Dreamteck Splines – User Manual

http://dreamteck-hq.com

https://www.facebook.com/dtkhq/

Contact the team: team@dreamteck-hq.com

Contact support: support@dreamteck-hq.com

Contents

1. Introduction 5
1.1. Key Features 5
2. Getting Started 5
2.1. Install & Configuration 5
2.2. Creating a new Spline 6
2.2.1. Adding control points 6
2.3. Selecting points 8
2.3.1. Selecting multiple points 9
2.4. Deleting points 9
2.5. Closing and breaking a spline 9
2.5.1. Closing 9
2.5.2. Breaking 10
2.6. Changing the spline type 10
2.7. Spline precision 10
3. Editing splines 11
3.1. Point components 11
3.1.1. Position 11
3.1.2. Normal 11
3.1.3. Size 11
3.1.4. Color 11
3.1.5. Tangents 12
3.1.6. Type 12
3.2. Editing points 12
3.2.1. Move tool 12
3.2.2. Rotate tool 12
3.2.3. Scale tool 13
3.2.4. Normal tool 13
3.3. Point operations 14
3.3.1. Center To Transform 15
3.3.2. Flat X, Y and Z 15
3.3.3. Mirror X, Y and Z 15

3.3.4. Distribute evenly 16
3.4. Spline Computer settings 16
3.4.1. Space 16
3.4.2. Spline type 16
3.4.3. Custom interpolation 18
3.5. Primitives and presets 18
3.5.1. Using primitives 19
3.5.2. Using presets 19
3.6. Configuring the Spline Computer editor 20
3.6.1. Custom Color 20
3.6.2. Always draw 21
3.6.3. Draw thickness 21
4. Morph states 22
4.1. Creating morph channels 22
4.2. Editing morph channels 23
4.3. Blending between shapes 24
5. Spline Users 24
5.1. Spline Follower 25
5.2. Spline Projector 26
5.3. Particle Controller 27
5.4. Object Controller 28
5.5. Spline Renderer 29
5.6. Path Generator 31
5.7. Tube Generator 32
5.8. Waveform Generator 32
5.9. Surface Generator 33
5.10. Extrude Mesh 35
5.11. Length Calculator 36
6. Nodes 36
6.1. Junctions 37
7. Editor Tools 39
7.1. Generating meshes using Spline Tools 41

7.2. Leveling terrain 41
8. Writing a custom SplineUser class 42
8.1. Protected and virtual Methods 42
8.1.1. Protected virtual void Run() 43
8.1.2. Protected virtual void Build() 43
8.1.3. Protected virtual void PostBuild() 43
8.2. Rebuilding 43
8.2.1. RebuildImmediate 43

1. Introduction

Dreamteck Splines is a Spline system and extension for Unity which comes with a collection of tools and components for mesh generation, particle control, object spawning and much more. For a full list of features, refer to Spline Users and Editor Tools. The tool was initially created to aid our company's developers (Dreamteck Ltd.) in the process of level editing and creating game mechanics. The tool has been growing ever since it was first created in 2013. Over the years it has been used for many different purposes in many different projects (action games, TCG games, racing games, children's games and more).

1.1. Key Features

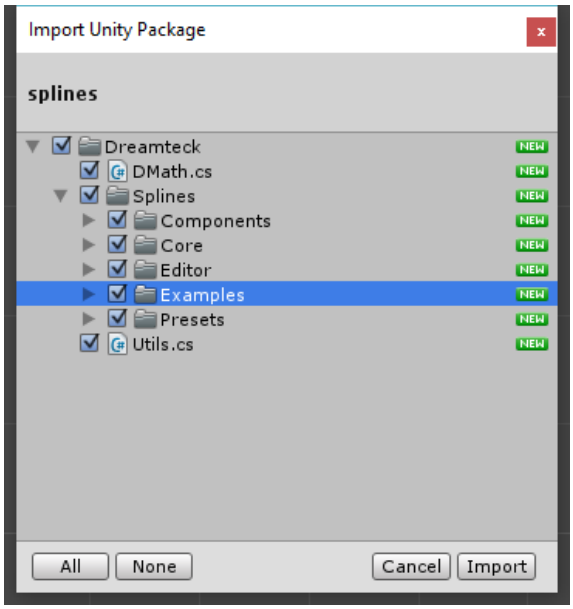
- Rapid spline creation and editing via custom editor in Unity.
- Four types of spline: Hermite, Bezier, B-Spline and Linear
- Procedural primitives and saving presets for later use
- Junctions
- Morph states
- On-the-fly mesh generation
- Multithreading
- Open source
- Easily expandable functionality

2. Getting Started

This chapter will guide the user through the installation, configuration and usage of the Dreamteck Splines system.

2.1. Install & Configuration

Dreamteck Splines comes packed in a .unitypackage file. Upon import, all the files and directories of the Dreamteck Spline system will be listed.



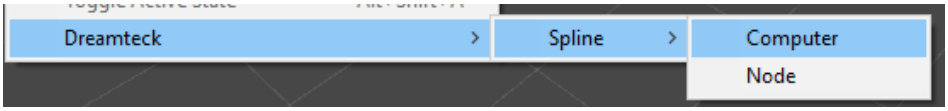
The "Examples" folder is optional. It contains example scenes which demonstrate the features of the system. It can be safely excluded from the import or deleted later if it isn't wanted.

