the brush appear.
- The mouse cursor snaps into place for the change.
- The current values will be displayed next to the mouse (**Brush Size** in bold).



As you move the mouse, **Embankment Size** will update. You can now

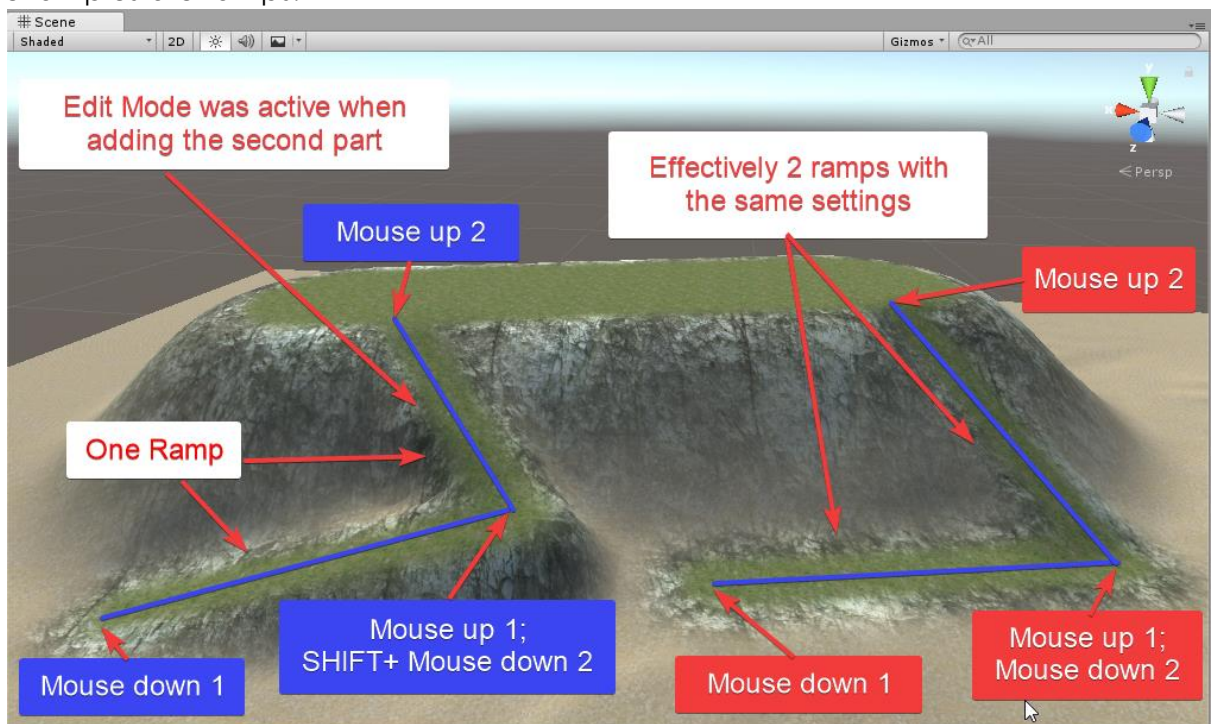
- set the **Embankment Size** with the Left Mouse button once you are happy with it, or
- cancel with the Right Mouse button or by pressing Esc.

Edit Mode

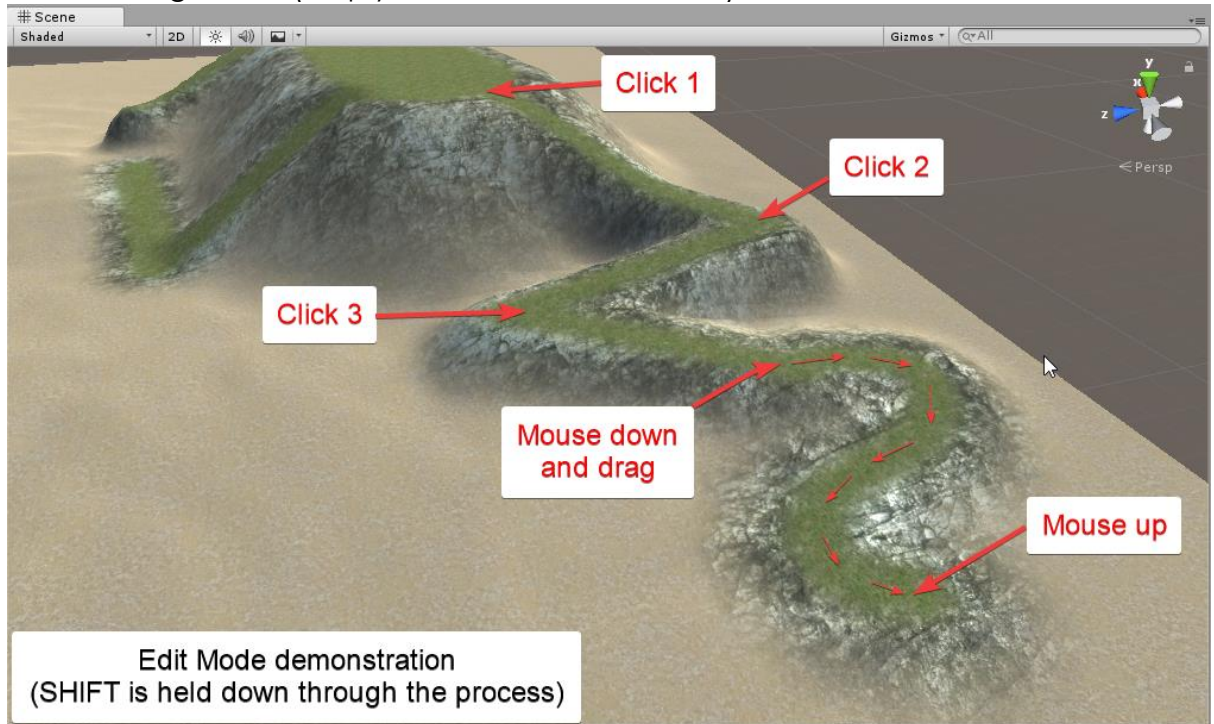
Edit Mode is a powerful feature of Path Painter. It modifies some of the functionality, like the *SHIFT* key on your keyboard. It's easy to remember this, because by default **Edit Mode** is mapped to the *SHIFT* key. It comes very handy when tweaking and when working on more complex, long paths, that need a lot of viewport navigation during creation.

Edit Mode can be used to

- Edit the active stroke and get live visual feedback.
- Add to the active stroke. This can be important when strokes are needed to be handled as one by the **Terrain Follow** setting (see [Brush Settings](#)). Good examples are ramps.



- Create straight lines (simply click while in **Edit Mode**)



Edit Mode applies if there is an active stroke and it does the following

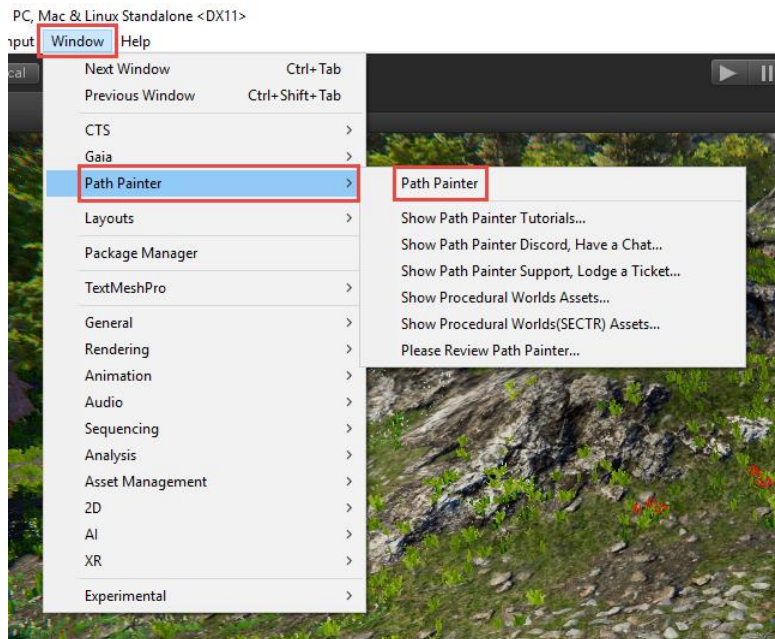
- Instead of starting a new line, the active stroke is going to be continued.
- Adjustments are instantly applied to the active stroke without the need to push the **Apply Changes** button.

Introductory Workflows

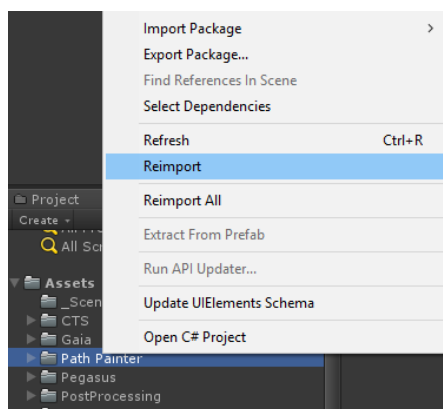
Opening Path Painter

When you first install Path Painter, please wait a little time for the import process to complete.

To open Path Painter select Window -> Path Painter -> Path Painter



If the menu is not there, it is possible that Path Painter has not yet been completely imported into your project. Please wait for a few moments. If it is still not there then try re-importing Path Painter.

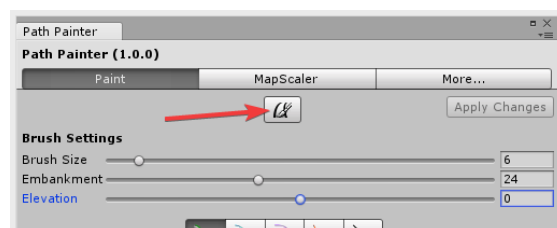


Note: These examples don't cover a lot of traditional path creation techniques. Their aim is to help the user understand how Path Painter works. After completing these, the user will be familiar enough with Path Painter to create any kind of paths.

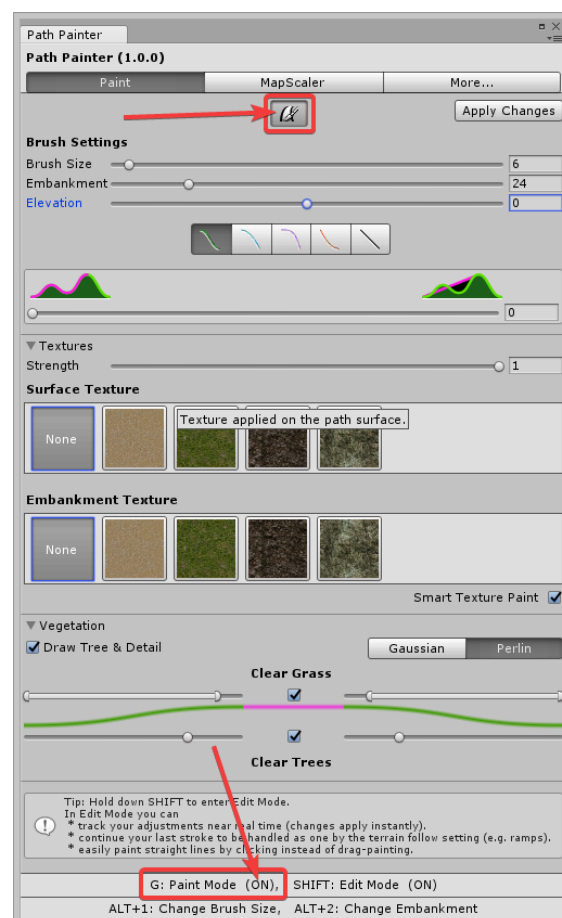
The first path

In this example we are going to be working on the *Demo Terrain* found in the *Demo* folder of the package.

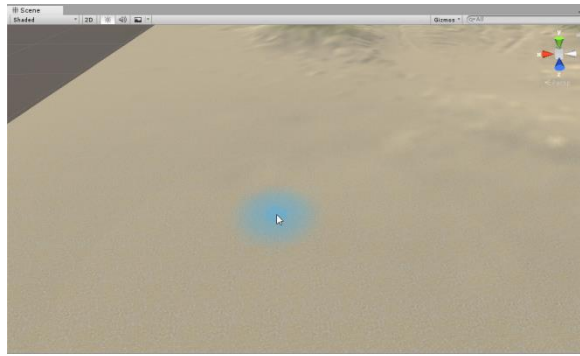
After opening Path Painter activate **Paint Mode** by pushing the button (or press G).



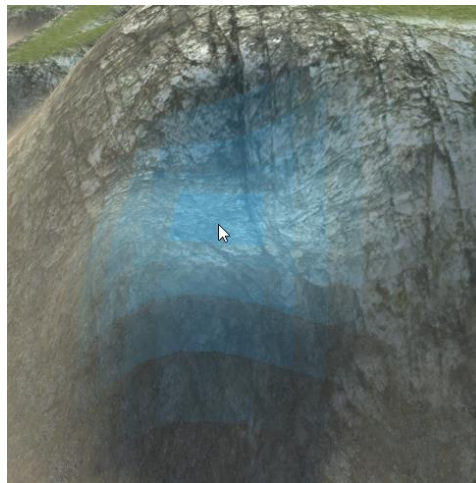
Both the button and the [Status Bar](#) will update to show that **Paint Mode** is active.



The brush will also be visible on the terrain

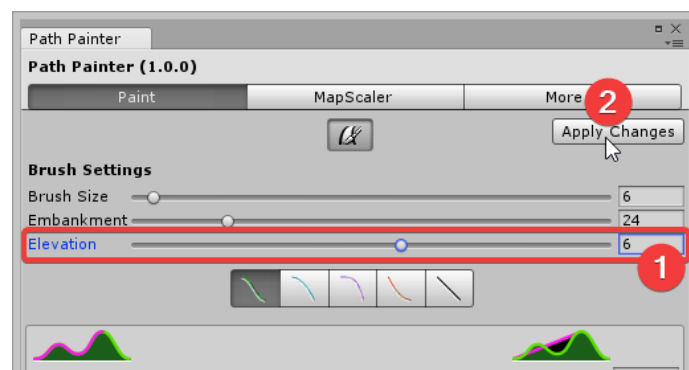


1. Now click on the terrain. - Nothing major will be visible on a flat terrain, but clicking does create a levelled flat surface and it can be used to create terrace spots.

