Spline Mesh Renderer v2.1
User Manual

WSM GAME STUDIO

Doing all the hard work for you!

2020

SUMMARY

1 - Intro

Thank you for purchasing “Spline Mesh Renderer”!

This package contains all that you need to harvest the power of Splines to:

- Generate long curved meshes (Spline Mesh Renderer)
- Spawn objects along paths (Spline Prefab Spawner)
- Make objects follow paths (Spline Follower)

It’s really simple to use and customize.

Follow this tutorial or watch it on youtube if you like:

Youtube Tutorial

1.1 - How it Works

This package is composed of four main components.

- Spline
- Spline Mesh Renderer
- Spline Prefab Spawner
- Spline Follower

The Spline component is an Editor Extension for creating and editing Splines Paths on the Unity Editor. It allows you to create a reference path of any length composed by sections of any curvature.

The Spline Mesh Renderer Component is responsible for generating and exporting meshes. It works by extruding a base mesh segment along a Spline path. This process allows the procedural generation of long and curved meshes such as roads, railroads, power lines, pipes, etc.

The Spline Prefab Spawner Component is responsible for spawning objects along a Spline Path. It can be used to easily populate your scene with objects that repeat along a path, such as light poles, bridge pillars, check points, etc.

The Spline Follower Component is responsible for moving objects along one or multiple Spline Paths. It can be used to easily create dynamic moving props for visual or gameplay purposes, such as complex platforms movement, orbiting behaviour, etc.

2 - Spline

The Spline component is an editor extension for creating Bezier Splines on the Unity Editor. A Bezier Spline is a curved path (open or closed) composed by one or more Bezier Curves.

Each curve is composed of 4 points (2 control points and 2 handles). The control points are represented on the Scene Window as green circles. They mark the start and end of each bezier curve.

The handles are represented by circles attached to the control points by red lines. They are used to adjust the curvature. The handle color changes accordingly to the Handle Alignment mode selected (See the Handles section for more info)

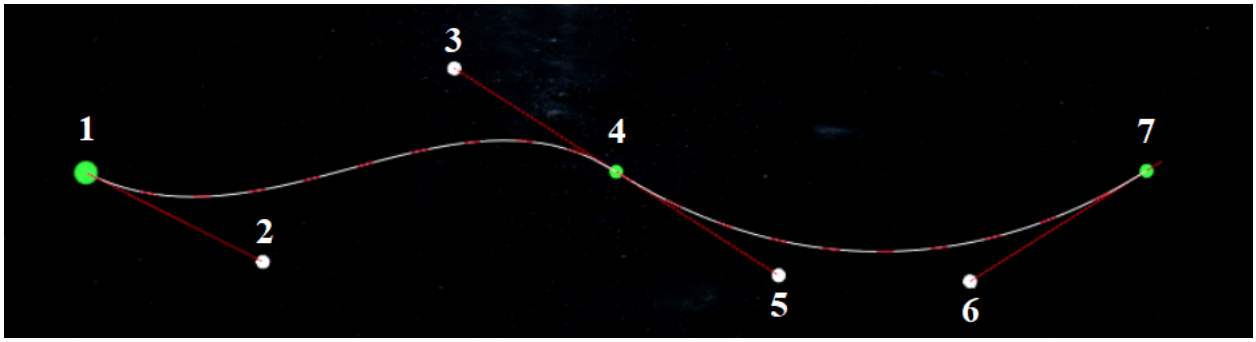

Figure 1 - spline composed by 2 bezier curves

In the example above, you can see a spline composed of 2 bezier curves. Note that the green points mark the beginning and the end of each curve and the curvature is defined by the position of the handles (white circles connected by red lines).

To better understand how Bezier Splines works, take a look on the following affirmations about the spline shown in the sample image:

- This **spline** is **composed by** 2 **bezier curves**
  - **Curve 1** is composed by **points 1, 2, 3 and 4**
  - **Curve 2** is composed by **points 4, 5, 6 and 7**
- Points **1, 4 and 7 are control points** that define the beginning and the end of each curve (Green points).
- The first control point is bigger than the others to mark the beginning of the spline
- Points **2, 3, 5 and 6 are handles** that defines the shape of each curve
- Each **handle is attached to** its corresponding **control point by a red line**
  - Point 2 is a handle from control point 1
  - Points 3 e 5 are handles from control point 4

    - Point 6 is a handle from control point 7
- **All control points are** positioned **inside the** generated **curve**

By using splines it is possible to create curves of any shape or length. This is very useful not only for game development, but also has a large number of practical applications in the real world. For more information about the mathematics behind splines, take a look on this article.

2.1 - Creating a Spline

You can create a new spline by using the default object creation menu:

- **Hierarchy Window**
    - Right Click on Hierarchy window
    - WSM Game Studio > Spline
- **GameObject Menu**
    - Click on GameObject menu
    - WSM Game Studio > Spline

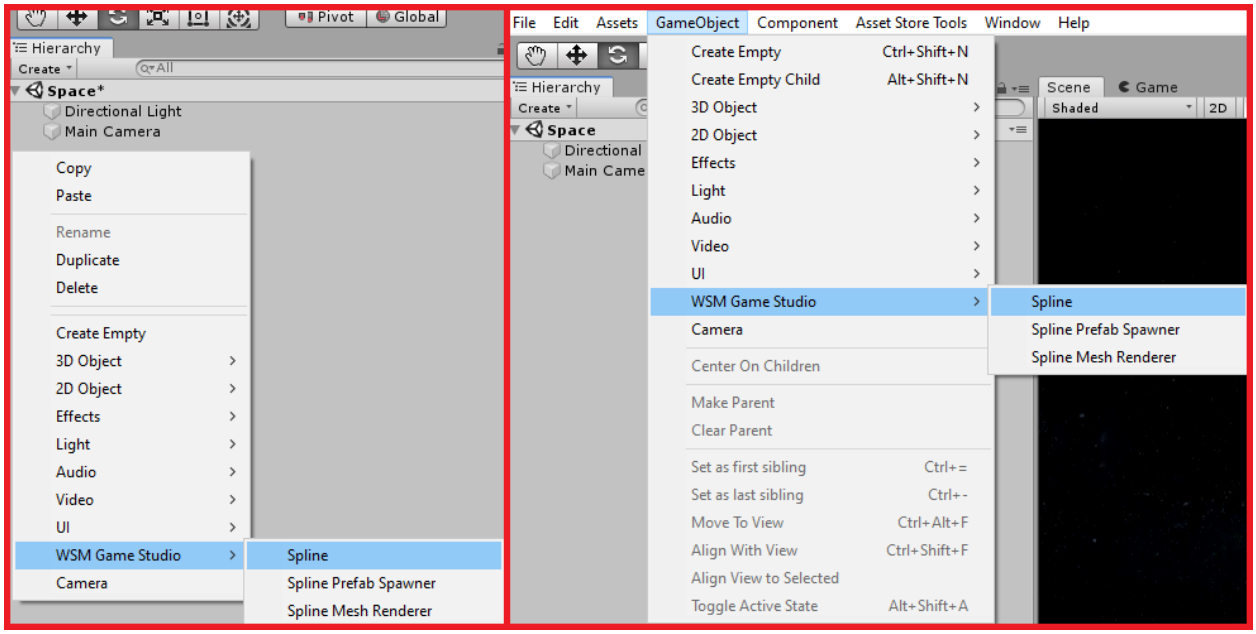

Figure 2 - Creating a new spline

Alternatively, you can also Drag & Drop one of the following prefabs to your scene.

- **SimpleSpline:** a simple spline with no other components attached to it
- **SplineMeshRenderer:** ready to use spline prefab for procedural generation
- **SplineSpawner:** ready to use spline prefab for object spawning

These prefabs are located under “Assets/WSM Game Studio/Spline Mesh Renderer/Prefabs/Legacy”.

Figure 3 - Spline prefabs

NOTE: Prior to version 2.1, these prefabs were used to create new spline instances. They are not needed for this purpose anymore, however they were kept for the sake of backwards compatibility.

You should never change these prefabs directly or save alterations made on them. They were created with the sole purpose of being used as a starting point.

Since the Spline is an editor extension, it will only be visible when an object that has a Spline component attached to it is selected on the Hierarchy. By default, the newly created spline is composed by only one bezier curve.

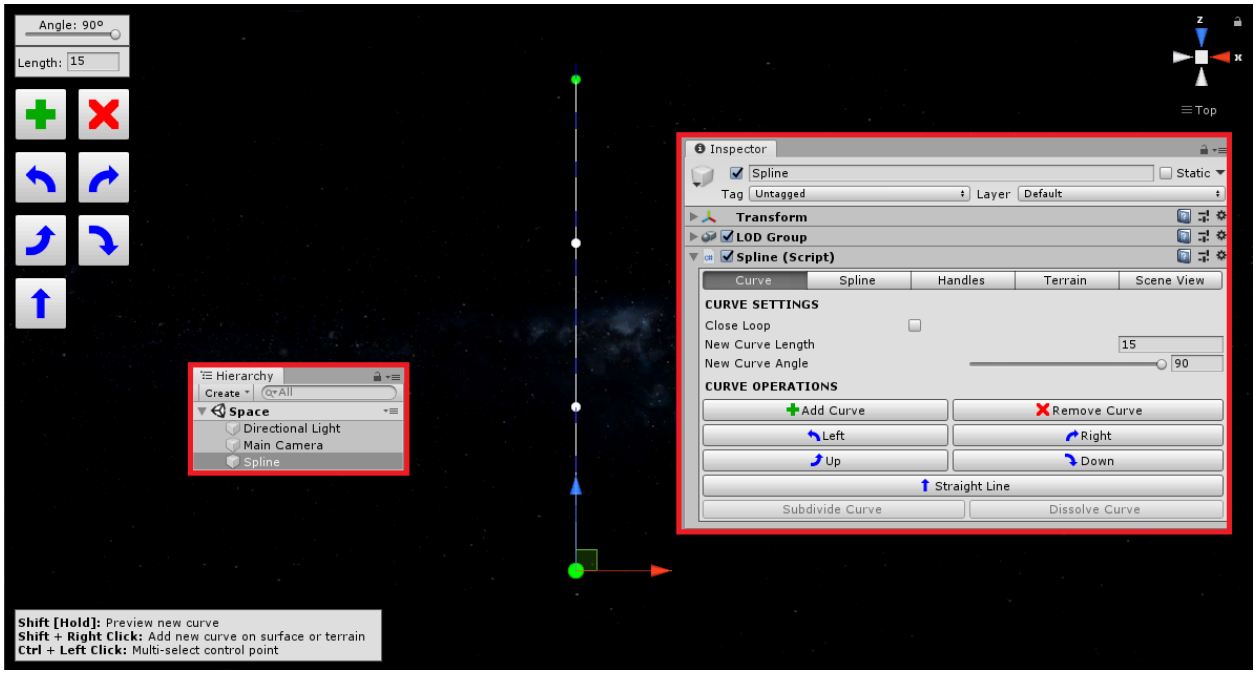

Figure 4 - Newly created spline

NOTE: By default, only selected control point handles will be visible. Please take a look at the Handles Visibility section for more details.

2.2 - Editing the Spline

There are 3 spline editing methods available:

- Quick Access Building Menu
- New Curve Projection
- Manually editing the control points and handles

2.1 - Quick Access Building Menu

The Quick Access Building Menu was designed to be powerful and user friendly at the same time. It allows the user to quickly create complex splines with a few button clicks, while maintaining mathematical precision for both curve length and angle.

When a spline is selected on the Hierarchy, the quick access building access menu will appear on the top left corner of the Scene View.

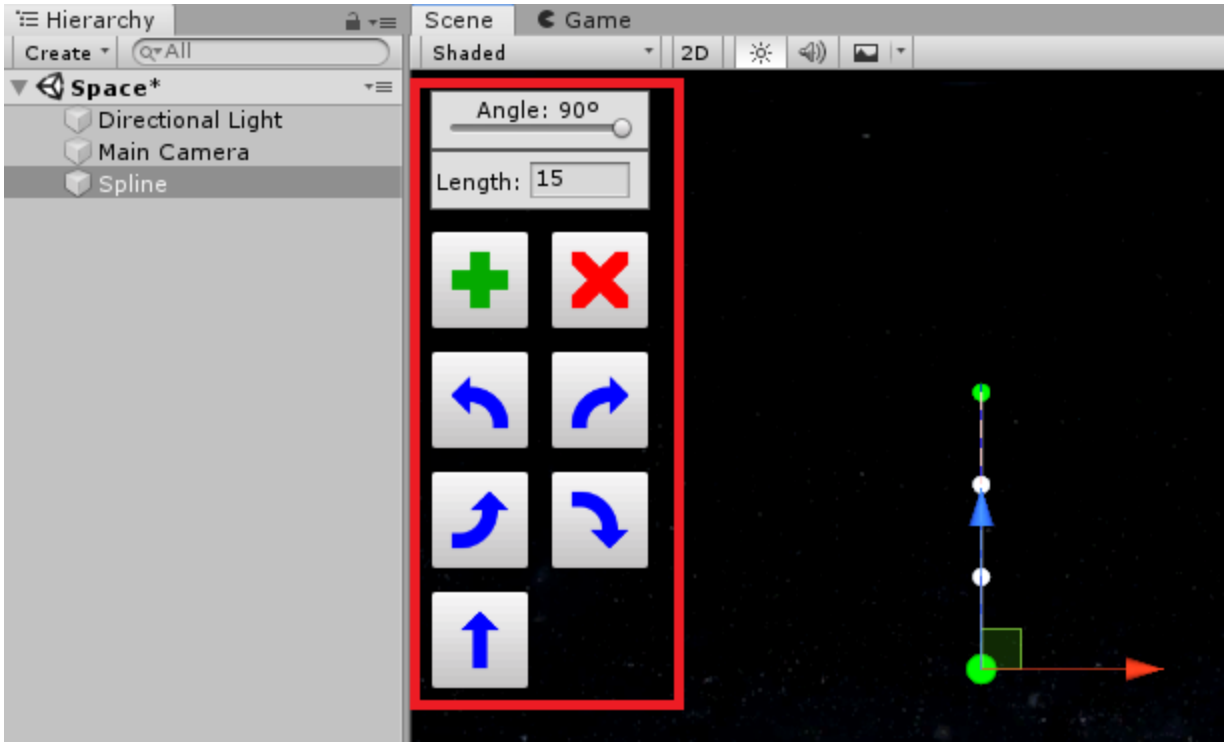

Figure 5 - Quick access building menu

By using this menu, you can quickly create, delete and reshape curves to the desired length and angle.

2.1.1 - New Curve Angle and Length

The new curve length and angle can be adjusted on the quick access menu and/or the unity inspector.

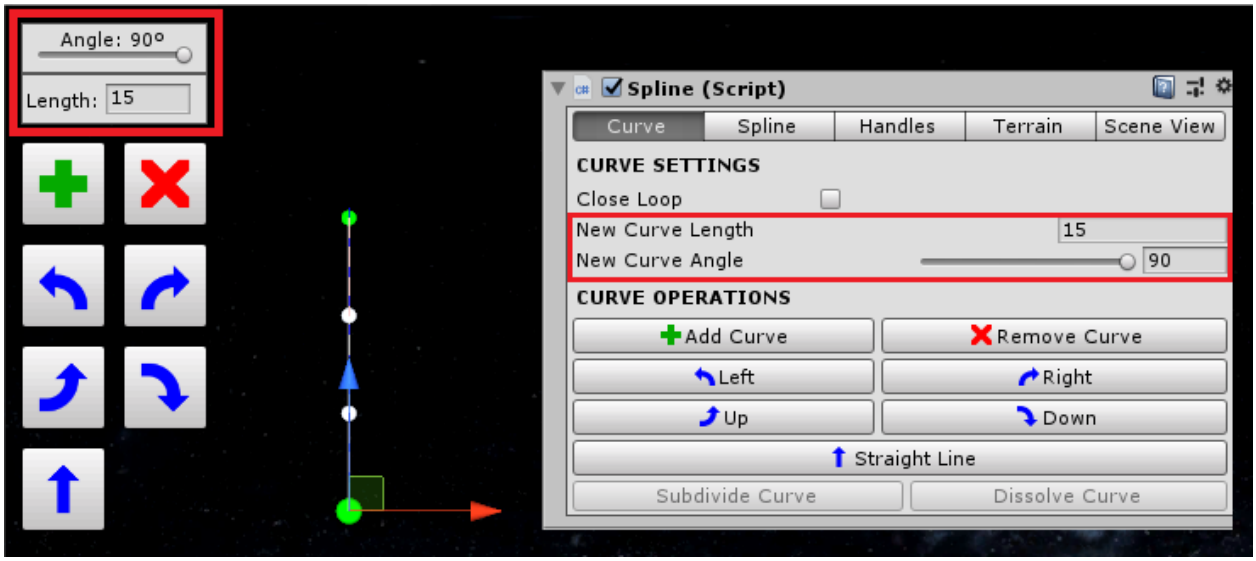

Figure 6 - New curve length and angle

The curve length is measured in meters and affects both the Add Curve and Curve Shaping operations. The angle property is measured in degrees and affects only the Curve Shaping operations. By using both, it is possible to create curves of any length and curvature.

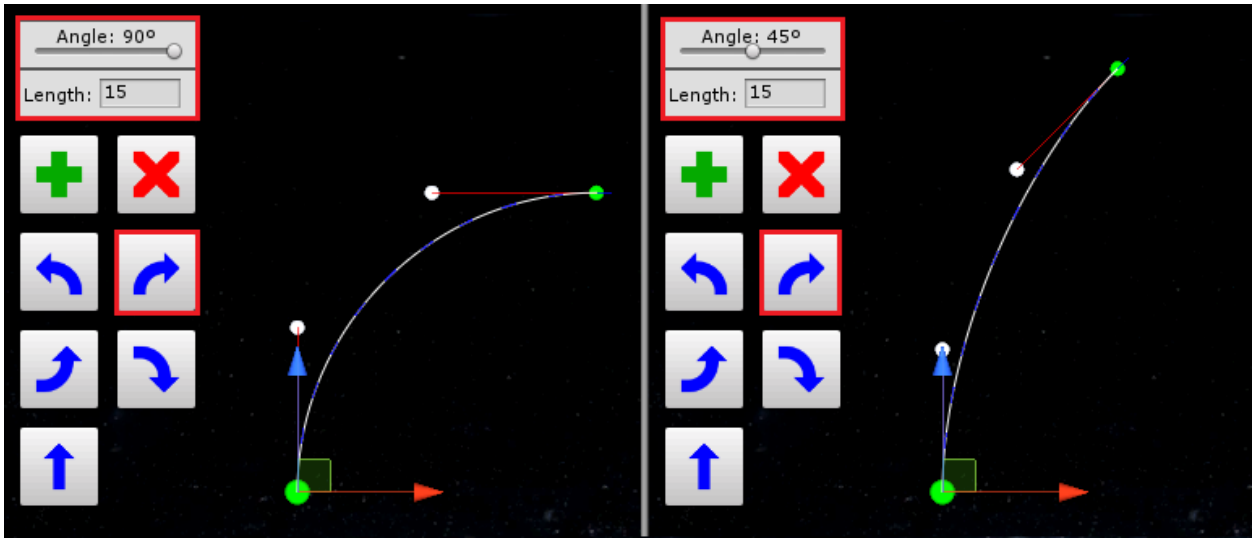

Figure 7 - Curve length and angle sample

In the sample image above, these properties were used to create a 15m long 90º curve (left) and a 15m long 45º curve (right). In order to reproduce this example, all you need to do is to adjust the length and angle values and click on the button with the “right arrow” icon in order to create a curve to the right.

The angle property is limited to 90º, in order to make it easier to create perfect circular shapes in which the bezier handles are located in the optimal position, according to bezier splines best practices. However, if you need to create curvatures greater than 90º it is possible to achieve this by using multiple curves and adding the curve angles values.

For example, if you wish to create a 60m long 180º curve, you can either use two 30m long 90º curves, four 15m long 45º curves or even three 20m long 60º curves.

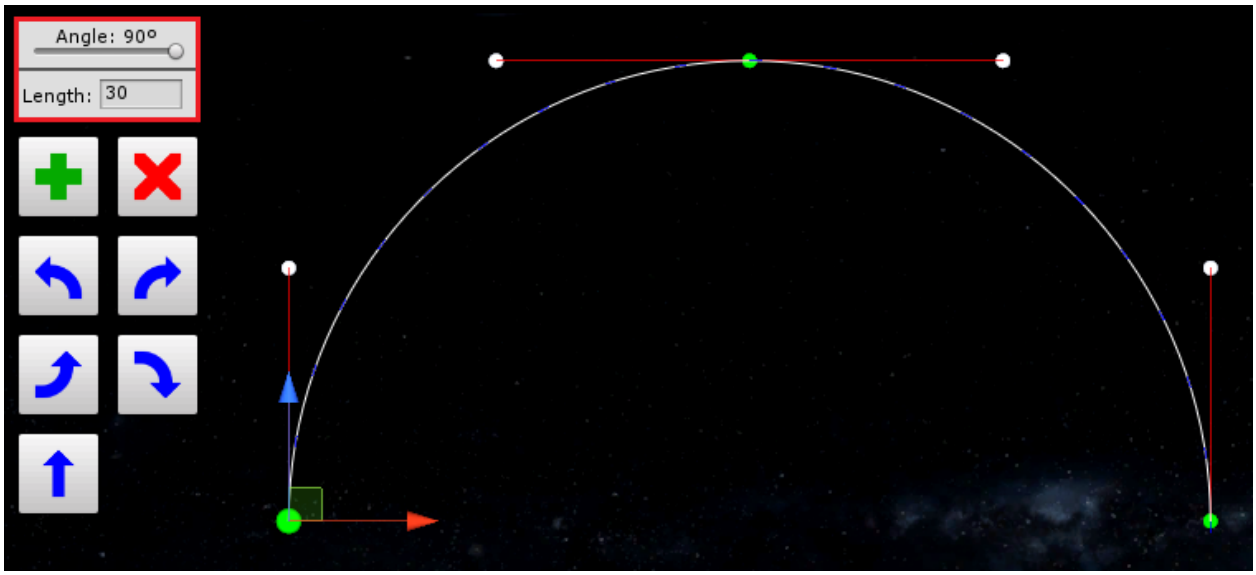

Figure 8 - 60m 180º spline composed by two 30m 90º curves

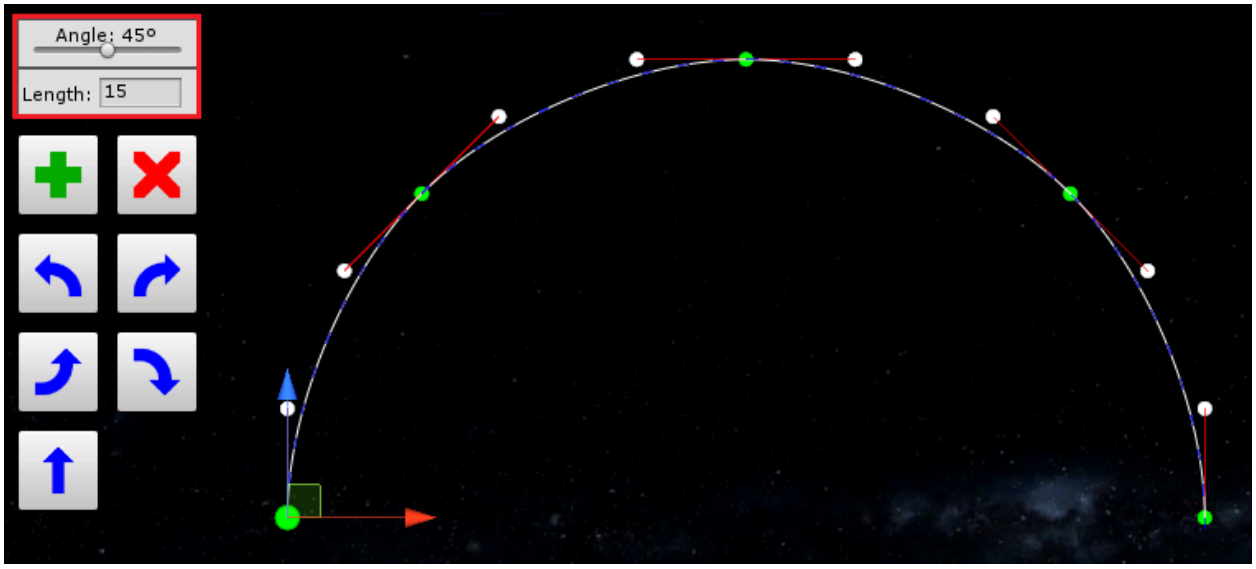

Figure 9 - 60m 180º spline composed by four 15m 45º curves

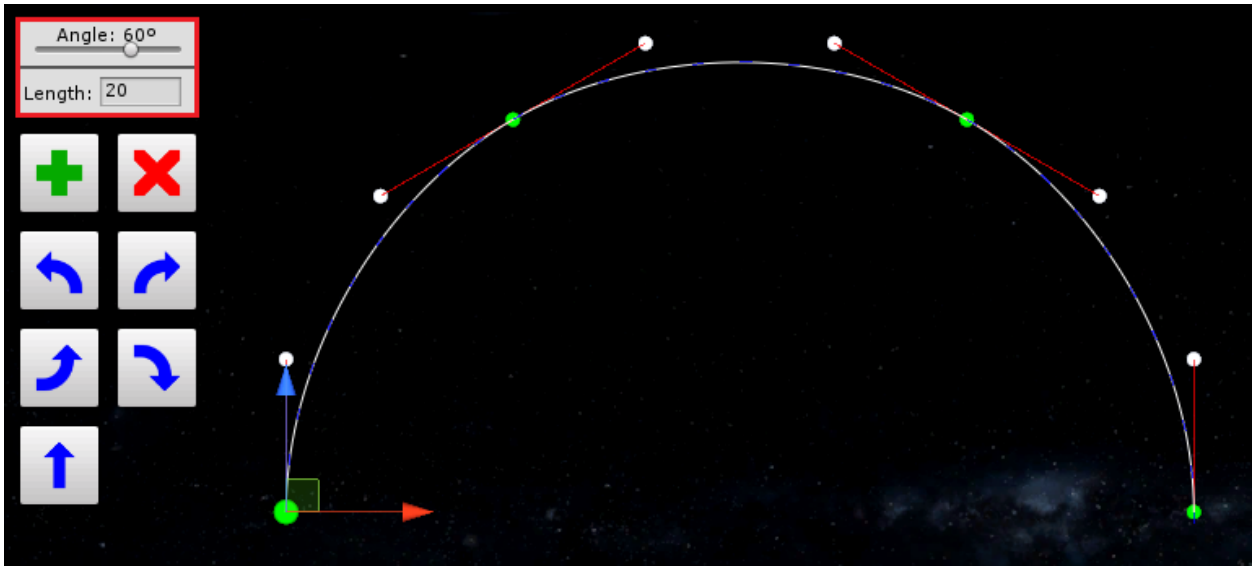

Figure 10 - 60m 180º spline composed by three 20m 60º curves

2.1.2 - Add Curve

The “Add Curve” button creates a new straight “curve” at the end of the spline, pointing in the same direction of the tip of the last curve. As mentioned before, the length of the newly added curve is defined by the “New Curve Length” property

2.1.3 - Delete Curve

The “Remove Curve” button deletes the last curve of the spline.
This operation only applies for splines composed by 2 or more curves

2.1.4 - Curve Shaping Operations

The “Left” button reshapes the last curve of the spline into a perfect curve pointing to the left. The curvature angle is defined by the "New Curve Angle" property and can be adjusted at the Quick Access Building Menu.

The “Right” button reshapes the last curve of the spline into a perfect curve pointing to the right. The curvature angle is defined by the "New Curve Angle" property and can be adjusted at the Quick Access Building Menu.

The “Up” button reshapes the last curve of the spline into a perfect curve pointing upwards. The curvature angle is defined by the "New Curve Angle" property and can be adjusted at the Quick Access Building Menu.

The “Down” button reshapes the last curve of the spline into a perfect curve pointing to down.The curvature angle is defined by the "New Curve Angle" property and can be adjusted at the Quick Access Building Menu.

The “Straight Line” operation reshapes the last curve of the spline into a straight line pointing to the direction of the first handle of the last curve. It is very useful to create straight segments pointing at any direction.

NOTE: Please take a look at the Curve Operation Examples section for practical examples on how to use this operations.

2.2 - New Curve Projection (Shift+Click)

The Quick Access Building Menu is a very powerful and precise tool, however, it is restricted by the New Curve Length and New Curve Angle properties. Of course this is very useful in any situation in which curve precision is required, but it can be a bit of a pain when you need to create irregular shapes quickly.

For example, if you're using your spline to create a path or a road on a terrain, curve irregularity would make it look much more natural then perfect curves.

By using the new curve projection feature, you can manually create curves of any length and/or curvature in your scene very quickly. All you need to do is hold shift and move the mouse around to preview the curve and right click to confirm its creation.

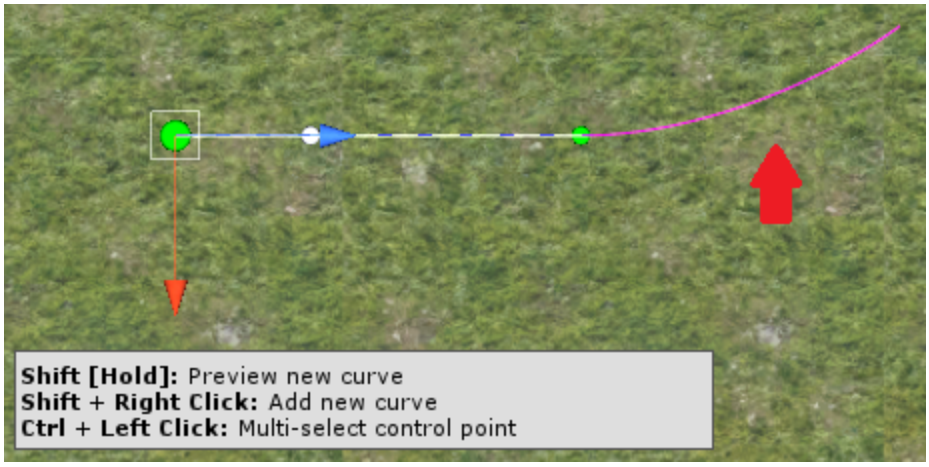

Figure 11 - New curve preview (Line in magenta)

When a spline is selected on the hierarchy, the following shortcut menu will be visible at the lower left corner of the scene view in order to remind you of this feature.

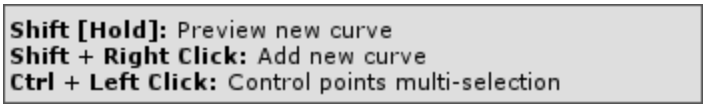

Figure 12 - Scene view shortcut menu

2.1 - Terrain Projection

When using this technique to add new curves on terrains, the spline control points will always intersect with the terrain height and a smooth curvature will be created in between the control points.

This can be very useful to create ramps, roads, paths, railroads or any other object that requires smooth vertical alignment.

NOTE: If you wish your terrain elevations to follow your spline or vice versa, please take a look at the Terraforming and Follow Terrain sections.

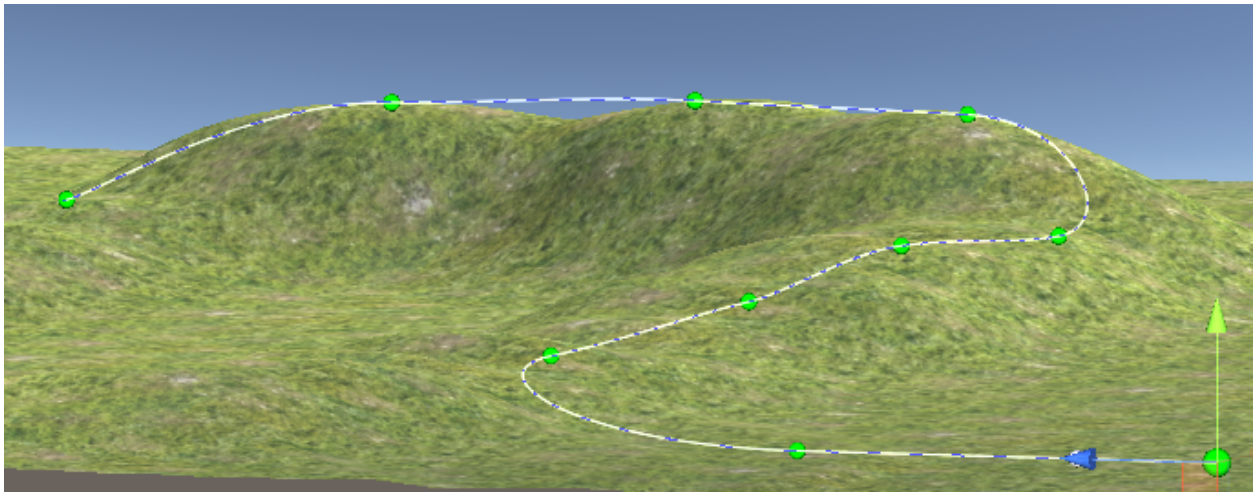

Figure 13 - Spline path following terrain elevation

When projecting into terrains, by default the spline handles will be aligned horizontally for better results.

NOTE: This feature is not restricted only to terrains, you can also project on top of planes and/or any other colliders in your scene.

2.2 - 2D Projection

If you don't have a terrain or object to project on, you can also project on "air", but in this case the projection will be made into an imaginary **horizontal** or **vertical** plane.

By default, if you're looking at your scene on a random orientation, the new curve will be projected into a **horizontal** imaginary plane.

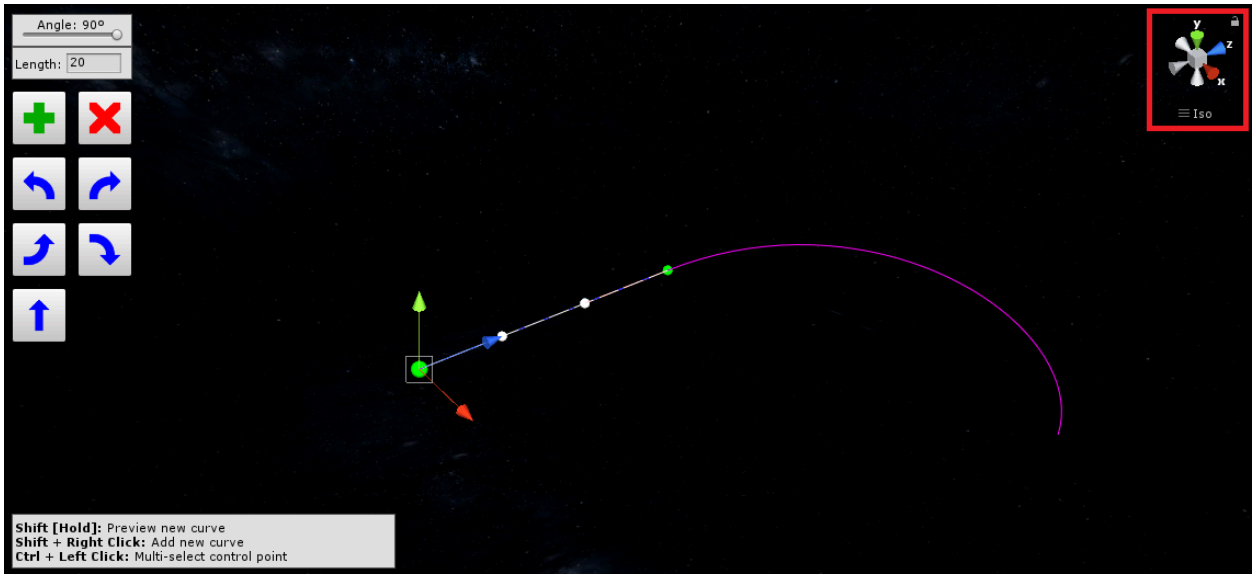

Figure 14 - default horizontal projection

However, for better results it is recommended to use the **Top Down** view when you wish to project horizontally.