The "Presets" folder contains an example preset file, saved with the preset manager. This folder can also be deleted but it will be automatically created as soon as a new preset is saved.

Clicking "Import" will decompress and import all of the selected files in the current Unity Project.

2.2. Creating a new Spline

To create a new spline, select GameObject in the toolbar and navigate to Dreamteck->Spline->Computer



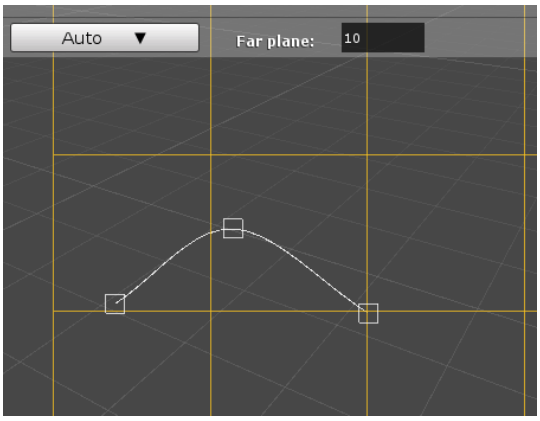
This will create a new Game Object called “Spline” with an empty **Spline Computer** component.

Once created, the object will be automatically selected and the currently selected editor tool for movement, rotation or scaling will be turned off. A new toolbar will appear in the scene view.

2.2.1. Adding control points

A spline needs at least two control points to exist. To add control points click the Plus button on the left of the toolbar. This will enter point creation mode and the button will get highlighted in blue. An additional toolbar will appear with the following options: Place method, Normal orientation and Far plane.

To create a new point, simply left click with the mouse somewhere in the scene. Where the point is created is defined by the place method. By default, the place method is set to Camera plane with far plane set to 0. This will create points in the position of the editor camera. The normal orientation option defines the direction of the normal of each new point upon creation. “Auto” will automatically pick a direction, appropriate for the selected placement method.



To exit point creation mode either press the Plus button again or click the right mouse button and without releasing it click the left mouse button too, then release both buttons – this is a shortcut for exiting point creation mode.

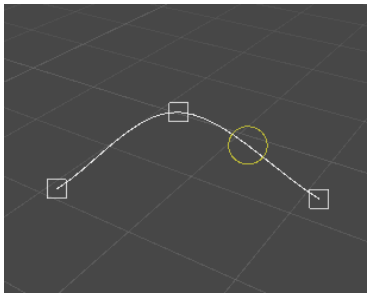
Dreamteck Splines offers six placement methods to accommodate a big range of developer needs related to point creation. To choose a different method, click on the Place method dropdown menu and select a different method.

2.2.1.1. Camera plane

The camera plane placement method places points on a plane, perpendicular to the scene camera. Far plane controls how far from the camera is the plane on which the points are created. The further the plane is, the bigger it becomes. If Far plane is set to 0, then the points will be created at exact position of the scene camera.

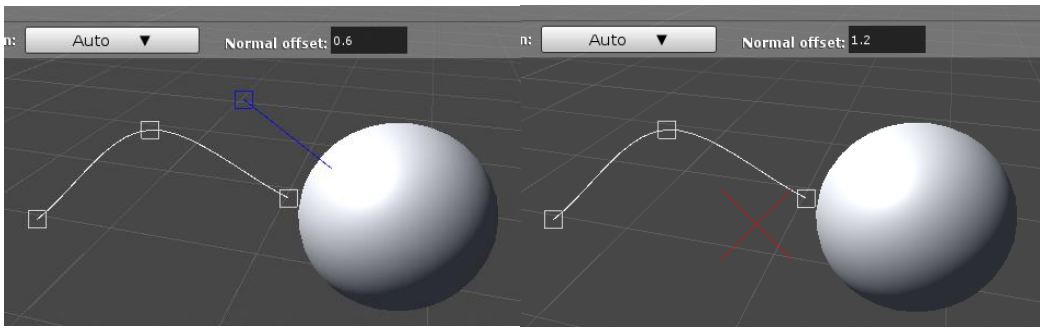
2.2.1.2. Insert

Insert inserts a new point between two points of the spline. When this method is selected, a circular button will appear in the scene. Moving the mouse around will move the button along the spline. Clicking the button will insert the point in the spline.