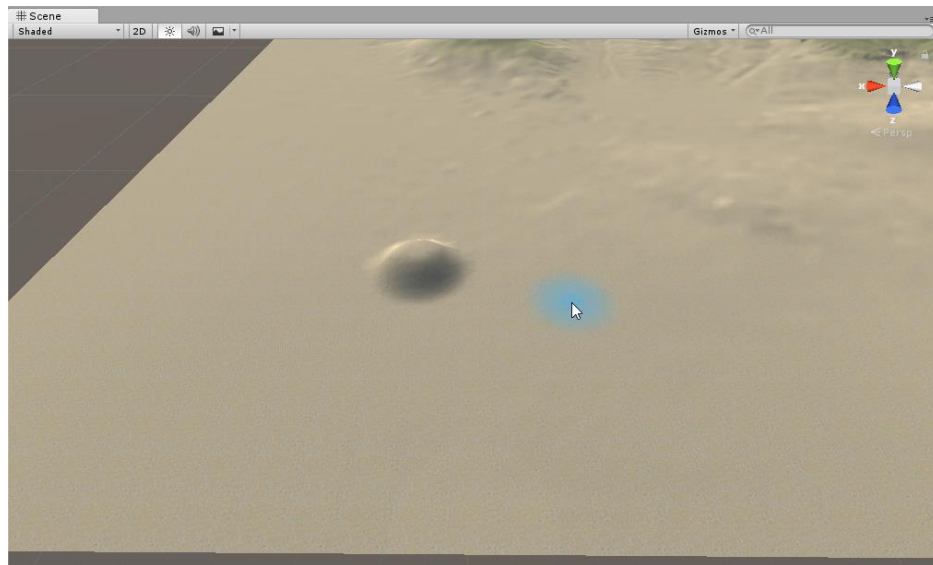


Back to our Demo terrain:

2. Set **Elevation** to **6** and press **Apply Changes**.



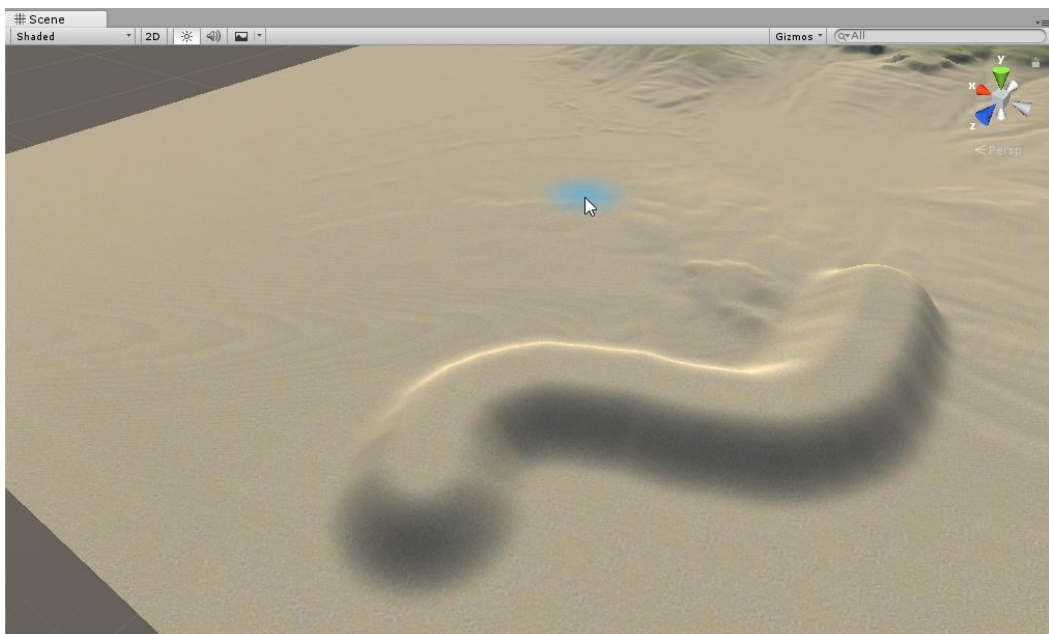
Congratulations! You created your first little hill.



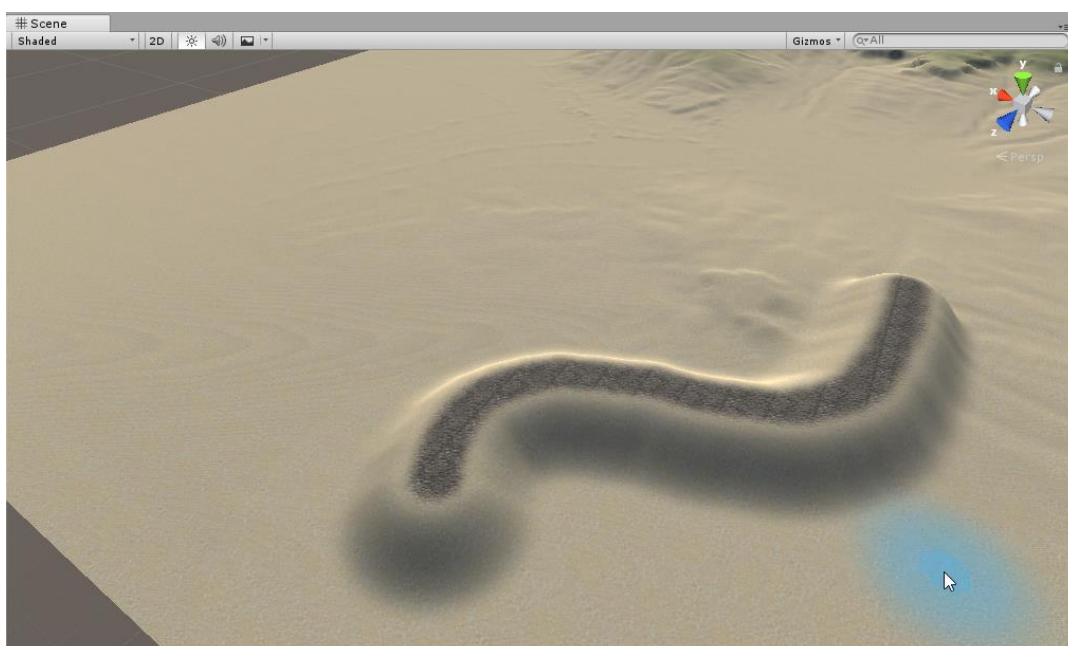
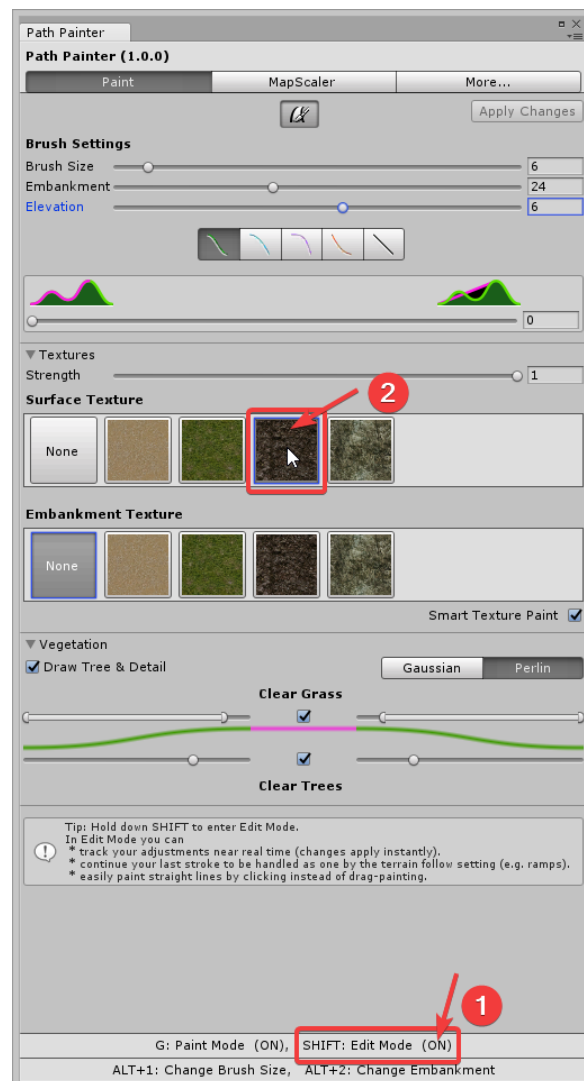
3. Let's get rid of it. **Press CTRL+Z**. You can find yourself doing this often.

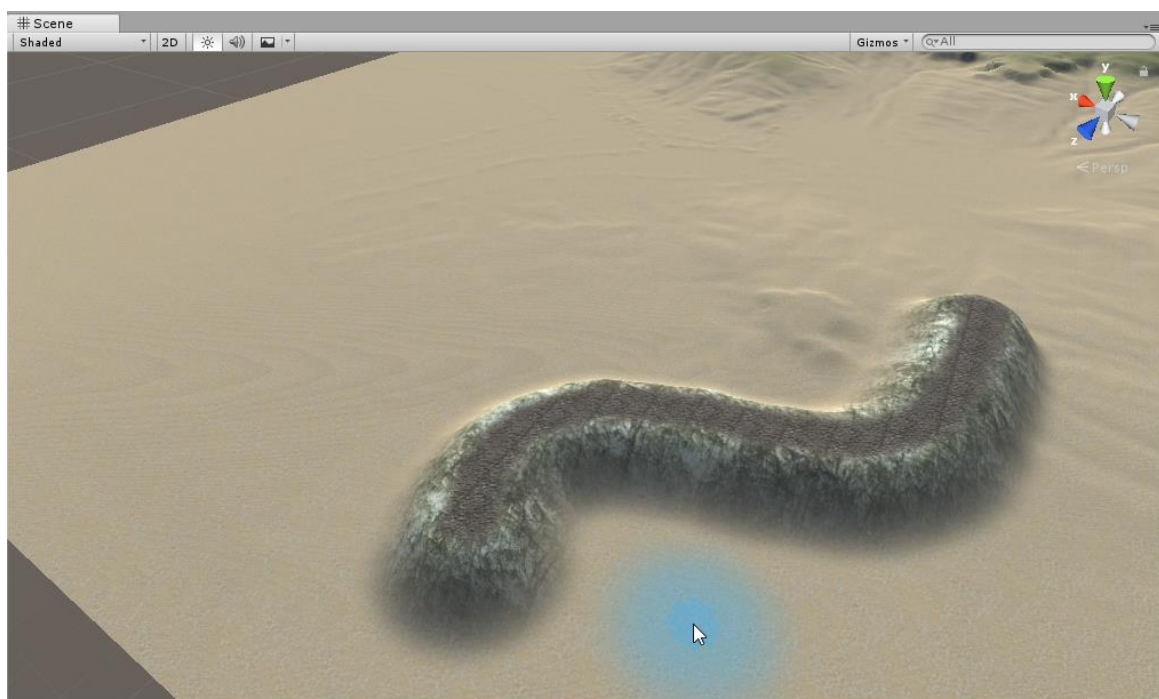
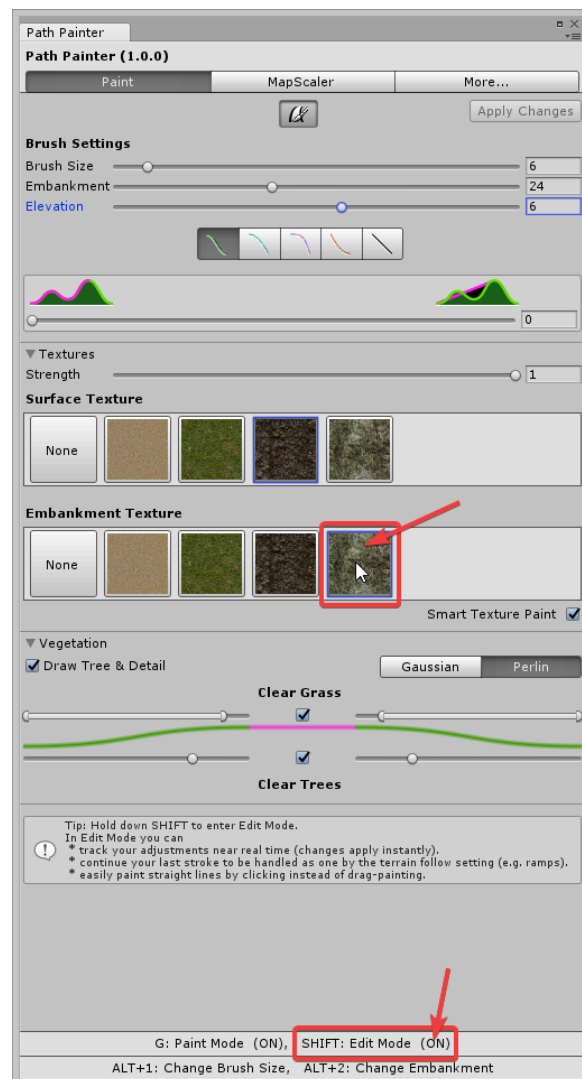
1. First painting a quick prototype to tweak your settings to what you want to do next.
2. Removing the prototype
3. And painting the paths you had in mind.