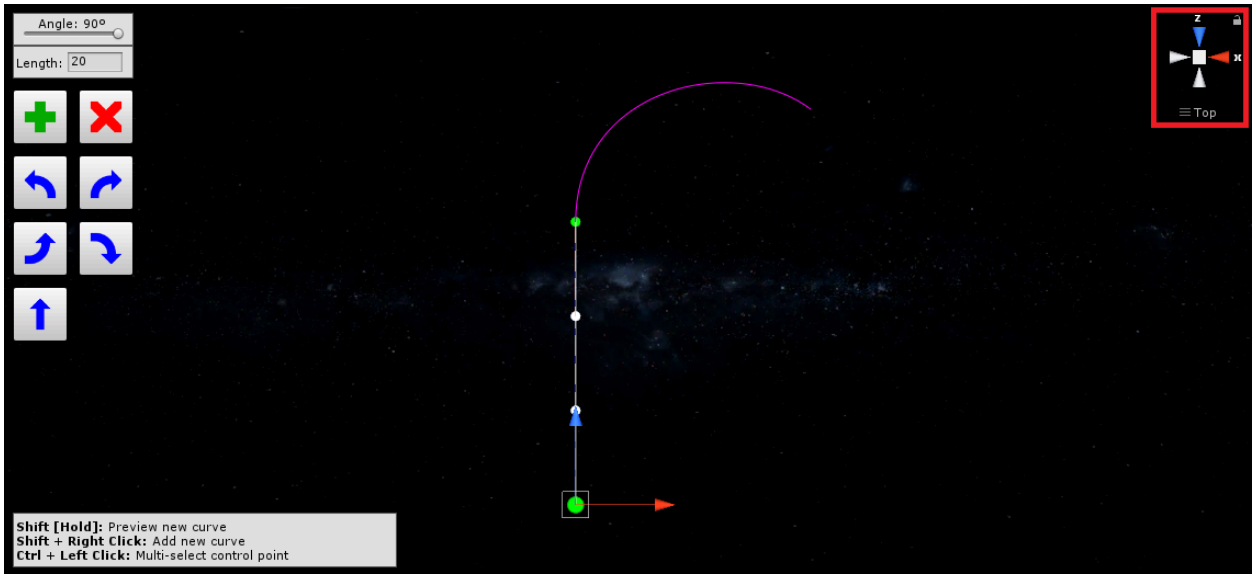

Figure 15 - Top Down horizontal projection

You can also make vertical projects, in that case you need to use the **Right View** in order to do that. Also, when working with vertical projection, make sure your spline is oriented along the Z or Y axis.

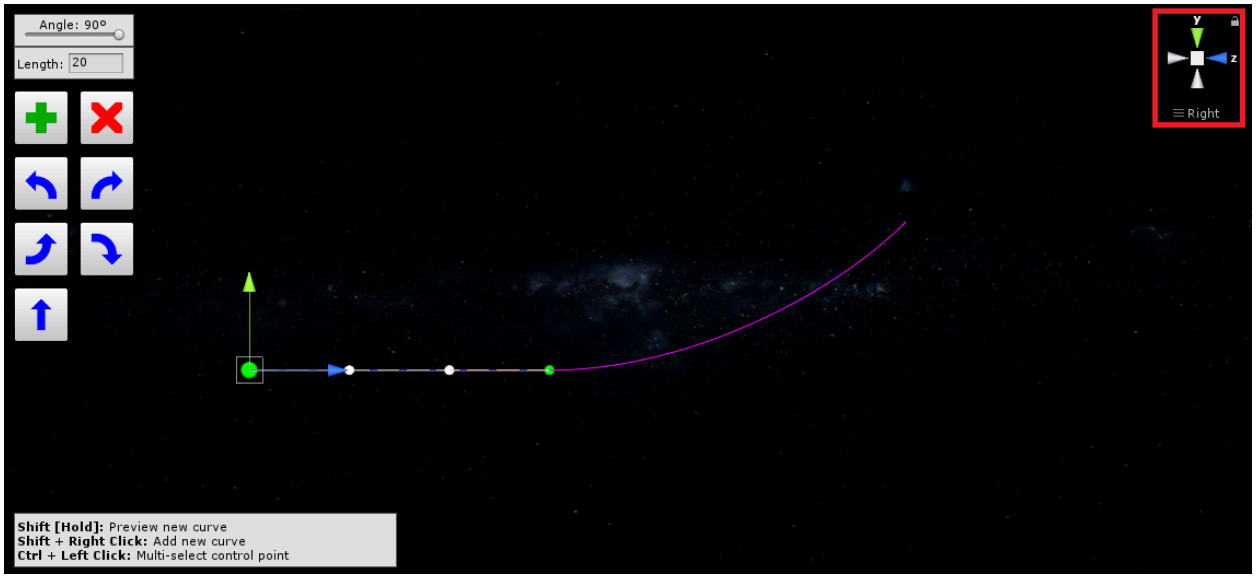

Figure 16 - Right View vertical projection

NOTE: If you are working on a 3D scene and need a vertical spline in your scene that is not oriented in the Z or Y axis, you can use the **Right View** orientation in order to create your projection, then rotate your spline transform to the desired direction once it is done.

2.3 - Manually Editing the Spline

As mentioned on the Spline section, the shape of the curve is defined by the position of the control points and handles.

2.3.1 - Single Point Editing

To manually change the position of a control point or handle, first you need to select it with a left mouse click.

When a point is selected, its position or rotation handle will appear on the Scene View, depending on what tool is selected in the Unity Editor.

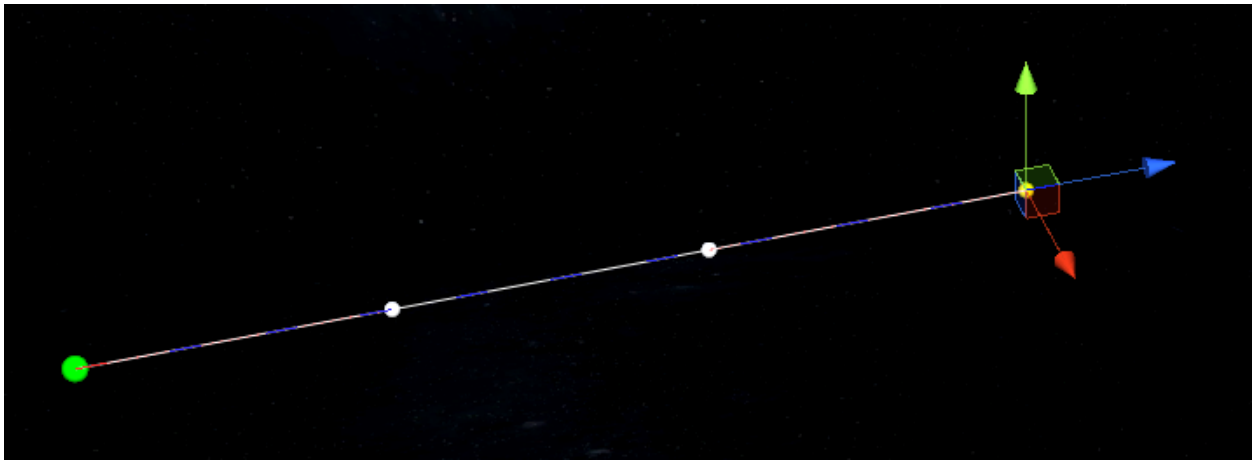

Figure 17 - Selected control point Position handle

In order to use the rotation handle, you need to select the Rotate Tool in the Unity Editor.

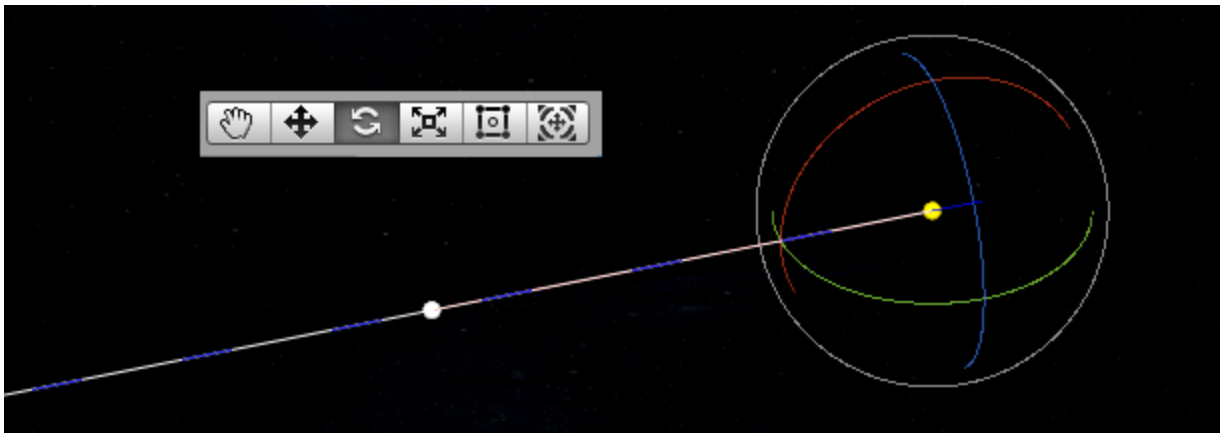

Figure 18 - Selected control point Rotation handle

Use these handles to manually adjust the position and rotation of the selected point.

By changing the position of the points, you can create any shape of curve you want.

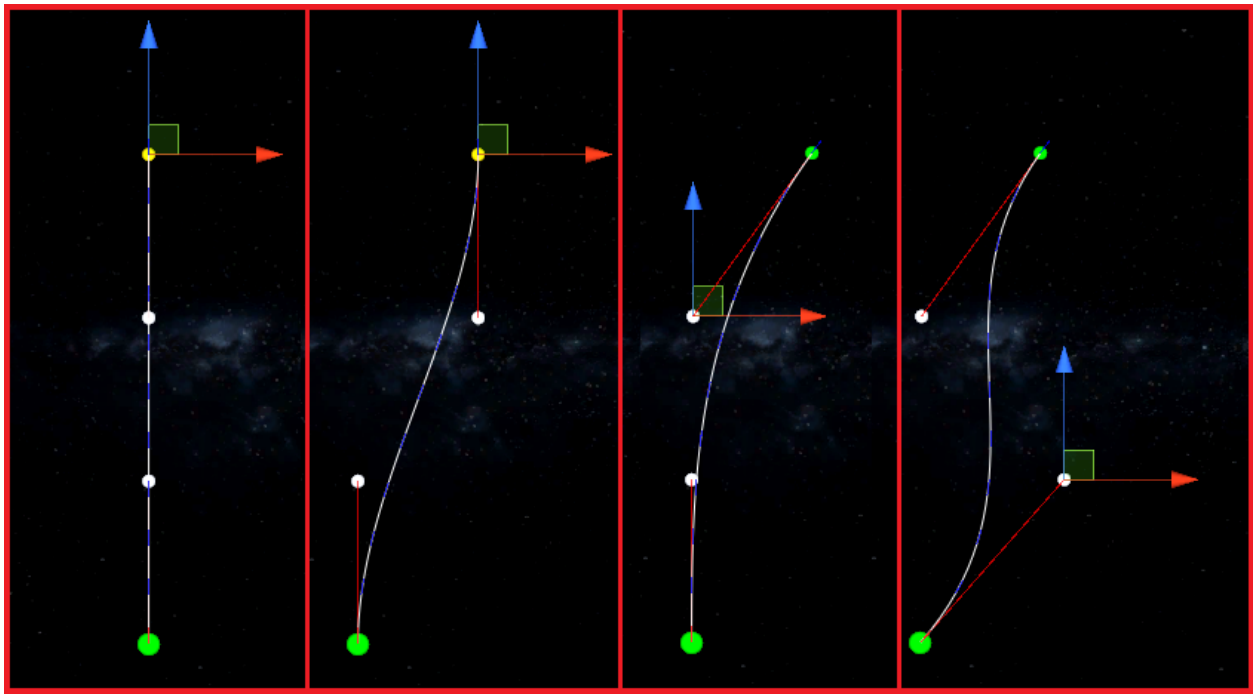

Figure 19 - Manually adjusting the curvature

The selected point position and normal values can be seen under the Handle tab of the Spline Component.

It is also possible to change these values directly on the Inspector if you wish so, this is very useful if you need mathematical precision for any purposes.

Figure 20 - Position & Normal values in the Inspector

The Normal property identifies the local upwards direction for that point, it is used as reference to automatically calculate the rotation along the spline.

NOTE: The normal of a point has different applications for each component:

- Spline Mesh Renderer
  - Applies custom rotation on a mesh segment close to the point
- Spline Follower
  - Applies custom rotation to the object movement when passing through the point
- Spline Prefab Spawner
  - Adjust the spawned object rotation for objects spawned close to the point

2.3.2 - Multi-selection

Starting with v2.1, you can now edit multiple control points at once. In order to do that, hold Ctrl and left click to select/deselect control points.

When a spline is selected on the hierarchy, the following shortcut menu will be visible at the lower left corner of the scene view in order to remind you of this feature.

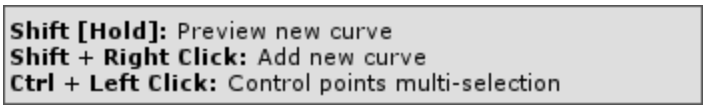

Figure 21 - Scene view shortcut menu

- **Selection Color Code**
  - **Selected**: Yellow
  - **Unselected:** Green

In the sample image below, the multi-selection was used to move points 1, 2 and 3 at the same time.

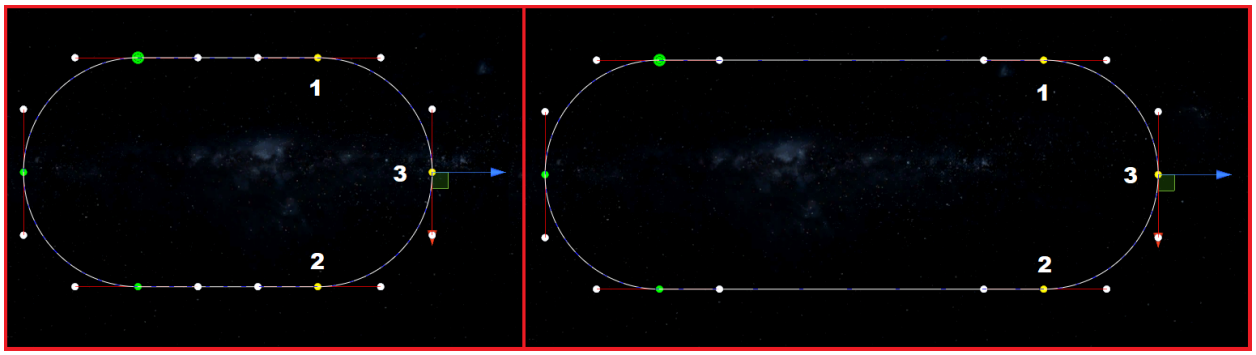

Figure 22 - Editing Multiple Control Points

NOTE: Within the context of a bezier spline there is no practical use for handles multi-selection. For this reason, handles cannot be multi-selected, only control points can (green points).

However, handles will inherit changes made to the corresponding parent control point for the sake of consistency.

Now that you know how to manually adjust the curvature, let's learn how to use the spline component features to create complex splines.

The spline component is segmented into five main categories, each corresponding to which elements of the spline are affected by its settings and operations. These elements are:

- Curve
- Spline
- Handles
- Terrain
- Scene View

2.3 - Curve Settings

All the curve related settings and operations can be found under the "Curve Settings" tab.

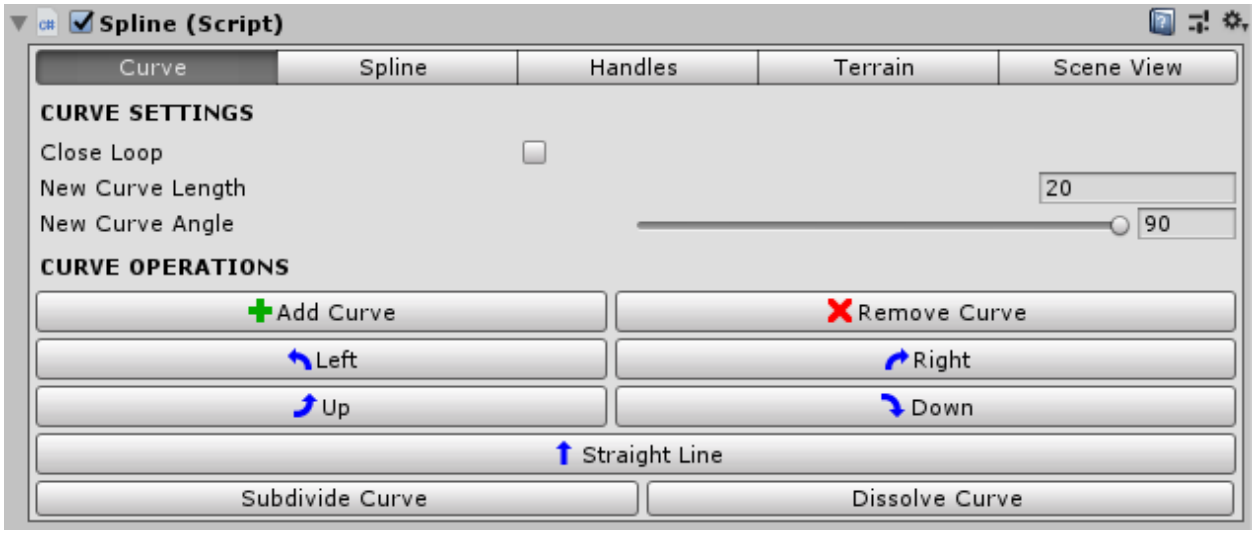

Figure 23 - Curve Settings & Operations

This settings and operations allows you to easily control the length and shape of the bezier curves that compose your spline.

2.3.1 - Close Loop

The “Close Loop” property defines if the spline consists of an open or closed path. By default, this option is disabled, when enabled, it will automatically adjust the last curve of your spline to create a closed path.

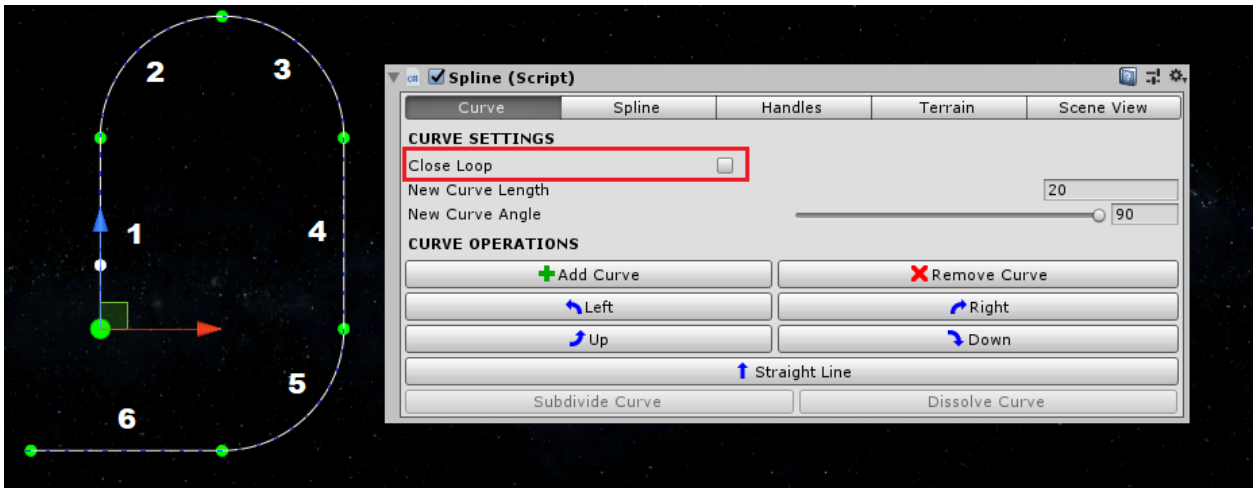

Figure 24 - Open spline path

In the sample image above, you see an open spline path composed by 6 bezier curves. Now, let’s see what happens when the “Close Loop” property is enabled.

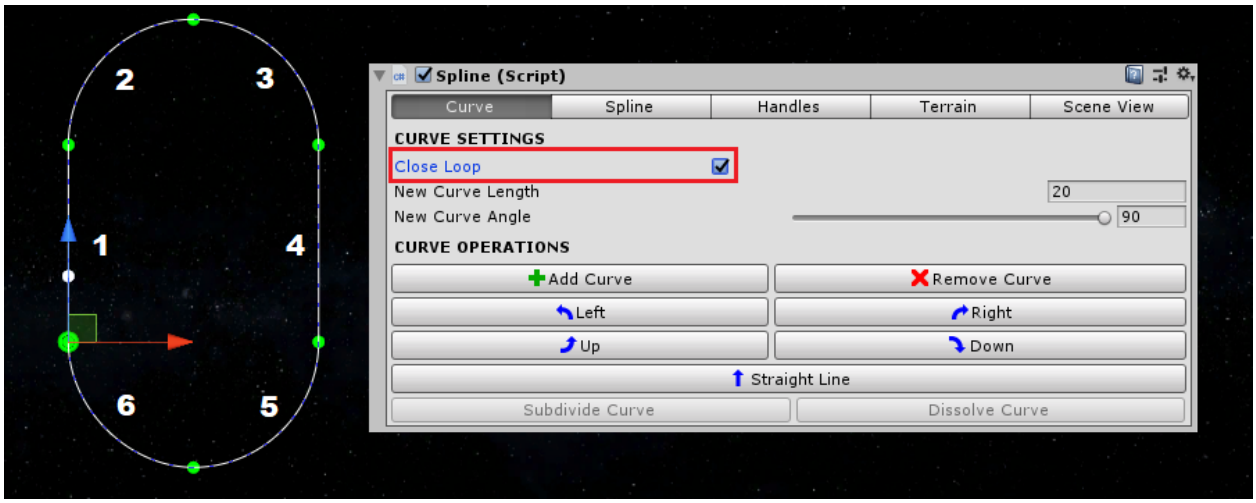

Figure 25 - Closed spline path

As you can see, the last curve (number 6), was adjusted to create a closed spline path.

NOTE: When the Close Loop property is disabled, the last curve will be disconnected and reseted into a straight line.

2.3.2 - New Curve Length

The “New Curve Length” property defines the length of newly created curves and curves shaped using any of the predefined curves operations.

2.3.3 - New Curve Angle

The “New Curve Angle” property defines the curvature angle of curves shaped using any of the predefined curves operations.

> **NOTE:** The New Curve Length and New Curve Angle properties are used to create mathematically precise curves. They can also be adjusted on the Quick Access Building Menu to speed up the spline building workflow.

2.4 - Curve Operations

There are nine curve operations in total. These operations can be separated in three groups:

- **Curve Creation**
  - Add Curve
  - Subdivide Curve
- **Curve Removal**
  - Remove Curve
  - Dissolve Curve
- **Curve Shaping**
  - Left
  - Right
  - Up
  - Down
  - Straight Line

The curve operations can be found under the “Curve Settings” tab. The most used operations are also available on the top left corner of the Scene View window of the Unity Editor for quick access.

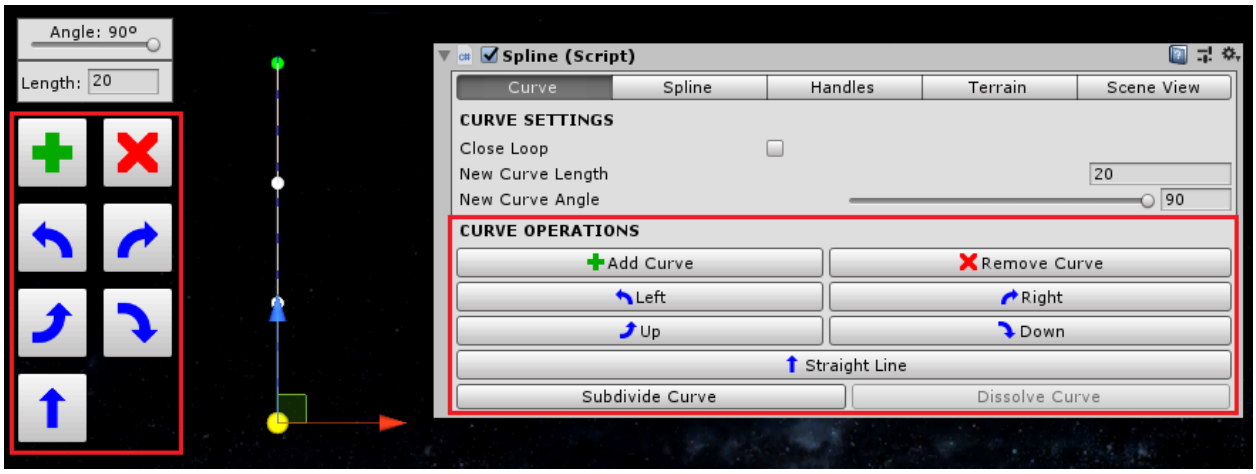

Figure 26 - Curve operations

NOTE: All the quick access operations available on the Scene View are mapped to the corresponding operation on the Unity Inspector by icons. Only quick access operations buttons have icons assigned to them. For more information about these operations, please take a look at the Quick Access Building Menu section.

2.4.1 - Subdivide Curve

The “Subdivide Curve” operation divides the selected curve in half, creating two distinct curves. To select a curve, just select the first control point of the curve. It is very useful to create new curves on the middle of the spline.

2.4.2 - Dissolve Curve

The “Dissolve Curve” operation dissolves the selected control points, transforming two curves into one. It is very useful to remove curves in the middle of the spline.

Now, let’s take a look at some examples.

2.4.3 - Curve Operation Examples

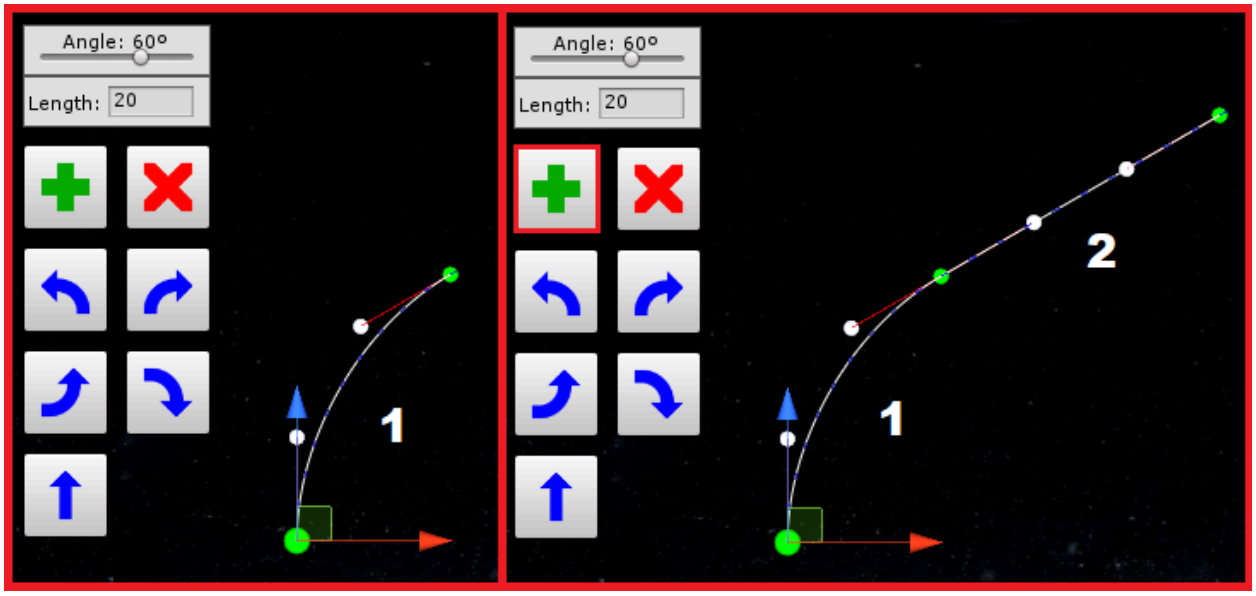

Figure 27 - “Add Curve” operation example

On the left side of the sample image above, you can see the spline before the "Add Curve" operation, on the right side you can see the spline after the operation. Note how the newly added curve (2) follows the direction from the previous curve (1).

By inference, we can affirm that by using the "Remove Curve" operation at this point, would revert the spline to the previous stage by removing the last curve (2).

By using the “Straight Line” operation we can reset the curve shape to a line pointing at any direction. This can be very useful to build ramps for example.

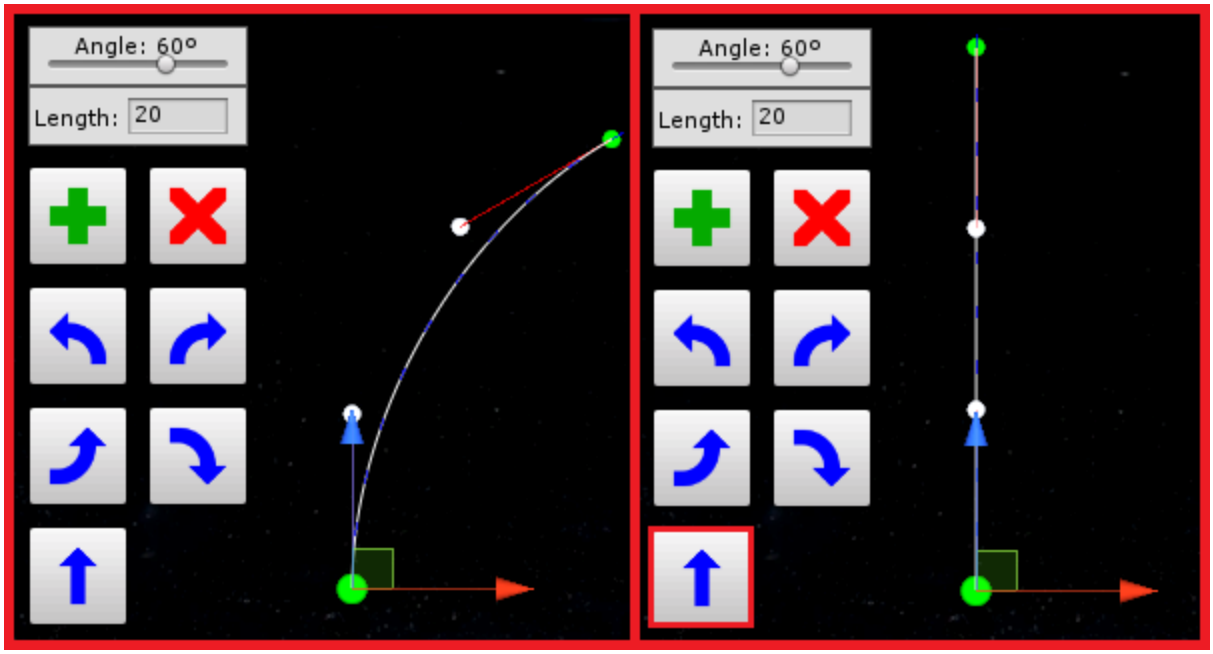

Figure 28 - “Straight Line” operation example

NOTE: All curve shaping operations are affected by the “New Curve Length” property. If the length is set to 15 for example, the curve length will be 15 meters after reshaping.

If you wish to create a perfectly round curve, you can use the predefined curves operations (right, left, up and down). They will convert the last curve on a perfect circle section of the desired angle. In the sample image below, a perfect 90º curve was created with a single button click.

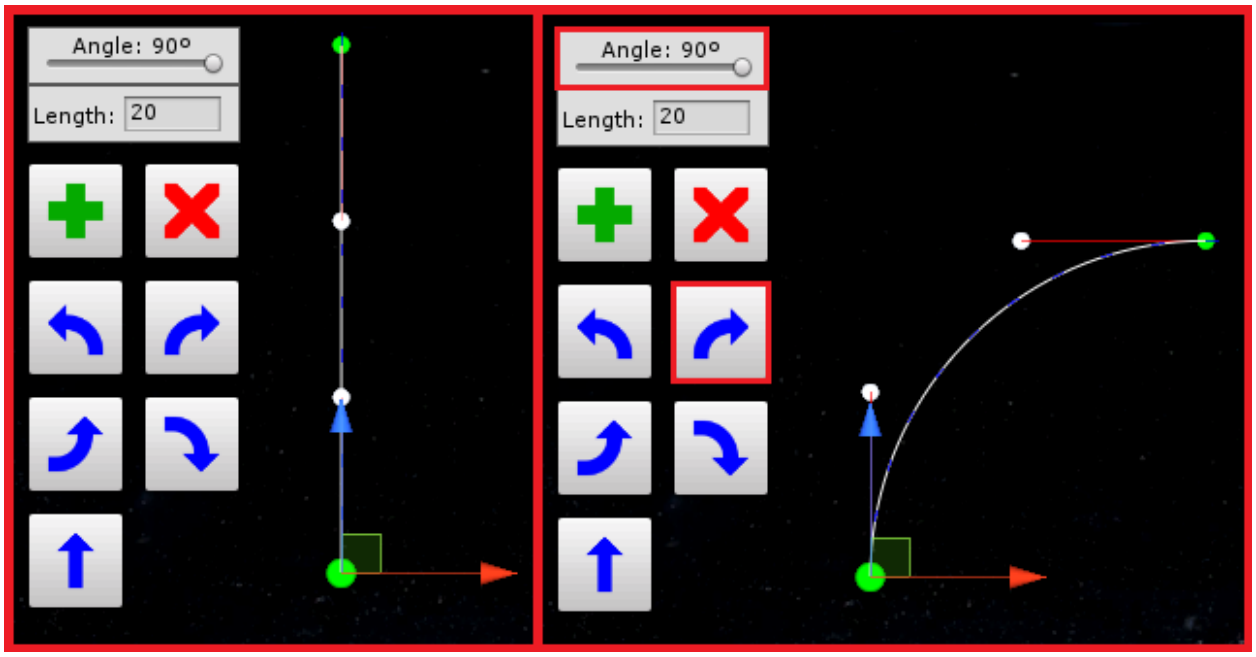

Figure 29 - 90º curve to the right example

The final curvature direction is based on the curve start direction. This allows you to create perfect curves in any direction.

If your camera is not aligned with the curve, at first it can be a little confusing to predict the curvature direction, but everything will make sense once you get used to these operations.

A good technique to understand this concept faster, is to always imagine that your camera is pointing in the same direction of the curve starting point, this way all arrow icons will be pointing to their respective curvature direction.

Alternatively, you can also look at your spline from the top while using the Left and Right operations and look at your spline from the right view while using the Up and Down operations.

For example, if you are looking at your spline from the top view, by combining the “Add Curve” and the “Right” curve shaping operation, you can create a perfect circle, as can be seen on the sample image below.

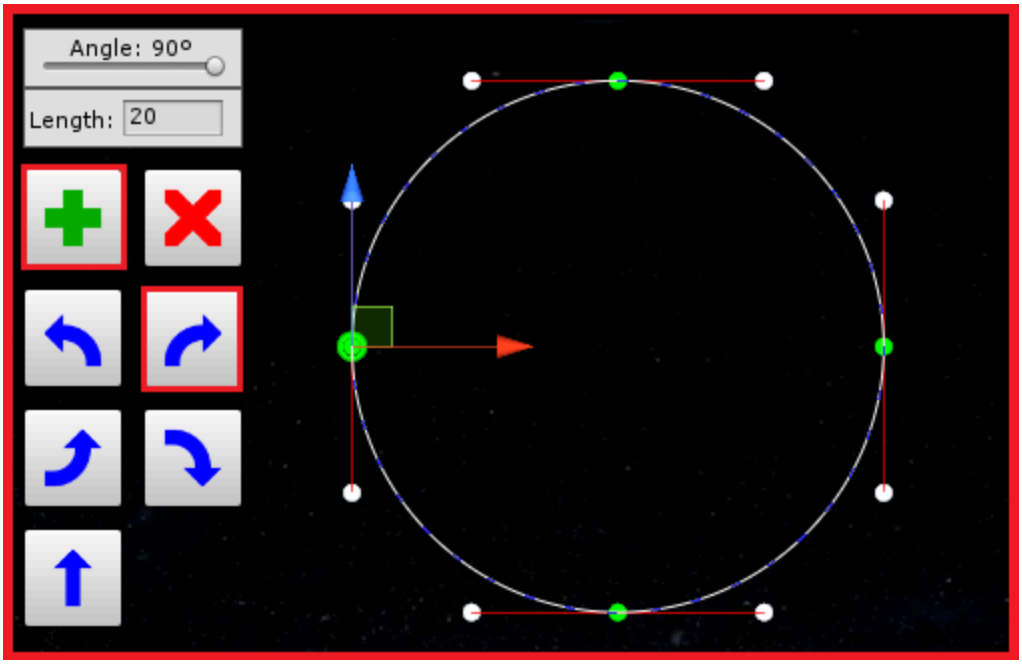

Figure 30 - Perfect circle create using “Add Curve” and “Right” curve operations

Now, let’s take a look at the “Subdivide” and “Dissolve” curve operations. Take a look at the sample image below, this spline is composed of three bezier curves. Imagine that you wish to split curve number 2 into two new curves, so you can add more detailed curvature to this segment of your spline.