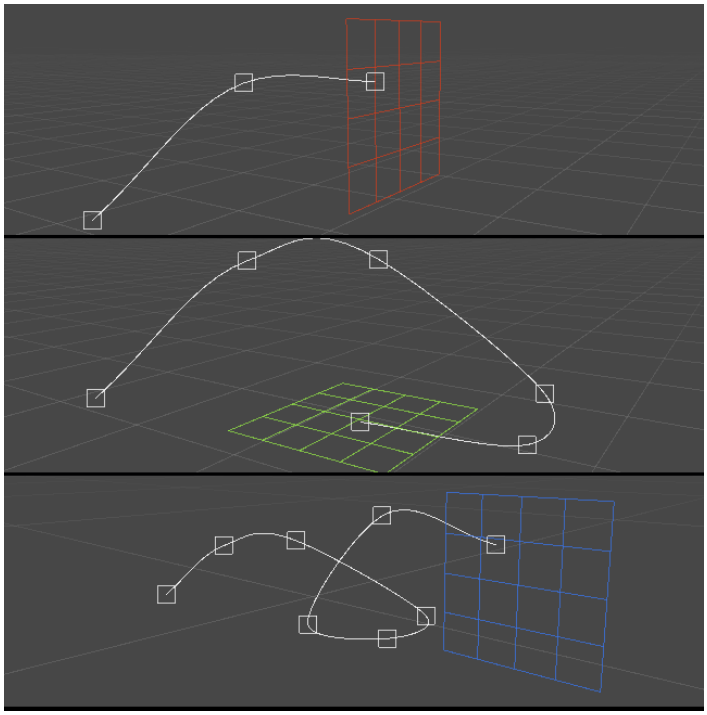
2.2.1.3. Surface

The surface method requires the scene to have at least one object with a collider. It will place the new point on the surface of the collider under the mouse upon left clicking. Normal offset controls the offset along the surface normal where the point will be created. If normal offset is 0, then the point will be created exactly on the surface of the collider. If the mouse does not hover over an object with a collider, a red cross will be displayed at the mouse position indicating that a point cannot be created.



2.2.1.4. Plane-X, Y and Z

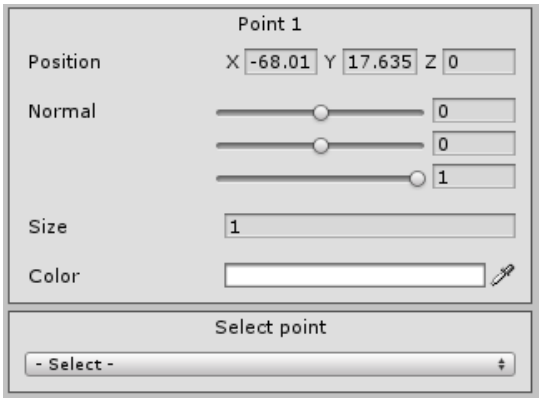
This placement method will project the mouse position onto infinite planes positioned at (0,0,0) world coordinates. This method is useful when the splines should be aligned to one of the three axes. In order to create a point the mouse must hover over the selected plane. If the mouse doesn't hover over the plane, a red cross will be drawn indicating that a point cannot be created. Plane offset controls the offset along the plane axis.



2.3. Selecting points

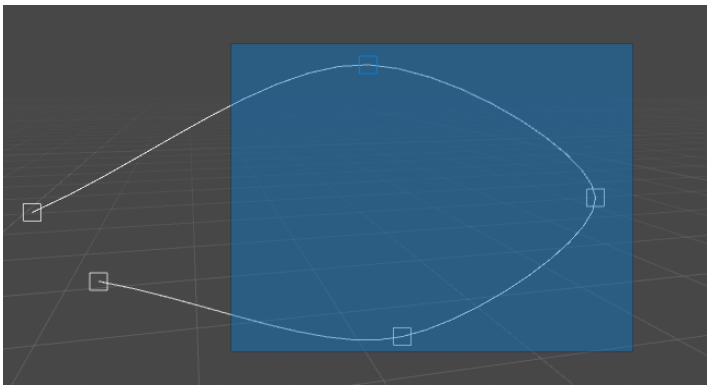
To select a point either click on it in the scene view or select it from the inspector using the point selection drop down menu.

When a point is selected, it's parameters will be displayed in the inspector:



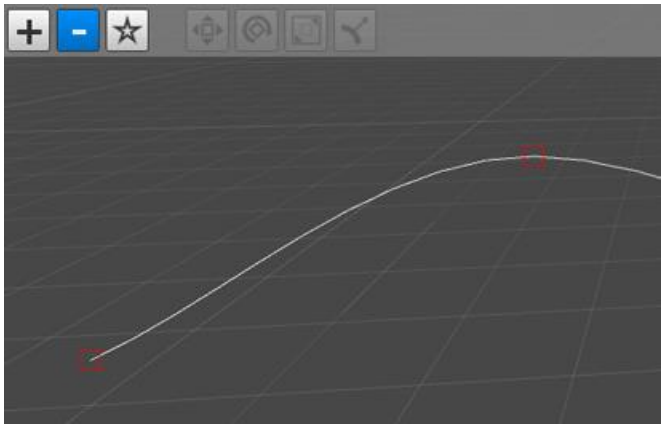
2.3.1. Selecting multiple points

To select multiple points either drag-select them in the scene view or select them one-by-one while holding Ctrl on the keyboard.



2.4. Deleting points

Points can be deleted by selecting them and pressing Delete on the keyboard or by entering Delete point mode.



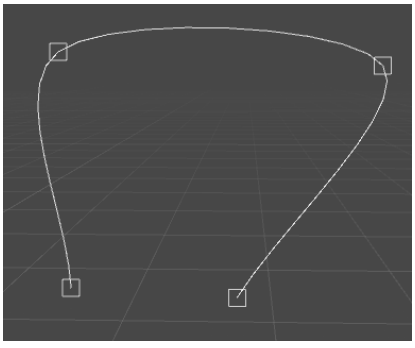
To enter Delete point mode, click on the Minus button in the scene toolbar. The button will get highlighted in blue and the control points will turn red. Clicking on a point in delete mode will delete the point.