4. Now paint a line.



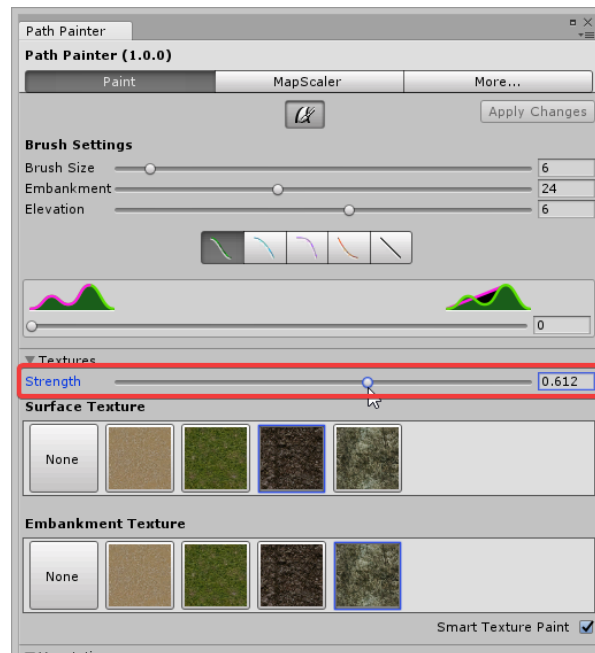
5. Let's add textures and use **Edit Mode** this time (hold down **SHIFT**) to avoid getting repetitive strain injuries from clicking the **Apply Changes** button all the time.



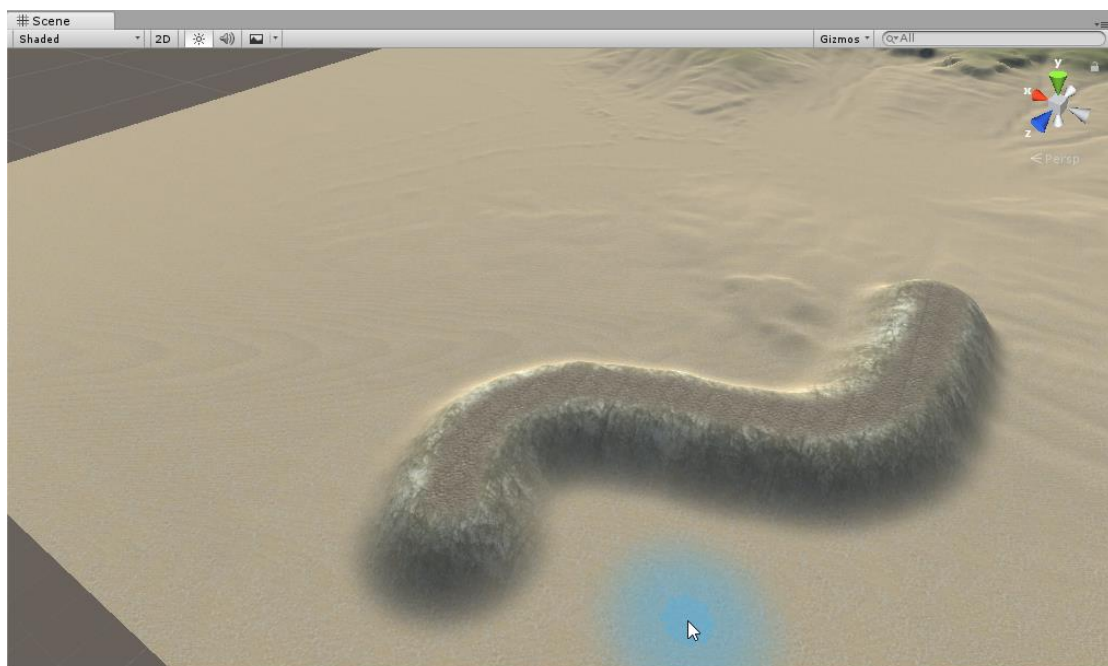
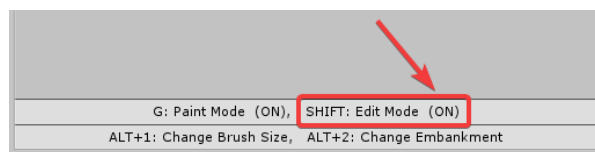


Well done.

6. Play with the **Texture Strength** slider to see what it does. Do this in **Edit Mode** to get live feedback.

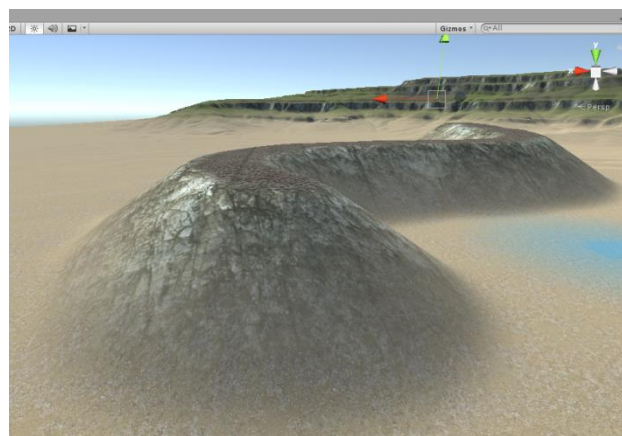
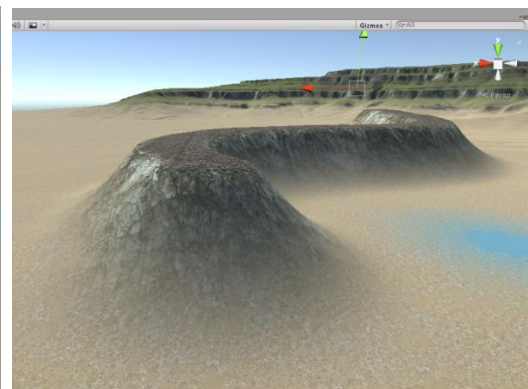
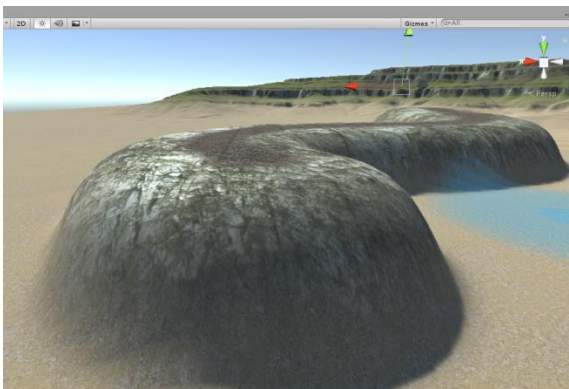
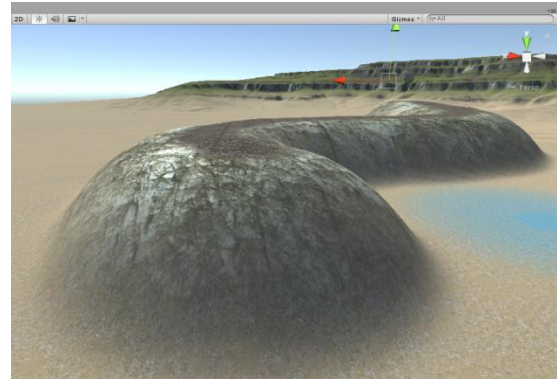
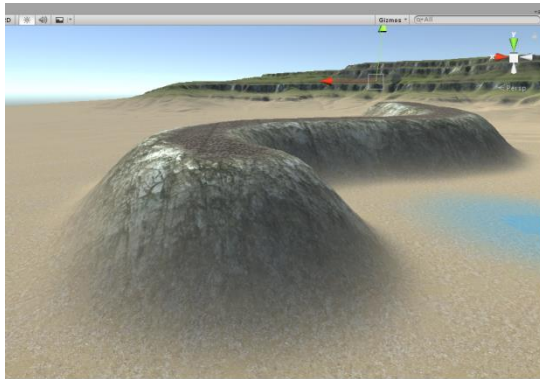
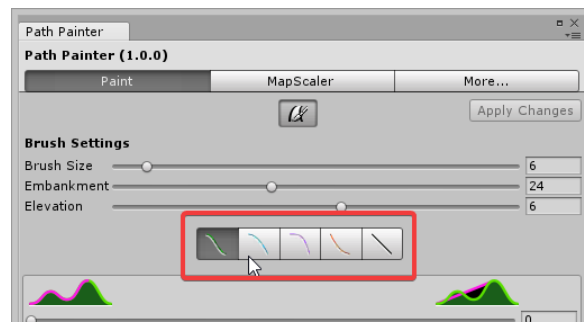


Hold down *SHIFT* to enter **Edit Mode** - you will see the **Status Bar** update



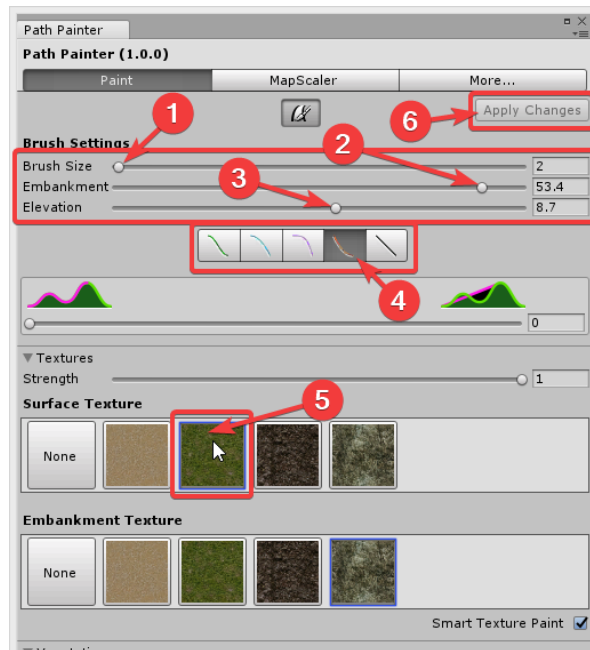
Set it back to 1 when you are done with it.

6. Play with **Embankment Curves** to see what they do. (Do this in **Edit Mode**).

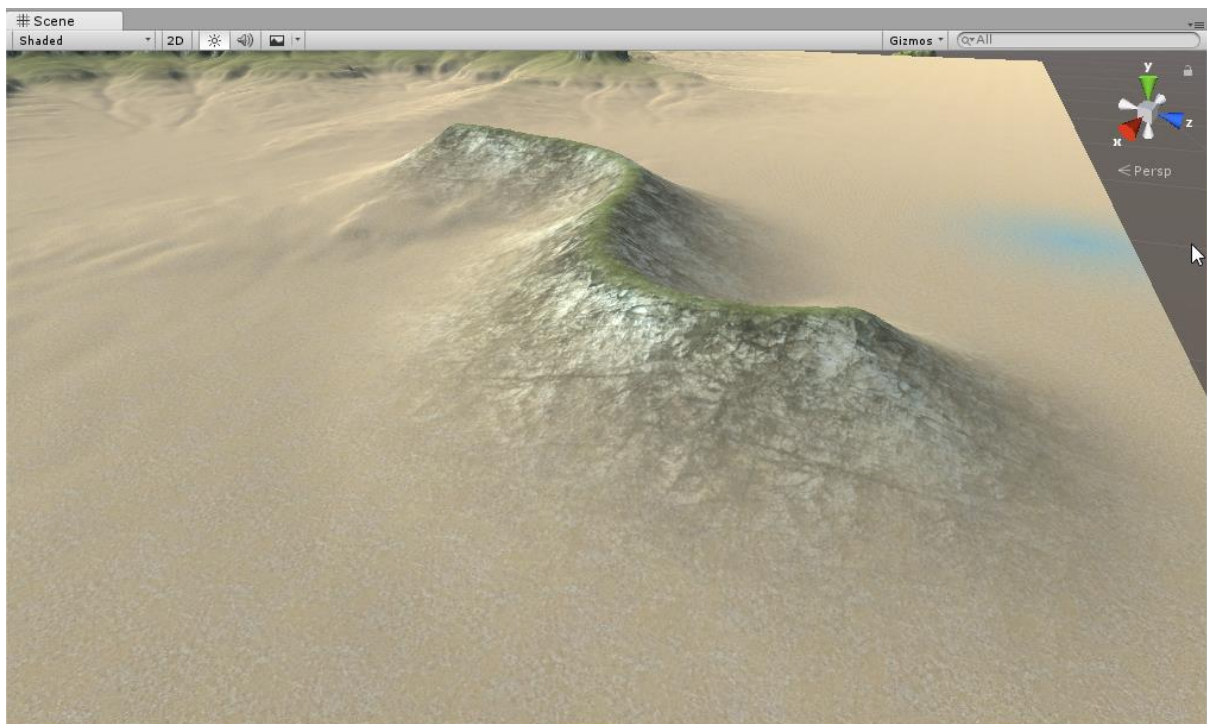


7. Change a few setting and see what we get. This time we will do the changes and apply them all at once. Set

1. **Brush Size:** 2
2. **Embankment Size:** 50 - 55
3. **Elevation:** 8 - 9
4. **Embankment Curve:** Sharp (second from the right)
5. **Surface Texture:** Grass texture
6. And click **Apply Changes**.