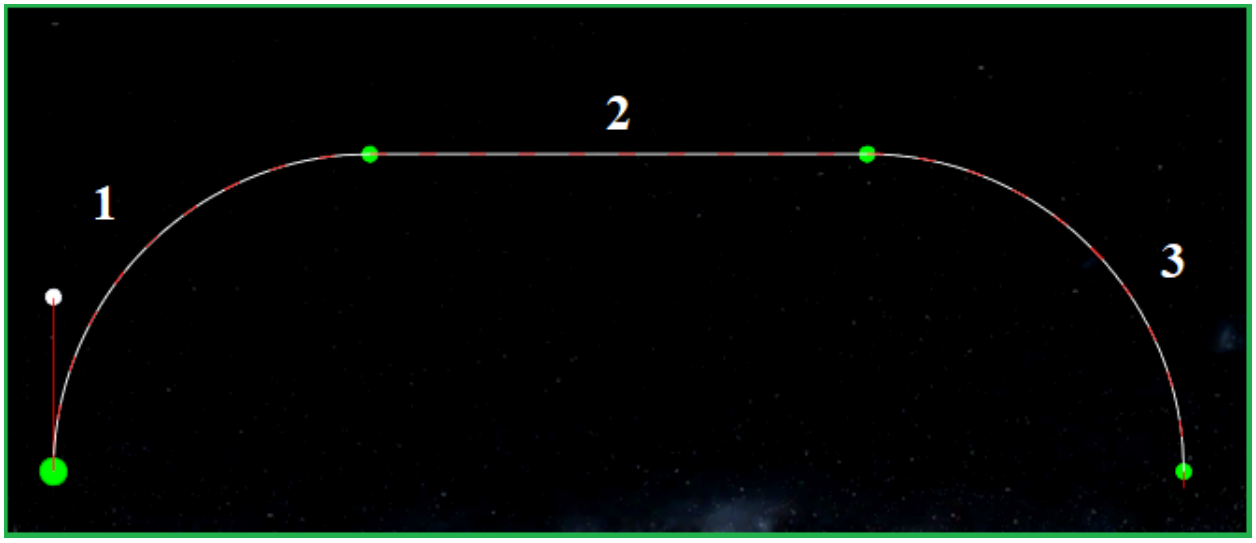

Figure 31 - Before subdividing

This can be achieved by selecting the control point at the start of curve number 2 and clicking on the “Subdivide” button on the inspector.

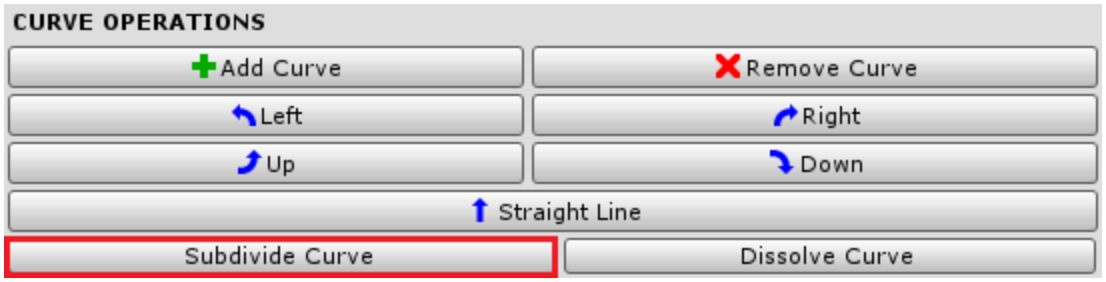

Figure 32 - Subdivide Operation

After the subdivision has taken place, this is how your spline will look like.

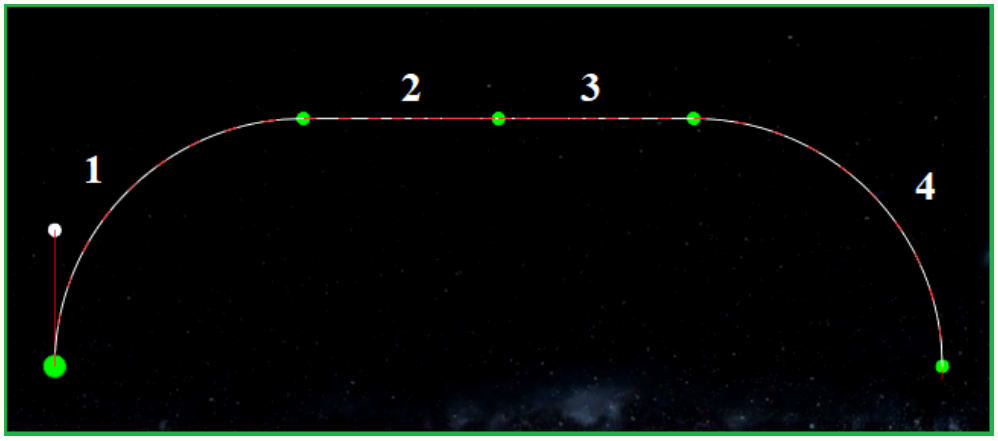

Figure 33 - Subdivided spline

Now that you have created your extra curve, you can adjust its position to create the desired shape.

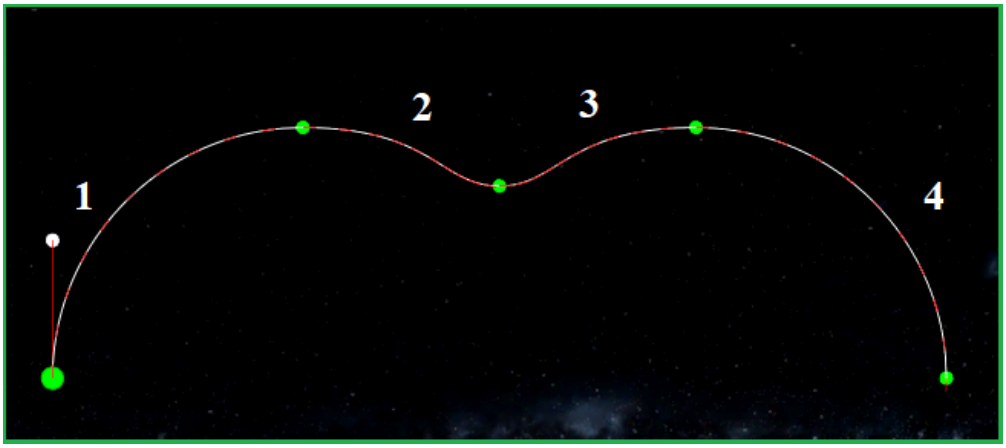

Figure 34 - Practical application of subdivision

If you change your mind and wish to remove the extra curve, you can use the “Dissolve” curve operation to get rid of it. To do that, simply select the control point that you want to dissolve and click on the “Dissolve” Button on the Inspector.

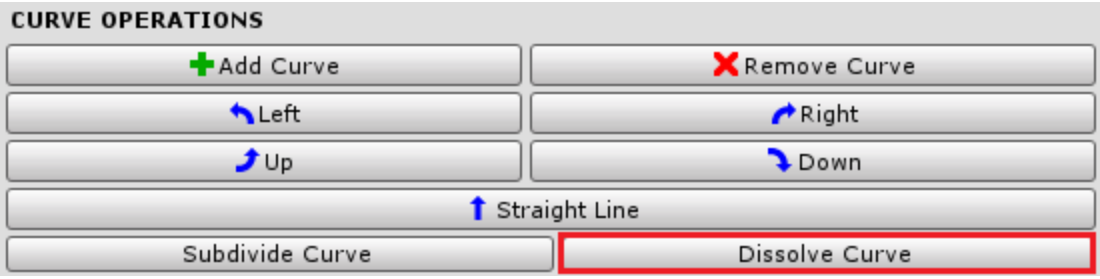

Figure 35 - Dissolve Operation

When a control point is dissolved, it will automatically connect the neighbors control points, merging two curves into a single one.

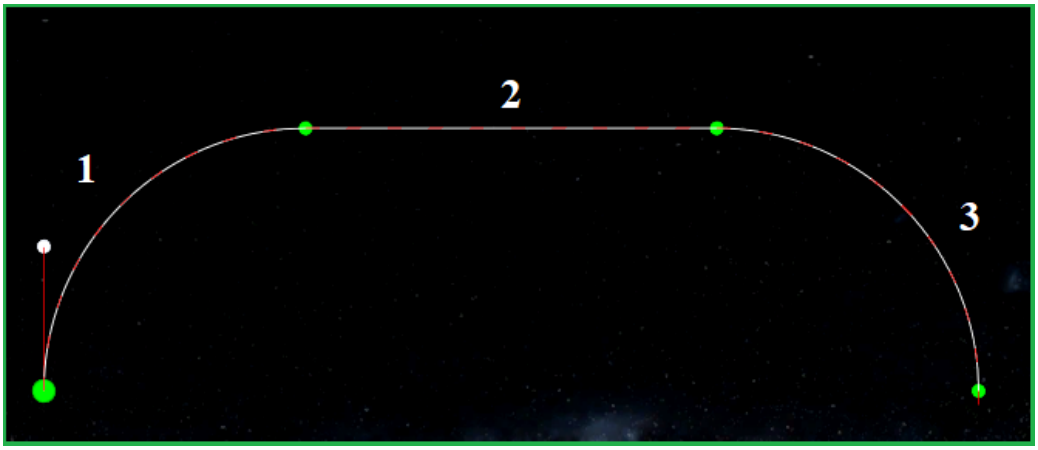

Figure 36 - After dissolving

That covers all the curve related settings and operations. Now, let's take a look at the Spline Settings tab.

2.5 - Spline Settings

All settings and operations that affect the entire spline at once are located under the "Spline Settings" tab.

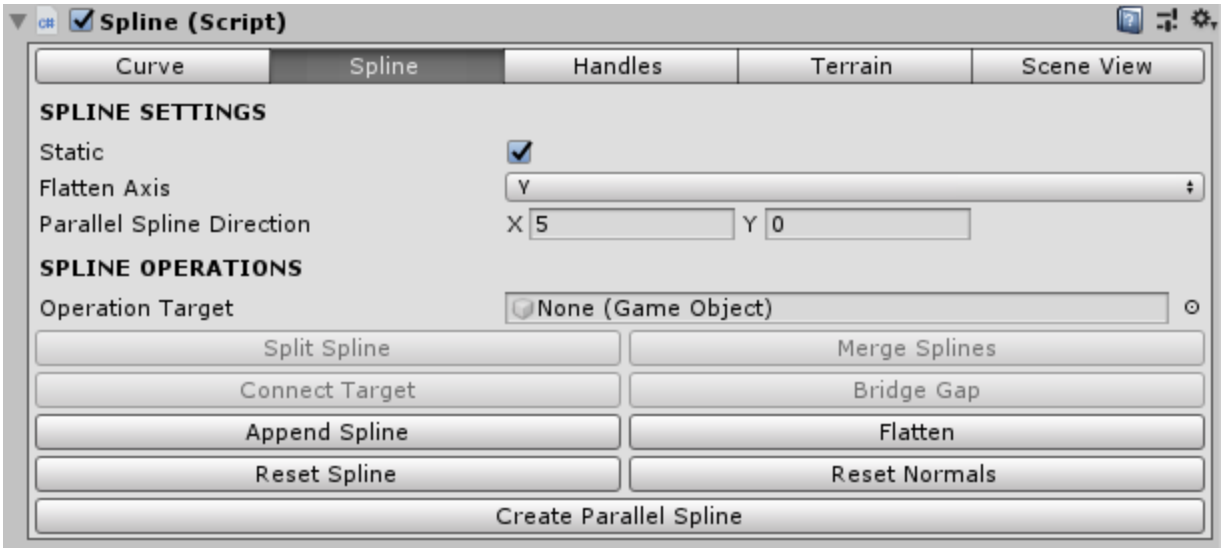

Figure 37 - Spline Settings & Operations

For a better understanding, all practical examples in this section were built using the “Spline Mesh Renderer” prefab (for more information please take a look at the Spline Mesh Renderer section).

2.5.1 - Static Spline

By default, all splines are set as “Static”, which means they will not make any unnecessary calculations at runtime for the sake of performance optimization. Therefore, if your spline will not move at runtime, keep the Static property enabled for better results.

However, if your spline needs to move at runtime. Uncheck the Static property in order to make sure all spline related components will recalculate properly.

For example, at the “Follower_SolarSystem” demo scene, the Moon Orbit Spline inherits the earth translation around the sun, therefore it is moving on the scene. In order for the Moon spline follower to calculate the following route correctly, Moon Orbit Spline cannot be static.

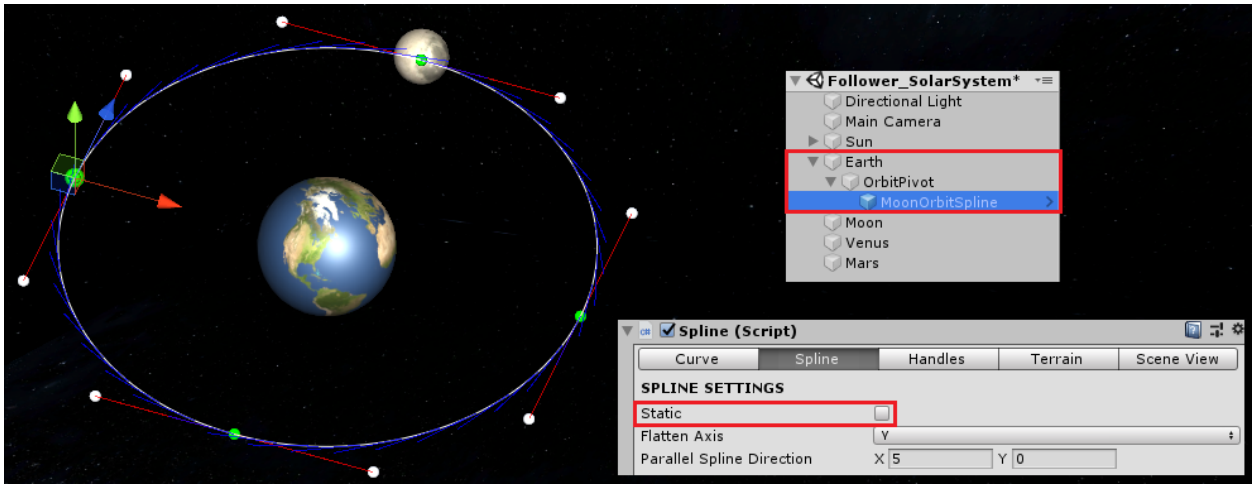

Figure 38 - Non-static spline sample

2.5.2 - Flatten Axis

The Flatten Axis property is used as reference by the Flatten operation, to identify in which world axis the spline needs to be flatten. By default it is set to the Y axis.

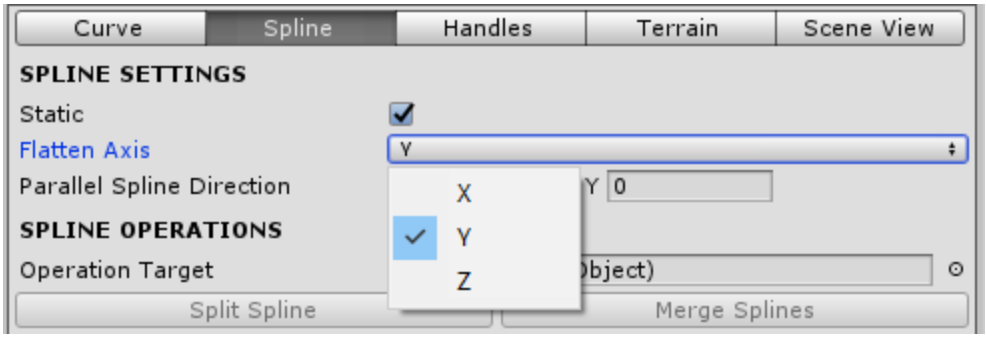

Figure 39 - Flatten axis

2.5.3 - Parallel Spline Direction

The Parallel Spline Direction property is used as reference by the Create Parallel Spline operation.

The X axis is relative to control points local horizontal directions, a positive X value means the corresponding parallel points will be created to the local right direction of the selected splines points. Negative X value means they will be created to the local left.

The Y axis is relative to the spline local vertical directions. A positive Y value means the parallel spline will be created above the current spline and a negative Y value means it will be created below the current spline.

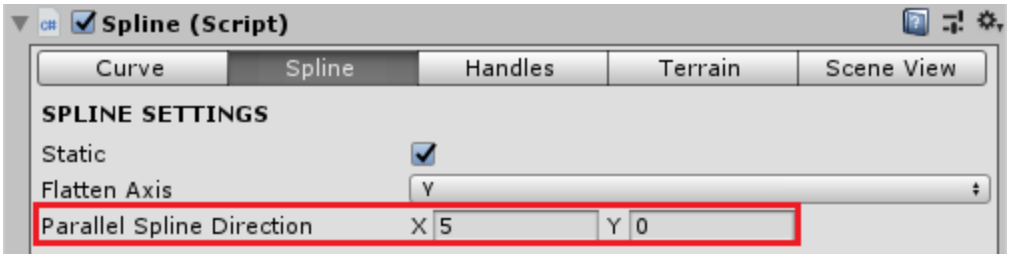

Figure 40 - Parallel Spline Direction

2.5.4 - Operation Target

Some Splines Operations affect or use other splines and/or game objects as reference. In order to use these operations, you need to use the Operation Target property to designate the desired target.

Currently, this property is used by the Merge Splines, Connect Target and Bridge Gap operations.

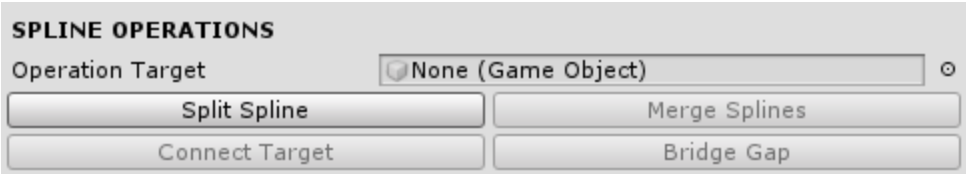

Figure 41 - Operation target

NOTE: If a target is not selected, these operations will be disabled on the Inspector.

2.6 - Spline Operations

2.6.1 - Split Spline

The “Split Spline” operation is used to cut one spline into two splines. This operation instantiates a new object on your scene that inherits all the characteristics and components from the original object. The newly created spline starts at the cutting point, connected at the new end point of the previous spline.

This operation is very useful for removing segments of a spline, or reducing the size of long objects to apply Occlusion Culling for example.

2.6.2 - Merge Splines

The “Merge Splines” operation is the opposite of the Split Spline operation. By using this operation you can turn two splines into one.

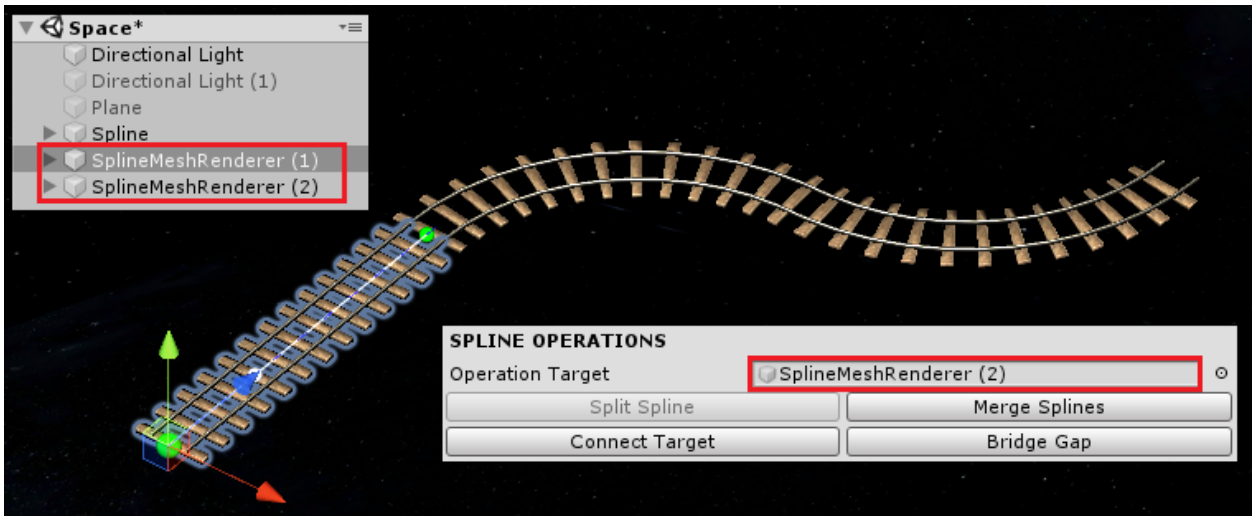

Figure 42 - Before Merging Sample

In the sample image above, the merging operation is being used to merge two railroad segments created using the Spline Mesh Renderer component into a single railroad segment.

In order to do that, the “SplineMeshRenderer (2)” was set as the operation target at the Spline component of the “SplineMeshRenderer (1)”.

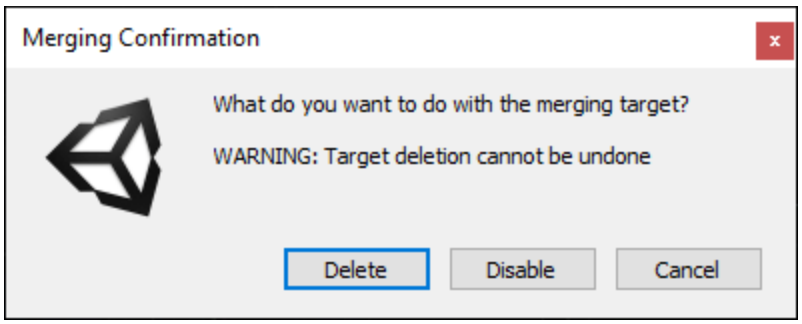

Figure 43 - Merging confirmation

Once you click at the “Merge Splines” button, the confirmation dialog above will be displayed in the Unity Editor. You can either choose to delete or disable the target object. In this example, the delete option was selected, therefore only the “SplineMeshRenderer (1)” object remains at the scene after merging as can be seen at the sample image below.

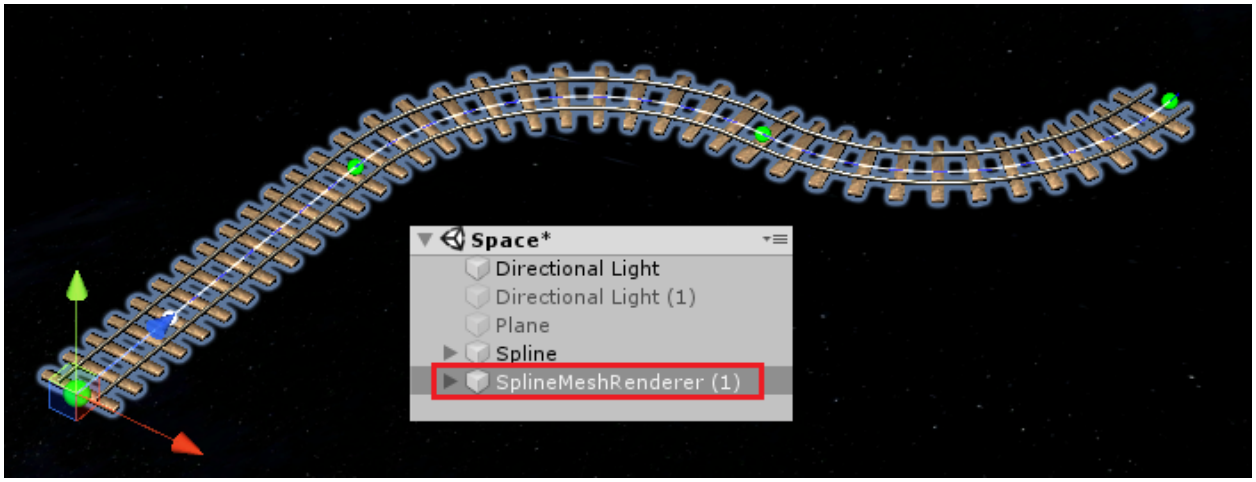

Figure 44 - After Merging Sample

In the sample above, the target spline was perfectly positioned at the end of the first spline. But, this may not be always the case. Eventually, you may wish to merge splines that are not connected to each other.

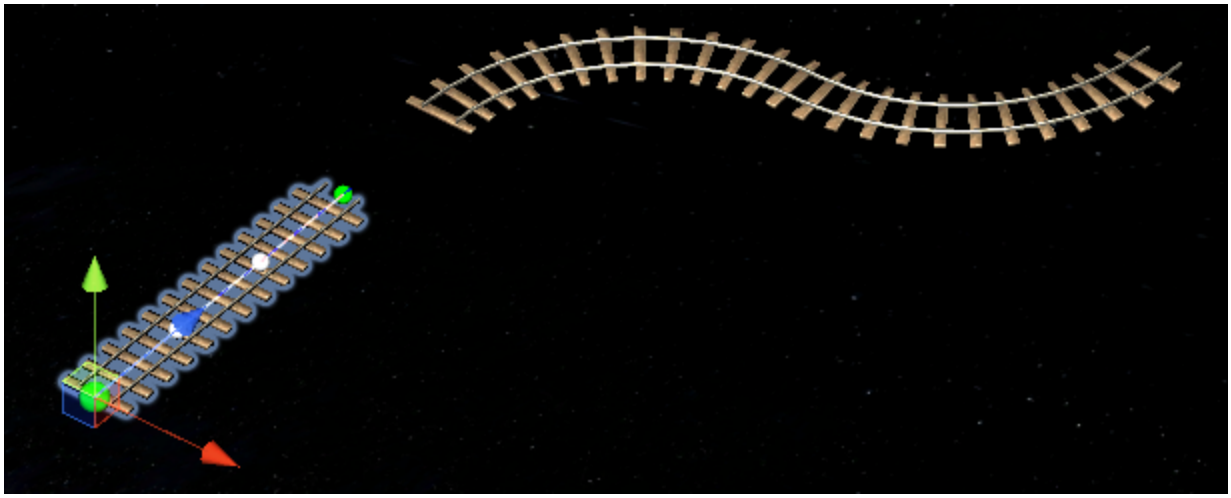

Figure 45 - Merging Disconnected Splines (before)

When merging disconnected splines, the gap between the splines will be closed automatically.

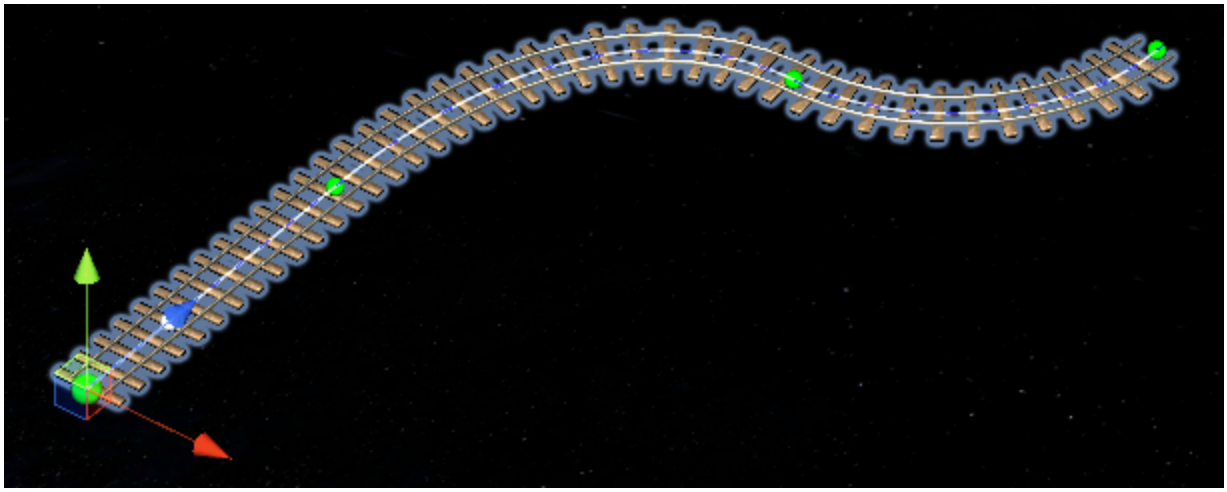

Figure 46 - Merging Disconnected Splines (after)

NOTE: The merging feature will make the best it can to preserve the curve shape of disconnected splines, if you don't want any distortion on the resulting spline you can either move the target spline position by using the Connect Target operation or close the gap prior to merging using the Bridge Gap operation.

2.6.3 - Connect Target

In some situations, you may need to reposition an object on your scene to the end of a spline. You can use the "Connect Target" operation in order to do that.

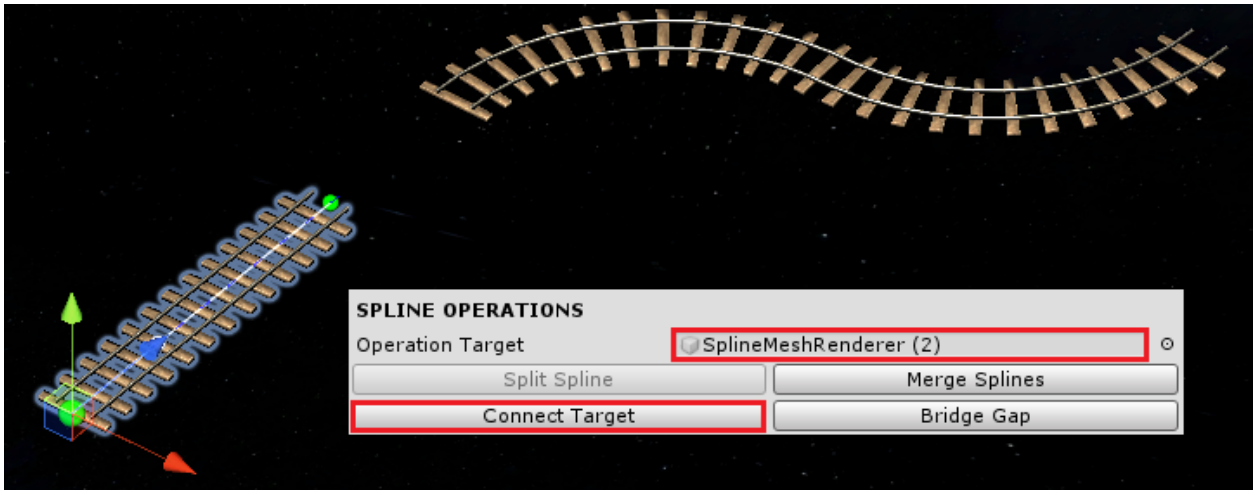

Figure 47 - Connecting target (before)

In the following example, the connect operation was used to move the "SplineMeshRenderer (2)" to the end of the selected spline, in order to align both railroad segments.

Figure 48 - Connecting target (after)

NOTE: This operation is very useful for level design, as it allows you to connect not only splines but any object to the end of your spline.

2.6.4 - Bridge Gap

Consider the following situation, you have a gap between your spline and another spline or game object in your scene and you want to close it using a new curve.

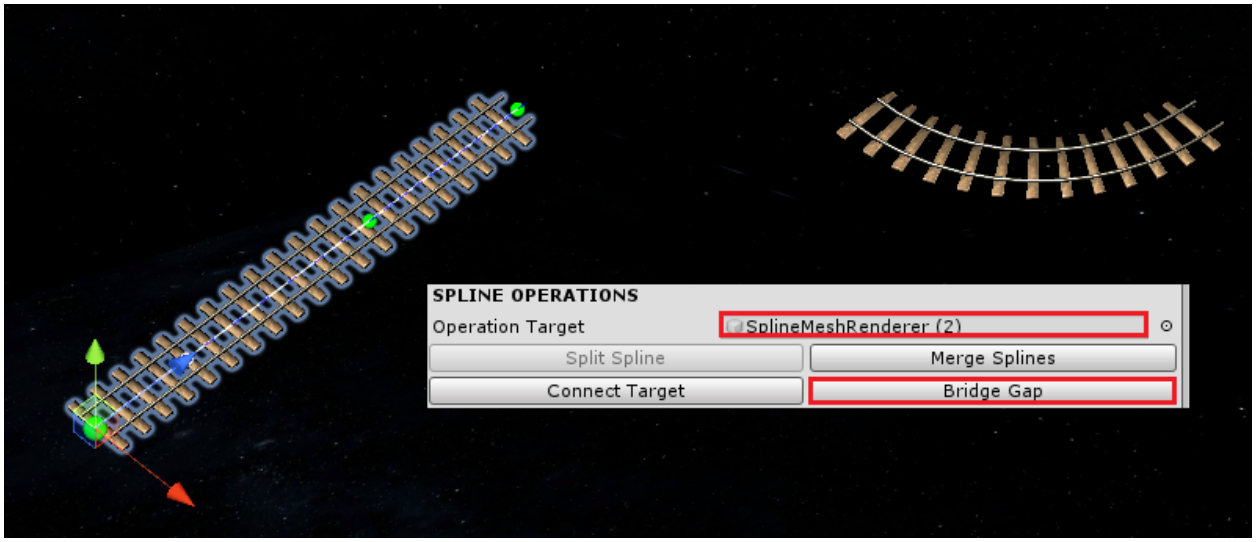

Figure 49 - Bridge gap operation (before)

In order to do this, you can use the bridge gap operation. It will create a new curve and automatically adjust the last point position to close the gap between the objects.

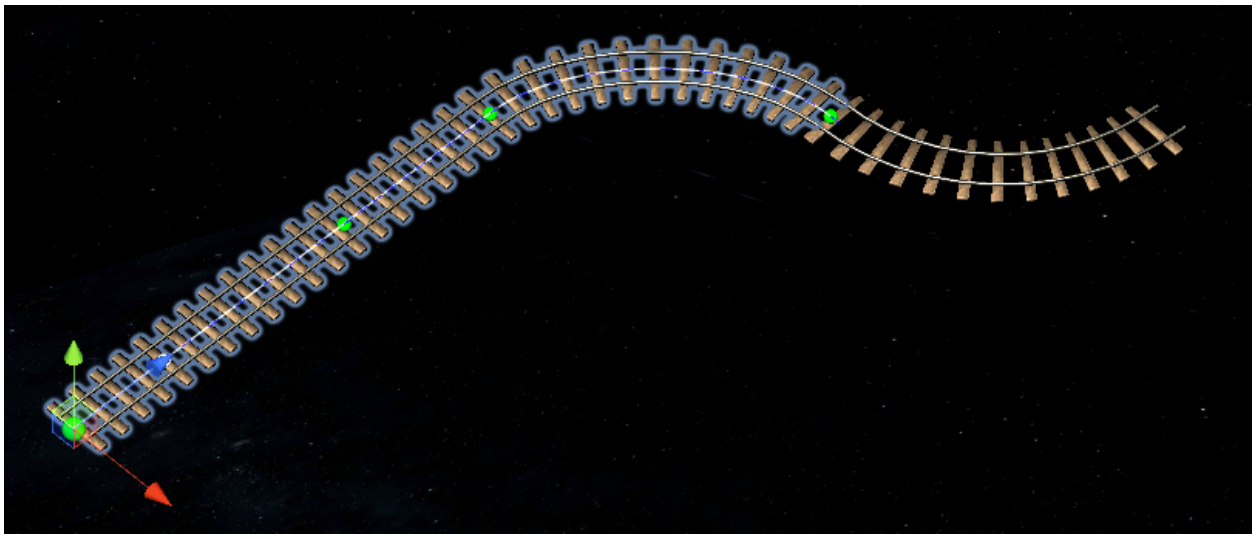

Figure 50 - Bridge gap operation (after)

2.6.5 - Append Spline

The “Append Spline” operation creates a new spline at the end of the current selected one. The newly created spline inherits all the characteristics and components from the original object. However, the new spline is generated with only one straight curve segment.

This operation is useful to create complex paths composed by multiple splines.

2.6.6 - Flatten

The “Flatten” operation sets all control points and handles at the same level of the first control point of the spline, transforming a 3D into a 2D spline. The spline dimension that will be “removed” when using this operation is defined by the Flatten Axis property.

> **NOTE:** The Flatten spline feature is a legacy feature from v1.0 that was originally designed as a quick way to undo the old Follow Terrain feature that was deprecated.
>
> On version 2.0, the Follow Terrain feature was completely re-written for better results. The Flatten operation was kept, since it still can be useful in some situations.
>
> On version 2.1, the Flatten Axis property was introduced in order to extend this feature usefulness to converting 3D splines into 2D splines.

2.6.7 - Reset Normals

The “Reset Normals” operation removes custom “rotation” of all control points and handles. It is very useful if you have rotated any points by mistake or wish to remove all custom rotations.

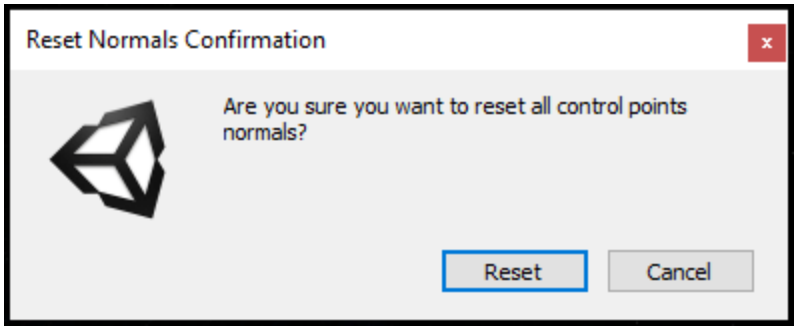

Figure 51 - Reset normals confirmation dialog

2.6.8 - Reset Spline

The “Reset Spline” operation completely resets the spline to its initial state: only one straight curve segment. This operation is useful if you want to recreate your spline from scratch.