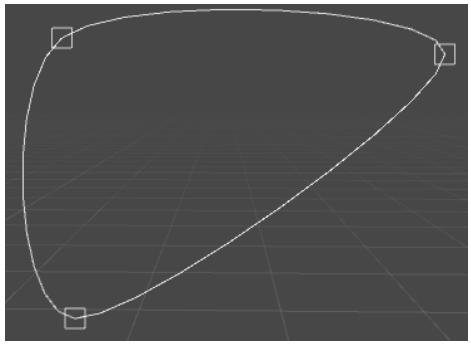
2.5. Closing and breaking a spline

A spline path consisting of four or more points can be closed. This will connect the first and the last points of the spline.

2.5.1. Closing

To close a spline, first ensure that there are at least four control points present, go to the inspector and click the “Close spline” button. The button will get replaced with a “Break path” button and the path will get closed.





2.5.1.1. Close spline during point creation

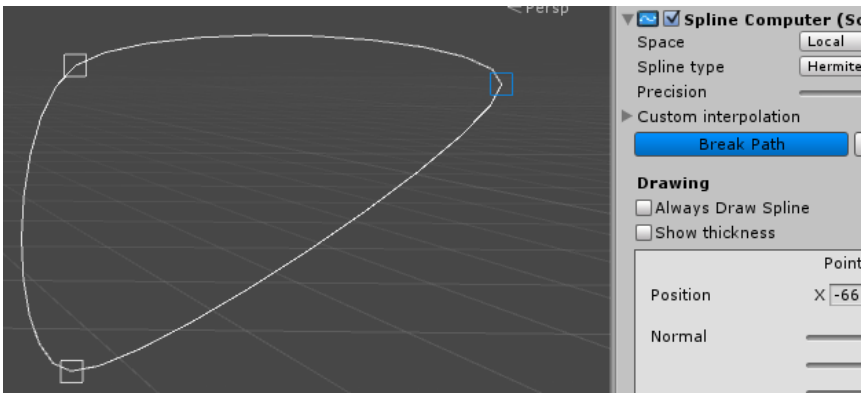
If a spline consists of three or more control points, it can be closed in point creation mode if the first point of the spline is clicked. Doing so will open a dialog asking whether or not the spline should be closed. This is convenient when the intention is to start off with a closed spline.

2.5.2. Breaking

To break a closed spline path press the “Break path button”. The two end points at which the path was closed will be able to move separately once again.

2.5.2.1. Breaking a path at point

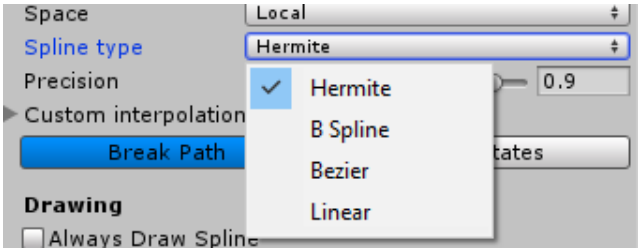
To break a spline path at a point simply select the point at which the path should be broken. The “Break Path” button in the inspector will get highlighted in blue. This indicates that the path will be broken at the selected point.



Make sure to select only one point, otherwise the path will break at the point it was previously connected.

2.6. Changing the spline type

Dreamteck Splines supports four types of interpolations: Hermite, Bezier, B-Spline and Linear. To change the spline type, navigate to the inspector and select a different type from the “Spline type” dropdown.

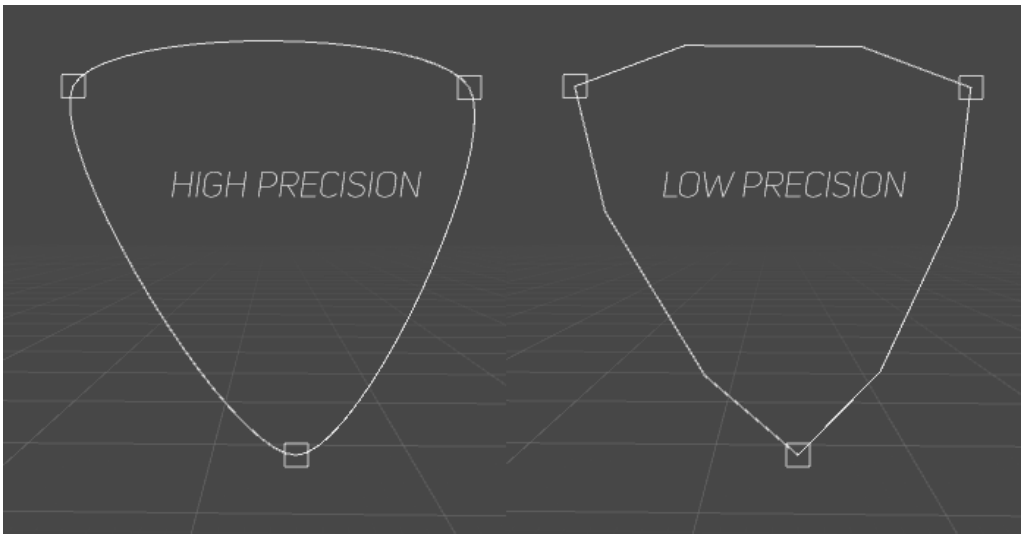


2.7. Spline precision

In computer graphics, splines aren’t endlessly smooth but are rather approximated at a certain rate. The higher the rate is the smoother the spline will be and the more computations it will require to approximate. It is usually a good idea to balance the precision so the spline looks visually smooth but it’s not too detailed.