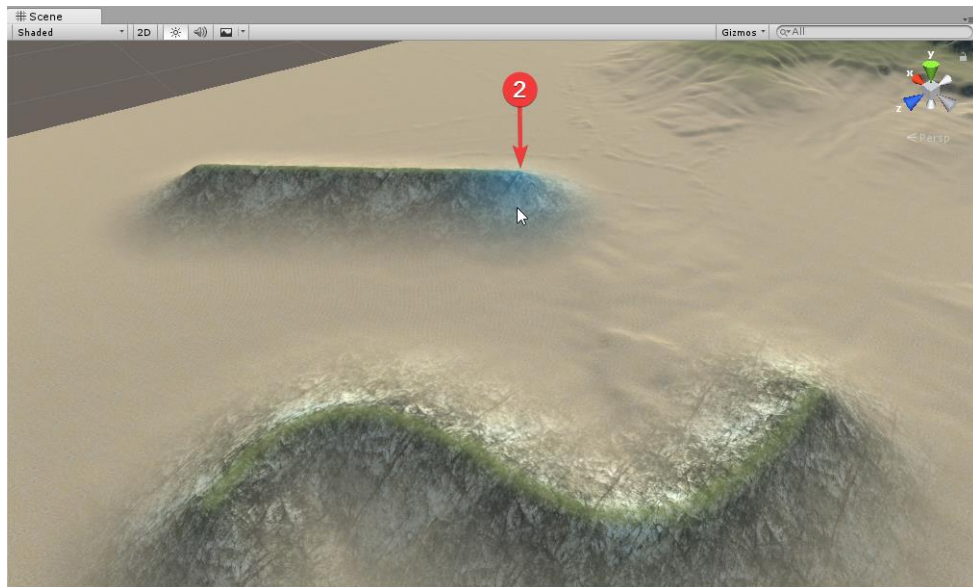
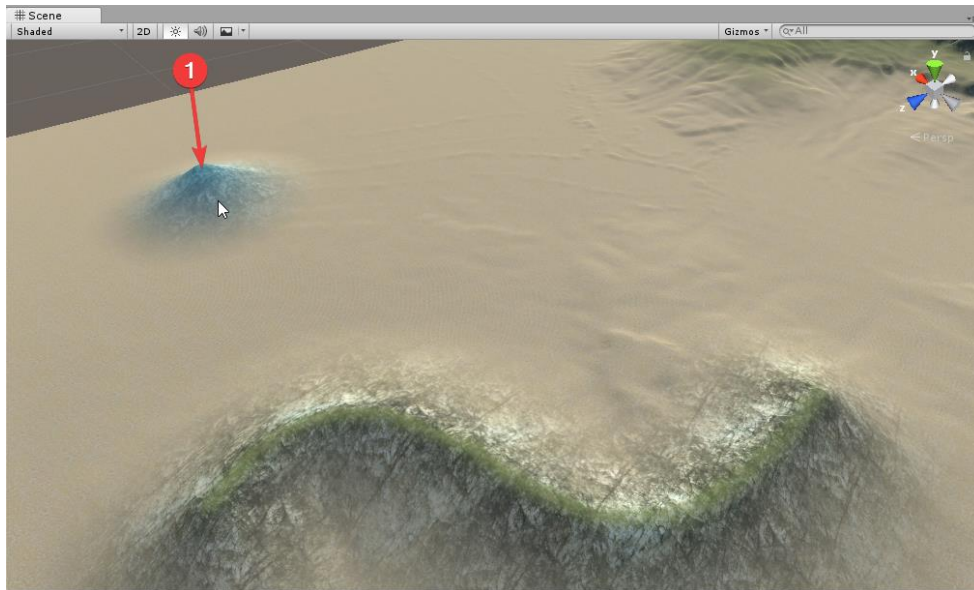
The result:



Ramps, Straight Lines, Mixed Lines

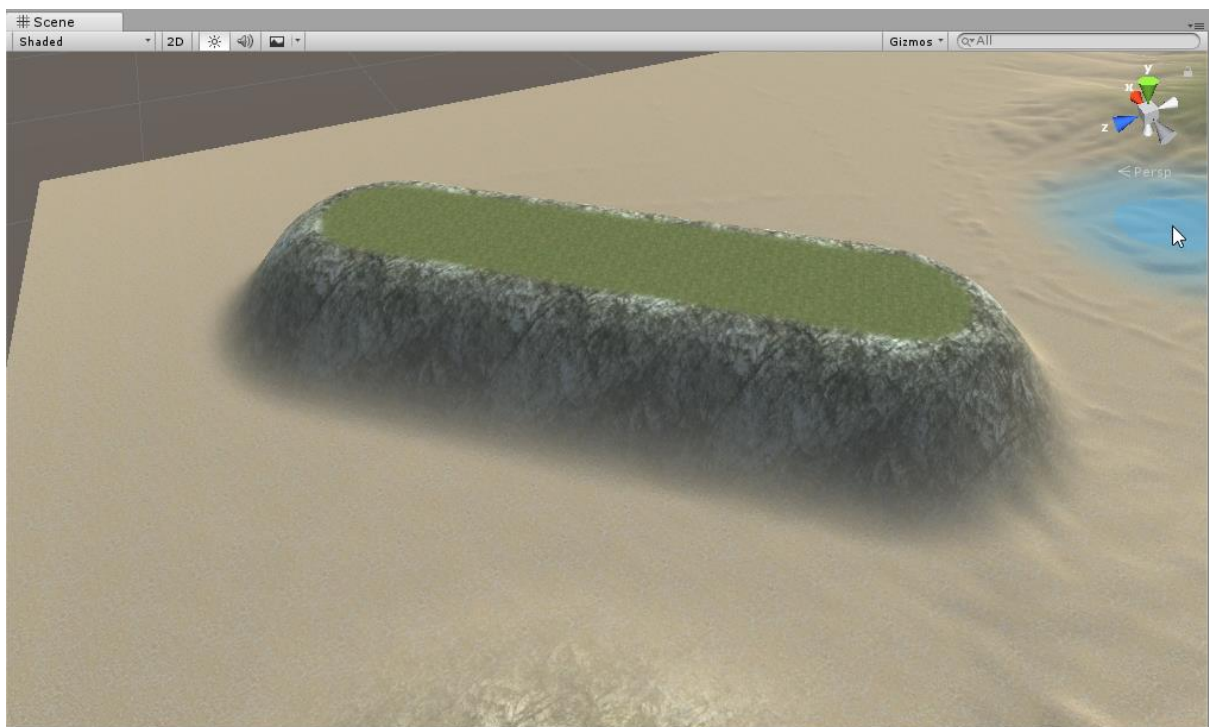
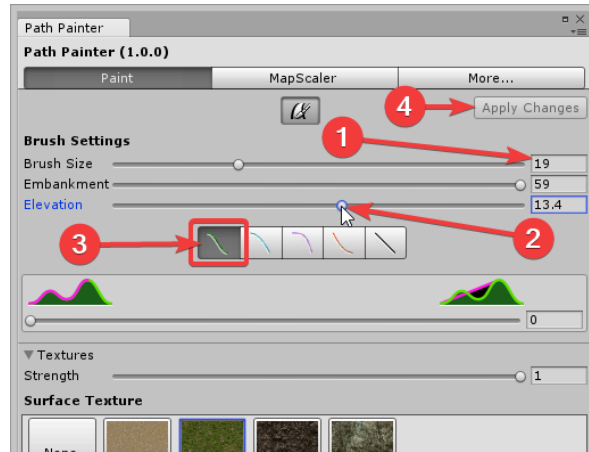
In this example we are going to be working on the *Demo Terrain* found in the *Demo* folder of the package.

1. Create a straight line by clicking a start and end point on the terrain in **Edit Mode** (We are adding to a point to get a straight line).



2. Update the line:

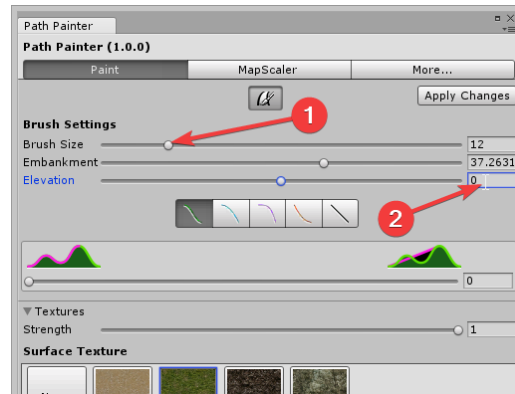
1. **Brush Size:** 19 (**Embankment Size** gets maxed out automatically)
2. **Elevation:** 13-14
3. **Embankment Curve:** Smooth (first on the left)
4. And click **Apply Changes**.



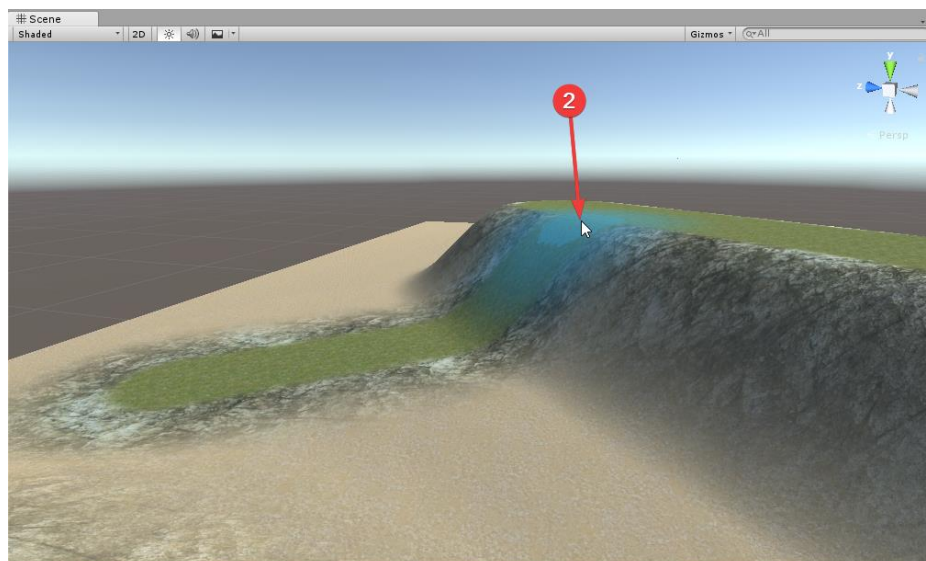
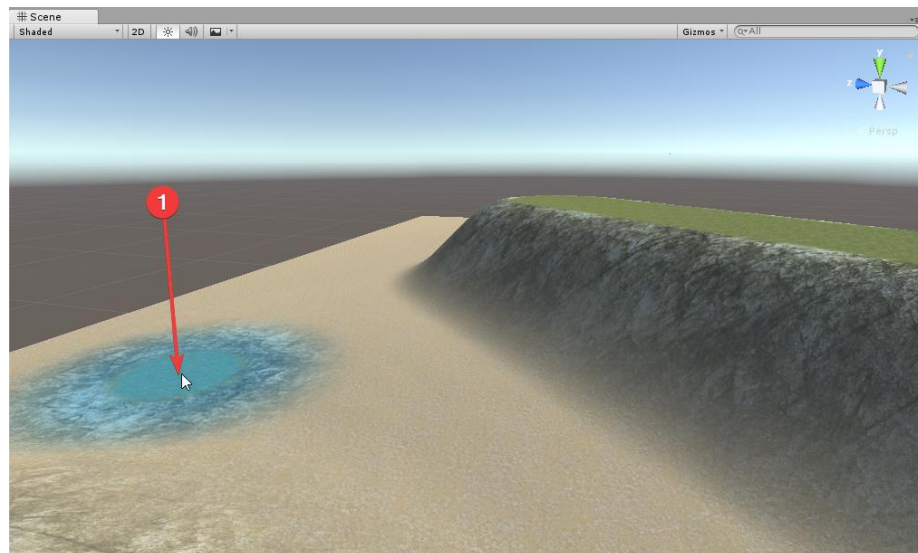
3. Creating a ramp - first step. Set

1. **Brush Size:** 12 (**Embankment Size** updates automatically)
2. **Elevation:** 0

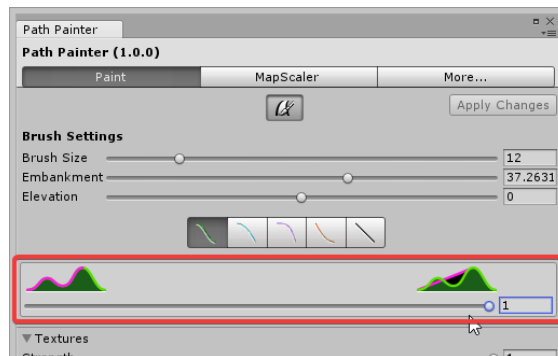
Do not **Apply Changes**. These settings are for the next path.



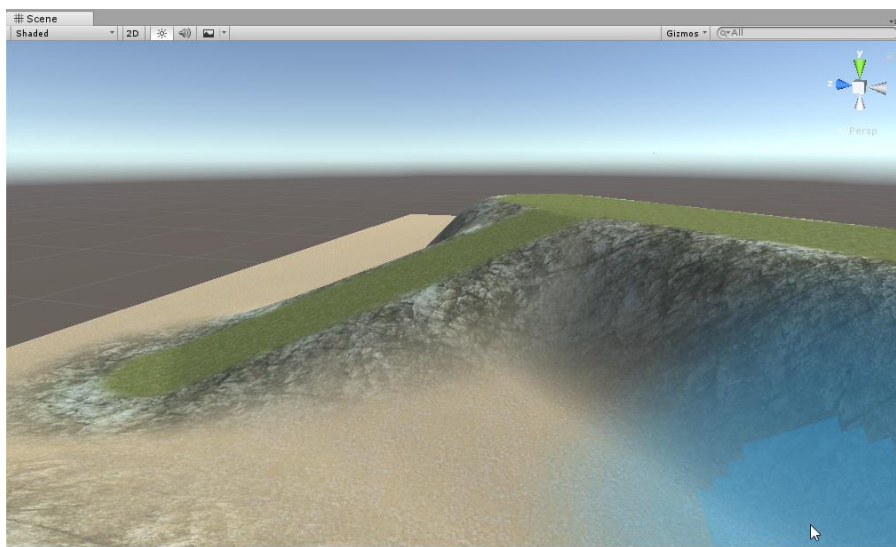
and paint a straight path up to the last one



4. Play with the **Terrain Follow** setting to see how it works and leave it at a setting you like. In the rightmost, even slope position



this is what the path will look like:

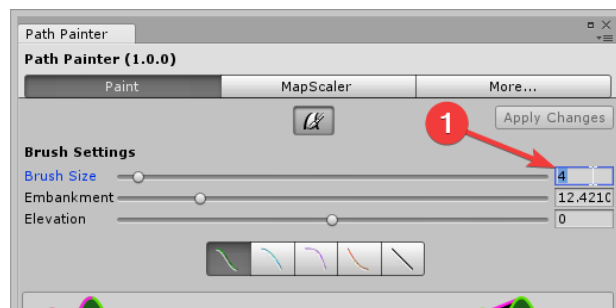


Adding to Paths

In this example we are going to be working on the *Demo Terrain* found in the *Demo* folder of the package.