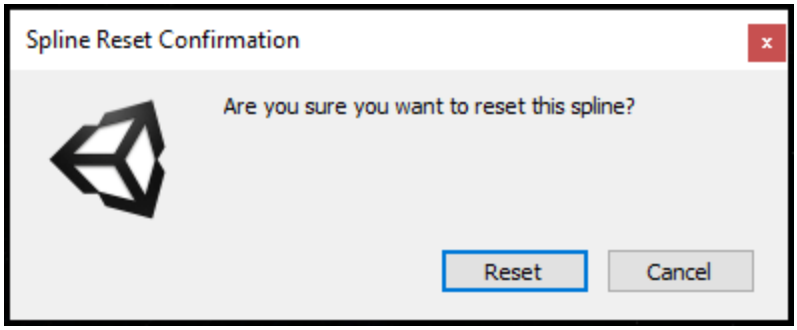

Figure 52 - Reset spline confirmation dialog

2.6.9 - Create Parallel Spline

If you need to parallel splines on your scene for any purposes, you can use the “Create Parallel Spline” operation instead of manually building another spline.

All you need to do is to feed the desired direction and distance on the Parallel Spline Direction property and click the “Create Parallel Spline” button.

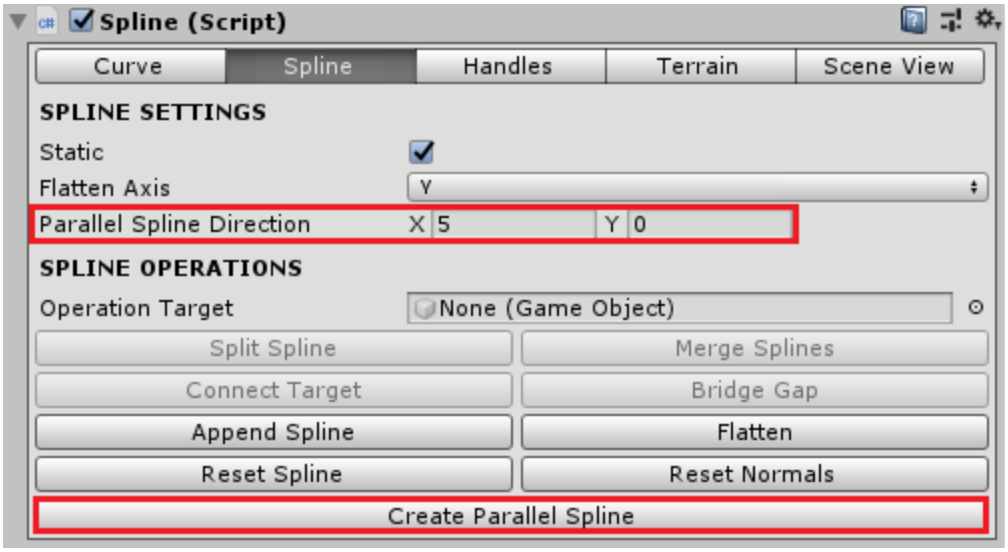

Figure 53 - Parallel spline settings

In the sample below, this operation was used to create a parallel spline 5 meters to the right of the original spline.

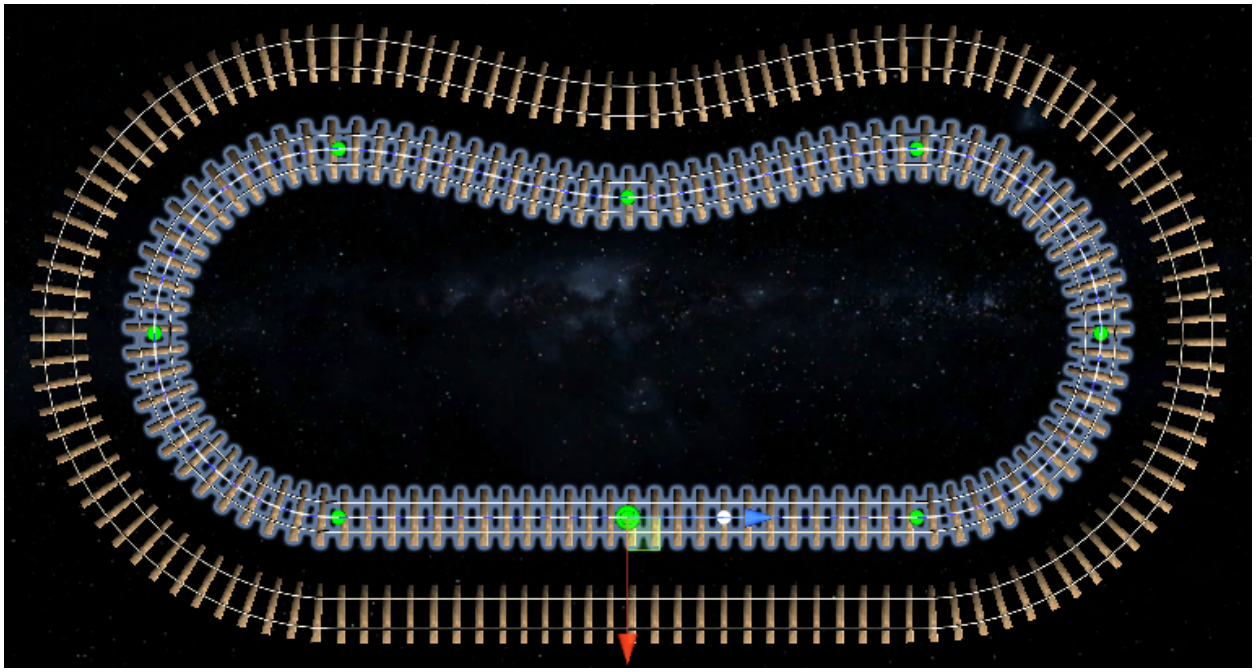

Figure 54 - Parallel spline creation sample

NOTE: In order to identify the local horizontal axis to create the parallel spline, this feature makes use of the control points normals. If you have any distortion while creating a parallel spline, adjust the control points normals and try again.

2.7 - Handles Settings

All settings and operations that affect the spline control points and handles are located under the “Handle” tab.

Figure 55 - Handle Settings

2.7.1 - Handles Visibility

The “Handles Visibility” property is used to hide/show handles on the Scene. This property affects only handles, control points are always visible.

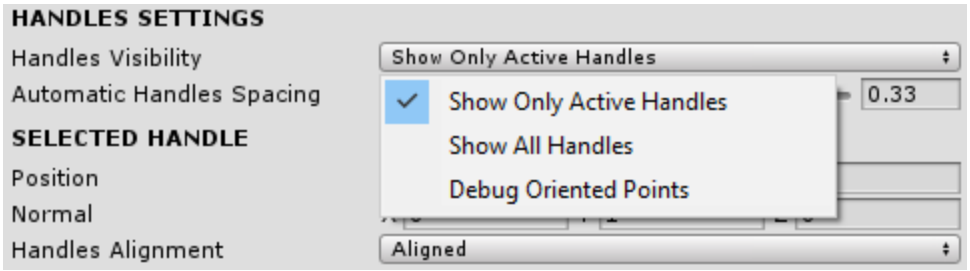

Figure 56 - Handles visibility property

You can either choose between “Show Only Active Handles” or “Show All Handles”. Active Handles are handles connected to the current selected control point.

NOTE: The “Debug Oriented Points” option can be used to visualize the reference path used by the Spline Mesh Renderer, Spline Follower and Spline Prefab Spawner components. It allows you to see the points position and direction vectors (forward, up and right). It is very useful to identify areas where the control points normals need to be adjusted.

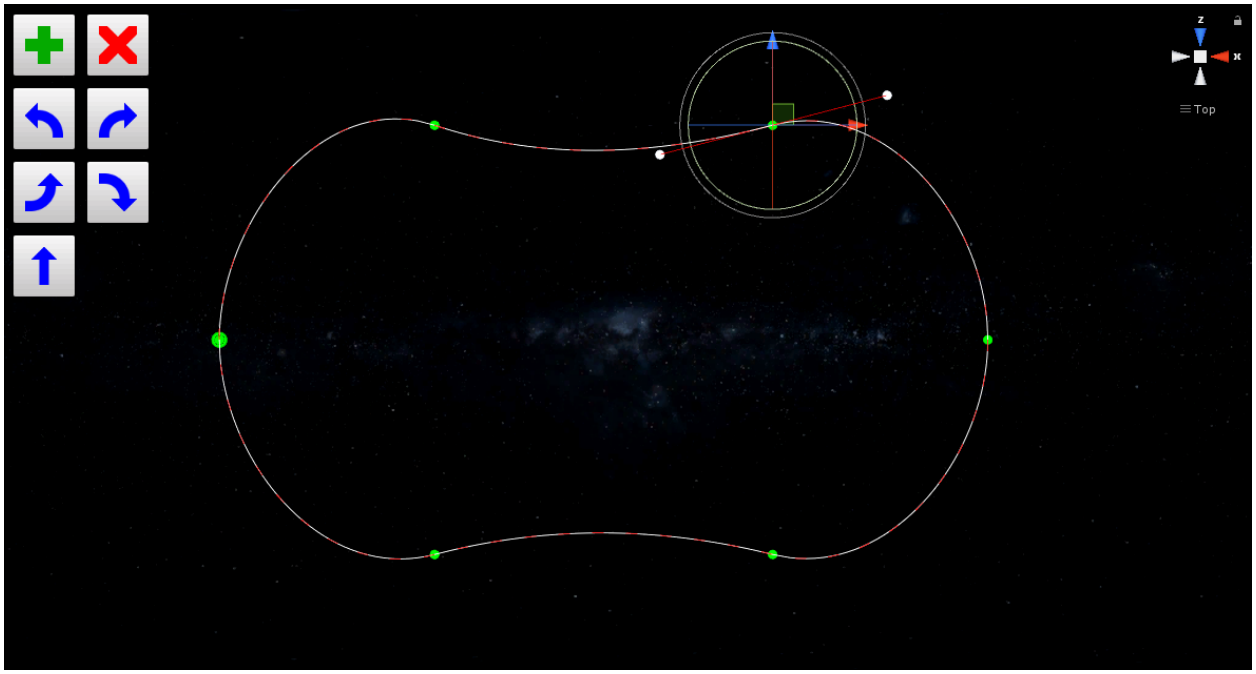

Figure 57 - Handles visibility: Show Only Active Handles

The “Show only Active Handles” is the default option, when it is selected the scene view is much more clean.

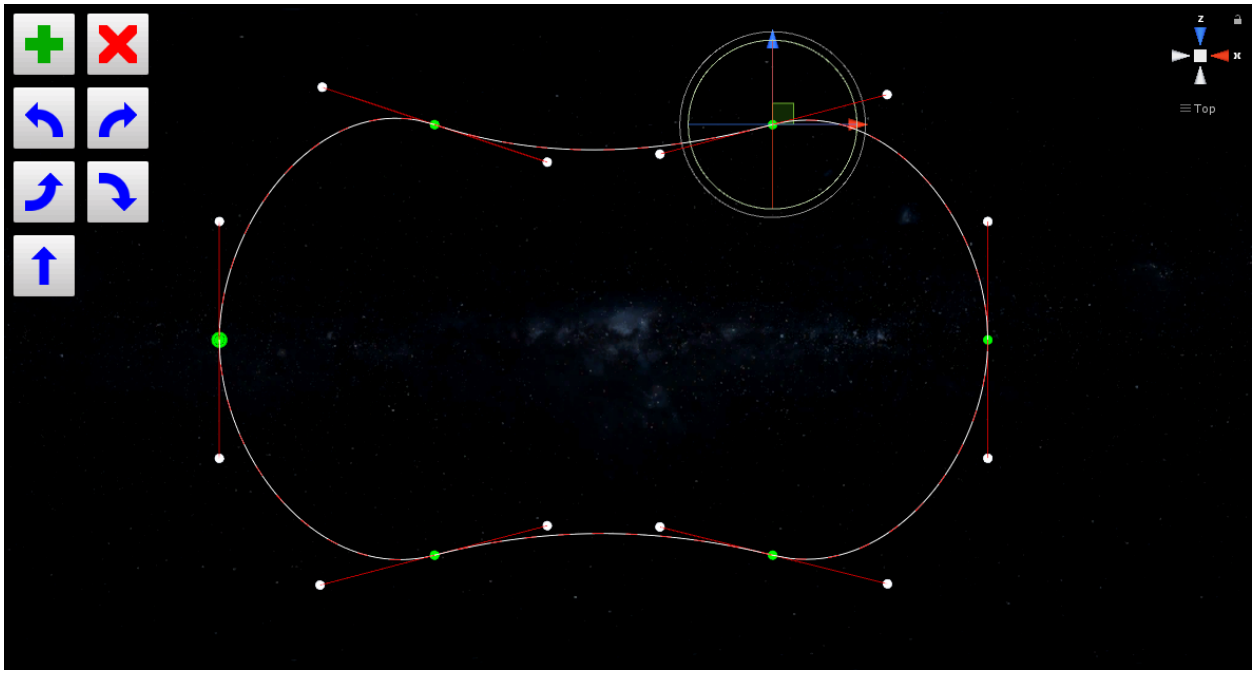

Figure 58 - Handles visibility: Show All Handles

The “Show All Handles” option is useful if you want to adjust the handles, without the need to pre-select their corresponding control point. It is also useful to check the handles alignment along your spline (See the Handles Alignment section for more details).

NOTE: Handles with “Automatic” alignment are not visible on either handles visibility modes. (see Handle Alignment section for more details)

2.7.2 - Selected Handle

The “Selected Handle” section shows the selected control point or handle properties.

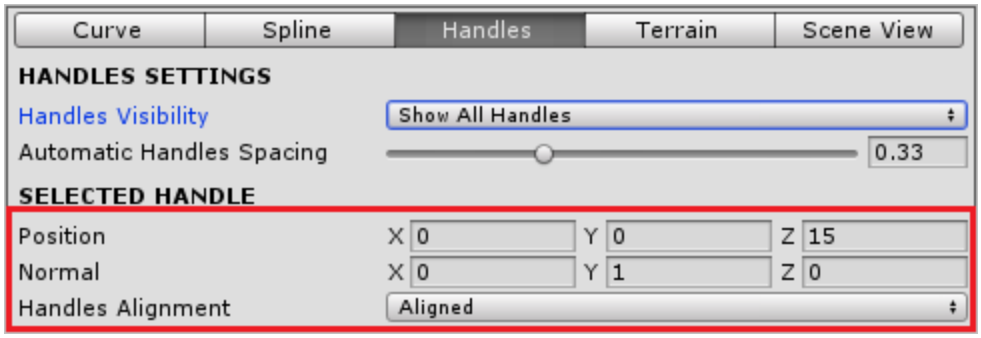

Figure 59 - Selected Handle properties

In this section you can verify and manually adjust the position, normal and alignment of any point.

As mentioned before, the position is responsible for defining the shape of the spline, the normal is used to customize the behaviour of other components. For example, The Spline Mesh Renderer uses the normal value to add custom rotation to the mesh area affected by the point.

2.7.3 - Handles Alignment

The “Handle Alignment” property defines how the bezier curves connections are shaped and aligned.

There are four alignment modes, which mode is represented by a different color and have different effects on the curve shape.

- **Aligned**
  - Represented by white circles
  - Positioned on opposite sides of the control point
  - Distance between control point and both handles can be different
  - Can be manually adjusted

- **Mirrored**
  - Represented by yellow circles
  - Positioned on opposite sides of the control point
  - Distance between control point and both handles is always equal
  - Can be manually adjusted
- **Free**
  - Represented by blue circles
  - Can be positioned anywhere
  - Distance between control point and both handles can be different
  - Can be manually adjusted
- **Automatic**
  - Automatic handles are not visible
  - Cannot be manually adjusted
  - Automatically adjusted accordingly to the control point position

NOTE: As mentioned before at the Spline section, control points are represented on the Scene View as green circles. They mark the start and end of each bezier curve.

Handles are represented by circles attached to the control points by red lines.

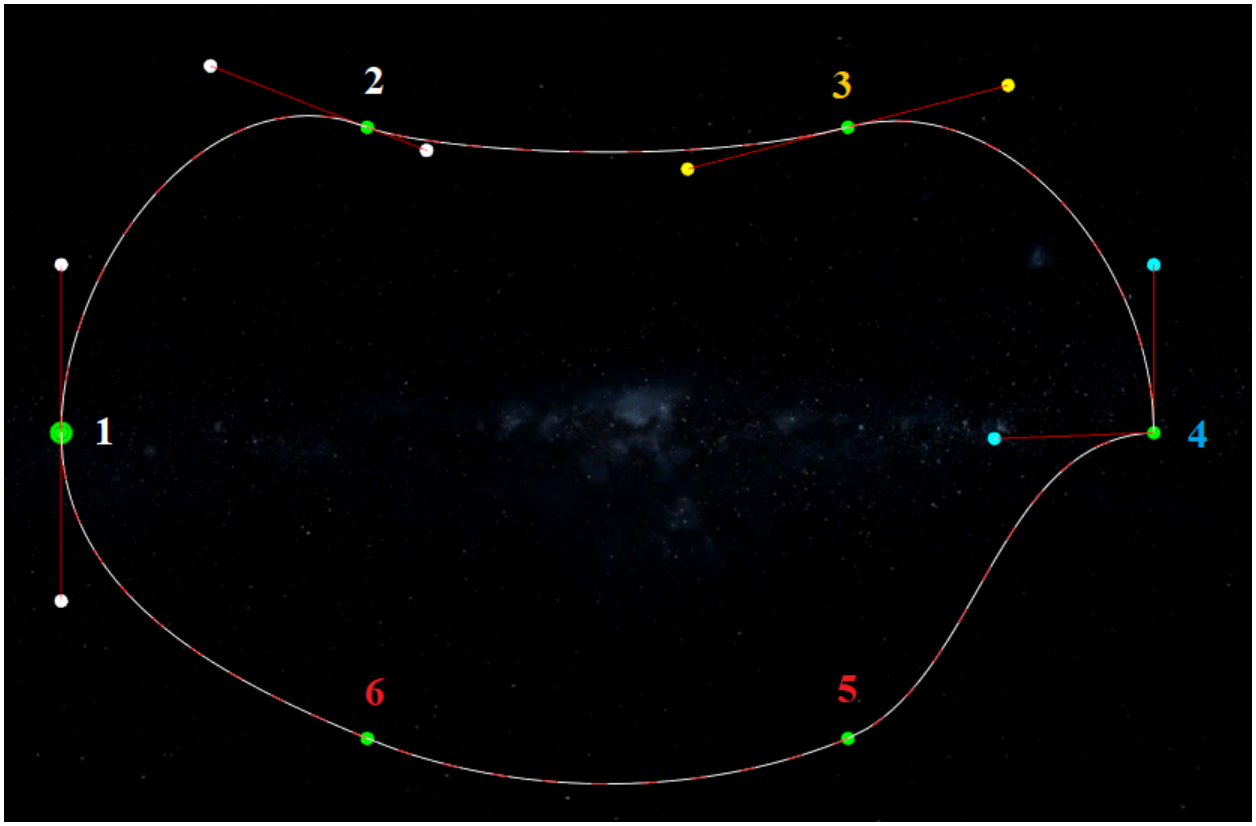

Figure 60 - Handle Alignments

Take a look at the sample image above, control points 1 and 2 handles are set to “Align”, control point 3 handles are set to “Mirrored”, control point 4 handles are set to “Free”, control points 5 and 6 handles are set to “Automatic”.

NOTE: When creating new curves, the newly created curve control point will inherit the handle alignment from the last control point. This is very useful if you want to build a spline that has the same alignment on all control points.

Align and Mirrored handles are good to create smooth curves by manually adjusting the handles positions.

Free handles can be used to create sharp curves, like can be seen in the sample image above.

Automatic handles automatically adjust themselves to create smooth curves based on the control points positions. They are useful if you want to adjust the entire spline shape at once. Automatic curves are adjusted by changing the “Automatic Handles Spacing” property.

2.7.4 - Automatic Handles Spacing

The “Automatic Handles Spacing” property affects the shape of the spline around control points that are set to the automatic handle alignment mode.

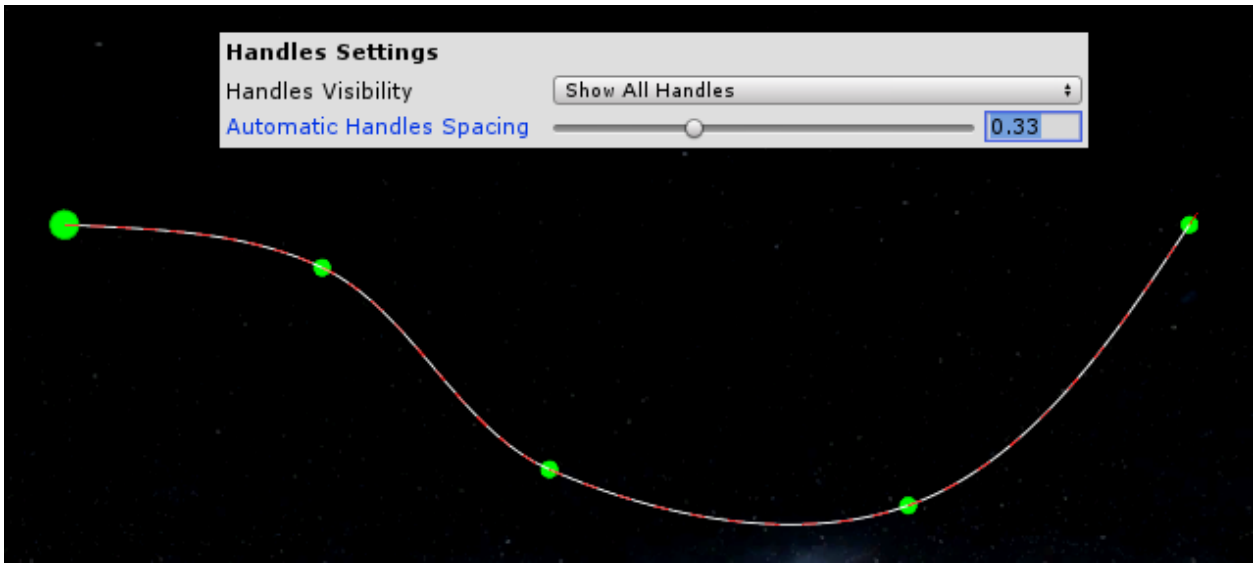

Figure 61 - Automatic Handles Spacing Sample 1

By default, it is set to 0.33, by using this value the shape of the curves will automatically be adjusted to create smooth curves, based on the position and distance between the control points.

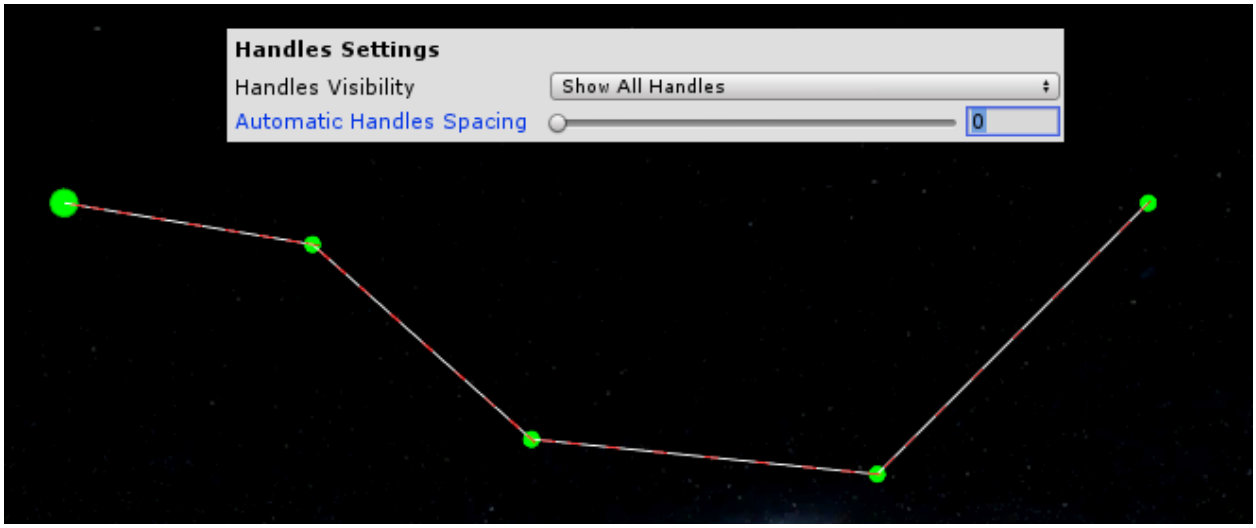

Figure 62 - Automatic Handles Spacing Sample 2

By reducing the spacing value, the spline shape will become less curved. A spacing value of zero will result on straight segments.

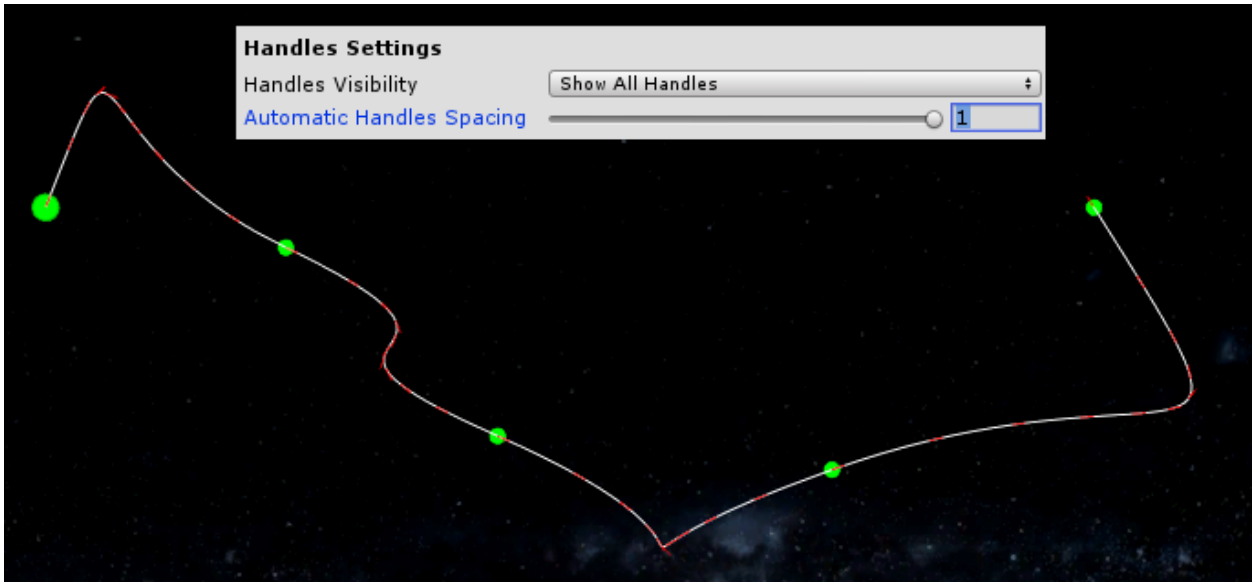

Figure 63 - Automatic Handles Spacing Sample 3

By increasing the spacing value, the spline shape will become distorted. A spacing value of one will result in completely distorted segments.

2.8 - Terrain

Regarding the relationship between splines and terrains, there are only two possible outcomes: you can either adjust the spline to the terrain **or** adjust the terrain to the spline. Luckily for you, this asset supports both.

You can use the Follow Terrain feature to project the spline to your terrain. Perfectly, adjusting the spline to the terrain elevations.

Alternatively, you can use the Terraforming feature to adjust the terrain to your spline.

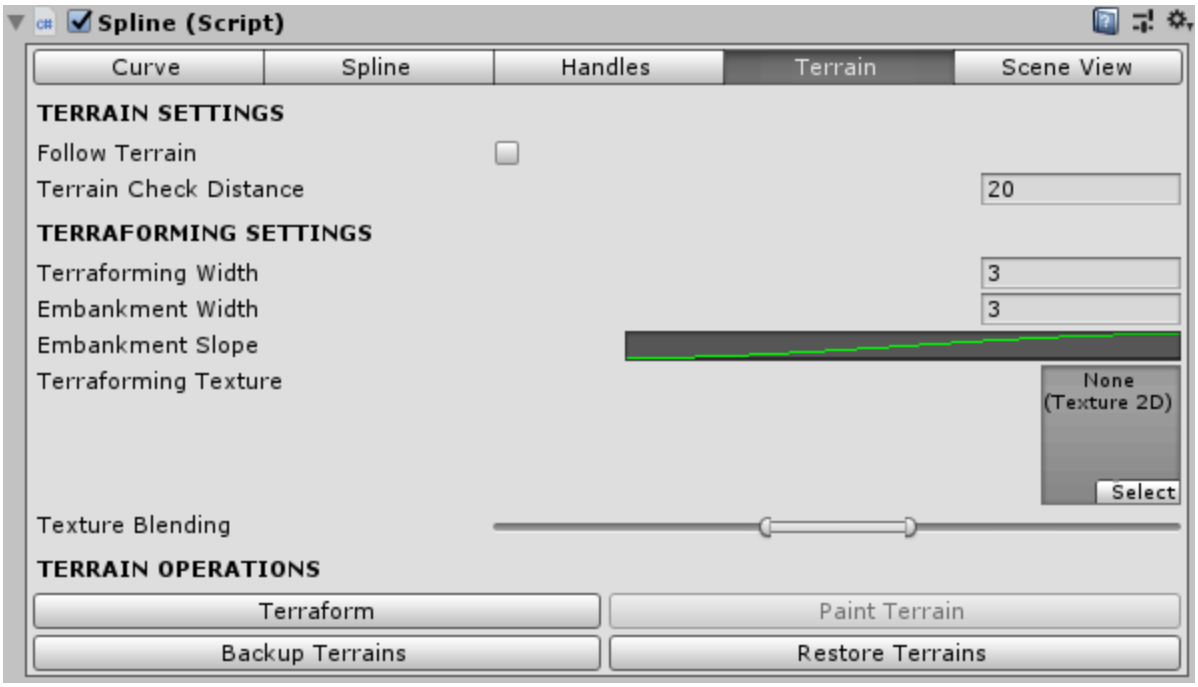

Figure 64 - Spline Terrain Settings

2.8.1 - Follow Terrain

The “Follow Terrain” property is used to add terrain elevation support **without** the need of manually adjusting your spline to fit your terrain. It works by creating a virtual projection of the spline on your terrain. The spline shape is preserved as it is, and the virtual projection can be used as a reference by the Spline Mesh Renderer, Spline Prefab Spawner and Spline Follower components.

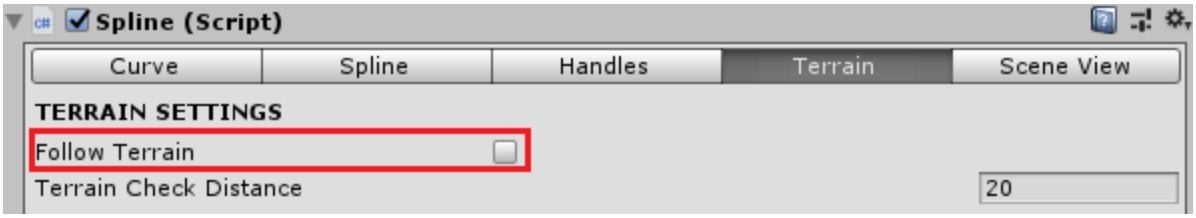

Figure 65 - Follow Terrain Property

This technique allows you to use simple flat splines to create complex level design elements that will automatically adjust to any type of terrain elevation.

For example, let's assume you wish to build a circular railroad using the Spline Mesh Renderer.

First, you build a simple flat spline on the desired shape (this can be quickly done by using the Curve Operations), but you realize that a segment of the railroad is passing through the terrain elevations, as shown in the sample image below.

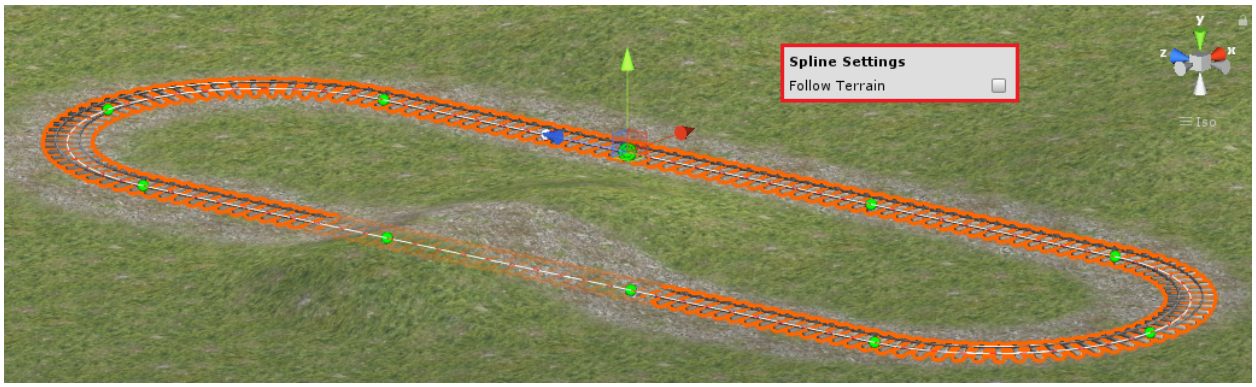

Figure 66 - Disabled Follow Terrain Feature

Since buried rails make no sense, you wish to adjust your railroad to your terrain. This can be easily achieved by enabling the "Follow Terrain" feature.

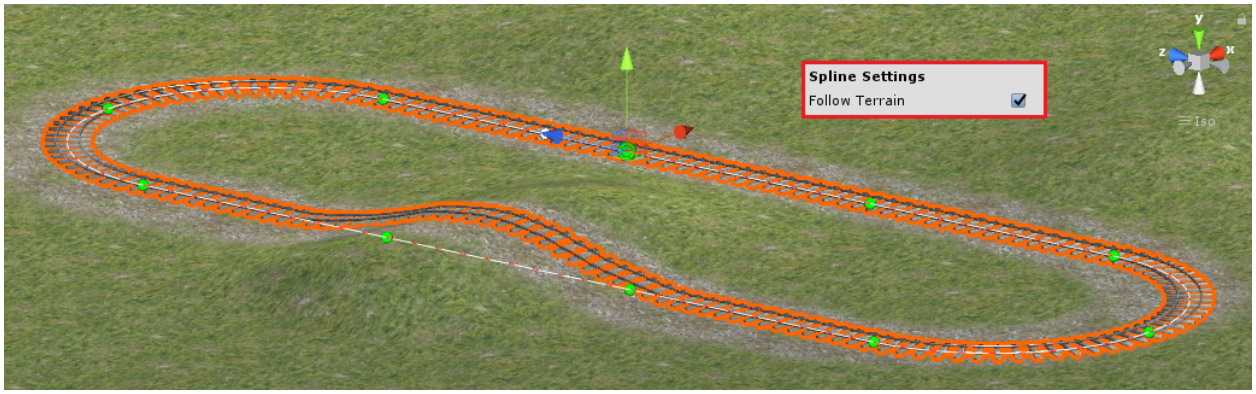

Figure 67 - Enabled Follow Terrain Feature

NOTE: The spline shape remains unchanged, even though the rails mesh is now being projected on the terrain.

2.8.1.1 - Terrain Check Distance

The Follow Terrain is a very powerful feature, however, when working with very long splines, it can become performance heavy for the Unity Editor, since it needs to check the exact position on space that the spline vertically intersects the terrain for every new segment created.

Since the vertical intersection can be located on an infinite number of spatial points below or above the spline, to avoid performance issues this feature is limited by the "Terrain Check Distance" property.

The “Terrain Check Distance” property defines how far from the spline the “Follow Terrain” feature should look for the terrain intersection.

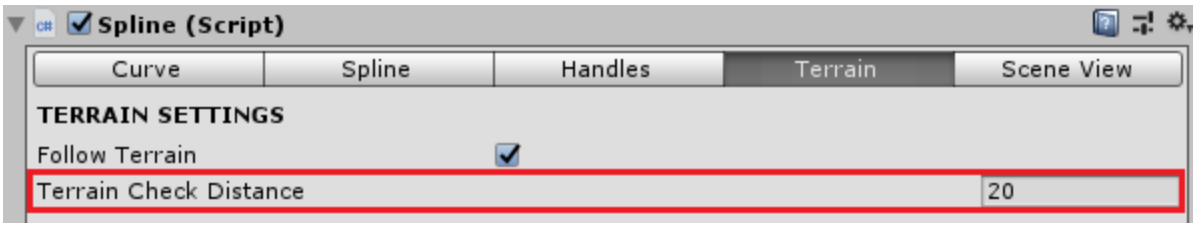

Figure 68 - Terrain Check Distance Property

NOTE: By default the “Terrain Check Distance” is set to 20 meters, which means it will look for the terrain 20 meters downwards and 20 meters upwards, covering a vertical range of 40 meters.

However, if your terrain height range is greater than 40 meters, you may need to adjust this value if, and only if, your spline projection does not reach a part of your terrain.