To change the spline approximation rate, use the Precision slider in the inspector.

Note that precision should never be 1 (that would mean dividing by zero) and therefore the maximum precision that can be set is 0.9999



3. Editing splines

Dreamteck Splines are highly configurable and come with a set of tools that aid the creation process. This chapter offers an in-depth look at those tools and explains how they work.

3.1. Point components

Each spline consists of a set of control points and each control point has a set of components that define it. These components get interpolated during spline evaluation.

3.1.1. Position

The position is a Vector3 point in space where the control point is located.

3.1.2. Normal

The normal is a Vector3 direction of the point normal. It is used to define object orientation along the spline.

3.1.3. Size

A float value defining the size of the point. It is used as a multiplier for scaling and offsetting.

3.1.4. Color

The color of the point.

3.1.5. Tangents

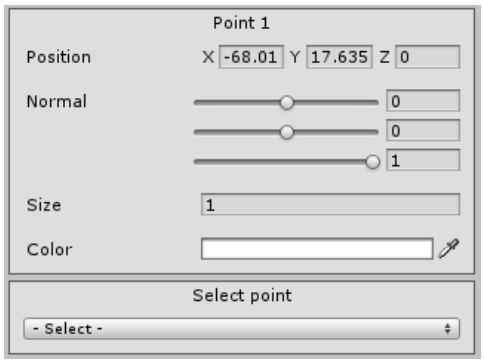
Each point has two additional Vector3 values for tangent position (tangent1 position and tangent2 position). The tangents are required by the Bezier algorithm otherwise do not have a function.

3.1.6. Type

The type of the point. Can be Smooth or Broken. This is used by the Bezier algorithm only to define whether or not the tangents will be free or locked.

3.2. Editing points

Selecting each point opens up the point menu in the inspector which exposes the point parameters. From there each point's position, normal, size, color and type can be edited. This menu is the only place where the point's size and color can be set however it isn't comfortable for setting the point's position and normal unless a precise value needs to be entered.



Dreamteck Splines comes with tools for movement, rotation scaling and editing normal. The buttons for those tools are located in the scene view toolbar and are unavailable by default. They become available once one or more points are selected.

3.2.1. Move tool

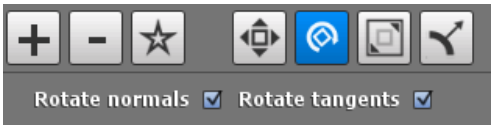
The move tool provides a standard move handle for each point. Clicking on the move tool button enables the move tool. Clicking on the button again, once it's highlighted disables the tool.

The move tool can be selected with by pressing "W" on the keyboard. This overrides Unity's built-in shortcut.

3.2.2. Rotate tool

The rotate tool provides a standard rotation handle for the selected points. Clicking on the rotate tool button enables the rotate tool. Clicking on the button again, once it's highlighted disables the tool.

The rotate tool can be selected with by pressing “E” on the keyboard. This overrides Unity’s built-in shortcut.



This tool has two options which appear in the toolbar extension below.

- Rotate normal – whether or not the normal of the point should be rotated
- Rotate tangents – whether or not the tangents’ positions should be rotated

Rotating a single point will only result in rotating the normal and tangents IF the appropriate options are selected in the toolbar.

When multiple points are selected, the rotation handle is positioned in the center of both points. Points’ positions are rotated around the position of the handle.

3.2.3. Scale tool

The scale tool provides a standard scale handle for the selected points. Clicking on the scale tool button enables the scale tool. Clicking on the button again, once it’s highlighted disables the tool.

The scale tool can be selected with by pressing “R” on the keyboard. This overrides Unity’s built-in shortcut.

This tool has two options which appear in the toolbar extension below.

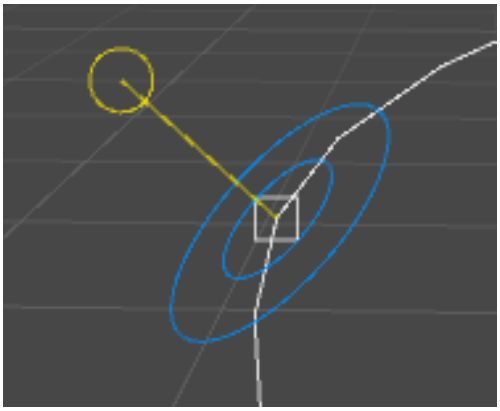
- Scale sizes – whether or not to scale the points’ size parameters
- Scale tangents – whether or not to scale the points’ tangents’ positions

Scaling a single point will only result in scaling it’s size and tangents IF the appropriate options are selected in the toolbar.