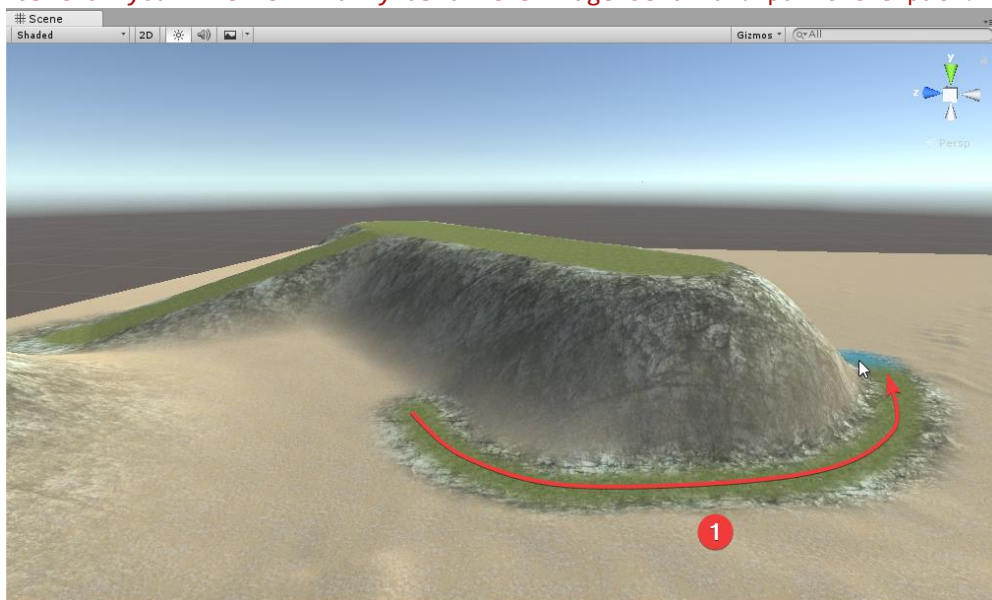
In the next steps we are going to look at when adding to a line comes especially handy. We are going to create a ramp with and without **Edit Mode** activated. In the first scenario we are going to try without using **Edit Mode**. We could fairly easily handle this example situation without **Edit Mode** by using the correct angle of view, but in real life scenarios this can be increasingly difficult.

1. Setup for thinner paths. Set **Brush Size** to 4.

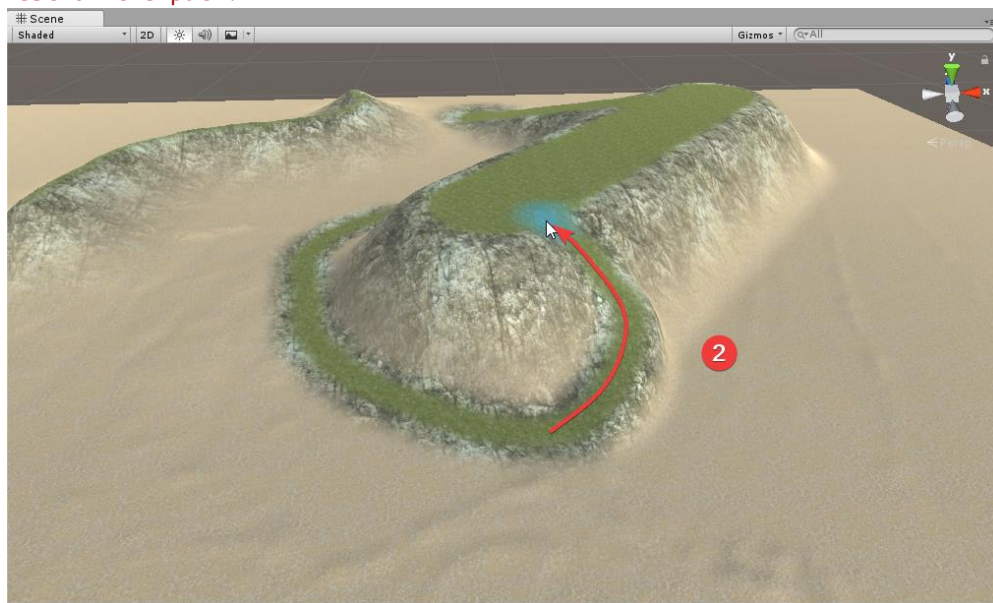


2. Painting the path in two.

1. Position your view similarly as on the image below and paint the path.



2. Now you will need to rotate your camera to see the other side and paint the rest of the path.

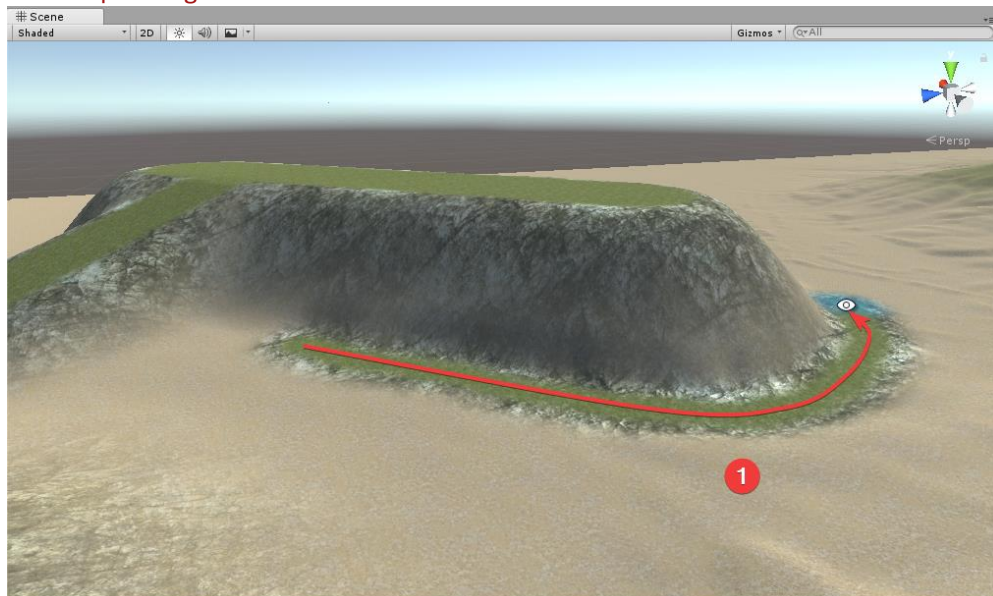


Notice where the actual ramp starts. In some cases this is not desirable, especially if the ramp ends up too steep.

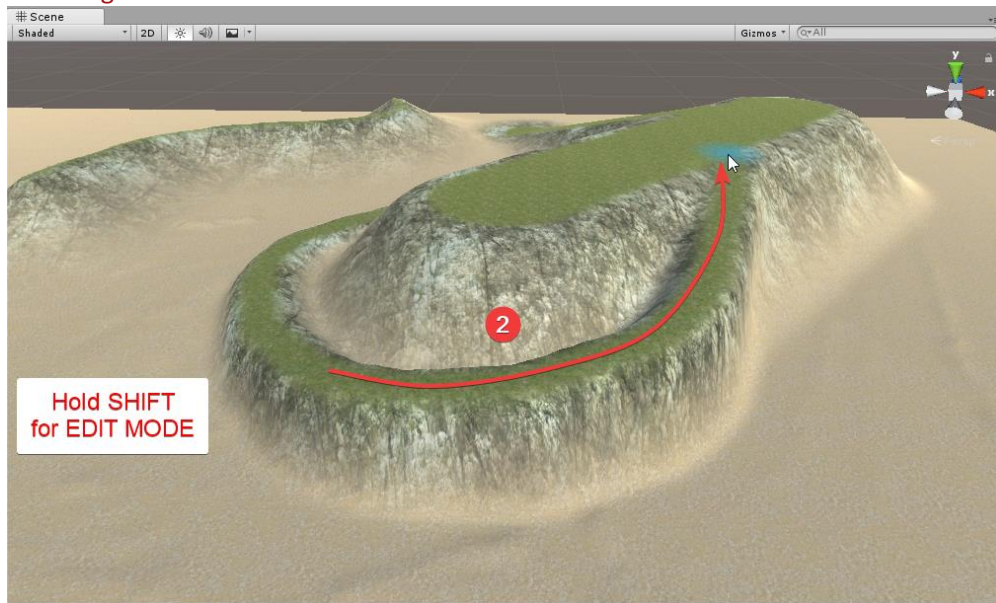
3. Use *CTRL-Z* twice to undo this before we try to do the same in Edit Mode.

2. Now let's imagine that we wanted the previously painter path to be a single ramp but the need to rotate the view made this difficult if not impossible.

1. Position your view similarly as on the image below and paint the first section of the path again.

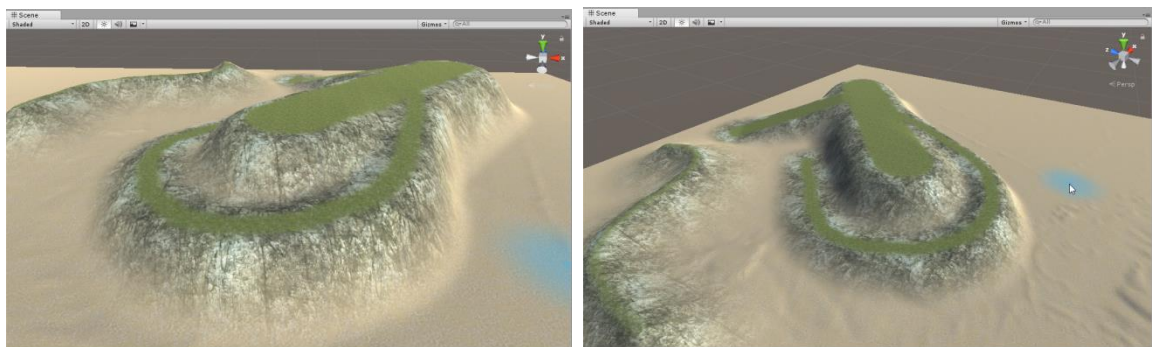
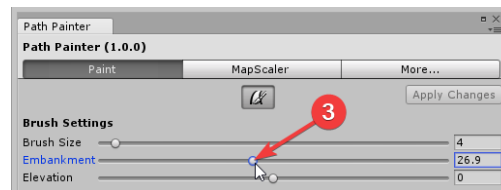


2. Rotate your camera to see the other side and paint the rest of the path, but this time hold down **SHIFT** for **EDIT MODE** to add to the first path instead of starting a new one.



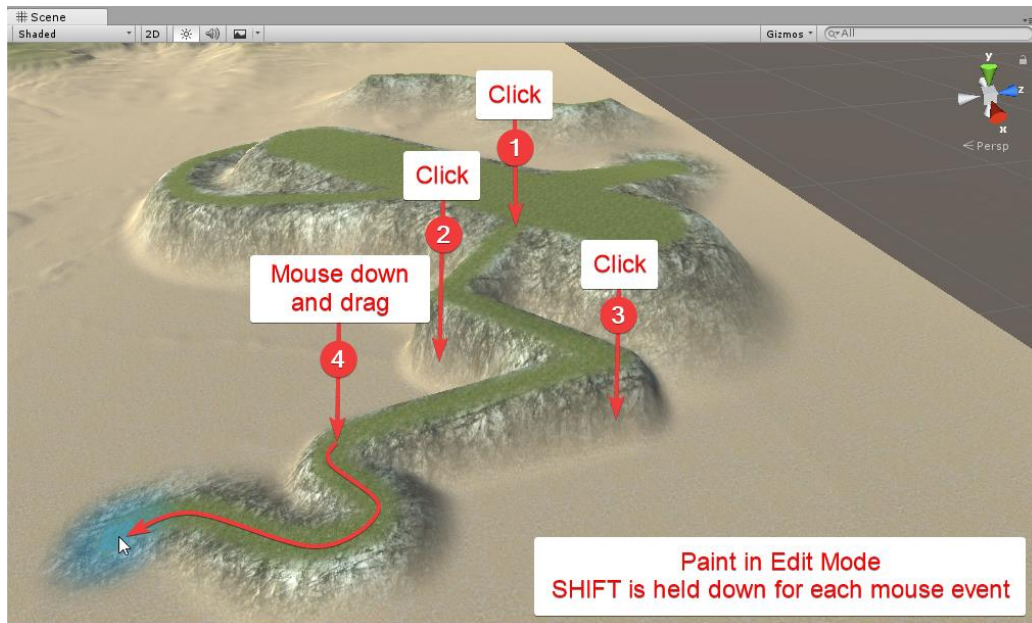
This time the whole path is a single ramp and it will be an easy walk up for characters.