Alternatively, you can also roughly adjust the spline to bring it closer to your terrain until it reaches the check range. This alternative is performance friendly and can be done quite quickly.

2.8.2 - Terraforming

In some situations, you may wish to adjust the terrain heights to match your spline. For example, let’s assume you are using the Spline Mesh Renderer component to build a railroad.

For the sake of train stability, you wish your railroad to be as flat and smooth as possible and for detailing purposes you also desire to create Track Ballast along your railroad.

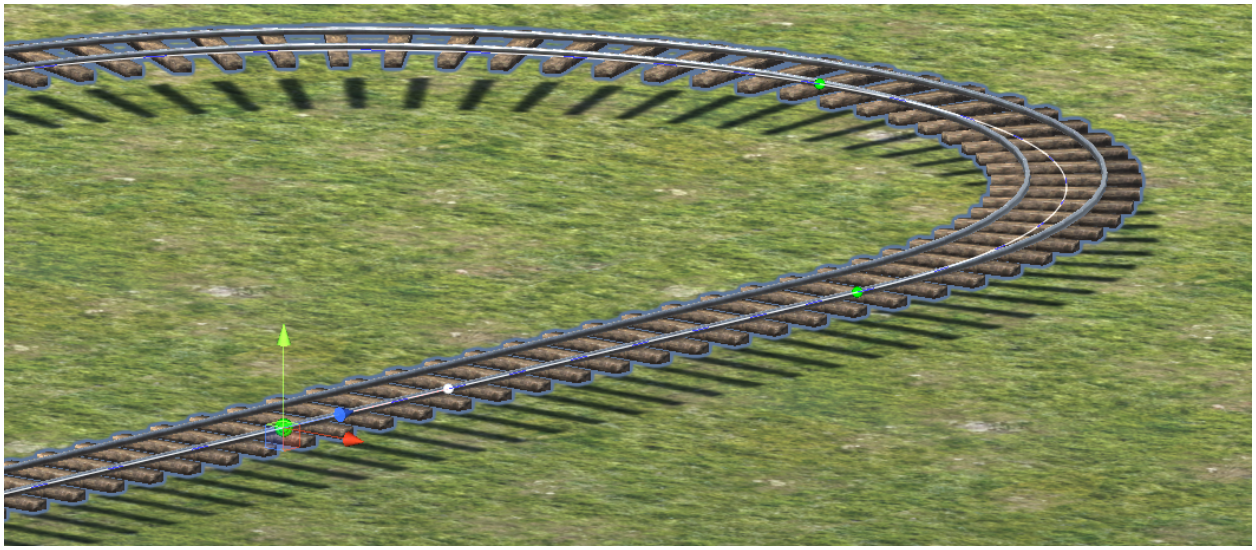

Figure 69 - Railroad track ballast sample (before)

Let's assume you already built your railroad by using a spline that is located 1 meter above the terrain. So, all you have to do now is to adjust the terrain height along the railroad and apply a rock texture to simulate the track ballast rocks. But, manually adjusting the terrain heights and textures would be very time consuming, that's why the terraforming operation was created to make this process as simple as a button click.

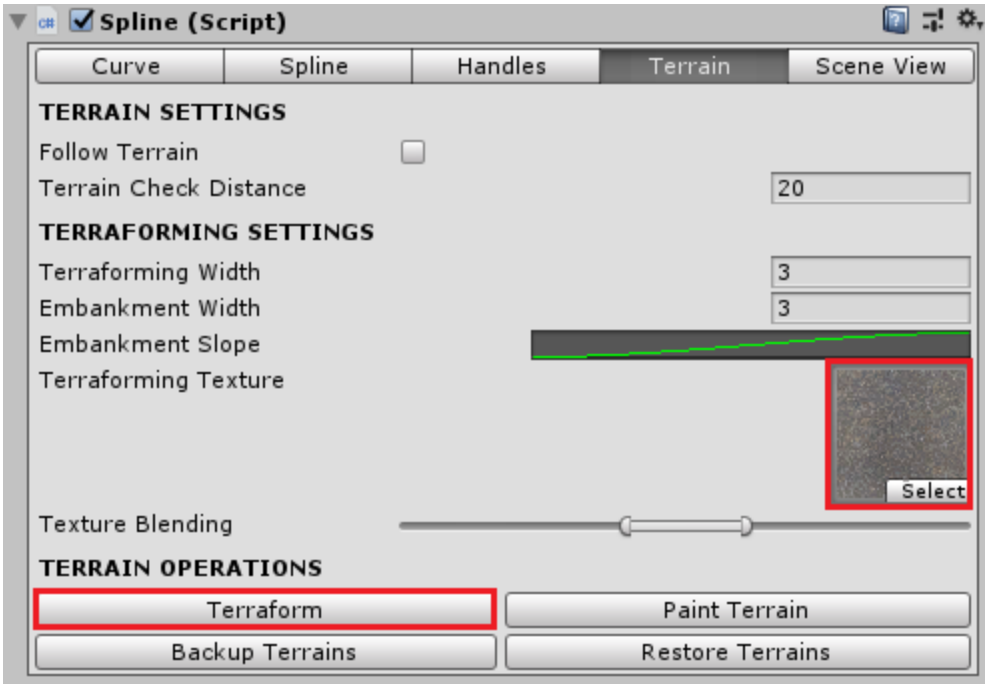

Figure 70 - Terraforming Operation

In order to create the track ballast, just select a rock terrain texture and click on the “Terraform” button.

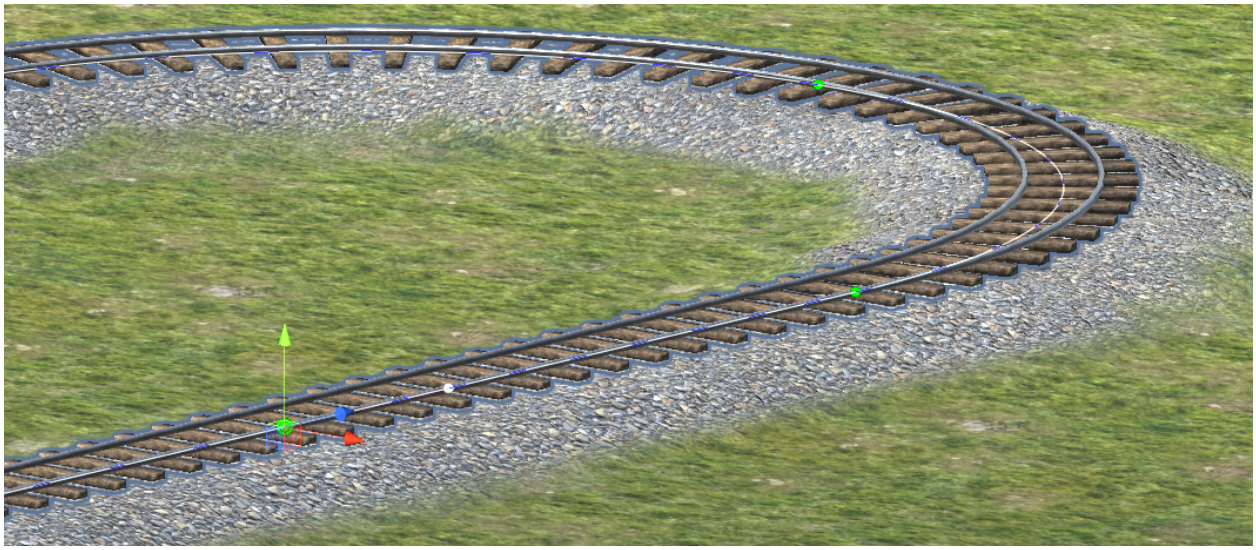

Figure 71 - Railroad track ballast sample (after)

NOTE: The Terraform feature will make the best it can to create a smooth result based on the selected Terraforming Settings. However, your terrains with lower heightmap resolution may present some terraforming artifacts (non-smooth areas). In these cases, it is recommended to use the Unity default Smooth Height brush to make minor adjustments where needed.

2.8.2.1 - Terrain Backup

Since terraforming terrain painting is a destructive operation (which means, ctrl+z will not undo terraforming), it is highly recommended to backup your terrains before proceeding with it. In order to remind you of the importance of backups, this confirmation dialog will pop up whenever you click on the “Terraform” button.

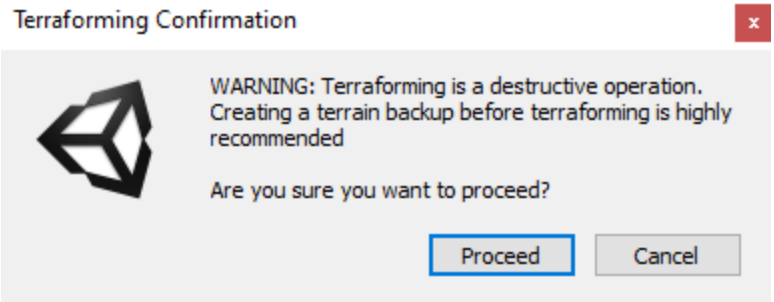

Figure 72 - Terraforming confirmation dialog

You can create and restore terrain backups by using the following buttons.

Figure 73 - Terrain backup features

A terrain backup is just a copy of its terrain data file, saved alongside the original terrain data. To keep things simple, just think of the backup file as a screenshot of your terrain settings at the moment of the backup creation.

Figure 74 - Terrain backup file

NOTE: Keep in mind that only 1 backup file per terrain is supported at any moment in time. Therefore, make sure to only update your terrain backups when you're happy with the modifications you've made.

In order for the "Restore Terrains" feature to work properly, the backup and the current terrain size, heightmap and splatmap resolutions **MUST BE THE SAME**. If you change any of your terrain settings in your scene or rename your terrains, it is recommended to update your terrain backups before using the Terraforming or Paint Terrain features.

NOTE: As a rule of thumb, it is recommended to always update your backups before using the Terraforming or Paint Terrain operations. That way, you can always use the Restore Terrains feature to undo the operation in case you wish to change the terraforming settings and try again.

2.8.2.2 - Terraforming Settings

The terraforming settings affect the "Terraform" and "Paint Terrains" operations.

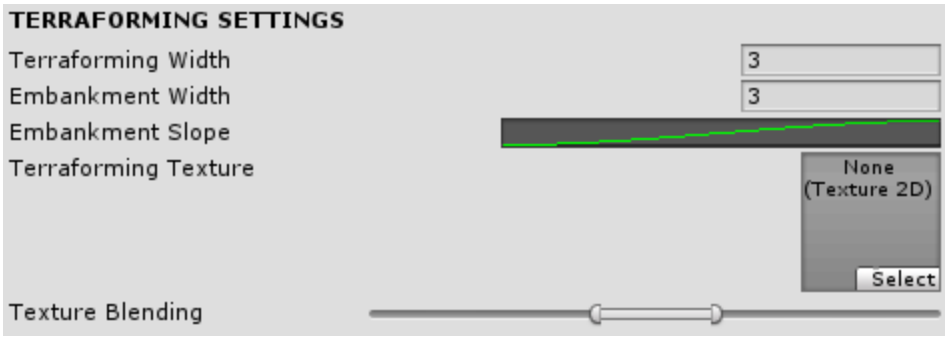

Figure 75 - Terraforming Settings

The “Terraforming Width” property defines how much of the terrain will be “flatten” along your spline when using the Terraform operation.

The “Embankment Width” defines how far the terrain smoothing will be applied, based on a smoothing curve defined by the “Embankment Slope” feature.

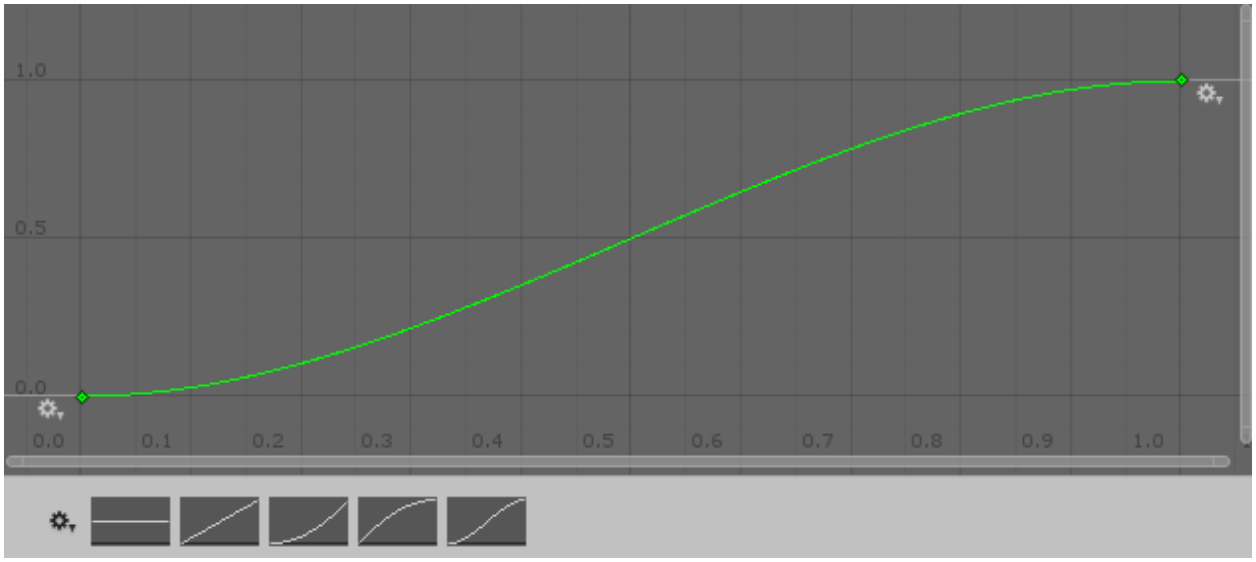

Figure 76 - Default embankment slope curve

If you click on the “Embankment Slope” property in the inspector, the Unity default curve editing dialog will be shown. You can choose the available presets or manually adjust the curve. However, keep in mind that the far left value must always be 0 and the far right value must always be 1.

NOTE: Both Terraforming and Embankment Width properties are measured in pixels. Therefore, the affected area may change based on your terrain’s heightmap and splatmap resolutions.

The “Terraforming Texture” is used to repaint the area affected by the “Terraform” and/or “Paint Terrains” features.

NOTE: The “Terraforming Texture” property is used to identify which terrain layer should be applied on the affected region. In order for the terrain painting to be successful, your terrain must have a corresponding terrain layer in which the Diffuse texture is the same as the selected terraforming texture.

You can also adjust how much “Texture Blending” will be applied while repainting. The blending is more visible at the spline starting and ending points. By using the default blending, it will produce a round texture blending on both ends. This is useful while painting more natural paths.

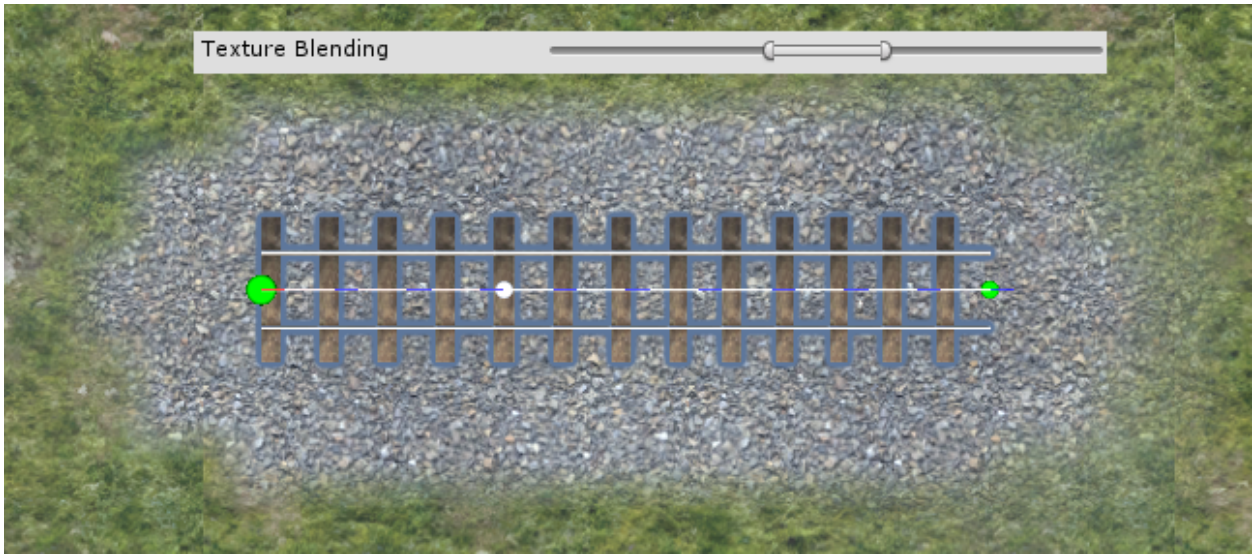

Figure 77 - Default Texture Blending

However, if you wish your path to be squared on both ends, you can drag the blending range all the way to the end in order to produce this effect. This can be useful for creating “human made” paths.

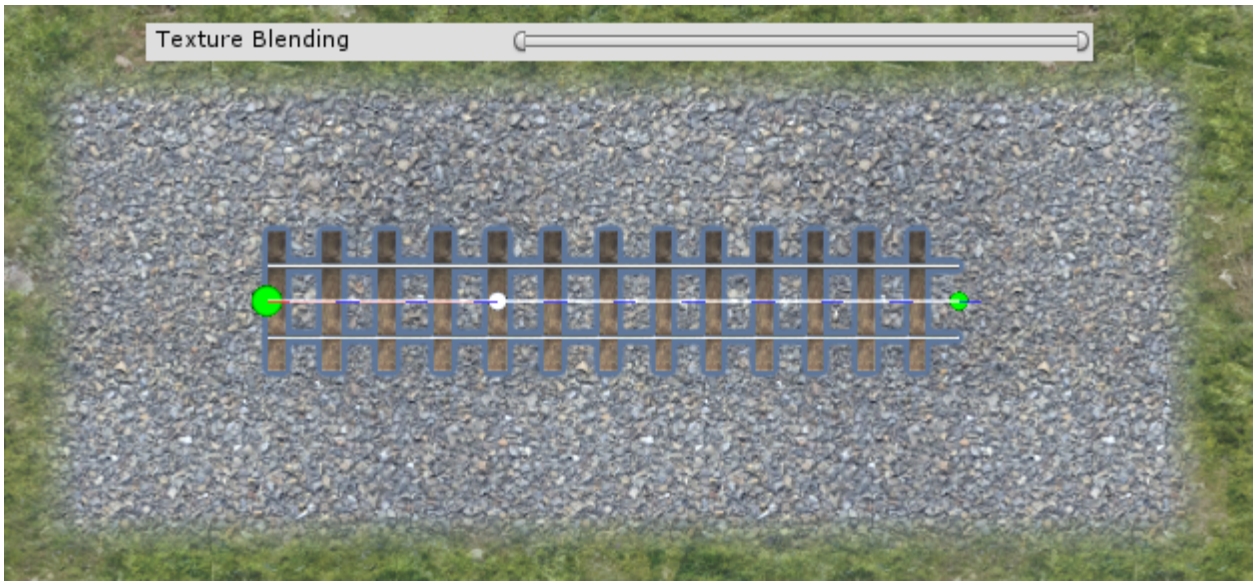

Figure 78 - No Texture Blending

2.9 - Scene View

All settings related to the spline behaviour on the Scene View are located under the “Scene View” tab.

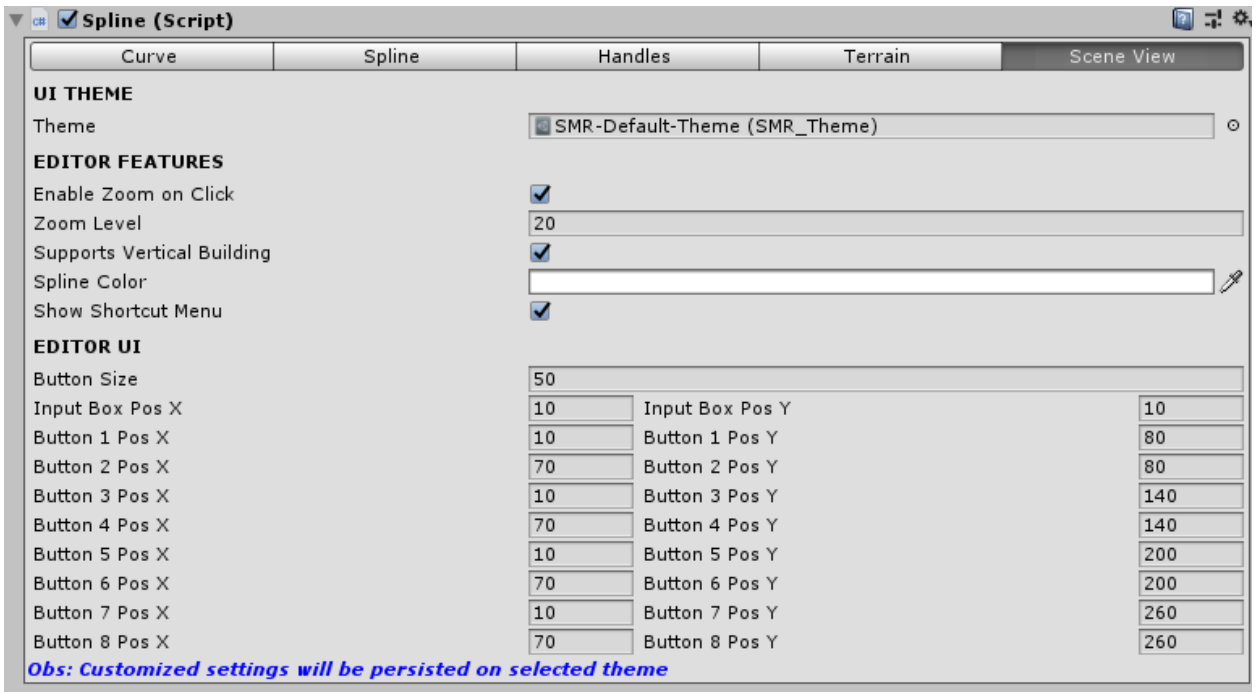

Figure 79 - Spline Scene View Settings

2.9.1 - UI Theme

The Theme property holds a reference to the settings file that persist all settings located under the Scene view tab. The theme defines which editor features are enabled and the position, size and icons for the buttons that compose the Quick Access Building Menu.

Figure 80 - Default UI Themes

The default theme files are located under “**WSM Game Studio > Spline Mesh Renderer > Themes**” folder. But, you can also create custom themes by right clicking inside the folder and

using the file creation menu: “**Create > WSM Game Studio > Spline Mesh Renderer > Theme**”

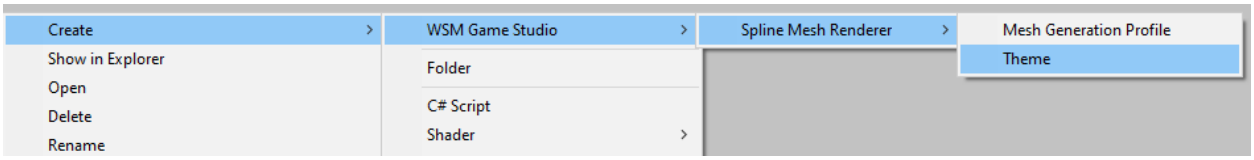

Figure 81 - Creating a Custom UI Theme

Since the Quick Access Building Menu relies on the theme settings, it will only be visible when a theme is selected. Otherwise, the following message will be shown in the Scene View whenever the corresponding spline is selected.

Figure 82 - Theme not selected warning

In order to fix that, just navigate to the Scene View tab and select a theme.

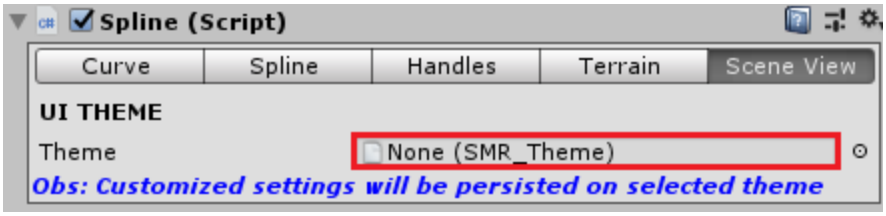

Figure 83 - Theme not selected (inspector)

NOTE: In order to visualize the Scene View tab properties, a theme must be selected.

2.9.2 - Editor Features

In this section you can enable/disable Scene View related Editor Features.

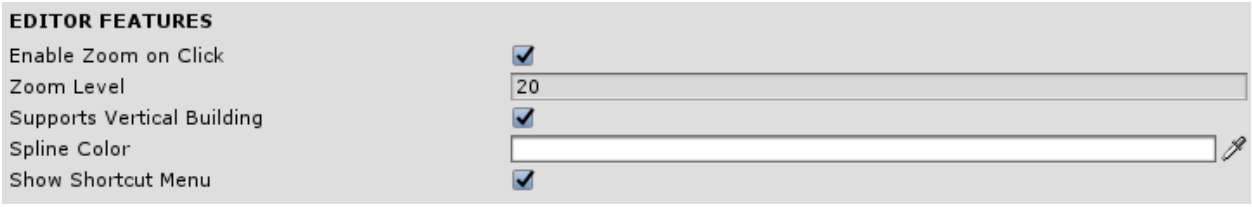

Figure 84 - Editor Features

Here you can decide whether you want to zoom when clicking on control points, how much zoom should be applied, customize the color of your splines, disable vertical building in the Quick Access Building Menu (hides curve up and down buttons) and enable/disable the shortcut menu.

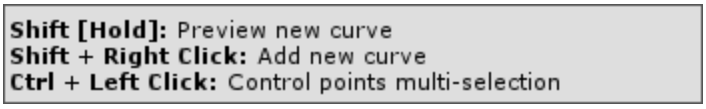

Figure 85 - Scene view shortcut menu

2.9.3 - Editor UI

If you're not satisfied with the Quick Access Building Menu buttons size and positions, you can create a custom theme and choose where you want the buttons to appear in the Scene View.

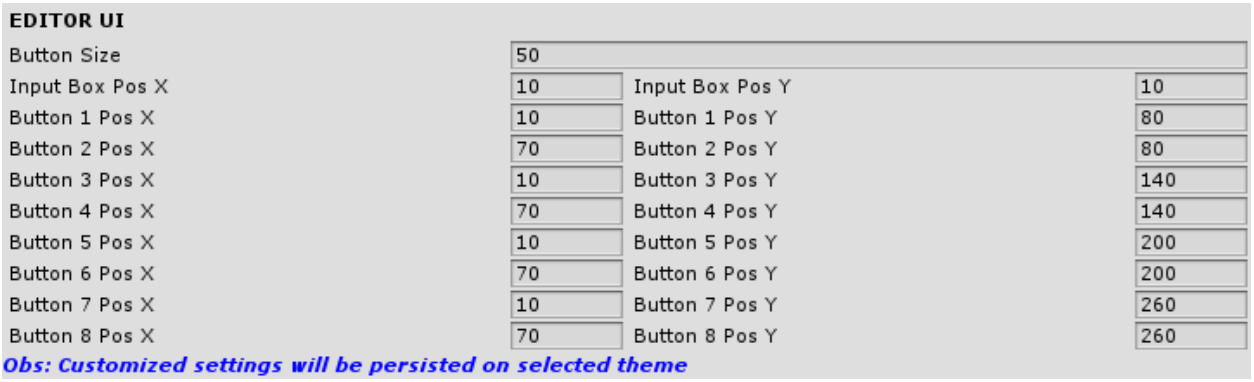

Figure 86 - Editor UI Settings

It is also possible to edit these settings directly on the theme file. However, by editing in the scene view tab of a spline object instance, you can visualize the changes at runtime in the Unity Editor, making it much easier to customize the buttons size and positions.

NOTE: You can also customize the buttons icons if you wish so, but these settings were kept in the theme file only.

3 - Spline Mesh Renderer

The "Spline Mesh Renderer" component is a level design tool that allows you to create dynamically generated meshes that can be adjusted on the Unity Editor to fit your scene needs.

It works by extruding a base mesh segment along a Bezier Spline. This process allows the procedural generation of long and curved meshes such as roads, railroads, power lines, pipes, etc…

Therefore, all you need to create your dynamic procedural mesh is:

- Base Mesh Segment Model
- Base Mesh Materials
- Base Mesh Segment Collider (optional)
- Create your Bezier Spline using the Spline component

3.1 - Creating a Spline Mesh Renderer

You can create a new Spline Mesh Renderer by using the default object creation menu:

- **Hierarchy Window**
  - Right Click on Hierarchy window
  - WSM Game Studio > Spline Mesh Renderer
- **GameObject Menu**
  - Click on GameObject menu
  - WSM Game Studio > Spline Mesh Renderer

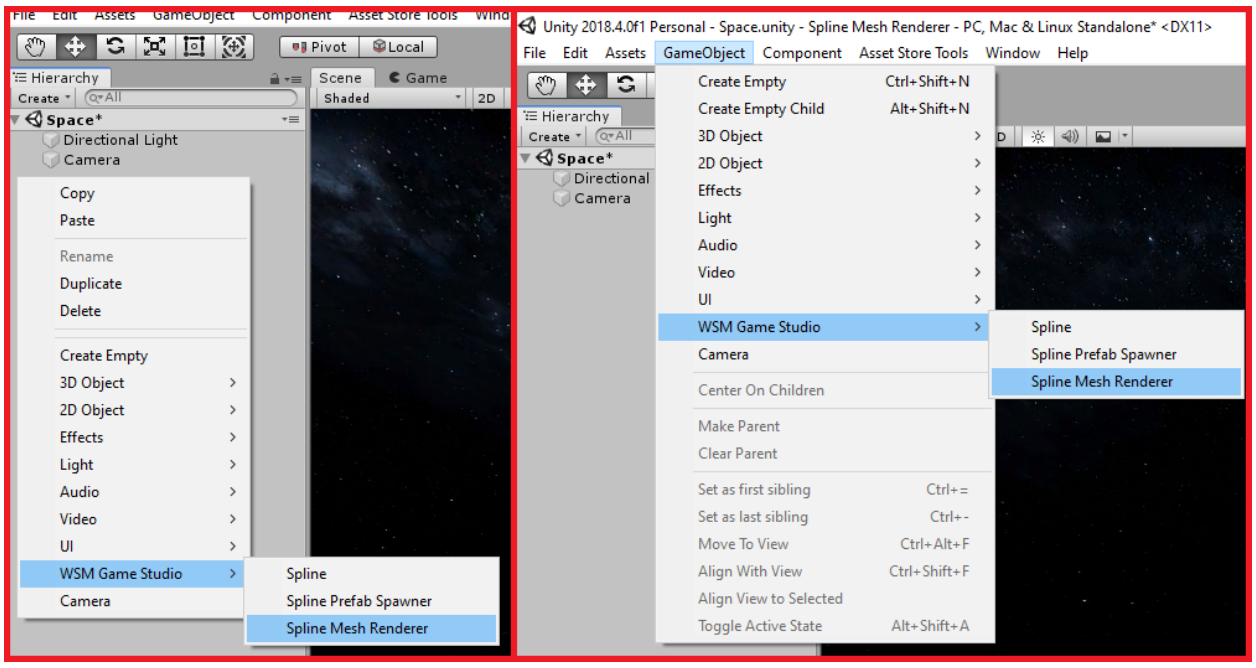

Figure 87 - Creating a new spline mesh renderer

Alternatively, you can also Drag & Drop the “SplineMeshRenderer” prefab to your scene. This prefabs is located under “Assets/WSM Game Studio/Spline Mesh Renderer/Prefabs/Legacy”.

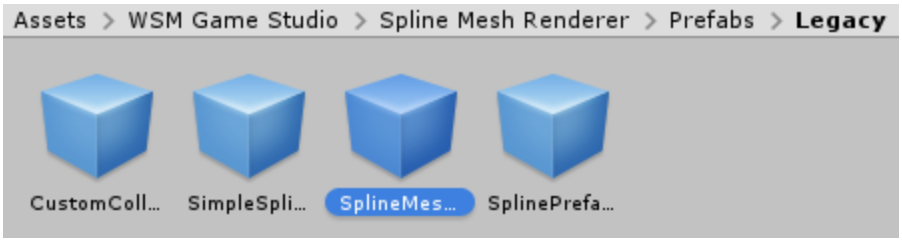

Figure 88 - Spline Mesh Renderer prefab

NOTE: Prior to version 2.1, this prefab was used to create new spline mesh renderer instances. It is not needed for this purpose anymore, however it was kept for the sake of backwards compatibility.

You should never change this prefab directly or save alterations made on it. It was created with the sole purpose of being used as a starting point.

The “SplineMeshRenderer” prefab is composed by the following components

- LOD Group
- Spline Script
- Spline Mesh Renderer Script
- Child Auxiliar Transforms (Aux1, Aux2)

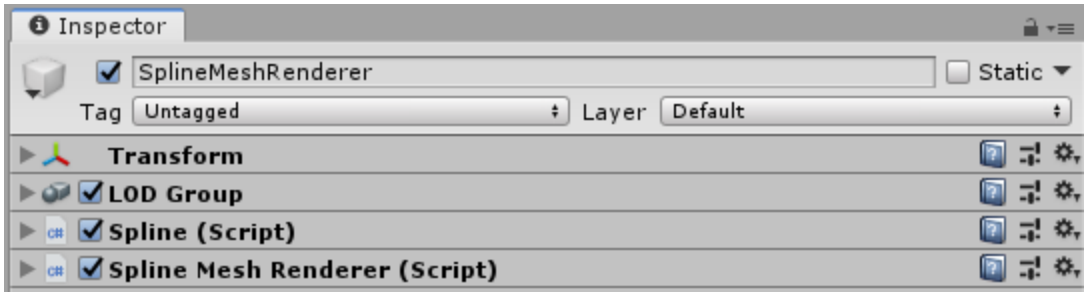

Figure 89 - SplineMeshRenderer prefab components

The LOD Group is a default Unity component. It was introduced on v2.1, to add LOD support to the Spline Mesh Renderer. For More information about LOD Groups, please check the official documentation.

- LOD Group Official Documentation

Note that the “SplineMeshRenderer” prefab already has a Spline component attached to it. By default, Spline Mesh Renderer component will use the attached spline component as reference. But, if you wish to use another spline on your scene as a reference, you can replace it on the Spline property.

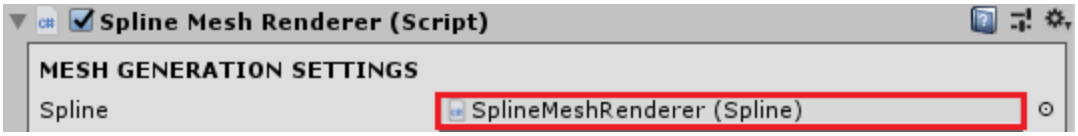

Figure 90 - Reference Spline

NOTE: The auxiliary transforms (Aux1 and Aux2) are used by the "Spline Mesh Renderer" script to calculate the position and length of each extruded mesh segment along the spline. **DO NOT** rename or delete these objects.

Now that you have a Spline Mesh Renderer on your scene, all you need to create your custom mesh is to select a Mesh Generation Profile and edit your Spline to shape the object as you see fit.

3.2 - Mesh Generation Settings

All settings and operations that affect the generated mesh are located under the "Mesh Generation Settings" section.

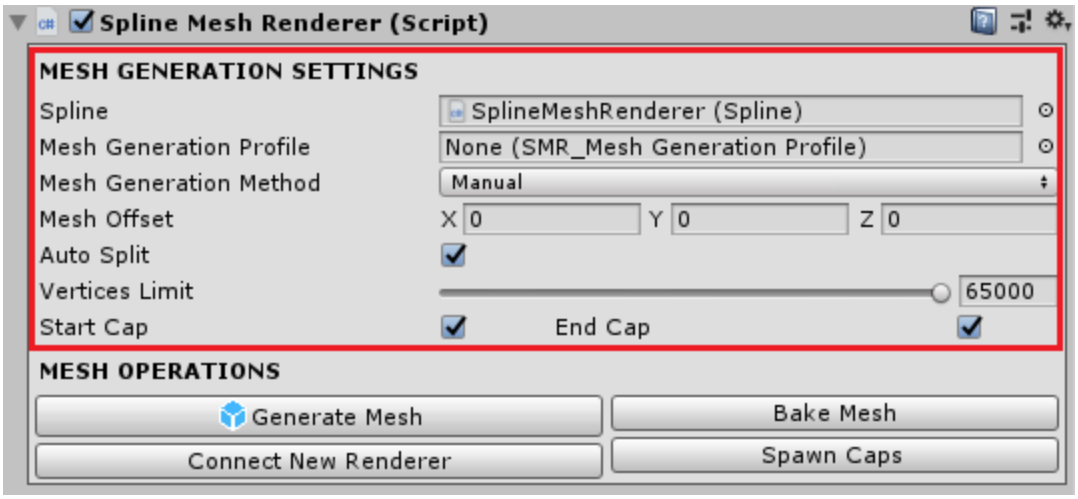

Figure 91 - Mesh Generation Settings

3.2.1 - Mesh Generation Profile

All the information about the mesh that will be extruded along the spline is stored on a Mesh Generation Profile.