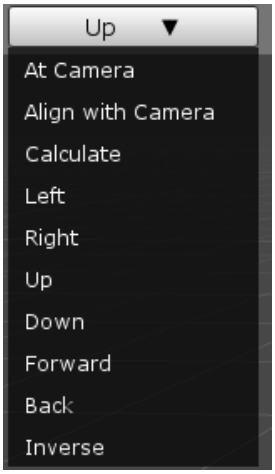
When multiple points are selected, the scale handle is positioned in the center of both points. Points’ positions are scaled away or towards the scale handle’s position.

3.2.4. Normal tool

The move tool provides a special directional handle for each selected point. Clicking on the normal tool button enables the normal tool. Clicking on the button again, once it’s highlighted disables the tool.



Dragging the yellow circular handle sets the normal's orientation.

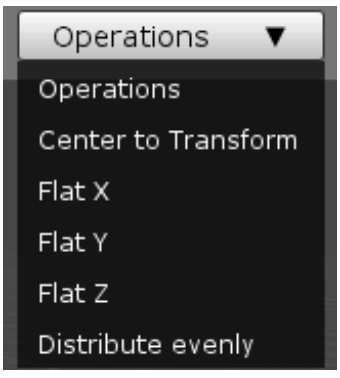


The normal tool comes with a "Set Normals" function in the toolbar extension. This function sets the normal to automatically point at a certain direction. To choose where the normal should point, use the dropdown menu next to the "Set Normals" button. The available options are:

- At Camera – sets the normal to point at the editor camera position
- Align with camera – aligns the normal with the forward direction of the editor camera
- Calculate – Calculates the normal based on the point's position
- Left – World left
- Right – World right
- Up – World up
- Down – World down
- Forward – World forward
- Back – World back
- Inverse – Inverses the normal's direction

Clicking on the button "Set normals" will set the normals of the selected points to the selected direction.

3.3. Point operations



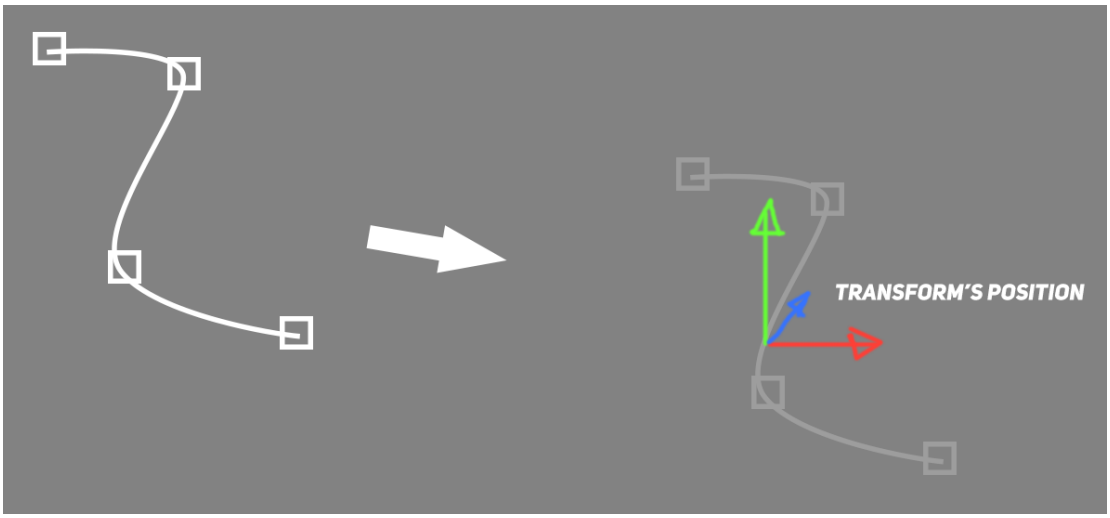
Dreamteck Splines provides a set of simple and handy operations for editing points. These operations can be found on the right side of the scene view toolbar and become available when at least one point is selected. Since most of the operations require two or more points to be selected, it's

recommended to select at least three points when examining this feature for the first time.

When an operation is selected from the dropdown menu it is immediately executed.

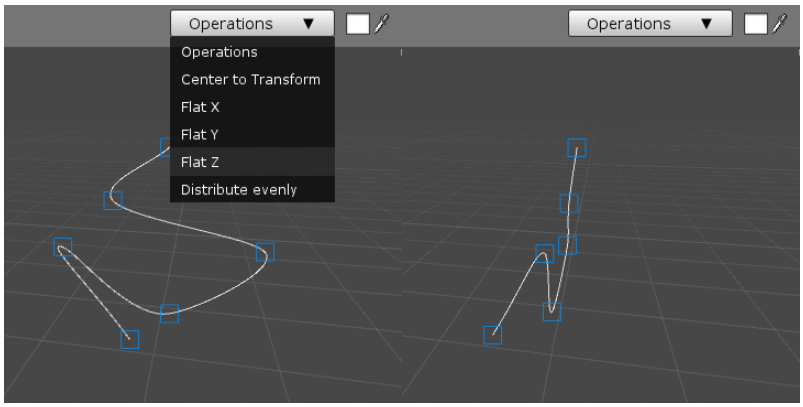
3.3.1. Center To Transform

When splines are created, their points can be positioned very far from the position of the Spline Computer's Transform. Sometimes the control points need to be brought to the center of their transform so proper transformation can be applied. This operation bring all of the selected points' positions to the center of the transform while maintaining their relative position.