3. You can increase the **Embankment Size** to make it seem like it's part of the feature.

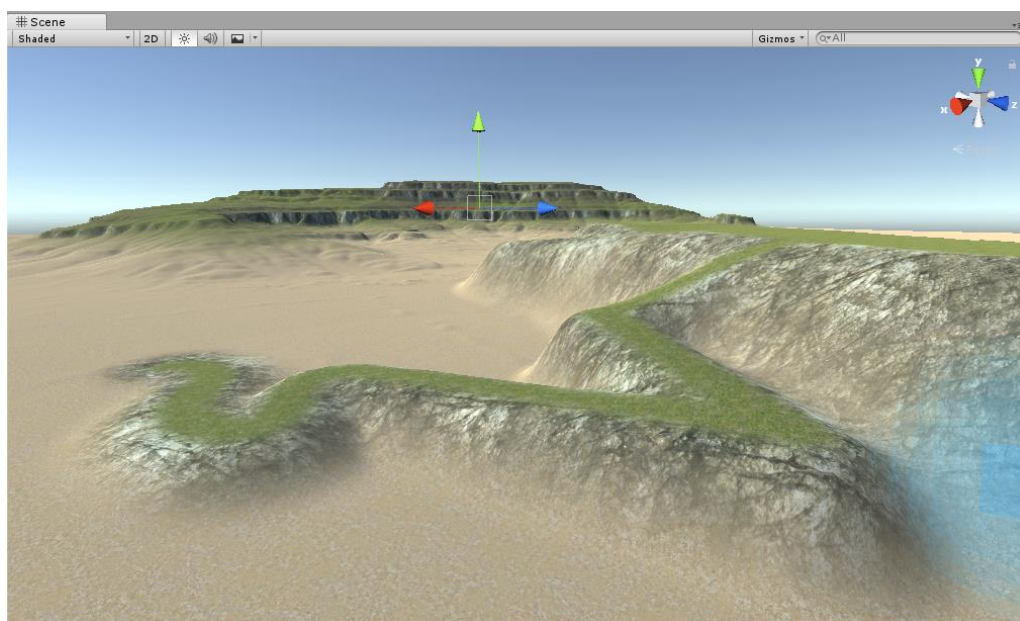
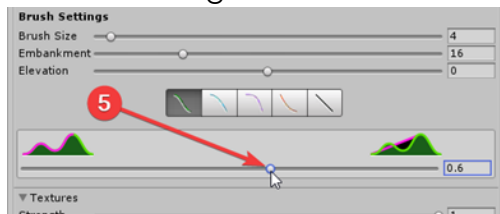


Creating a mixed (straight and curved) path.

1. Set the **Embankment Size** to 16 (don't **Apply Changes**).
2. Hold down SHIFT to add all the sections

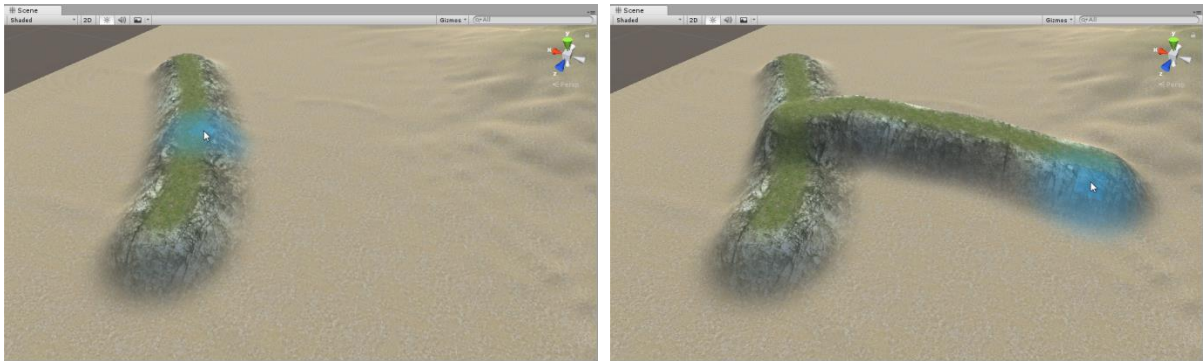


3. You can change the **Terrain Follow** setting to



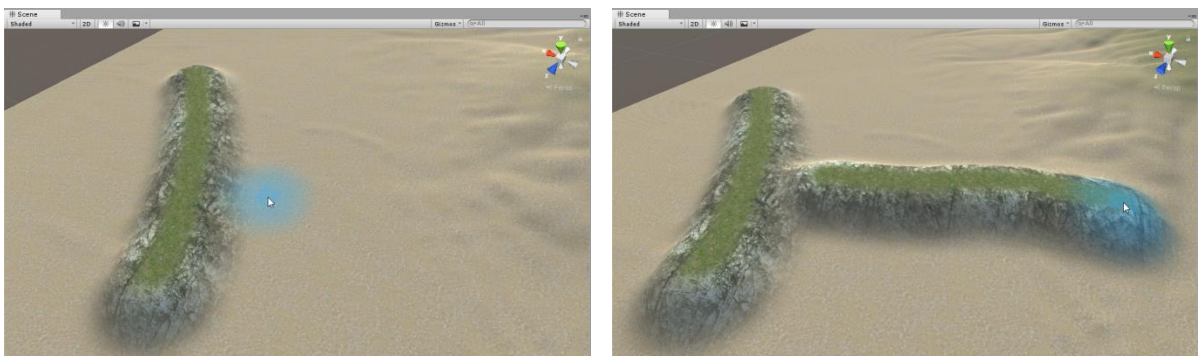
Connecting Elevated Paths, Riverbeds

Path Painter was created to paint. In this sense an existing elevated (or lowered) path/riverbed is just part of the canvas and the next path will apply its elevation according to this.

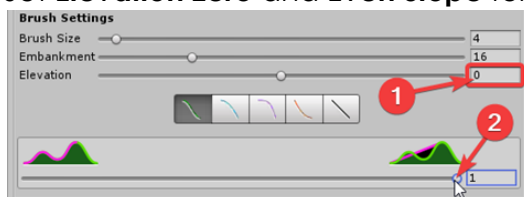


There are plans to improve this in the future. It's easy to get around this in the meantime. Even slope to the rescue! (CTRL-Z to Undo if you followed along)

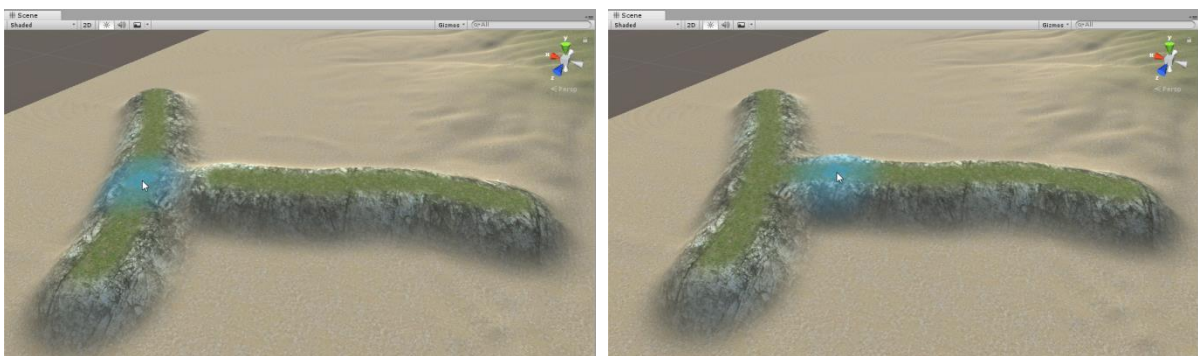
1. Create the paths so they don't cross one another.

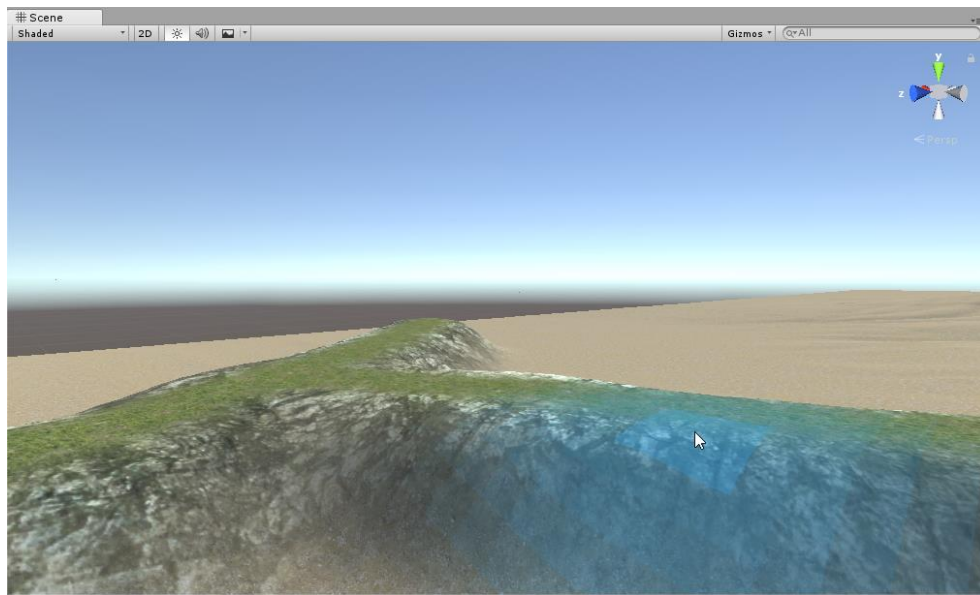


2. Set **Elevation** zero and **Even Slope** for your next step.



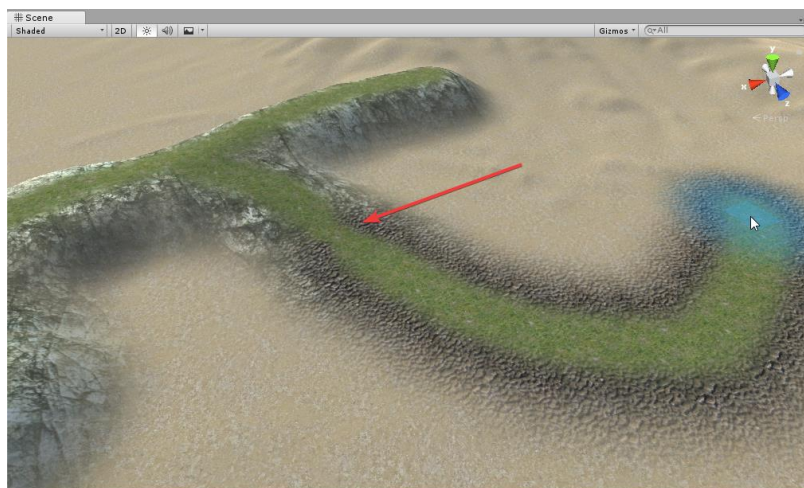
3. Connect the paths with this setting





Blending Embankment Textures

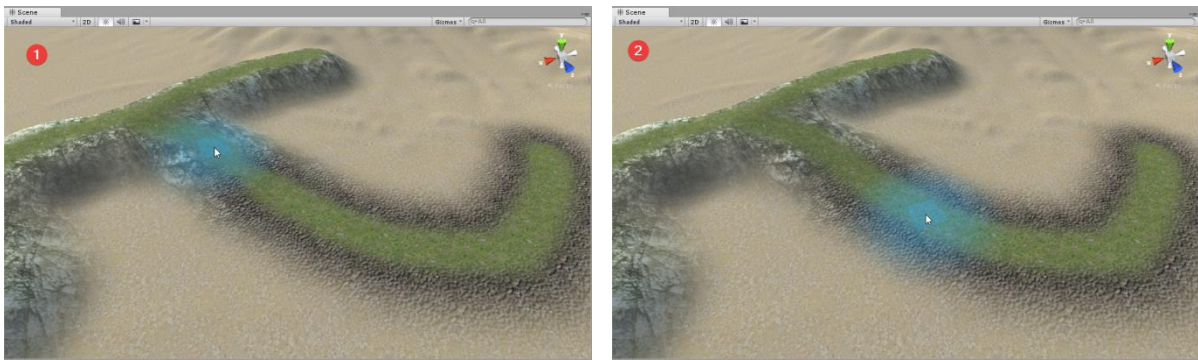
When a path switch to a different embankment texture, a little blending can become necessary.



This can be easily achieved.

- No **Embankment** (**Embankment Size** in the leftmost position, equal **Brush Size**)
- No **Elevation**
- **Terrain Follow** (Leftmost position)
- **Embankment Texture** -> None.

Paint over

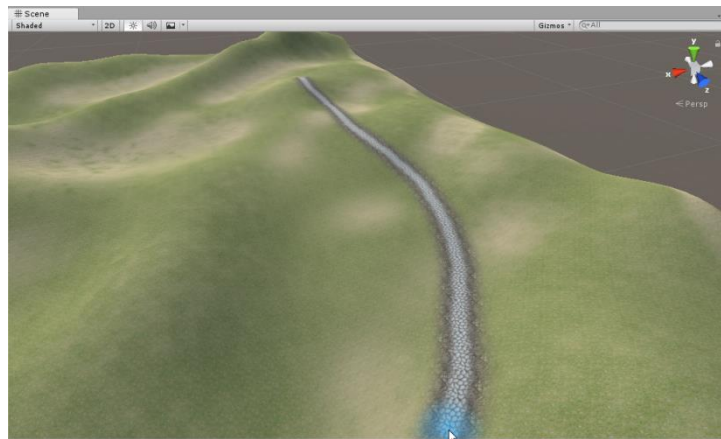


Path and Vegetation

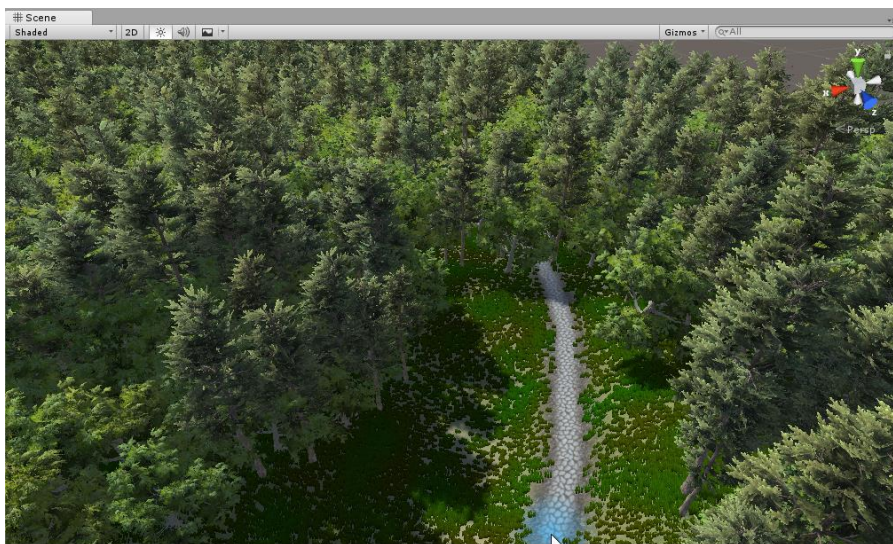
This example uses a terrain with some grass and trees. You can follow along if you have a similar terrain.