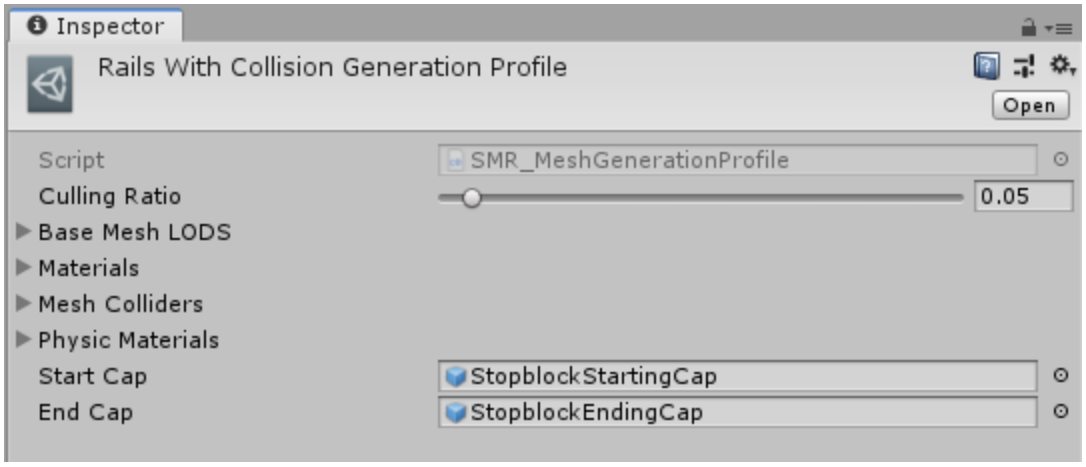

Figure 92 - Mesh Generation Profile

Profiles defines what base mesh segment will be extruded, how many LODs will be used, what materials should be applied, if and what colliders should be used, what physics materials should be applied for each collider and if there is any caps prefabs available to close the start and end of the generated mesh.

You can find some generation profile samples under the "**WSM Game Studio > Spline Mesh Renderer > Generation Profiles**" folder.

Figure 93 - Mesh Generation Profile samples

You can also create custom Mesh Generation Profiles by using the default file creation menu. Right click on the folder, then navigate to "**Create > WSM Game Studio > Spline Mesh Renderer > Mesh Generation Profile**".

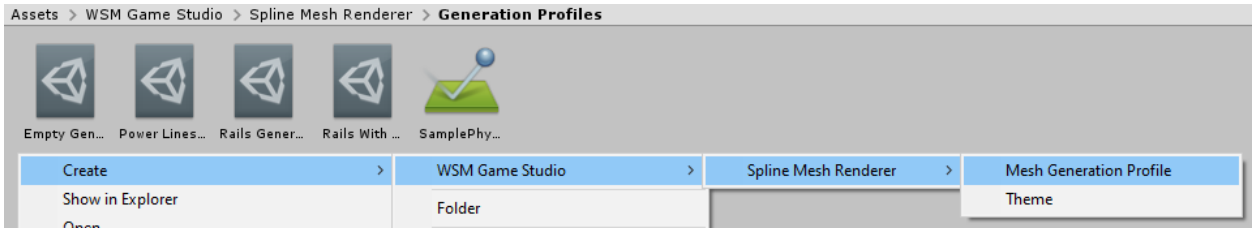

Figure 94 - Creating a new Mesh Generation Profile

NOTE: Mesh Generation Profiles were introduced on v2.1, in order to make it easier to share generation settings between Spline Mesh Render instances in your scene.

Since you can create multiple generation profiles and rename them as you wish, it also makes it easier to generate multiple objects on your scene. For example, you can have a generation profile for creating railroads, another one for roads, racing tracks, etc…

Whenever you create a new Spline Mesh Renderer instance on your scene, by default it will not have a generation profile selected.

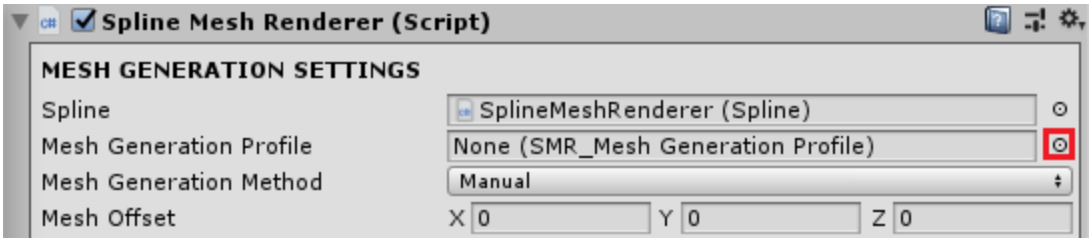

Figure 95 - Spline Mesh Renderer default state

To select a profile, you can drag & drop it to the corresponding property field in the inspector or, you can click on the small circle to see a list of all profiles available in the project.

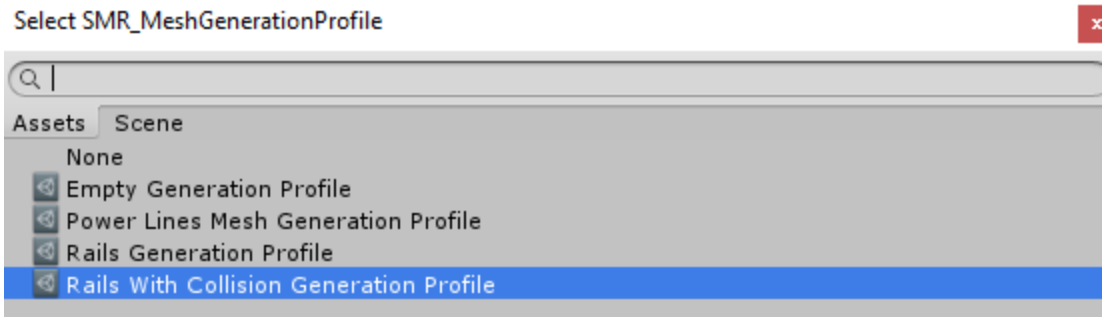

Figure 96 - Generation profile selection dialog

For this sample, let's select the "Rails With Collision Profile".

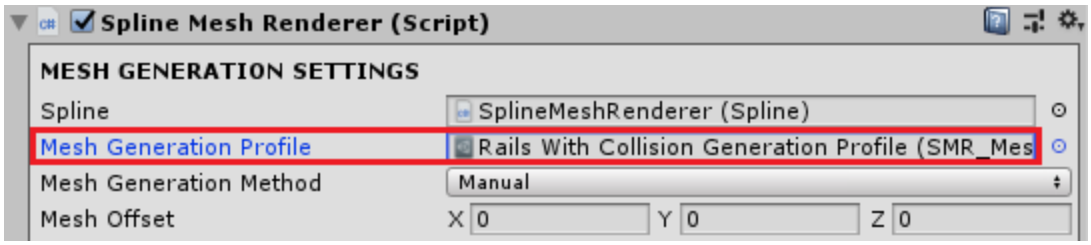

Figure 97 - Selected generation profile

Now that a generation profile is selected, in order to preview the generated mesh in the scene, just click the Generate Mesh button in the Quick Access Building Menu.

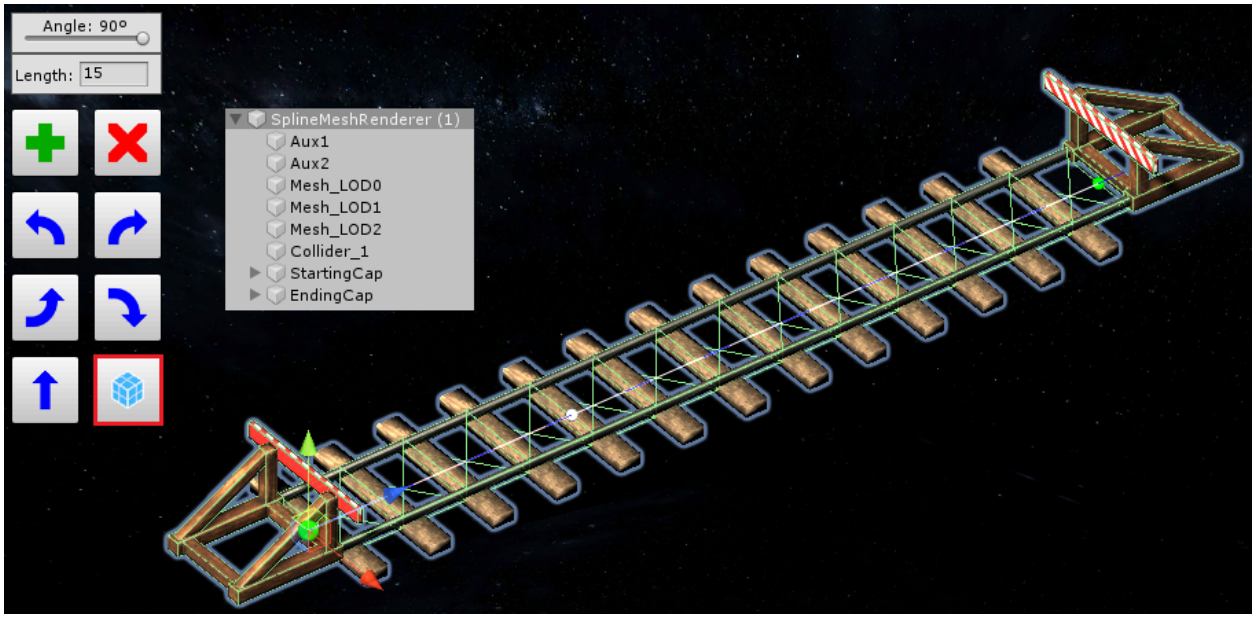

Figure 98 - Procedurally Generated Mesh Sample

As can be seen in the sample image above, the generated LOD levels, colliders and caps configured on the selected generation profile are spawned as children of the spline mesh renderer instance.

Let's take a closer look at the generation profile settings.

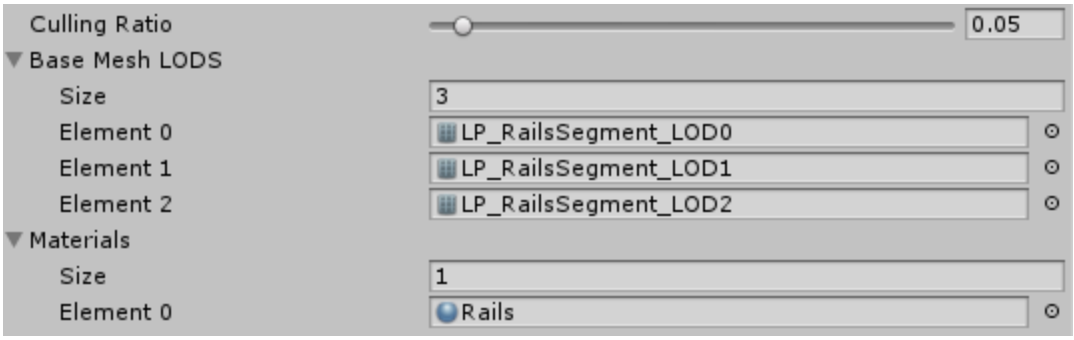

Figure 99 - Profile base mesh & materials

You can have multiple Level of Details for your procedural generated mesh. The LOD levels must be set at the "Base Mesh LODS" list in order, starting from the highest detailed to the lowested detail mesh. In this case, all LOD models share the same material, but you can also have multiple materials if you wish so.

The base mesh segments used on this sample are located under the "**WSM Game Studio > Spline Mesh Renderer > Models**" folder.

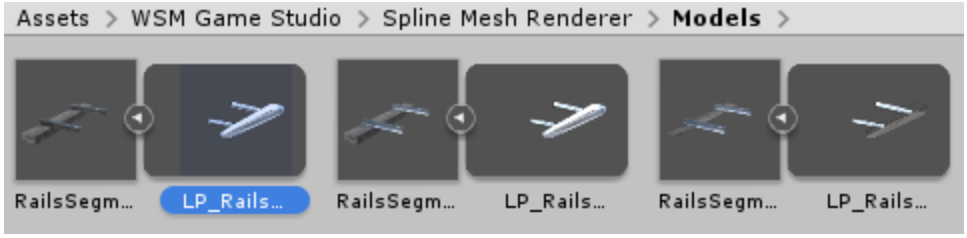

Figure 100 - Sample rails segments LOD levels

You can also generate colliders for your procedurally generated mesh and even apply custom Physics Materials to them if you wish. (optional)

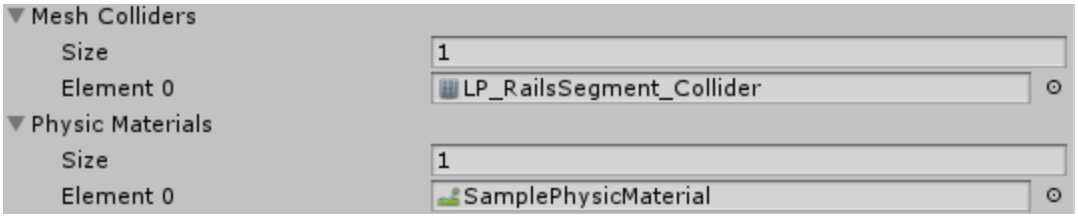

Figure 101 - Profile colliders & physics materials

NOTE: The "Mesh Colliders" and "Physic Materials" list have a one-to-one relationship. Which means that the physics material will be applied on the corresponding mesh collider of the same index.

The sample rails base mesh collider is located in the same folder of the LOD meshes.

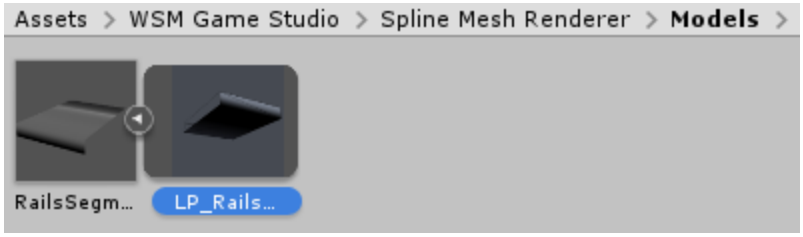

Figure 102 - Sample rails collider

And last, you can set references to start and end caps prefabs, to be spawned on the spline extremities in order to "close" your generated mesh.

Figure 103 - Profile start & end caps

For this sample, some railroad stop blocks were used as start and end caps.

Figure 104 - Start & end caps prefabs

Now that you already know how mesh generation profiles work, all you need to do to create your railroad is to edit your spline and the Spline Mesh Renderer will render the rails along the spline path.

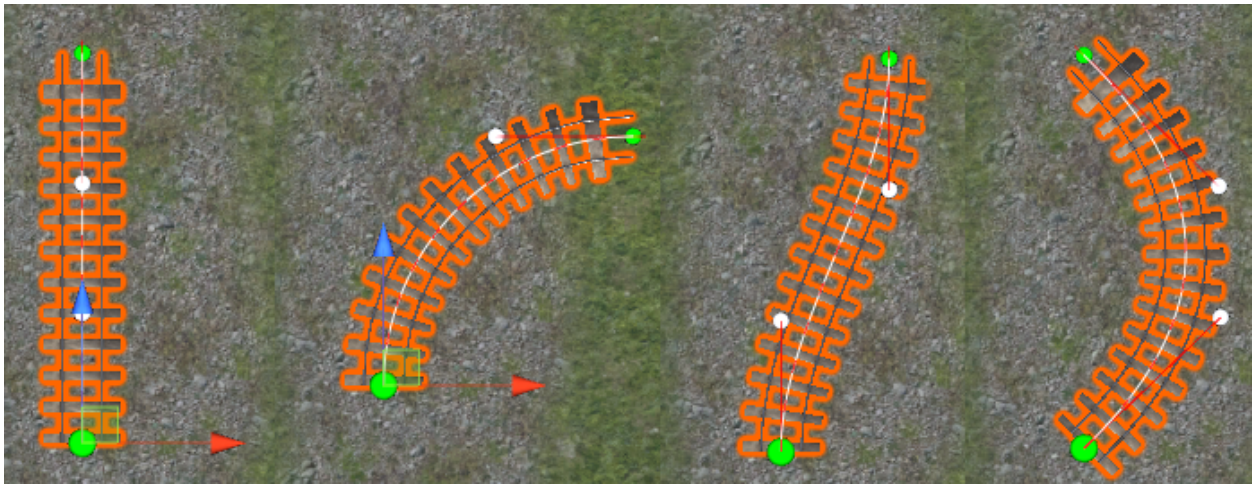

Figure 105 - Curve examples

By adjusting the spline control points normals, we can also apply custom local rotation along the generated mesh.

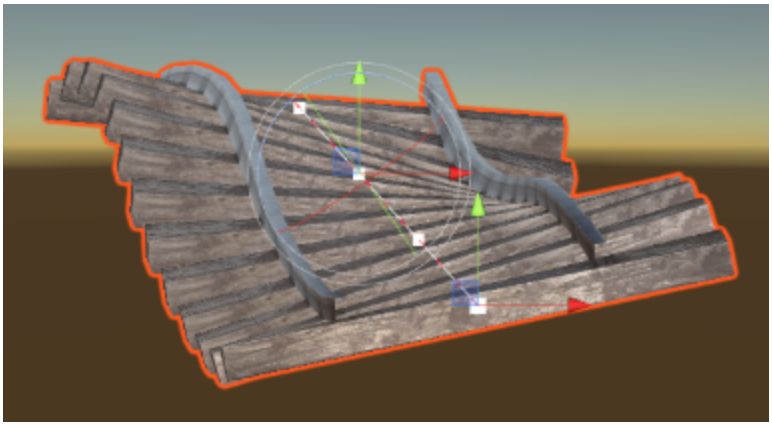

Figure 106 - Custom local rotation

This allows the creation of twisted meshes, as the example below.

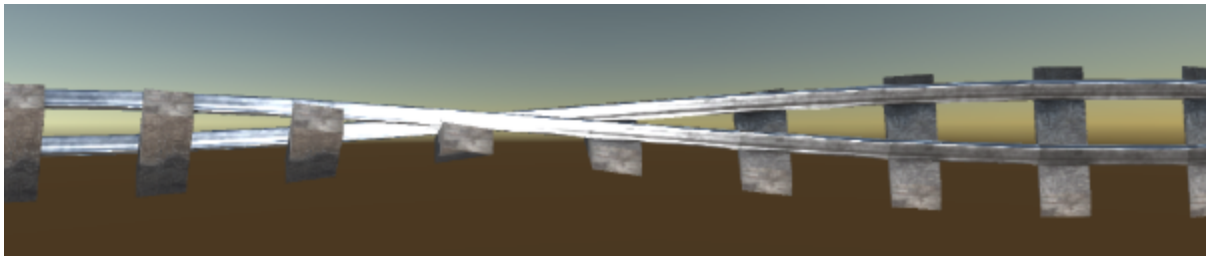

Figure 107 - Twisted mesh sample

Please take a look at the Spline section for more information on how to manipulate the spline shape.

3.2.2 - Mesh Generation Method

The “Mesh Generation Method” property defines how often the mesh procedural generation is executed. You can choose between manual and real time generation mesh generation.

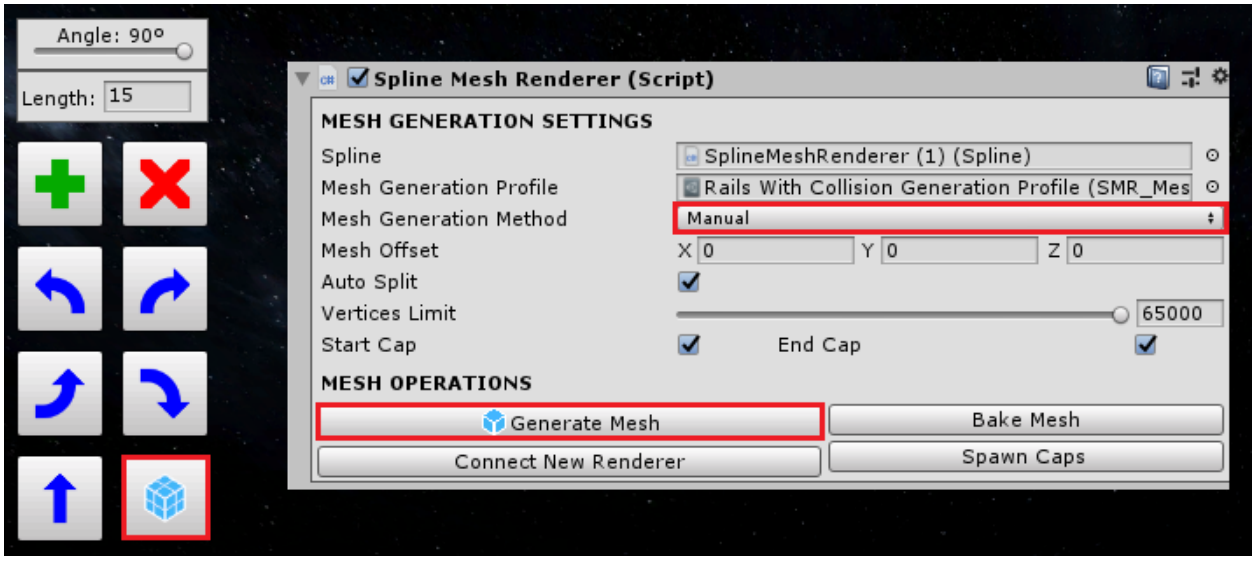

Figure 108 - Manual mesh generation

When the “Manual” generation is selected, every time you change the spline shape, you need to click the Generate Mesh button on the Inspector or on the Quick Access menu at the scene view to update the mesh.

This option is selected by default, since it allows a performance friendly workflow on any computer.

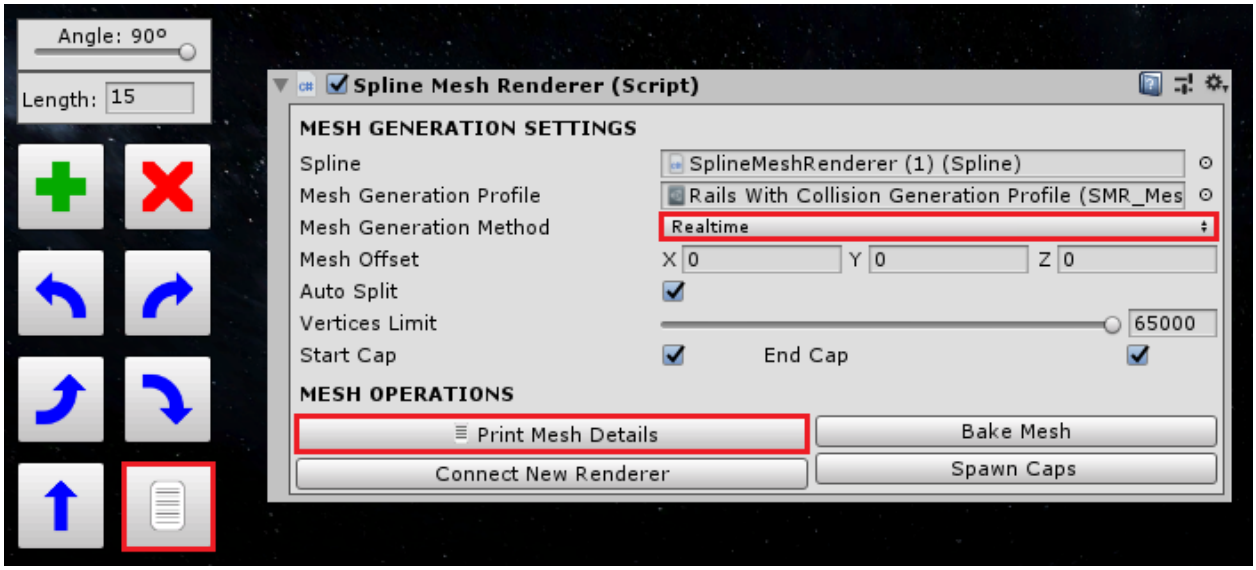

Figure 109 - Real Time mesh generation

When the “Realtime” generation is selected, all changes made to the spline shape are applied instantly to the mesh. However it can be performance heavy when building long meshes.

When this option is selected, the Generate Mesh button is replaced by the Print Mesh Details button on the Inspector and the Quick Access menu.

The real time generation is recommended for fine tuning the mesh shape when needed or when building short mesh segments. Otherwise, always prefer using the manual generation workflow to avoid performance issues on the Unity Editor.

3.2.3 - Mesh Offset

By default, the Spline Mesh Renderer will generate the mesh on top of the spline based on the base mesh model pivot position, but it is also possible to offset the mesh generation by using the “Mesh Offset” property.

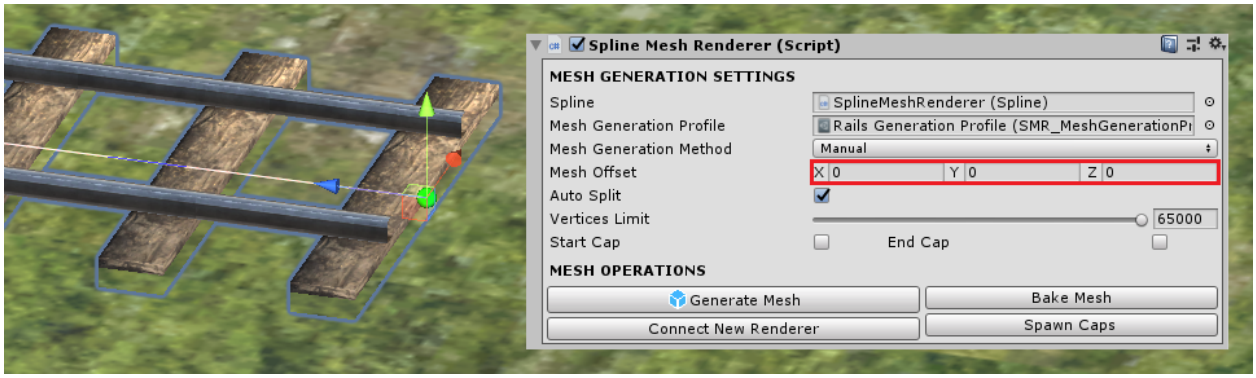

Figure 110 - Mesh offset default value

In the sample, this property was used to slightly raise the mesh to match the terrain elevation. By using this property with the Follow Terrain feature, you can fine tune the mesh position as you wish.

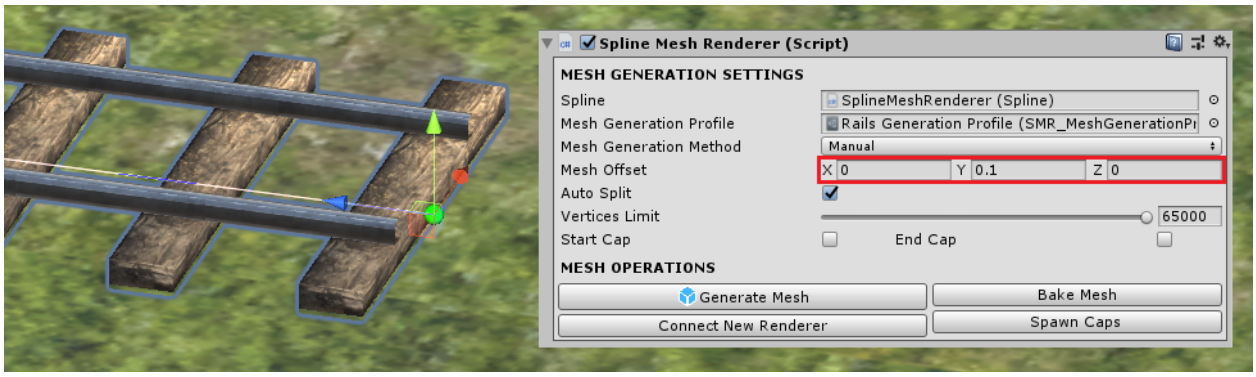

Figure 111 - Mesh offset sample

3.2.4 - Auto Split

Since Unity has a limit of 65000 vertices per rendered mesh, if you are building very long meshes, eventually you may need to split your Spline Mesh Renderer in order to avoid reaching the limit.

You can manually split by using the Split Spline feature. However, in order to make things easier, you can also use the Auto Split feature and control vertices limit for each generated segment.

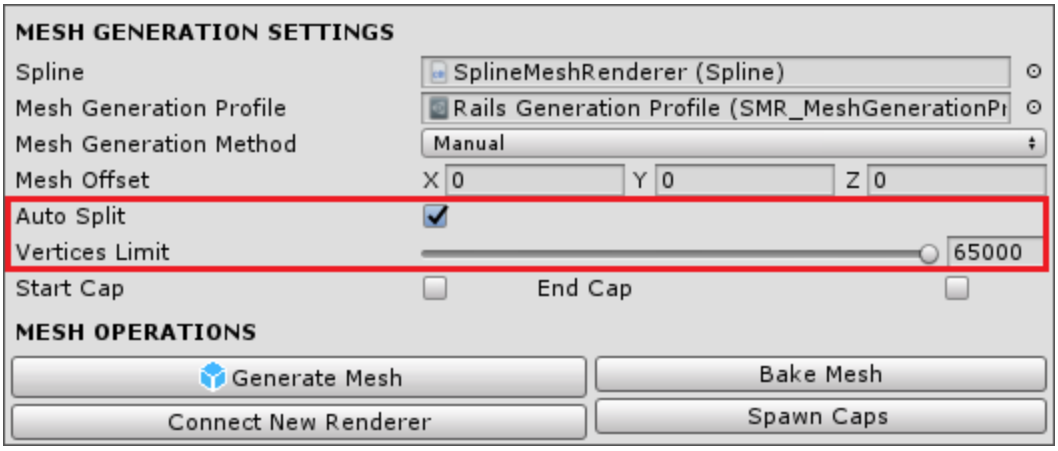

Figure 112 - Auto Split & Vertices Limit

Whenever you create a new Spline Mesh Renderer, the “Auto Split” feature is enabled by default and the “Vertices Limit” is set to the Unity default limit (65000).

3.2.5 - Start & End Caps

Even though the caps prefabs are defined by the Mesh Generation Profile, if you don't wish to include caps on your generated mesh, you can use the "Start Cap" and "End Cap" properties in order to control which caps should be spawned.

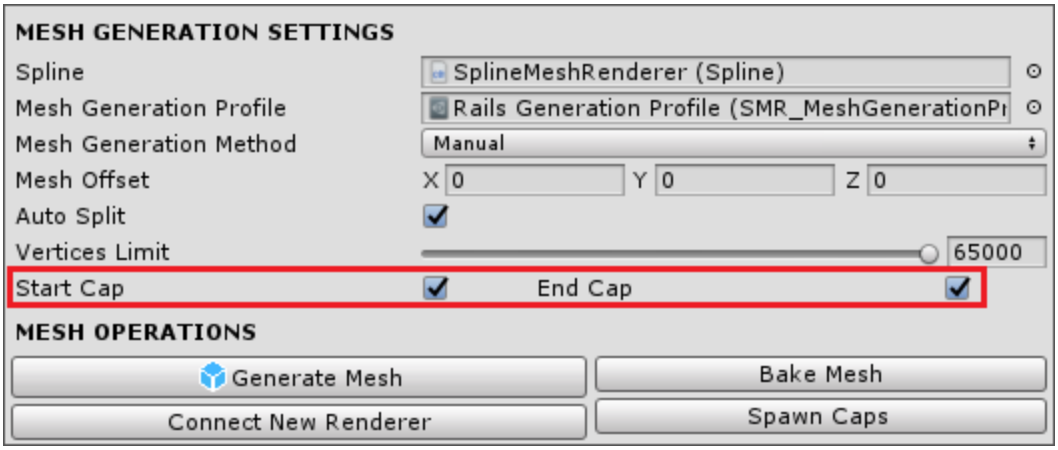

Figure 113 - Start & End cap properties

Whenever you change these properties, you can either regenerate the mesh or use the Spawn Caps operation to apply the changes.

3.3 - Mesh Operations

3.3.1 - Generate Mesh

The "Generate Mesh" operation is available when the manual Mesh Generation Method is selected. This operation submits the changes made on the spline shape to the generated mesh.

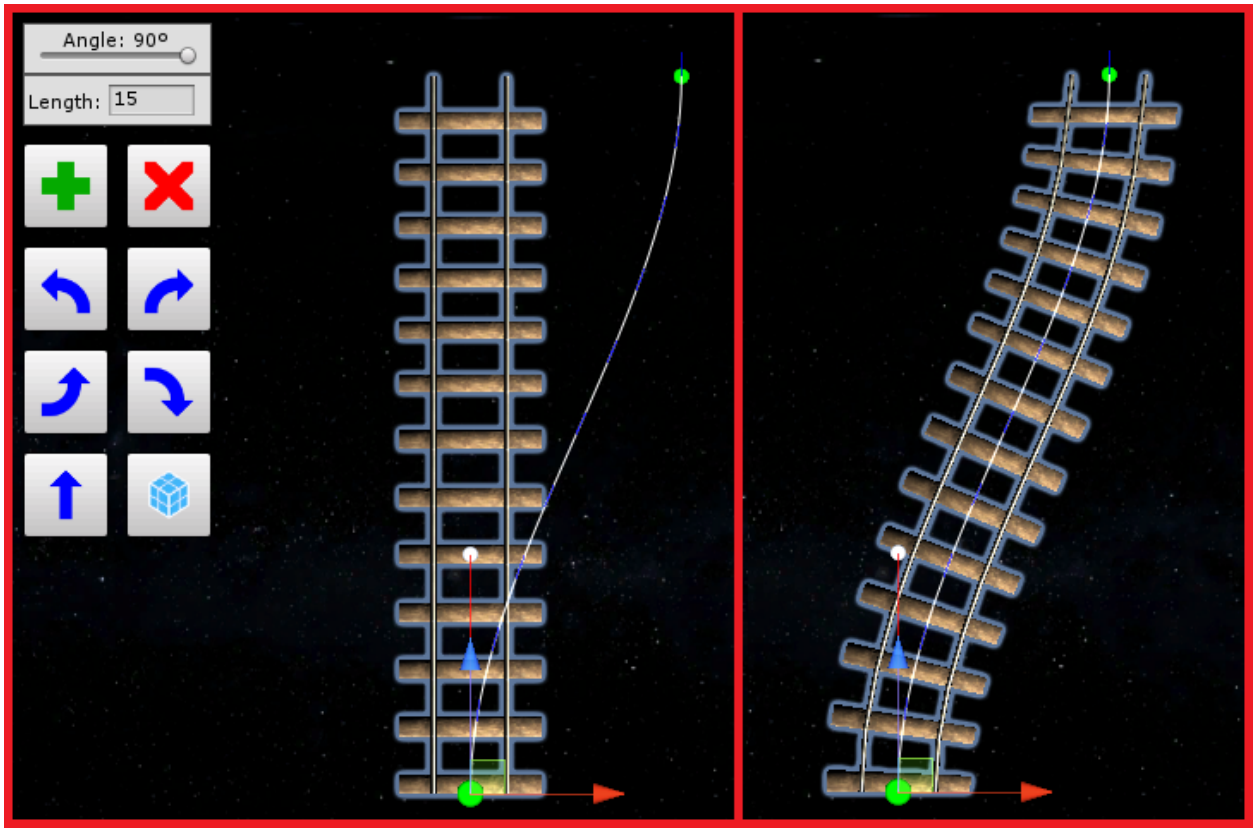

Figure 114 - Before x After “Generate Mesh” operation

In the sample imagem below, you can see the mesh before and after the Generate Mesh operation is performed to submit the changes made to the spline shape to the generated mesh.

The fastest way to use this operation is by clicking on the quick access menu as shown on the sample image above, but it is also available on the Inspector under the Mesh generation Settings tab.

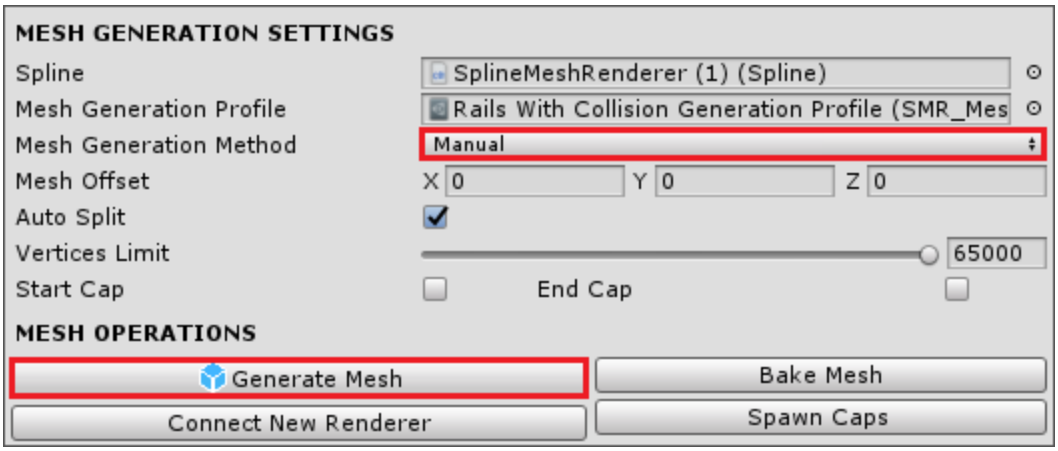

Figure 115 - Generate Mesh on the Inspector

3.3.2 - Print Mesh Details

The “Generate Mesh” operation is available when the real time Mesh Generation Method is selected. This operation prints information about the generated mesh on the Console Window.