3.3.2. Flat X, Y and Z

These operations flat the selected points' positions along the given axis.



If the spline type is Bezier, a dialog will appear upon selecting the flat operation asking if the points' tangents should be flatten too.

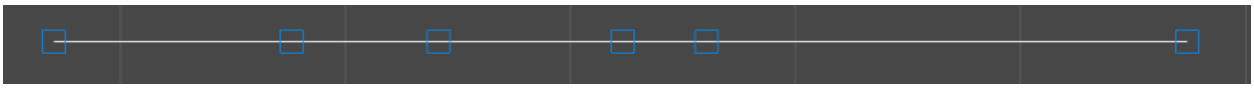
3.3.3. Mirror X, Y and Z

Mirroring is introduced in version 1.0.3 and mirrors the selected spline points along the given axis.

3.3.4. Distribute evenly

As the name suggest, this operation will attempt to distribute the points at even distances from each other.

Before:



After:

3.4. Spline Computer settings

The Spline Computer component has several very important settings that can change the way it works. These settings can be found in the inspector.

3.4.1. Space

She Space of the Spline Computer can be either world or local and defines how the it's spline is computed. By default, the space is set to local which applies the transform's position, scale and rotation to the spline resulting in a transformed spline. Setting the space to World will cause the spline to ignore the object's transform much like the Space attribute of Unity's Built-in Line Renderer.

When changing the space of a Spline Computer, the spline's coordinates are preserved into the new coordinate system resulting in the spline staying the same.

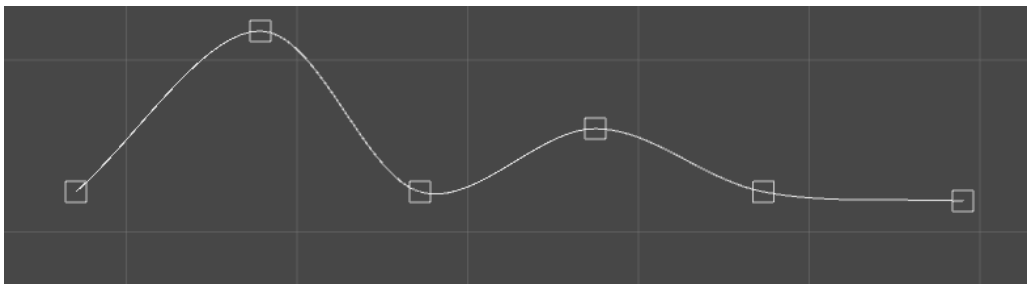
Example case:

A spline with five control points is created;
The Spline Computer's space is set to local;
The Spline Computer's scale is set to (2, 2, 2) (twice the initial). This results in a scaled spline;
The Spline Computer's space is set to world. The spline's size stays the same;
The Spline Computer's scale is set back to (1, 1, 1). The space is set to world so the spline's size stays the same;
The Spline Computer's space is set back to local; The spline's size stays the same.

3.4.2. Spline type

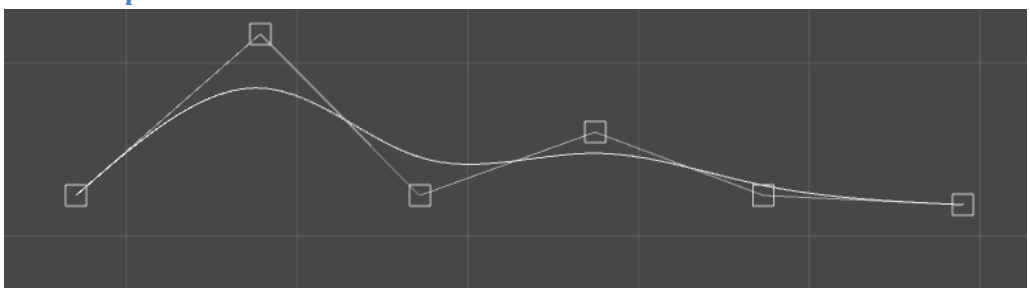
The spline type defines which interpolation method should be used to calculate the spline.