While painting, trees and grass are not visible to aid the painting process.

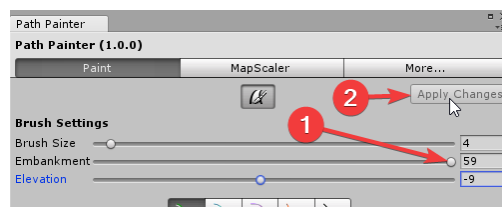


When painting is done, vegetation will be visible again (if drawing for them is enabled; see the [Vegetation](#) section).



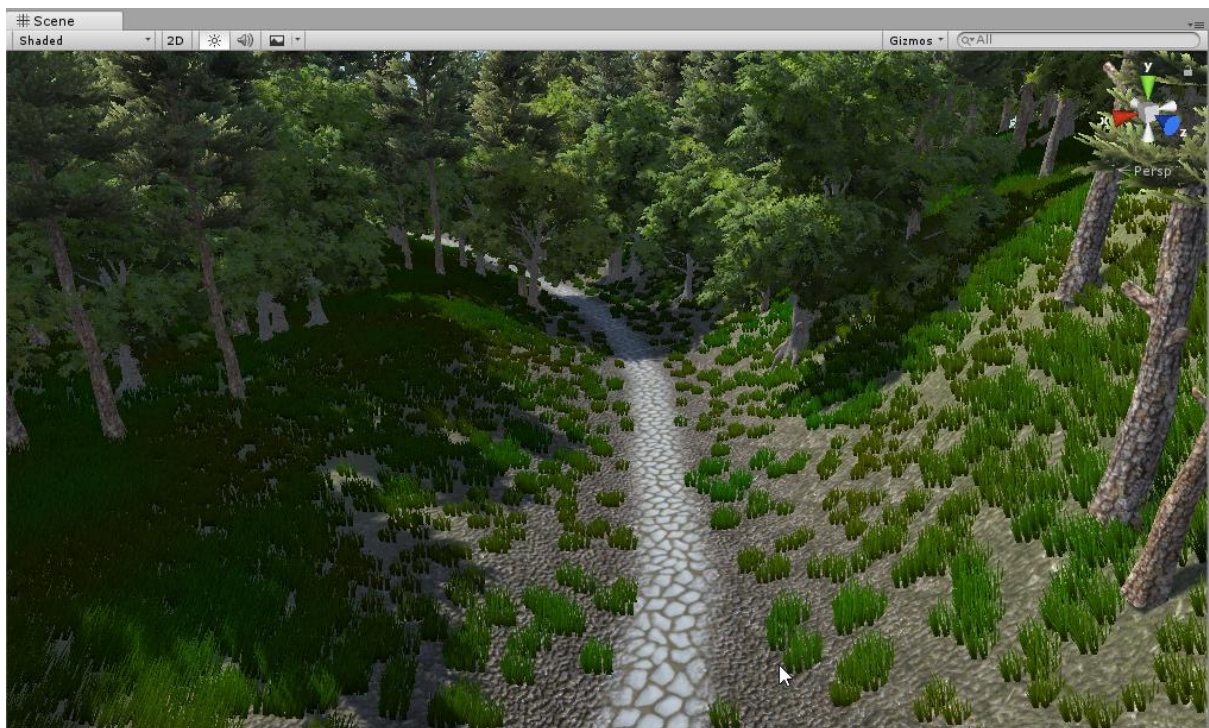
Edit Mode with **Paint Mode** activated will also hide vegetation to aid tweaking. An exception from this is when the vegetation settings themselves are being tweaked.

I maxed out the embankment on the path to better show the thinning and clearing options.

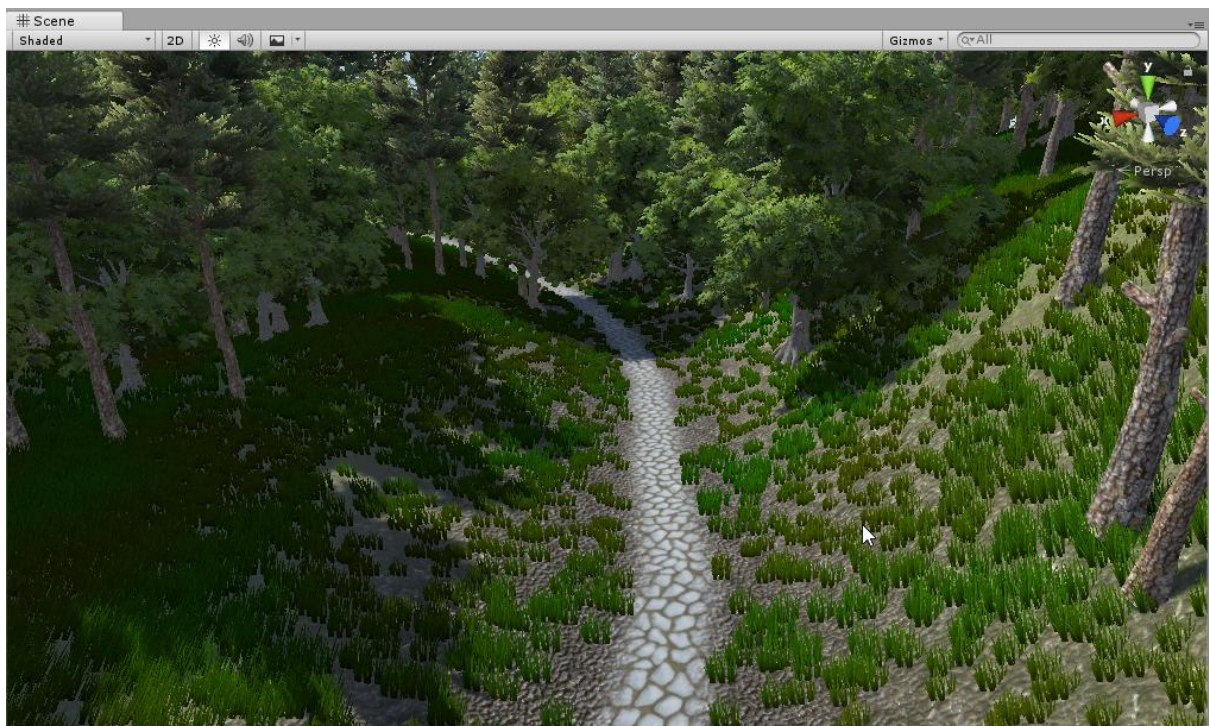
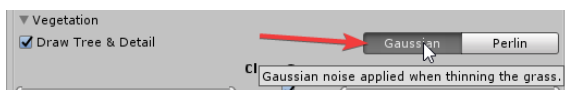


Do the grass tweaks in **Edit Mode** (SHIFT held down). The effect of clearing and thinning is easily visible that way.

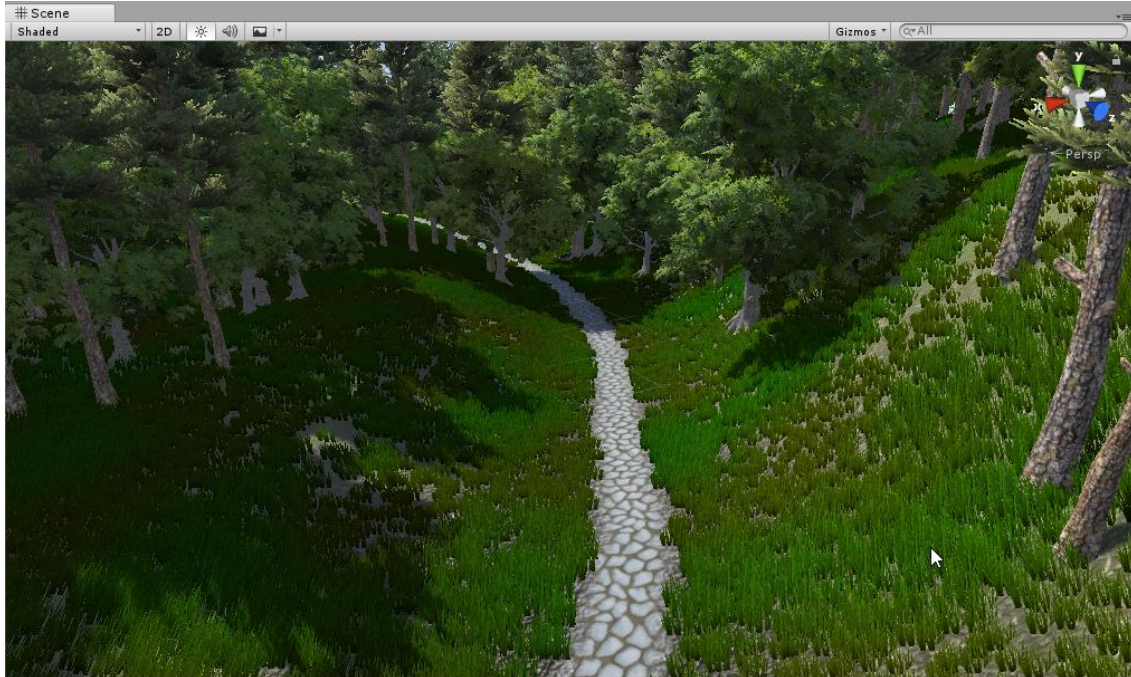
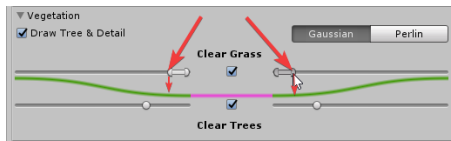
Grass thinning now have **Perlin** Noise applied



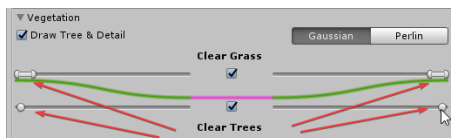
Change to **Gaussian**



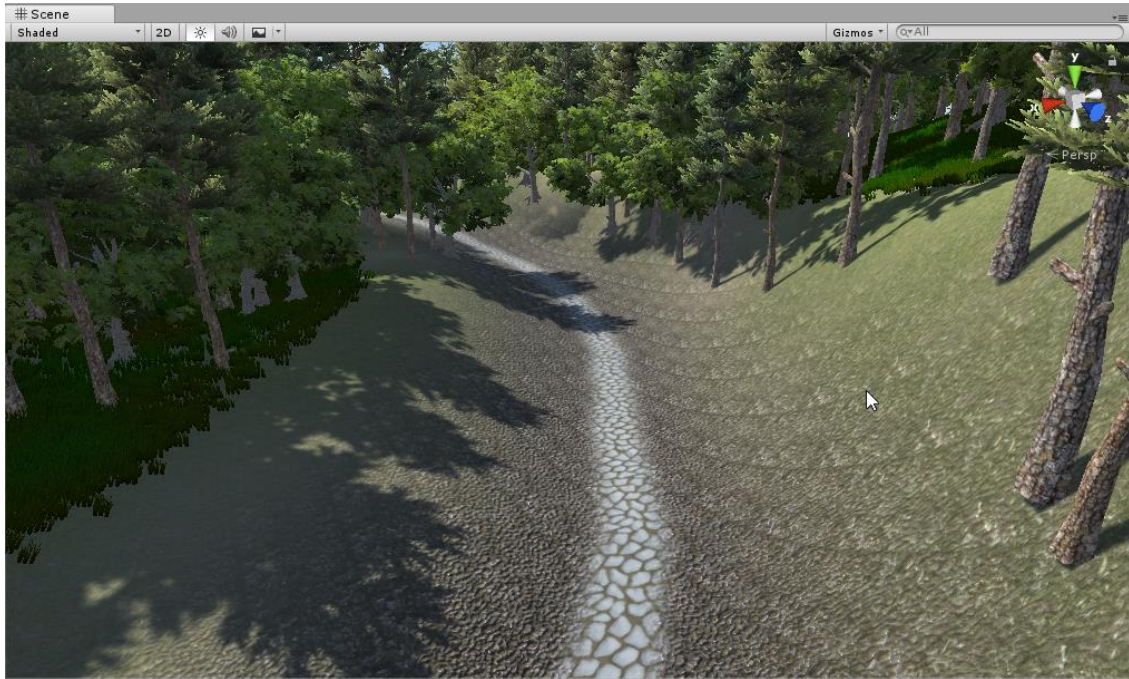
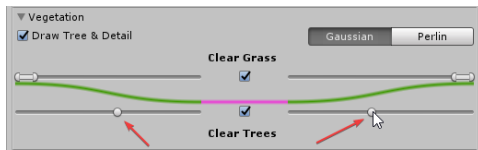
Reduce the thinning range