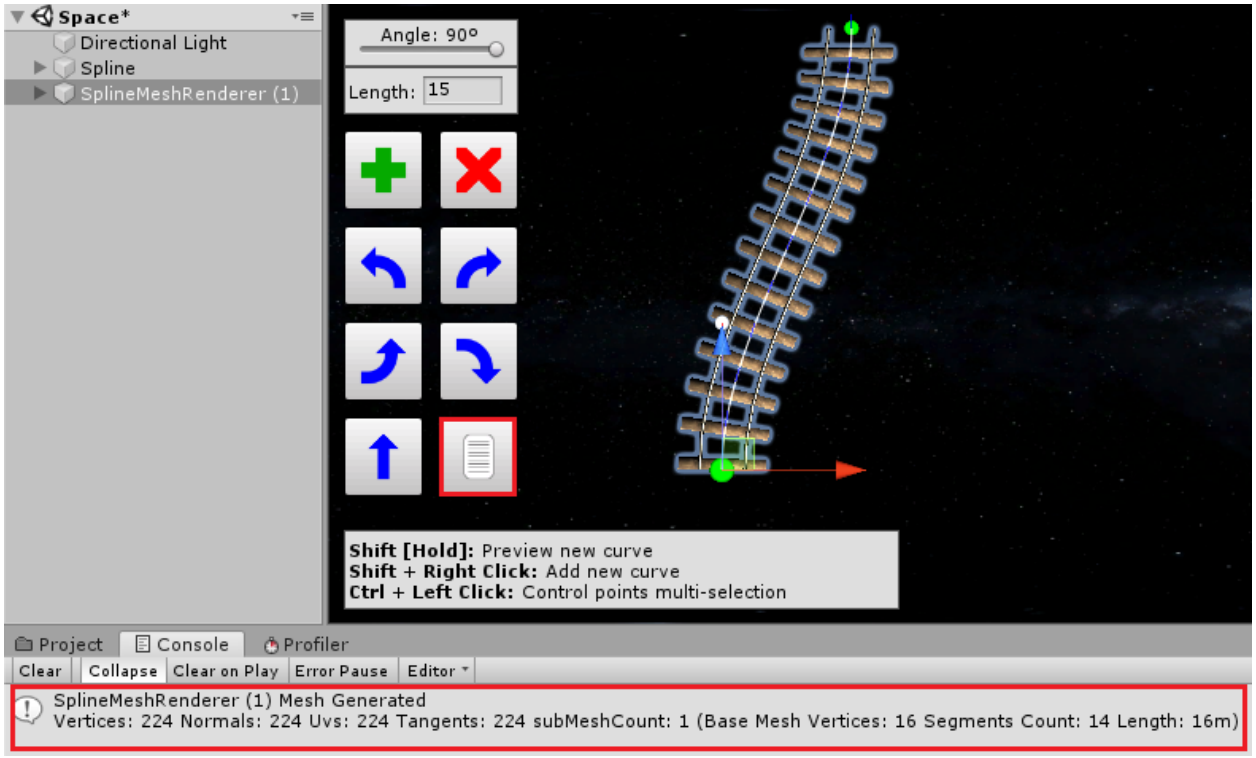

Figure 116 - Mesh details sample

This operation is also available on both the quick access menu and the Inspector.

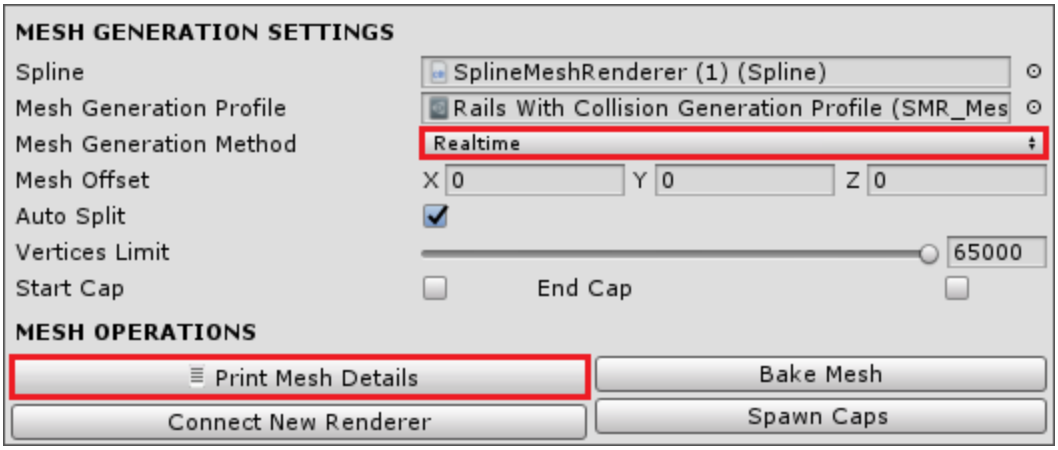

Figure 117 - Print mesh details on the Inspector

NOTE: The mesh details log message is also printed on the console every time the Generate Mesh operation is performed, when the real time generation is selected the log information must be manually printed to avoid too much information being outputted on the console window when editing the mesh.

3.3.3 - Bake Mesh

When you finish editing your mesh, you can bake a static version of it to improve the performance of your game. This feature saves the generated mesh and creates a ready to use prefab that can be used to replace the Spline Mesh Renderer on your scene.

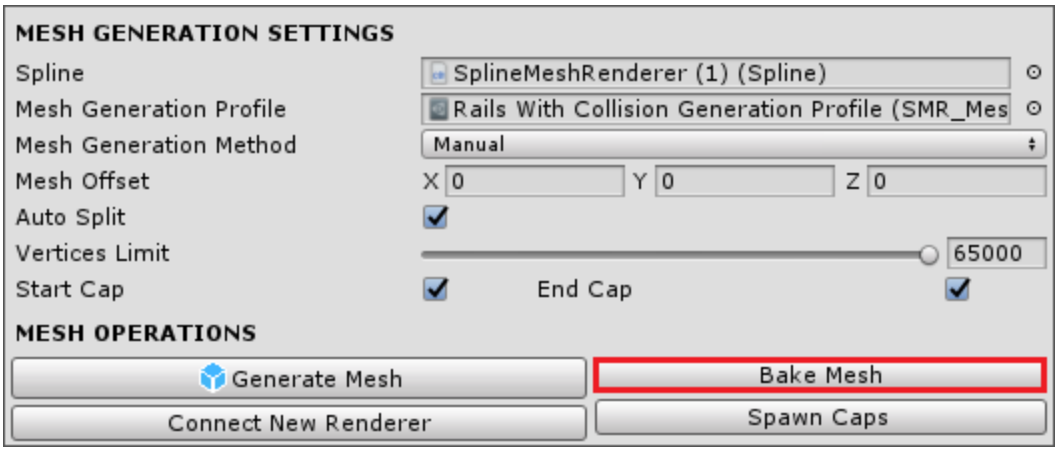

Figure 118 - Bake mesh operation

By clicking on the “Bake Mesh” button the Mesh Baker window will appear.

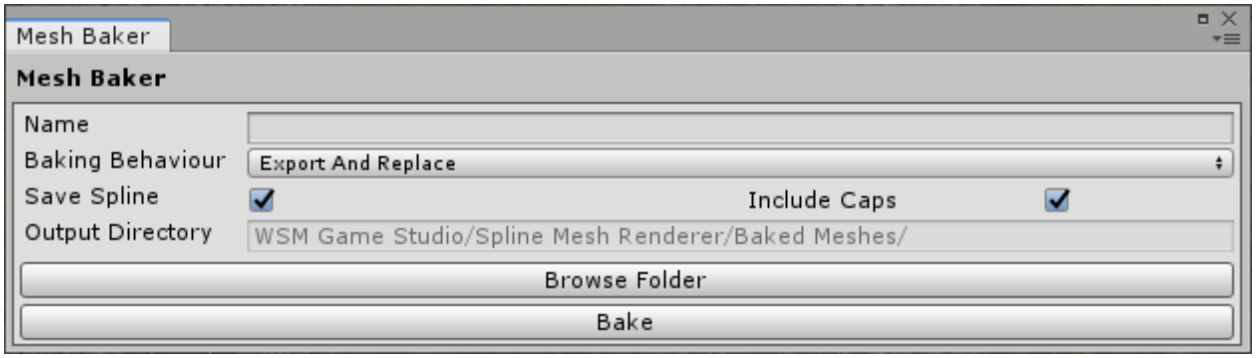

Figure 119 - Bake Mesh window

This window can also be found under the window menu tab “Window > WSM Game Studio > Mesh Baker”.

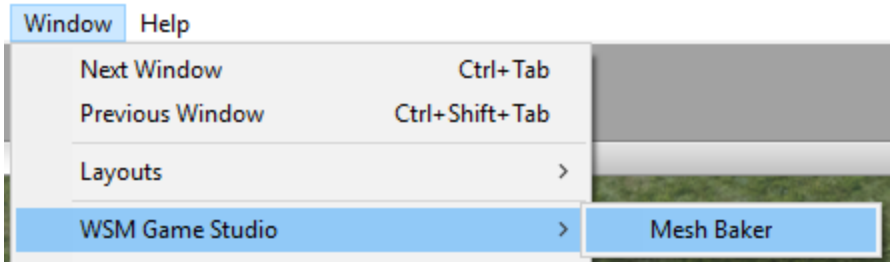

Figure 120 - Mesh Baker menu

By default, the prefabs will be baked into the “Baked Meshes” folder, but you can change the Output Directory by using the “Browse Folder” button.

To start the baking process, all you need to do is select a name for your prefab, decide if you want to save the spline and include the caps or not and click on the “Bake” button.

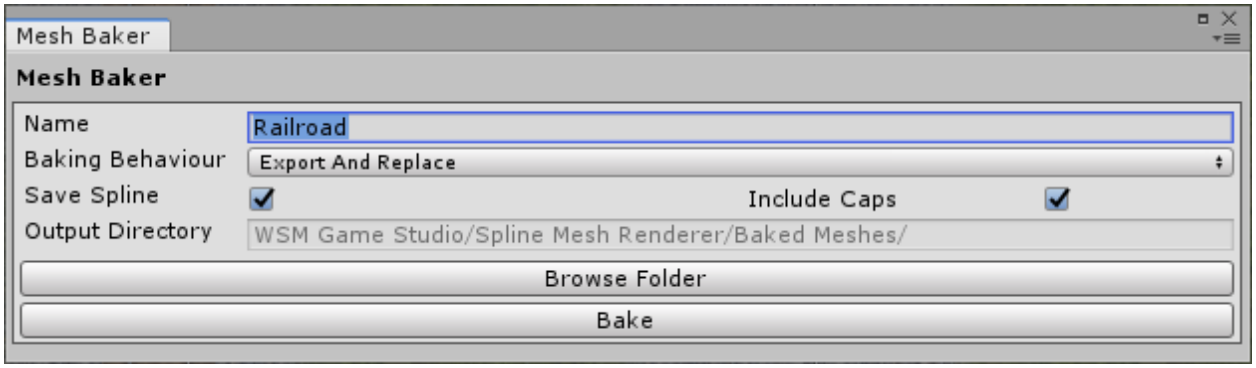

Figure 121 - Mesh baker sample

The “Bake” feature will automatically save the meshes and the prefab on the selected Output Directory. (The meshes are save on a “data” folder alongside the prefab)

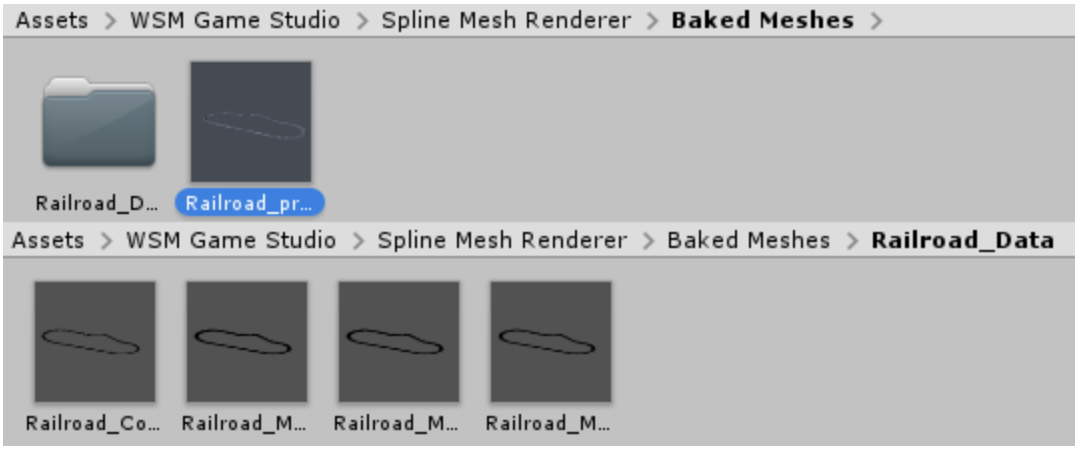

Figure 122 - Baked meshes & prefab

The generated prefab can be used to replace the Spline Mesh Renderer on your scene. This will reduce the scene loading time and increase performance. If you've let the default "Export And Replace" baking behaviour, the spline mesh renderer will be disabled and an instance of the baked prefab will be created on your scene to replace it automatically. If you don't want the automatic replacement to happen, select the "Just Export" option instead.

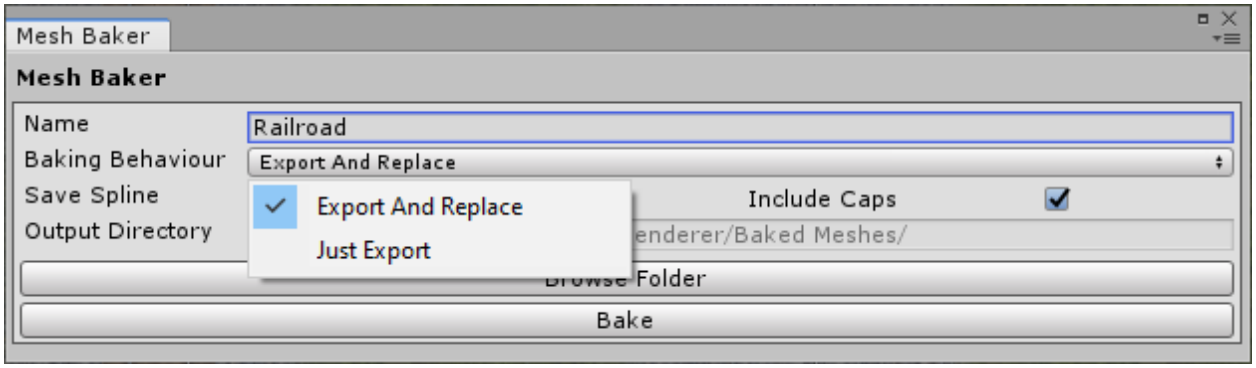

Figure 123 - Baking behaviour

NOTE: If you want to change your mesh in the future you can keep the Spline Mesh Renderer as a disabled object on your scene instead of deleting it. That way you can always adjustments and export the results again.

If you keep the "Save Spline" option enabled, a copy of the spline used to create this object will be saved as a child of the baked prefab.

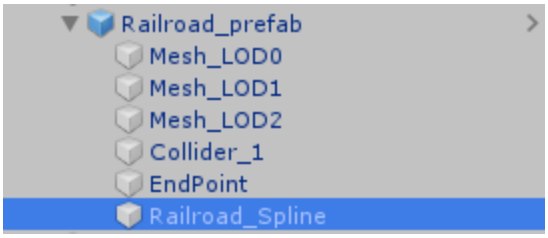

Figure 124 - Spline backup on baked prefab

NOTE: The saved spline is supposed to be a backup of the original spline used to bake the prefab, but it can also be used normally by the Spline Follower and Spline Prefab Spawner components.

3.3.3.1 - Baked Segment Component

Starting with v2.1, all prefabs created by the Mesh Baker will have a Baked Segment component attached to them by default.

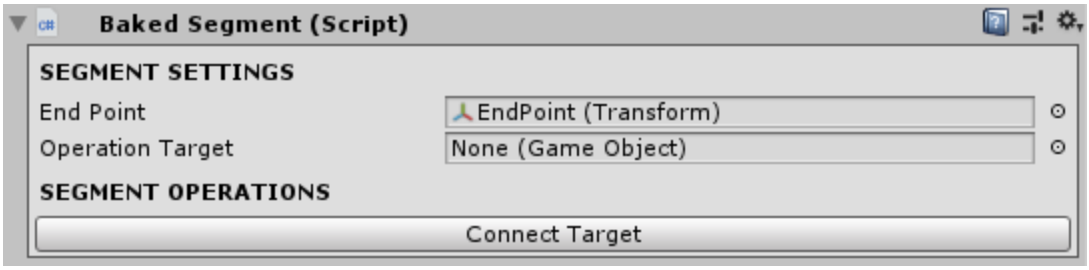

Figure 125 - Baked Segment component

This is an auxiliary component, that works just like the spline Connect Target operation. It was included to make it easier to connect baked mesh segments to each other, or to any splines instances on your scene.

NOTE: The EndPoint property holds a reference to the transform located at the end of the baked segment. It is used as reference by the Connect Target operation. **DO NOT** change the endpoint property reference or move the EndPoint child transform manually.

3.3.4 - Connect New Renderer

The “Connect New Renderer” operation creates a new Spline Mesh Renderer at the end of the current selected one. The newly created object inherits all the characteristics and components from the original object. However, the new spline is generated with only one straight curve segment.

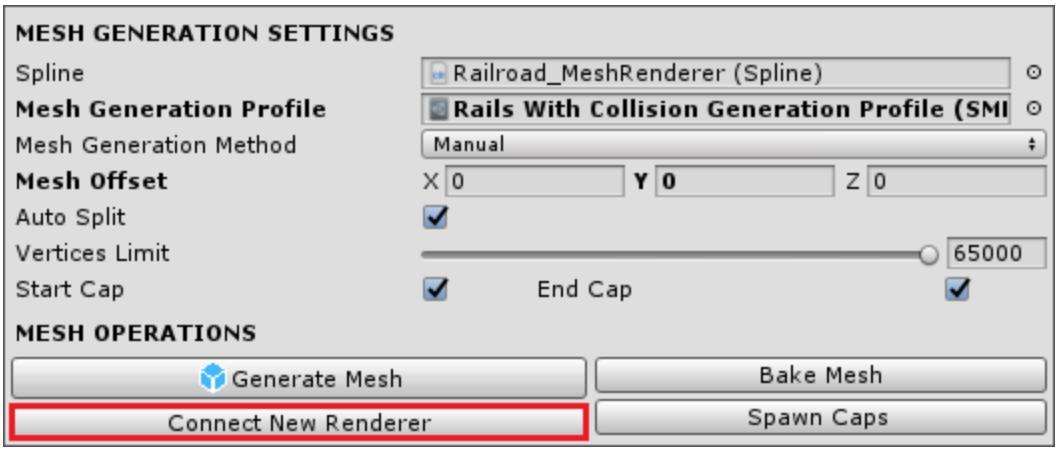

Figure 126 - Connect New Renderer operation

It is very similar to the Append Spline operation, both can be used to produce the same effect.

This feature is very useful if you want to create complex paths, composed of different meshes. For example, you can change the base mesh to create a railroad segment with a different kind of rails.

The new Mesh Renderer will always perfectly connect to the end of the previous one. Like can be seen in the sample image below:

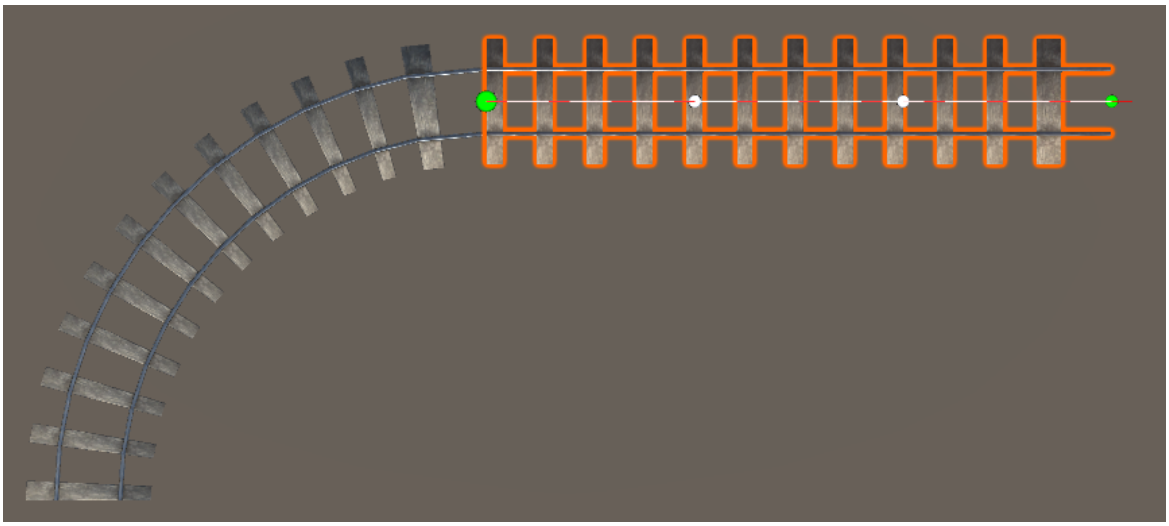

Figure 127 - Connect New Renderer sample

3.3.5 - Spawn Caps

Whenever you need to update the caps without regenerating the mesh, you can use the “Spawn Caps” operation in order to do that. This can be very useful, if you’ve enabled or disables the start or end caps, but haven’t made any changes to the spline.

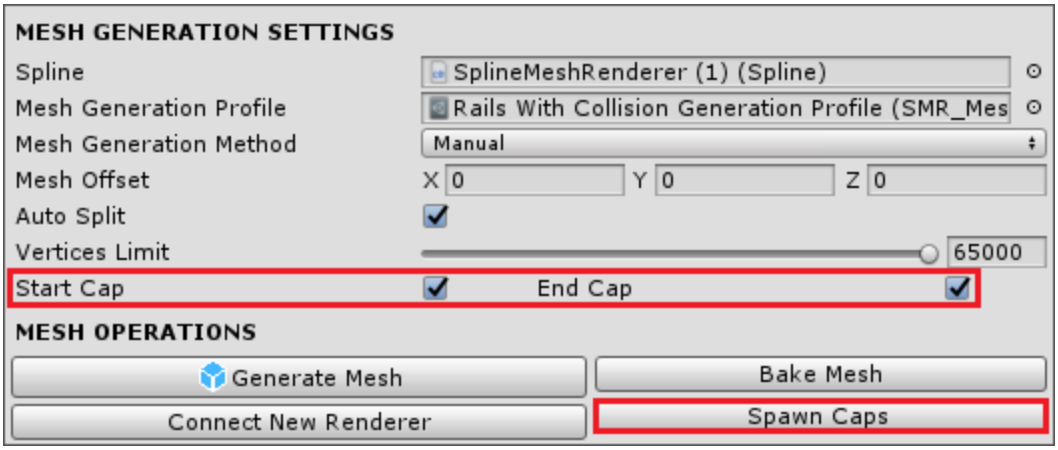

Figure 128 - Spawn Caps operation

3.5 - Mesh Vertices Limit

If you intend to build super long meshes for open world games for example, eventually you may see a warning like this:

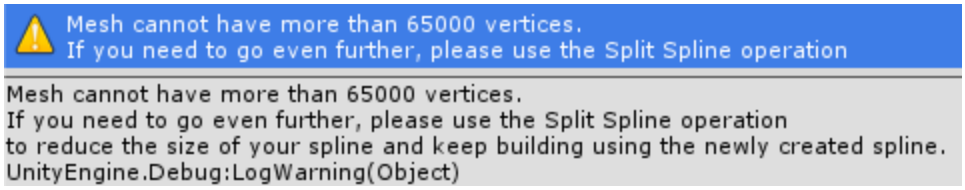

Figure 129 - Vertices limit reached warning

This happens because Unity has a limit of 65000 vertices per mesh. This limit exists to avoid performance issues when rendering large meshes. This limit is also very useful to apply Occlusion Culling on your railroad and increase your game performance. If you see this warning, just follow the instructions below to keep building your super long mesh.

> **NOTE:** The process described below is only necessary if the Auto Split feature is disabled, as it is a step-by-step guide on how to manually split long meshes manually to avoid the vertices limit.

With the Spline Mesh Renderer object selected, go to the end of your mesh and check if the mesh ending matches the last control point. It probably doesn't match, like in the sample image below.

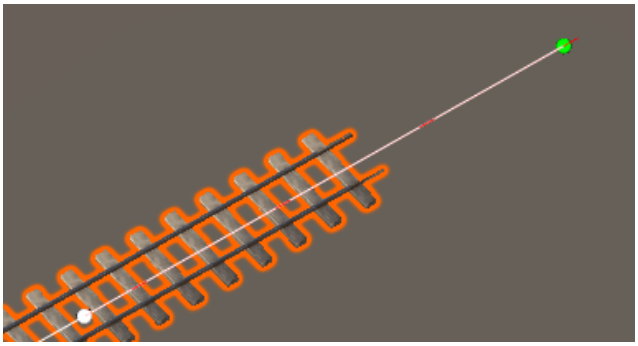

Figure 130 - Vertices limited rendering at the end of spline

To solve this issue and keep building, all you need to do is to select the last control point that was successfully rendered.

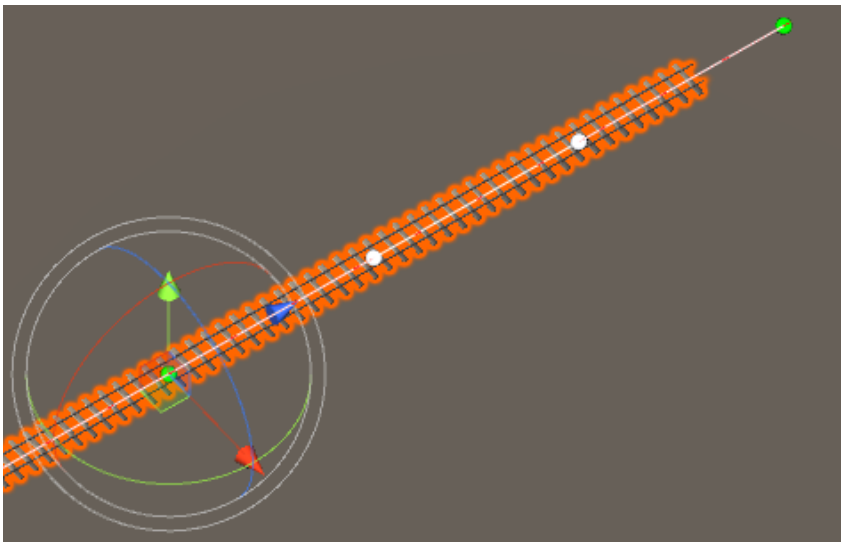

Figure 131 - Last successfully rendered control point selected

And then use the Split Spline operation to start a new section for your mesh.

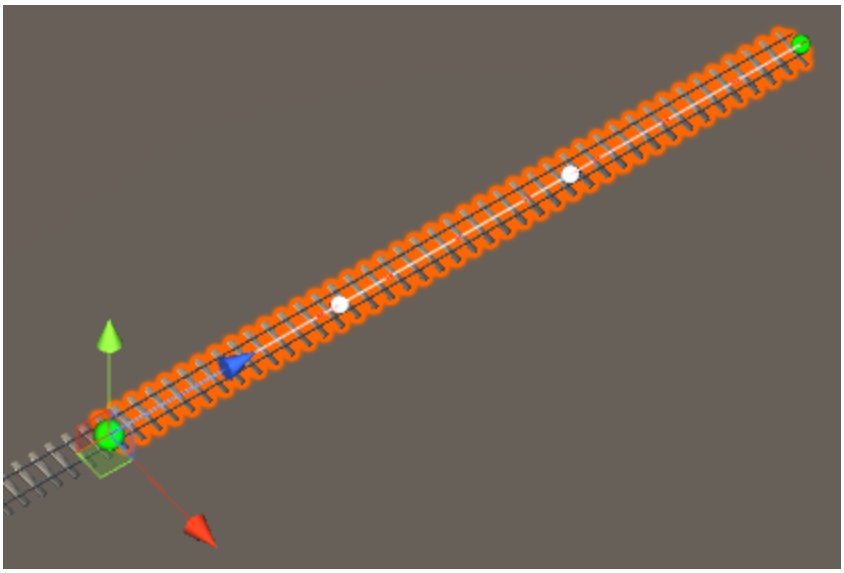

Figure 132 - New section created using the Spline Spline operation

NOTE: Real time mesh generation is disabled automatically when the max number of vertices is greater than 65000. This is the default behaviour to increase the editor performance while building your mesh, since processing large meshes data in realtime is costly.

You can change back to the real time mesh generation manually after reducing the rails, but, for very long meshes it's recommended to use the manual mesh generation button to update the mesh after adjusting your splines curves.

The max length of each segment depends on the base mesh vertices count. For example, the base rail mesh included in the package, has 56 vertices in total. By using it, it is possible to create railroad segments up to 1.35 Km each. To create railroads longer than 1.35 Km, use the technique described above to create multiple railroad segments connected to each other.

4 - Spline Prefab Spawner

The “Spline Prefab Spawner” component is a level design tool that allows you to easily spawn prefabs along a spline path to populate your scene.

It can be used to easily populate your scene with objects that repeat along a path, such as light poles, bridge pillars, check points, etc.

4.1 - Creating a Spline Spawner

You can create a new Spline Prefab Spawner by using the default object creation menu:

- **Hierarchy Window**
  - Right Click on Hierarchy window
  - WSM Game Studio > Spline Prefab Spawner
- **GameObject Menu**
  - Click on GameObject menu
  - WSM Game Studio > Spline Prefab Spawner

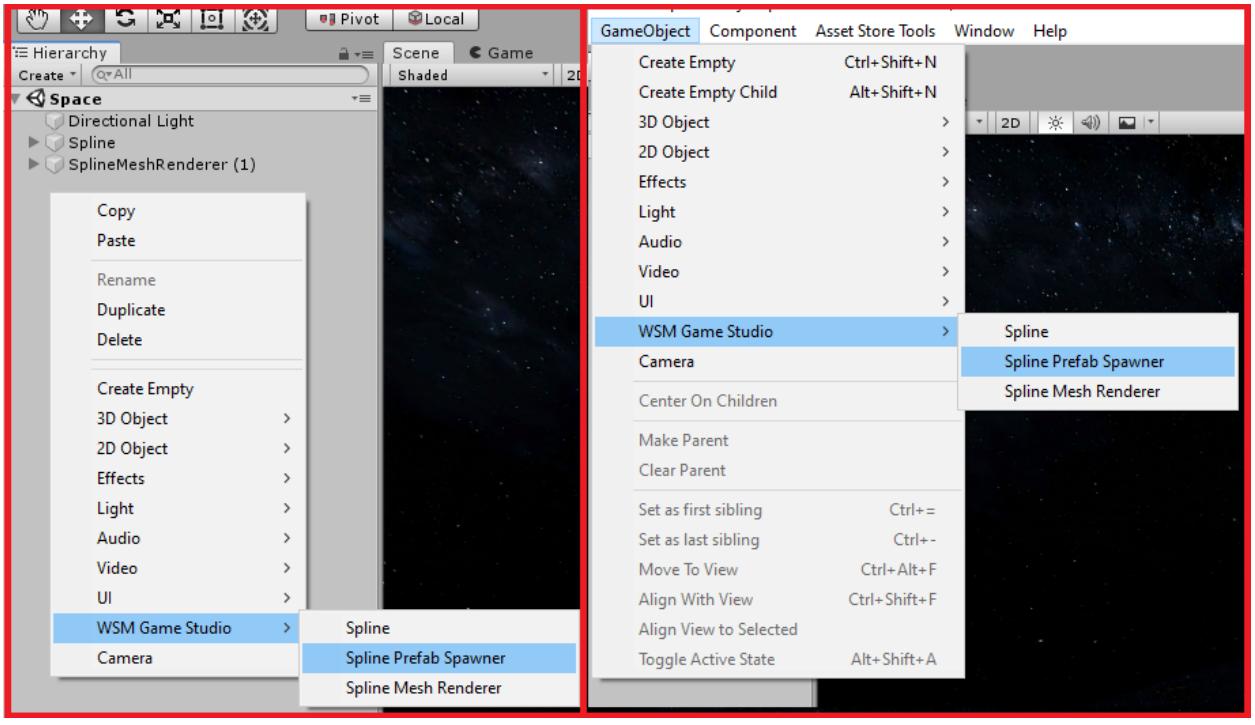

Figure 133 - Creating a new spline prefab spawner

Alternatively, you can also Drag & Drop the “SplinePrefabSpawner” prefab to your scene. This prefab is located under “Assets/WSM Game Studio/Spline Mesh Renderer/Prefabs/Legacy”.

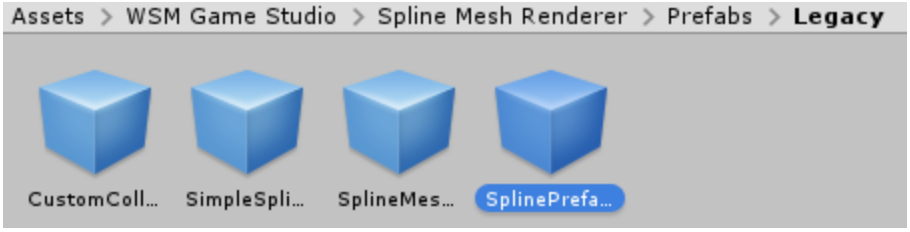

Figure 134 - Spline Mesh Renderer prefab

NOTE: Prior to version 2.1, this prefab was used to create new spline prefab spawner instances. It is not needed for this purpose anymore, however it was kept for the sake of backwards compatibility.

You should never change this prefab directly or save alterations made on it. It was created with the sole purpose of being used as a starting point.

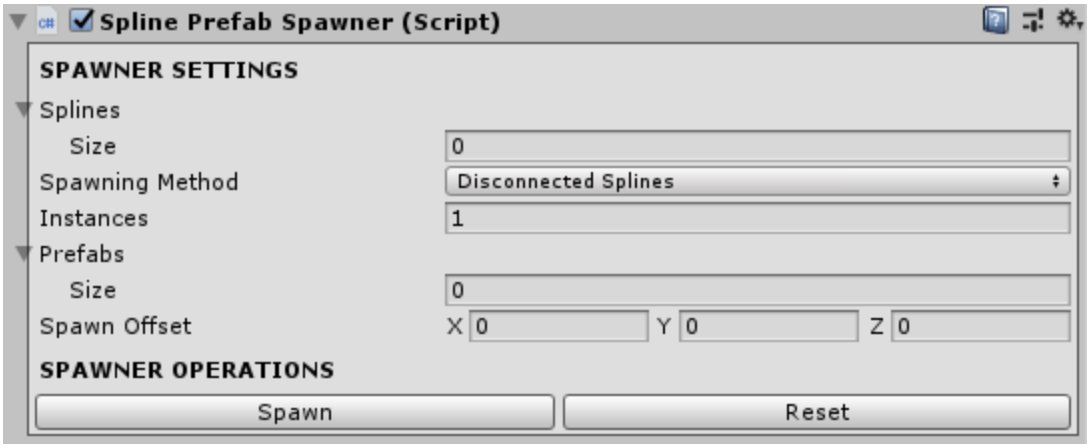

Figure 135 - Spline Prefab Spawner component

The “Splines” property holds a list of the target splines used as reference by this spawner.

The “Spawning Method” is used to identify if the target splines are connected or disconnected from each other, and avoid duplicates in case they are connected.

The “Instances” property defines how many instances of each prefab will be spawned along the spline.

The “Prefabs” property is a list of all the prefabs you want to spawn.

By default, the instances will be spawned on top of the spline, but you can use the “Spawn Offset” property to change the position of the spawned object if you wish so.

4.2 - Spawning Prefabs

Spawning prefabs is really simple, just fill the Prefabs and Instances property on the Inspector and click on the “Spawn” button.

If the Follow Terrain property of the reference Spline is enabled, the spawned objects will follow the terrain elevations as well.

In the sample image below, it was used to spawn 13 light poles along a single spline.

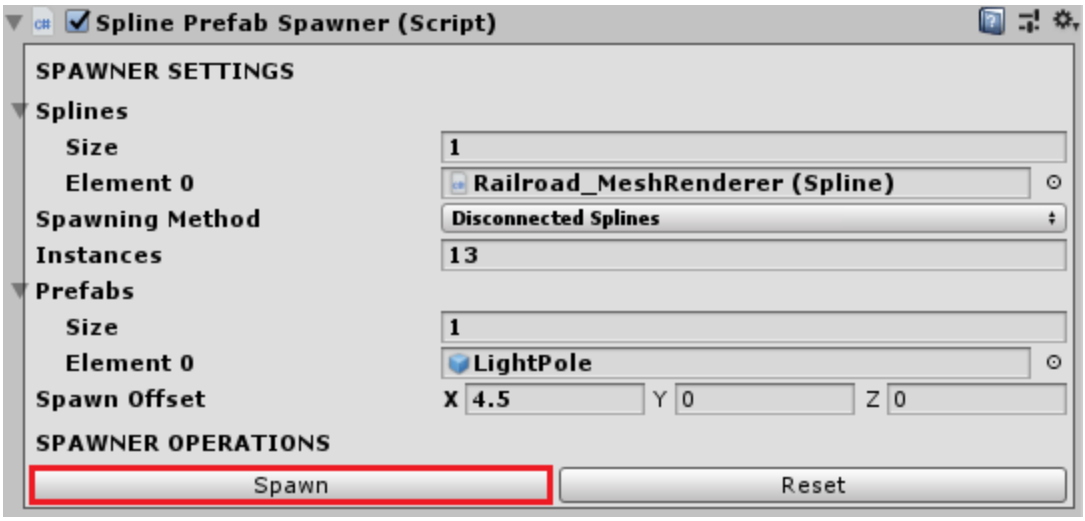

Figure 136 - Spline Prefab Spawner sample

The result can be seen on the image below. Note how the light poles were instanced based on the terrain elevation.

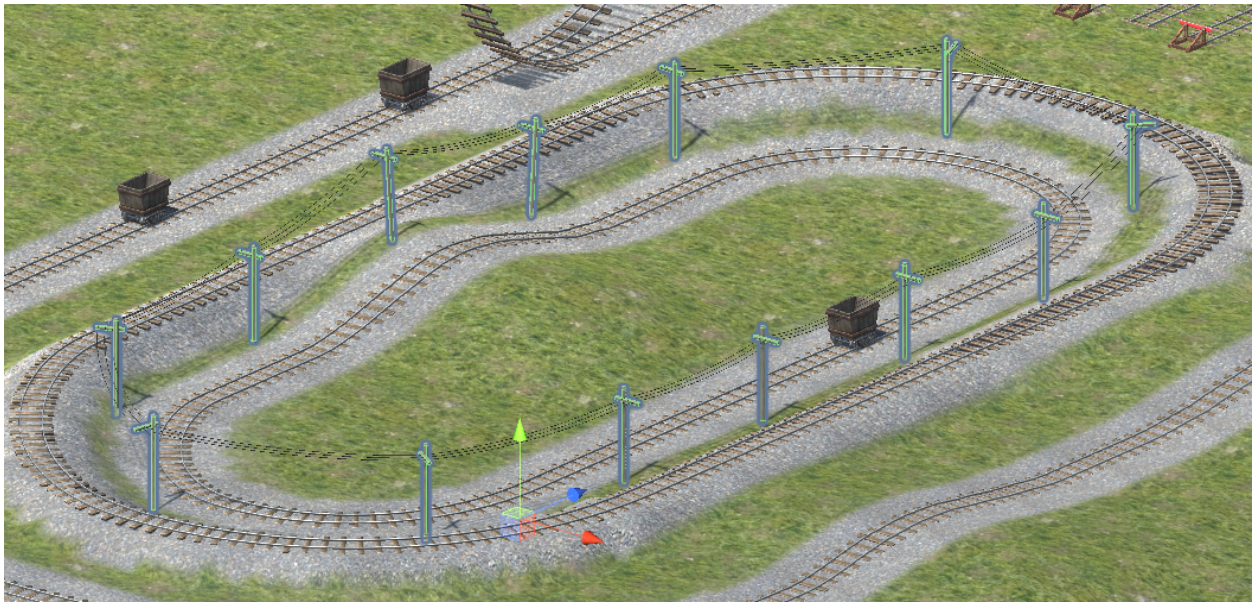

Figure 137 - Spawned light poles

If you decide to change the number of instances after spawning, there is no problem, just change it and click the Spawn button again and it will automatically respawn the new number of instances.

NOTE: All spawned instances are set as children of the Spline Prefab Spawner object. This is the default behaviour, to keep the objects Hierarchy organized and to easily identify what objects were spawned using the Spline Prefab Spawner component.

4.3 - Deleting Prefabs

Deleting spawned prefabs is also pretty simple. All you need to do is click the “Reset” button and all spawned instances will be deleted.

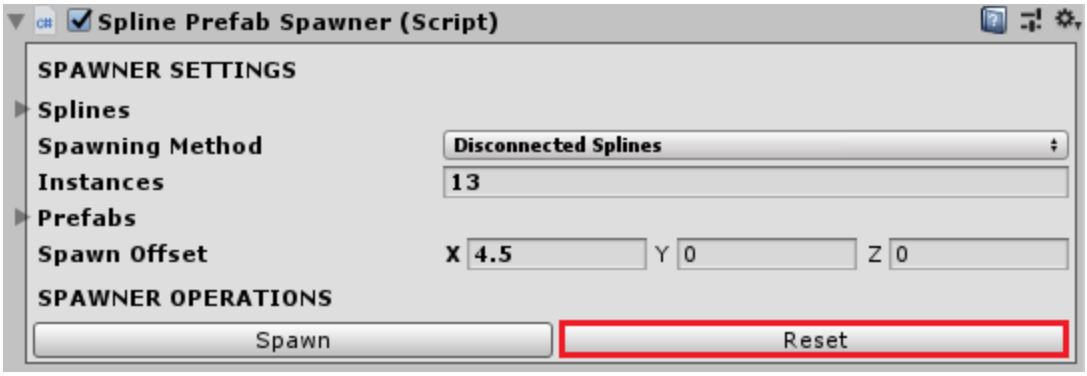

Figure 138 - Reset operation

NOTE: Spline Prefab Spawner component identifies what objects were spawned by it by looking at his children on the Hierarchy. **DO NOT** set other objects in your scene as children of an object that has a Spline Prefab Spawner component attached to it to avoid deleting objects unintentionally.

5 - Spline Follower

The “Spline Prefab Spawner” component allows any object on your scene to follow a path defined by spline. It can be used to easily create dynamic moving props for visual or gameplay purposes, such as complex platforms movement, orbiting behaviour, etc.

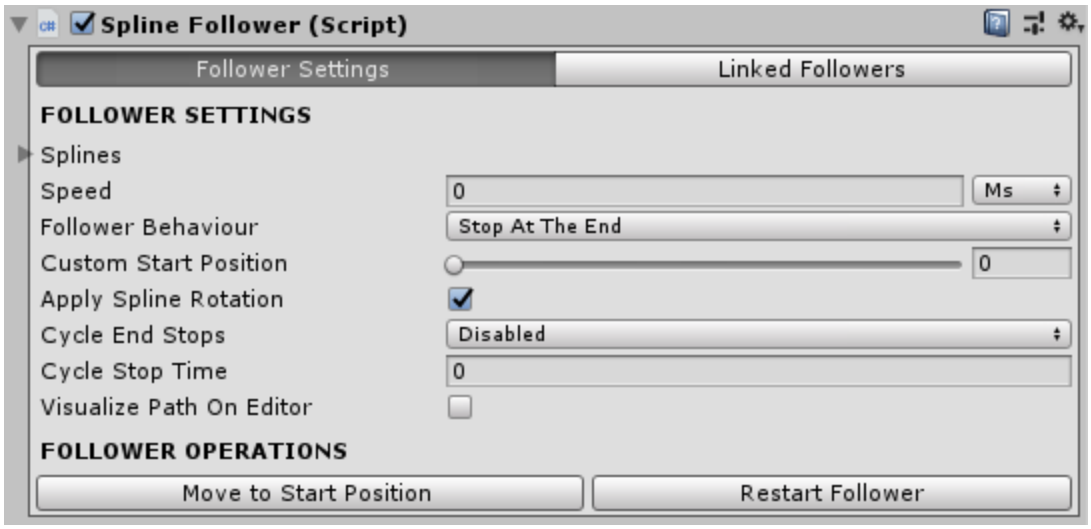

Figure 139 - Spline Follower component

5.1 - Creating a Spline Follower

To add the spline following behaviour to any object all you need to do is to add a “Spline Follower” component to it.

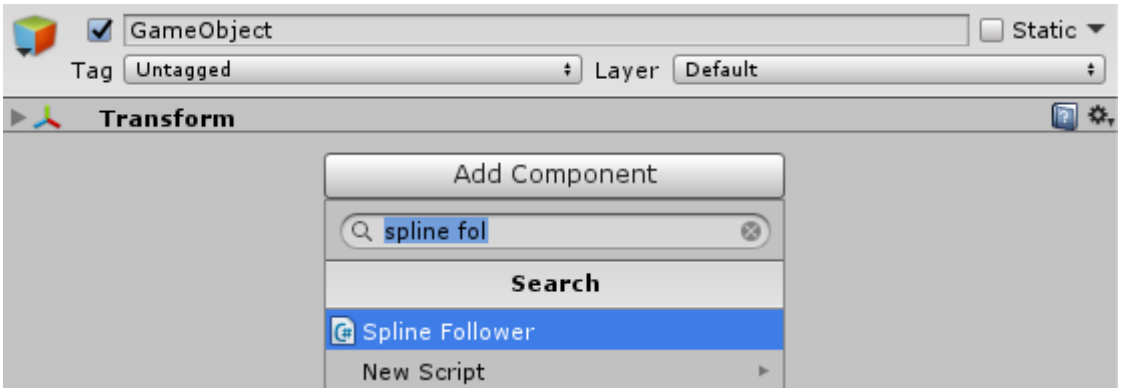

Figure 140 - Adding the Spline Follower component

It is possible to follow one or more splines. The “Splines” Property defines what splines will be followed. For multiple splines, it will always consider the order of the splines on the list.

5.2 - Following Speed

The “Speed” Property defines how fast the object will move regardless of how many splines it will follow. There are four speed units available:

- Meters per second (ms)
- Kilometers per hour (kph)
- Miles per hour (mph)
- Knots (kn)

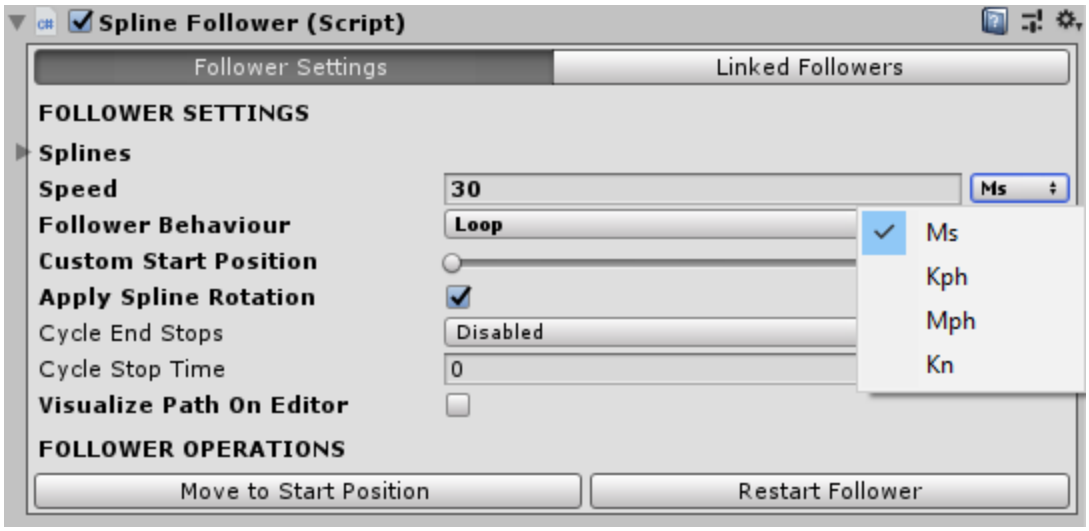

Figure 141 - Spline Follower speed units

5.3 - Following Behaviour

The “Follower Behaviour” Property defines how the object should behave once it reaches the end of the spline list.

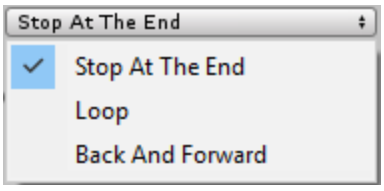

Figure 142 - Follower Behaviour options

The “Apply Spline Rotation” Property defines if the object should rotate while moving along the spline. This property is enabled by default, if you wish to create moving platforms for example you need to disabled this option to keep the platforms from rotating.

5.4 - Start Position

The “Custom Start Position” Property defines the percentage of the first spline the object will use as a start point position. For example, if you wish the object to start at the middle of the spline, you can set this property to 50%.

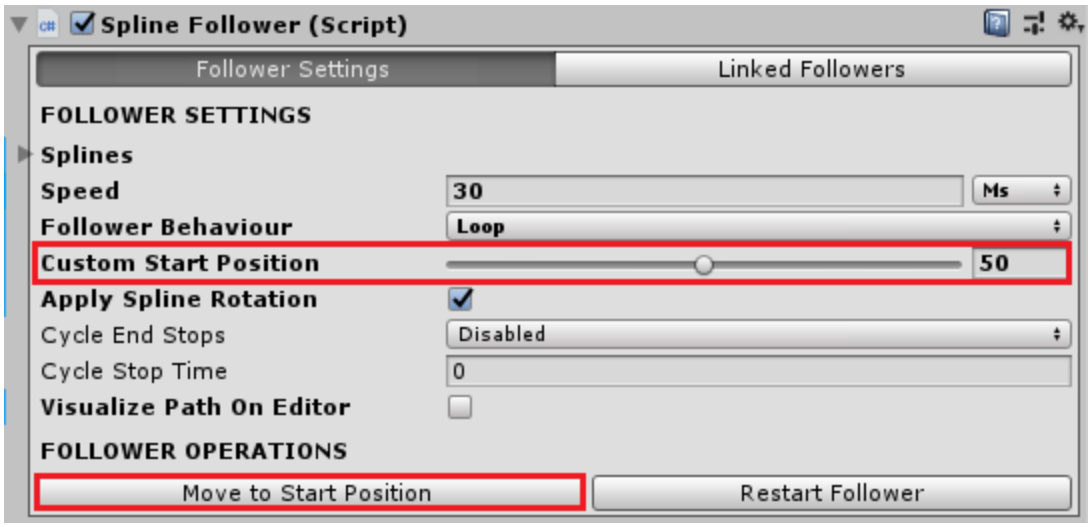

Figure 143 - Start position properties

After adjusting the position click the “Move to Start Position” button to preview where the object will start.

5.5 - Stops

The “Cycle End Stops” Property defines if and when the object should make temporary stops along the way.

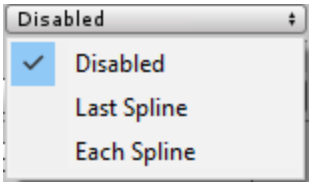

Figure 144 - Cycle End Stops options

The “Cycle Stop Time” Property defines how long temporary stops will last (in seconds).

5.6 - Linked Spline Followers

Linked followers are objects that follow a “master” Spline Follower by a fixed offset.

Figure 145 - Linked Followers Sample

One master can control multiple linked followers, creating a relationship very similar to wagons following a locomotive or roller coaster carts linked together.

> **NOTE:** If you’re looking for a complete train solution for your project, please take a look at the Train Controller asset collection.

To add the linked follower behaviour to any object all you need to do is to add a “Spline Linked Follower” component to it.

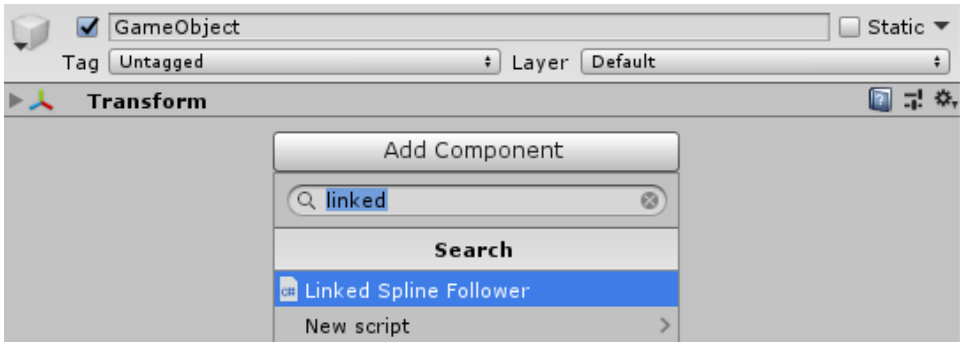

Figure 146 - Adding the Spline Linked Follower component

The Linked Follower component is very simple, all you need to define is the following offset that will be applied when following the master.

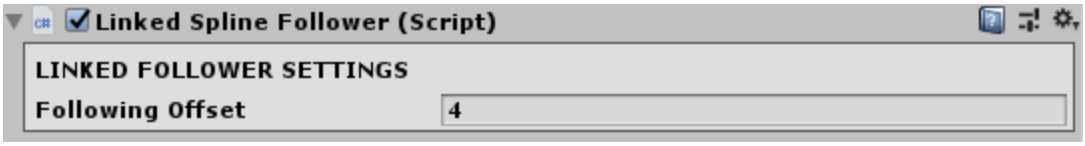

Figure 147 - Linked Spline Follower Component

Linked followers need to be referenced on the master under the Linked Followers tab. “Followers offset” property is the offset between the master and the first follower. By clicking on “Move to Start Position”, you can preview the starting position of all linked followers.

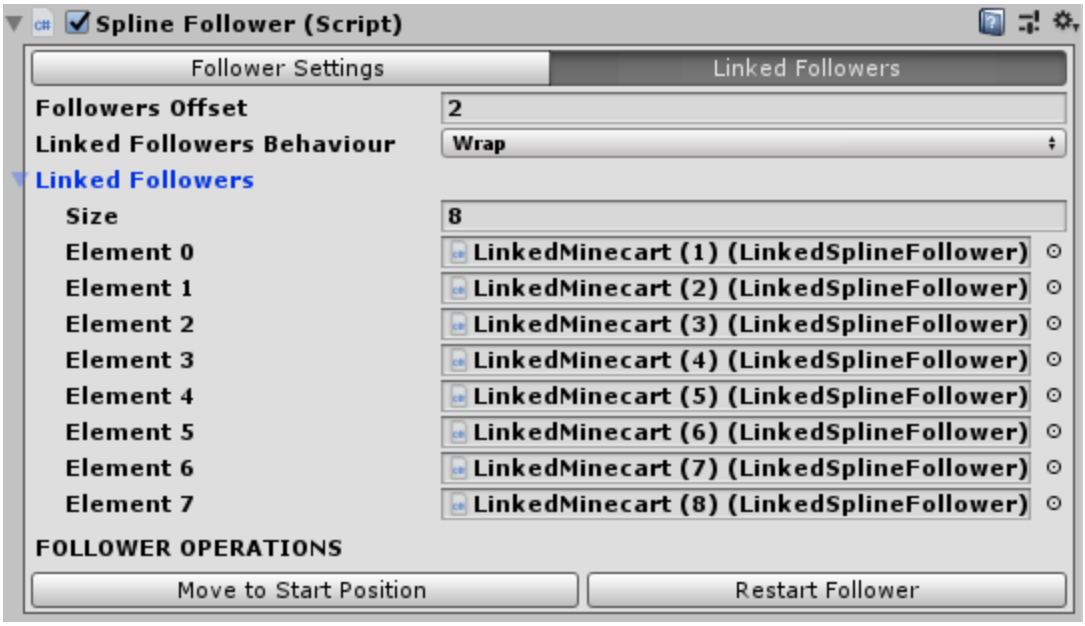

Figure 148 - Linked Followers references

The “Linked Followers Behaviour” defines what happens to linked followers when they are out of spline bounds. For example, if the mast follower is at the start of spline, since the the linked followers are always behind the master, they will be outside of the spline.

In this situation, you can choose between sending the linked followers to the end of the spline (wrap) or to align them on a straight line behind the master (overflow).

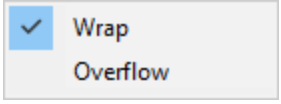

Figure 149 - Linked Followers Behaviour

NOTE: Whenever following closed splines loops it is recommended to use the “wrap” behaviour for better results.

6 - Practical Use Samples

This section contains some practical uses samples created with the Spline Mesh Renderer, Spline Follower and Spline Prefab Spawner.

6.1 - Minecarts Demo Scene

At the first demo scene, the Spline Follower and Spline Mesh Renderer were used to create a complex railroad path for minecarts.

Figure 150 - Minecarts sample

This demo demonstrates the power of the spline components when used together to create complex scenes. Note how the minecarts correctly follow the loop, jump and terrain elevations.

This complex multi-spline railroad was first created as a single Closed Loop spline, then it was divided into smaller segments using the Split Spline operation.

Each segment was then configured accordingly to the desired behaviour. The control point normals on the top segment of the vertical loop were customized in order to set the relative upwards direction of this spline segment.

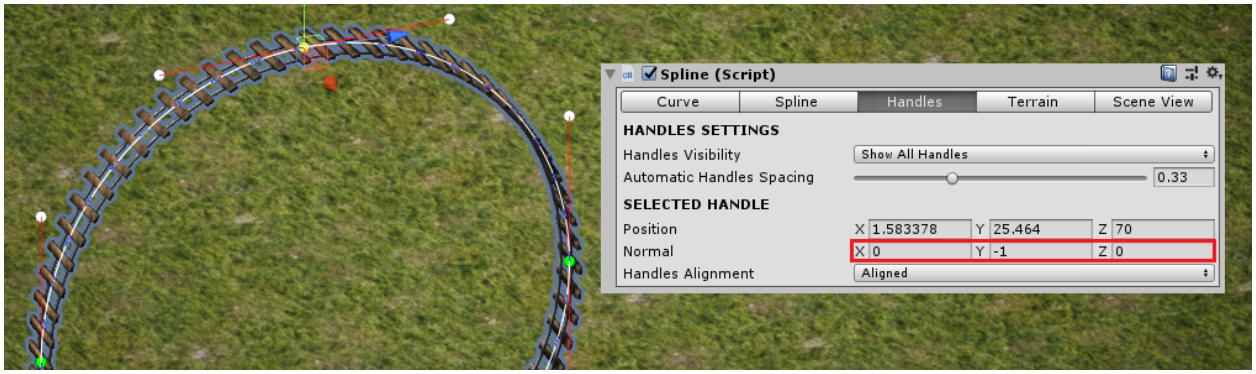

Figure 151 - Vertical loop settings

An empty generation profile was applied to simulate the “jump” behaviour on one of the splines.

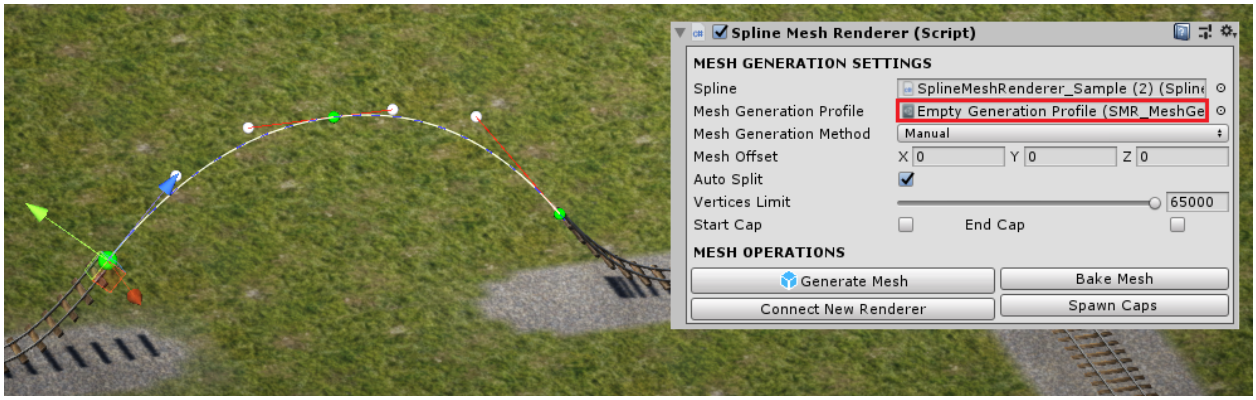

Figure 152 - Jump settings

The Follow Terrain feature was enabled on the segments that went through terrain elevations.

Figure 153 - Follow terrain

The Terraforming feature was used to create a rock track ballast for the railroad.

Figure 154 - Terraforming

The Spline Prefab Spawner was used to instantiate some light poles.

Figure 155 - Light poles

Linked Spline Followers were also used to connect several minecarts.

Figure 156 - Linked Spline Followers

6.2 - Solar System Demo Scene

At the “Follower_SolarSystem” demo scene the Spline Follower component was used to create planetary orbit behaviour.

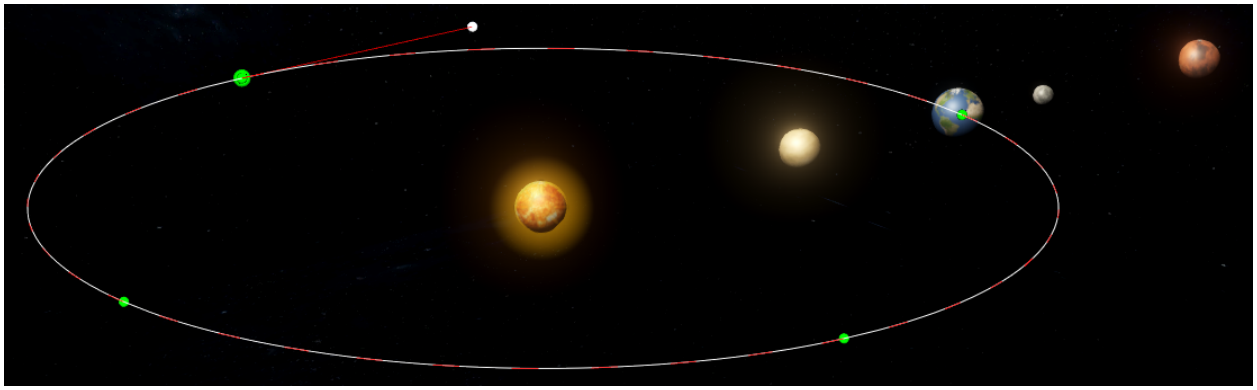

Figure 157 - Solar system sample

The Curve Shaping Operations were used to create perfect circular splines for the orbits. The orbits circles are composed of four 90º curves.

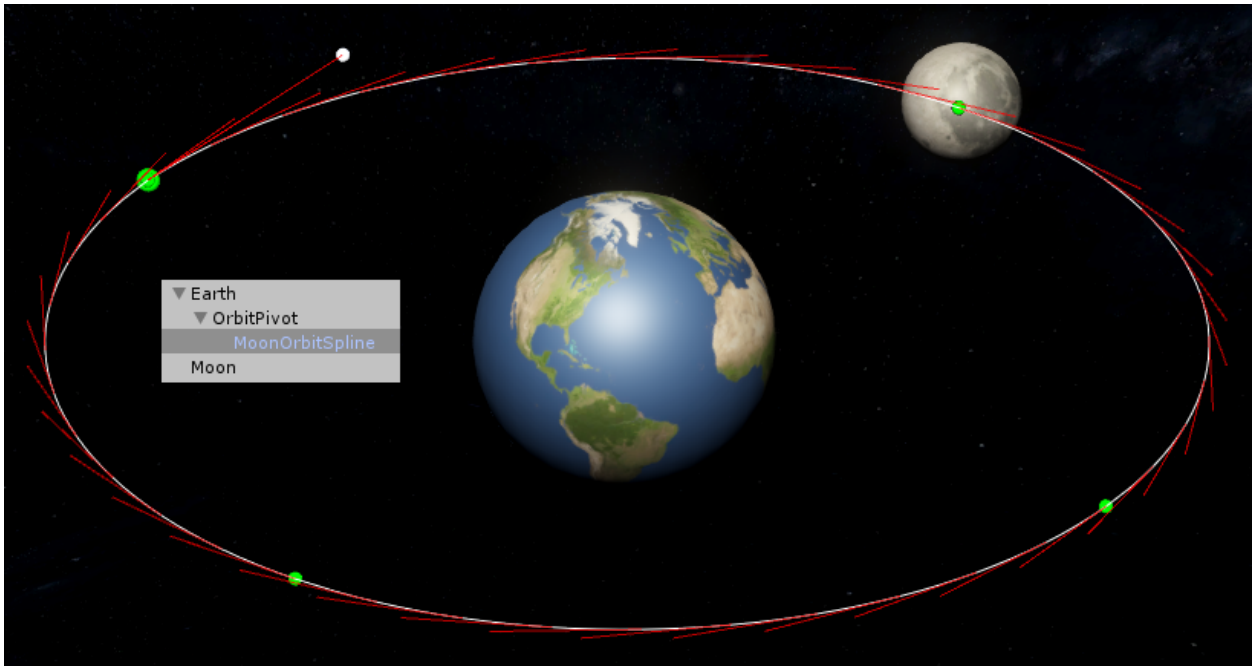

Figure 158 - Moon orbit

As you can see in the sample image above, the "Moon Orbit Spline" object is set as children of the Earth object. As any other game object, splines also inherit their parents movement, by using this in our advantage the moon orbits correctly around the Earth and inherits the Earth's orbit movement around the Sun. The moon orbit spline is configured as a non-static spline.

6.3 - Moving Platforms Demo Scene

At the "Follower_MovingPlatform" demo scene the Spline Follower component was used to create complex moving platform behaviour.

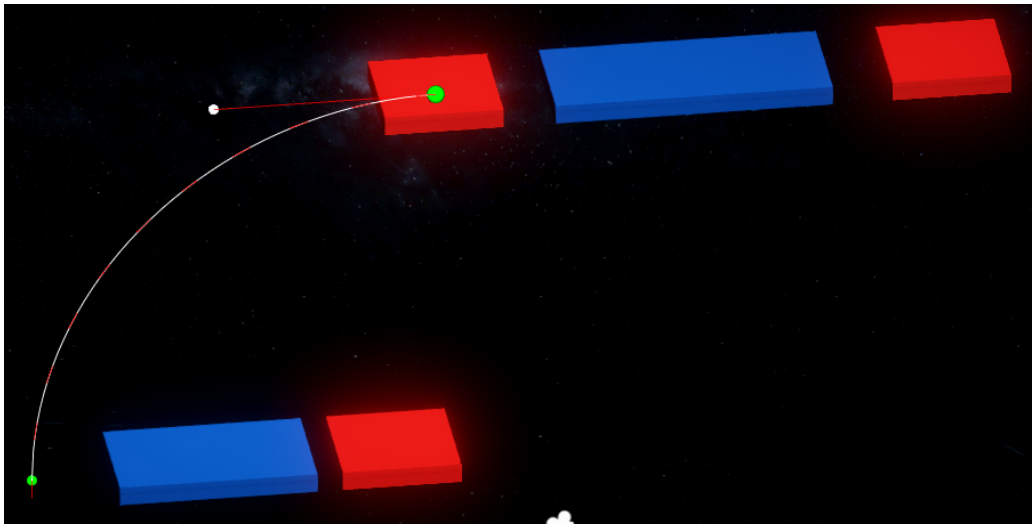

Figure 159 - moving platforms sample

In most games moving platforms don't rotate around themselves, so the “Apply Spline Rotation” property was disabled to ensure the platforms would still face the right direction all the time.

Cycle End Stops were also used to simulate stops at interest points, this can be used to let the player know that the platform has reached its destination.

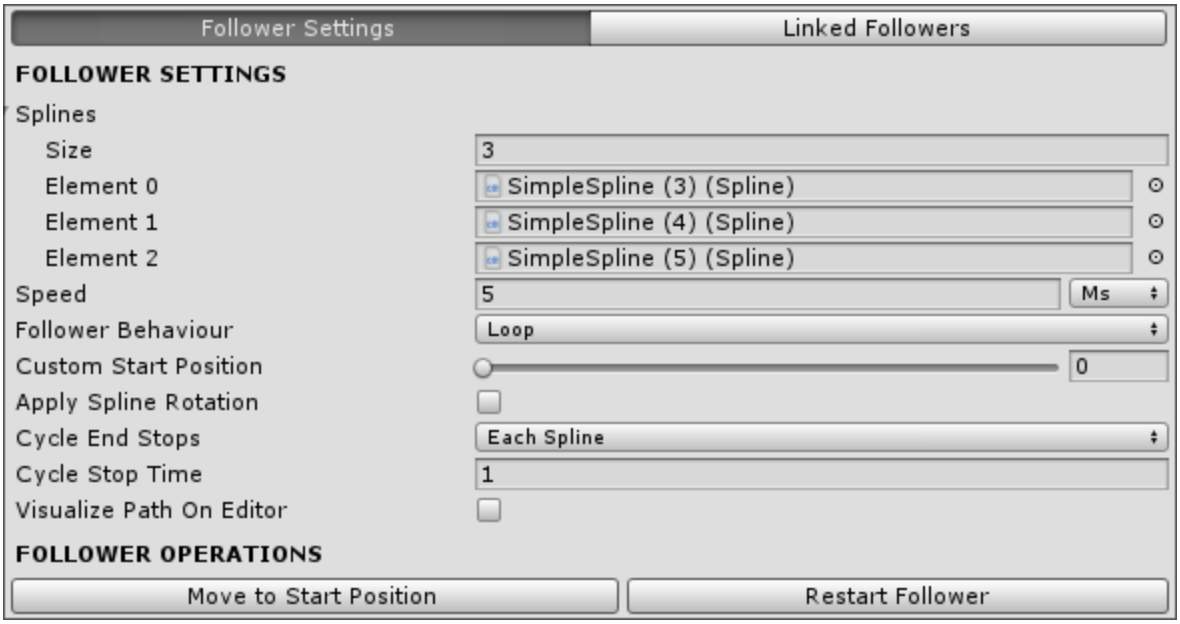

Figure 160 - Platform settings sample

In the sample image above, the platform was set up to follow multiple splines and stop for 1 second at the end of each one. When it reaches the end of the splines list, it starts all over again (Loop following behaviour).

7 - Performance Guidelines

Procedural mesh generation is powerful however it also is a costly operation. This section contains performance guidelines to avoid performance issues on the Unity Editor and improve the loading time and performance of your game.

7.1 - Manual x Realtime Generation

Always prefer using the manual Mesh Generation Method workflow. Although the real time generation is useful while working with short meshes it can be performance heavy for the Unity Editor while working with long meshes.

7.2 - Terrain Following

Avoid using the Follow Terrain feature and the real time mesh generation at the same time while working with long meshes.

Also, always prefer working with low Terrain Check Distance values, increasing its value is the easiest way to guarantee the spline will correctly identify the terrain all over your map, however, if any segment of your spline is not correctly projected in your terrain you can also roughly drag the spline closer to the terrain instead of raising the check distance property.

7.3 - Mesh Length

By using the Spline Mesh Renderer component it is possible to create long curved meshes. For example, it is possible to create railroad segments up to 1.35 Km long.

However, for better performance always prefer creating short mesh segments.

If you need to create long meshes, you can create multiple segments connected to each other by using the Connect New Renderer, Split Spline and Auto Split Features.

Also, by using short mesh segments you can take advantage of the Occlusion Culling feature to improve the performance of your game.

7.4 - Base Mesh Polycount

When creating custom Base Mesh models for the Spline Mesh Renderer, keep the polycount low.

By using low poly models you reduce the Unity renderer workload. Unity has a limit of 65000 vertices per rendered mesh, however it doesn't mean it is performance wise to have high poly meshes all around your scene.

Low poly base meshes, besides being more performance friendly, also allows you to create longer mesh segments on the Unity Editor.

7.5 - Prefab Baking

When preparing the final build of your game, always bake all meshes generated by the Spline Mesh Renderer as prefabs using the Bake Mesh feature.

This will drastically reduce the scene loading time. This happens because the Spline Mesh Renderer procedural mesh generation is executed on the start of the scene to ensure all meshes are correctly generated.

In the final build of your game, always replace the Spline Mesh Renderer components by baked prefabs.

If you are using the Spline component attached to the SplineMeshRenderer object on your scene and need it to still be active in play mode for gameplay purposes, make sure to keep the Save Spline option enabled when baking your meshes. That way, you will have a perfect copy of that spline as a child of the baked prefab. Use this new spline in your scene instead of the original one.

8 - License

By purchasing this asset you are allowed to use it for unlimited games and/or 3D projects (like animations, simulation softwares, etc). Both personal and commercial use.

You are NOT allowed to resell or distribute the assets components individually or as part of another asset package (including, models, scripts, etc).

For more information about licensing, please take a look at the asset store EULA and FAQ.

9 - Contact Info & Support

If you have some questions, need support or have some business inquiries, feel free to get in touch.

Asset Store
Sketchfab
Instagram
Facebook
Twitter
Youtube Channel