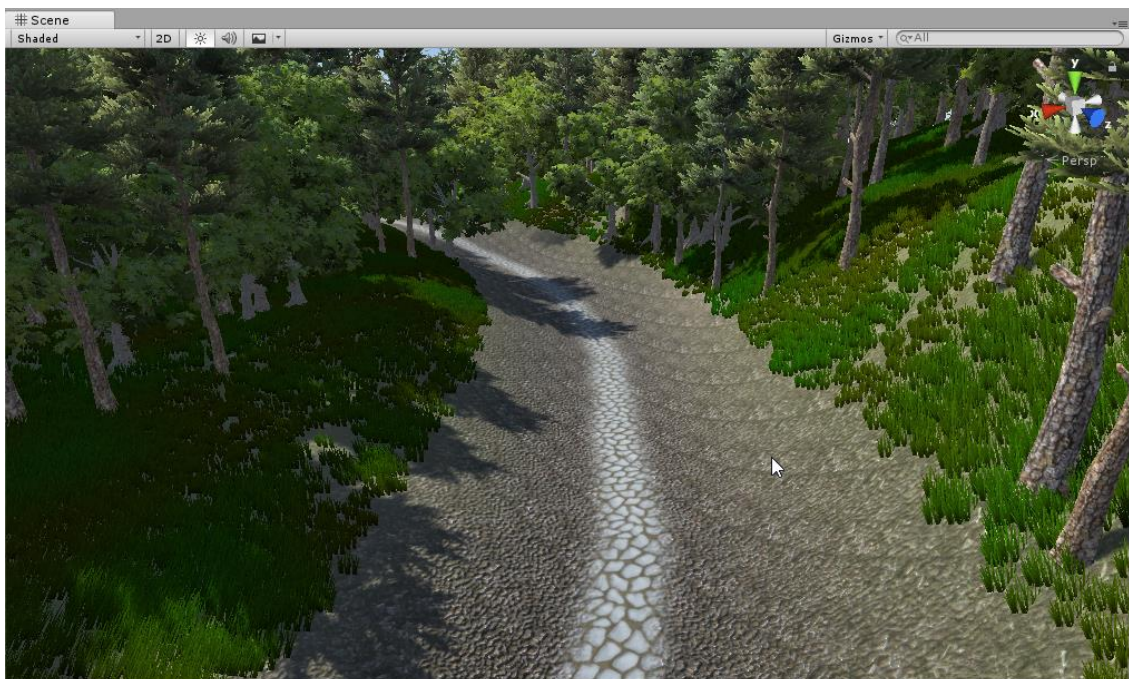
Grab and move the whole grass range to the outer edge of the embankment. Do the same with the tree slider.



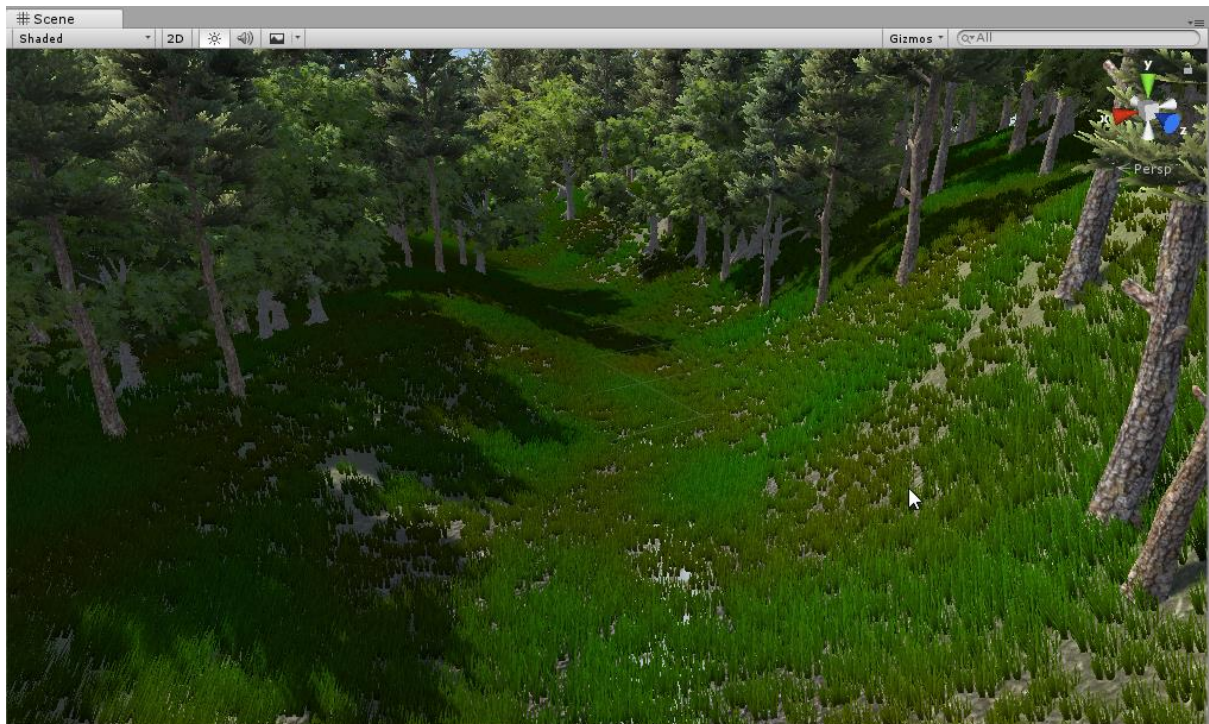
Click on the tree slider to move it back something like this.



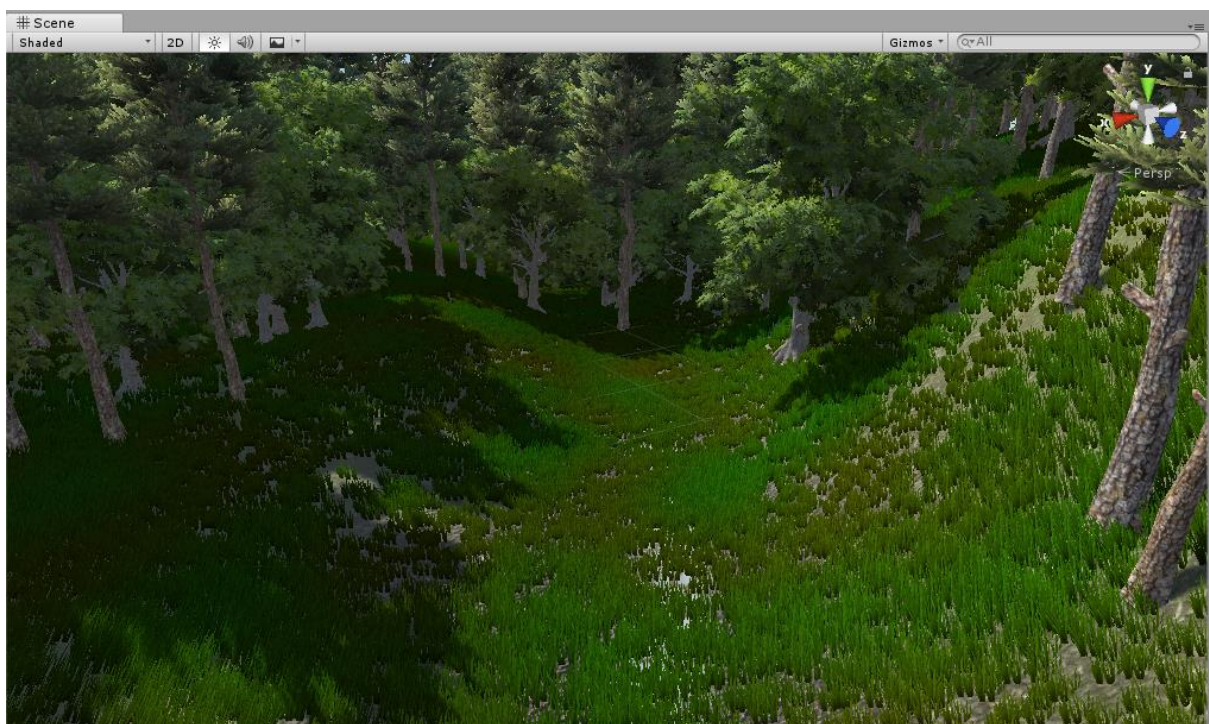
Pull the grass range back as well.



Disable Grass clearing:



And tree clearing:



This was a quick run through of the vegetation settings. It's recommended to play with them a little until it's clear what each of them does. You can also find more information in the [Vegetation](#) section.

MapScaler

Sometimes resolutions of terrains could turn out to be inadequate for the level of detail we would like to include in them. Imagine the scenario that you want to create 4 meter wide paths on a 2 km by 2 km terrain but the *Heightmap* resolution is 513 and the *Control Texture* resolution is 512. You can't create the paths and attempts would look awful. This is where MapScaler comes handy and why it's included in Path Painter. You can scale up each map with a click of a button and you are good to go.

Interface



The resolution of each map is displayed and they can be scaled up/down via the buttons.

Work Flow

Hit the / buttons at the map of your selection and acknowledge the action. MapScaler won't allow you to scale to resolutions that are not supported by Unity.

The resolution of each map is displayed and they can be scaled up/down via the buttons.

More Tab

The **More** tab provides quick access to more information.

Paint API

You are going to be using the Path Painter API for this:

```
using PathPainterAPI;
```

The API expects a list of world space nodes (`List<Vector3>`) that determine points of the path to be painted (similar to how mouse movement could be used by recording a point every time the mouse moves a certain distance). Note: Y values are ignored.

There are a few options to paint using the API.

Just Paint

The API stores its settings during a session. This means that a paint can be initiated without setting any options. The default settings are pretty similar to the ones in the GUI. The default settings can be restored any time by calling `Painter.SetDefaults()`.

Signature:

```
void Paint(List<Vector3> nodes)
```

The below code will use the current settings of the *Painter* API and paint the path determined by the nodes.

```
Painter.Paint(nodes);
```

You can also change the settings in-between Paint calls. This is handy when paths are needed with very similar settings. Example:

```
Painter.Paint(nodes);
Painter.Size = 5f;
Painter.EmbankmentSize = 25f;
Painter.Paint(nodes2);
Painter.EvenRamp = 0.3f;
Painter.Paint(nodes3);
Painter.EmbankCurve = Painter.EmbankmentCurve.Mound;
Painter.Paint(nodes4);
```

Change All Settings and Paint

This can be used for explicit painting to ensure that all the settings are explicitly set a call.

Signature:

```
void Paint(  
    List<Vector3> nodes,  
    float size,  
    float embankmentSize,  
    EmbankmentCurve embankmentCurve,  
    float elevation,  
    float evenRamp,  
    float textureStrength,  
    int textureIndex,  
    int embankmentTextureIndex,  
    bool smartTexturePaint,  
    bool clearGrass,  
    float grassClearingDistance,  
    float grassThinningDistance,  
    Noise grassClearingNoise,  
    bool clearTree,  
    float treeClearingDistance,  
    bool smoothPath = true)
```

Example:

```
Painter.Paint(nodes, 6f, 24f,  
    Painter.EmbankmentCurve.Smooth, 0f, 0.1f, 1f, 2, 3,  
    true, true, 0.1f, 0.3f, Painter.Noise.Perlin, true,  
    0.2f);
```

Change Any Settings and Paint

This is handy to change only certain settings but otherwise use the current Painter settings. Use the `PathPainterAPI.Paint` class to pass in the options.

Signature:

```
void Paint(List<Vector3> nodes, params PaintOption[] options)
```

For example, the below simple script was used in the project that was used for build tests. The script uses a Pegasus path to drive the path creation:

```
/// <summary>
/// Test Paint
/// </summary>
0 references
public void TestPaint()
{
    ResetTerrain();

    PegasusManager manager = GetComponent<PegasusManager>();

    List<Vector3> nodes = new List<Vector3>();
    Transform target = manager.m_target.transform;

    // Let's move to the beginning
    manager.StepTargetBackward(float.MaxValue);

    while (manager.m_totalDistanceTravelled < manager.m_totalDistance)
    {
        nodes.Add(target.position);
        manager.StepTargetForward(flowSlider.value);
    }
    manager.StepTargetBackward(float.MaxValue);

    Painter.Paint(nodes, Paint.Size(sizeSlider.value), Paint.EmbankmentSize(embankSizeSlider.value),
        Paint.Elevation(elevationSlider.value), Paint.EvenRamp(evenRampSlider.value),
        Paint.TextureIndex((int)indexSlider.value),
        Paint.EmbankmentTextureIndex((int)embankIndexSlider.value));
}
```

Troubleshooting

Below you will find some troubleshooting tips.

The shape of the path is jagged, not smooth

The possible reasons:

- Your *Heightmap Resolution* is too low for the fine details you are trying to create.
Solution: Increase your *Heightmap Resolution* [MapScaler](#).
- You are trying to create sharp edges or vertical walls that the Unity terrain can't handle.
Solution: Increase the Embankment Size until you are happy with the result.

The painted texture has sharp edges

The possible reasons:

- Your *Control Texture Resolution* is too low for the fine details you are trying to create.
Solution: Increase your *Control Texture Resolution* with [MapScaler](#).

Grass clearing is inaccurate and disproportionate

The possible reasons:

- Your *Detail Resolution* is too low for the fine details you are trying to create.
Solution: Increase your *Detail Resolution*.

Can't Nicely Connect Raised Paths or Lowered Riverbeds/Paths

See [Connecting Elevated Paths, Riverbeds](#) for a solution.

The Embankment is Not Being Painted Randomly

Try turning **Smart Texture Paint** off for those areas. See the [Textures](#) section for more information.

Performance Is Not Ideal When Painting Large Scale

If you find yourself constrained by performance while painting something big, you can try to paint with

- Texture painting disabled (set **Surface Texture** to **None** in the [Textures](#) section)
- **Draw Tree & Detail** off (checkbox in the [Vegetation](#) section)
- smaller brush size ([Brush Settings](#))

then apply your desired settings in one go to the path.

Terrain Editing Reset When I Update My Path

The following workflow is not currently supported:

- Paint a path (and possibly realise something to be changed on the terrain outside of Path Painter for example a missing tree).
- Edit terrain outside of Path Painter.
- Tweak the path.

You can either

- finalise your path before making the other change, or
- create the path only after you made the other change.

See which solution best fits your situation.