3.4.2.1. Hermite



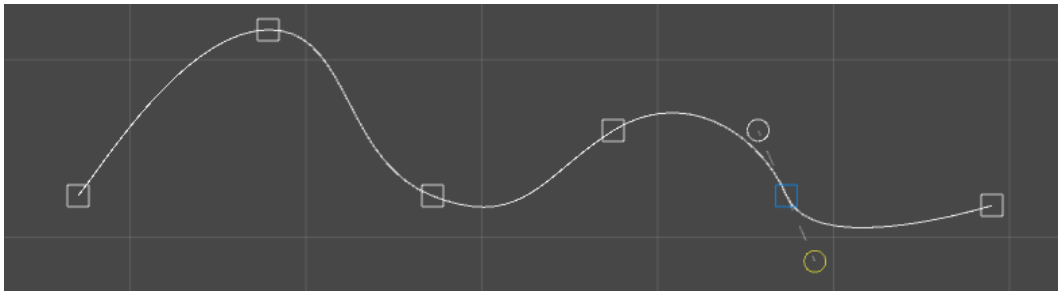
Interpolates all points

3.4.2.2. B-Spline



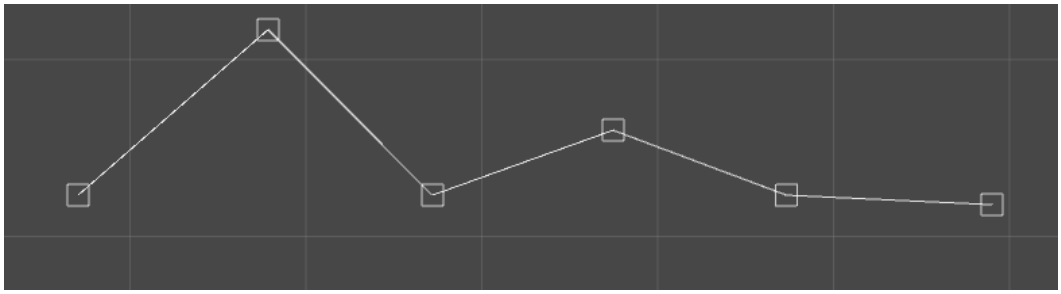
Approximates all points

3.4.2.3. Bezier



Interpolates points, approximates tangents

3.4.2.4. Linear

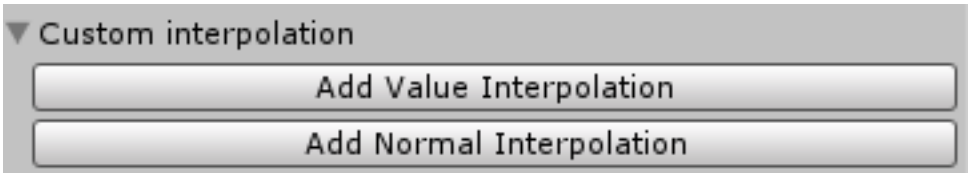


Linearly interpolates all points

3.4.3. Custom interpolation

By default the points' colors, sizes and normals are linearly interpolated which may not be desired in some cases.

To set an interpolation curve, expand the "Custom interpolation" foldout in the inspector.

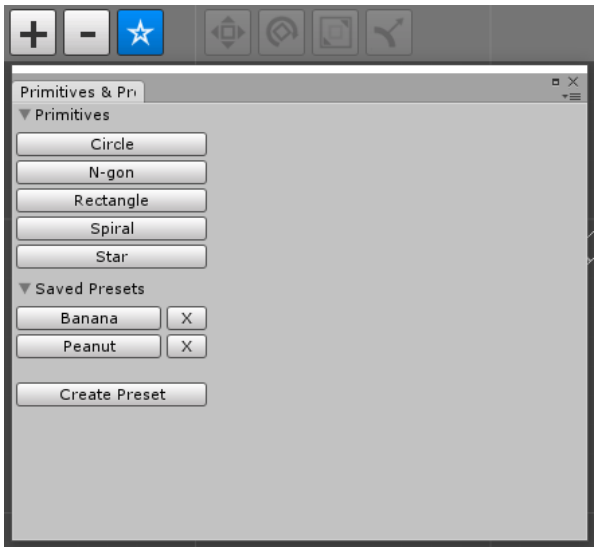


Then click on "Add Value Interpolation" to add a custom curve for the color and size interpolation or click "Add Normal Interpolation" to add a custom curve for the normal interpolation.

Doing so will create a default Ease-in, Ease-out curve which can then be edited to something else.

3.5. Primitives and presets

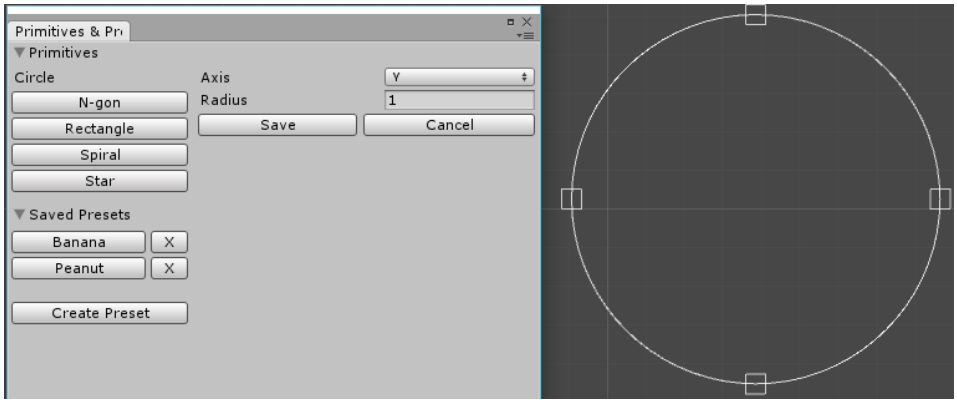
Dreamteck Splines provides a list of procedural primitives and a preset saving system. Both are located in the Primitives & Presets window which can be opened by clicking the Star button from the scene view toolbar.



The primitives are listed first with buttons. Clicking on a primitive button will open the primitive preview and menu. The saved presets are listed second and like the primitives, clicking on a preset button opens the preset preview and menu. Presets can be deleted by clicking on the button with a cross next to the preset button.

3.5.1. Using primitives

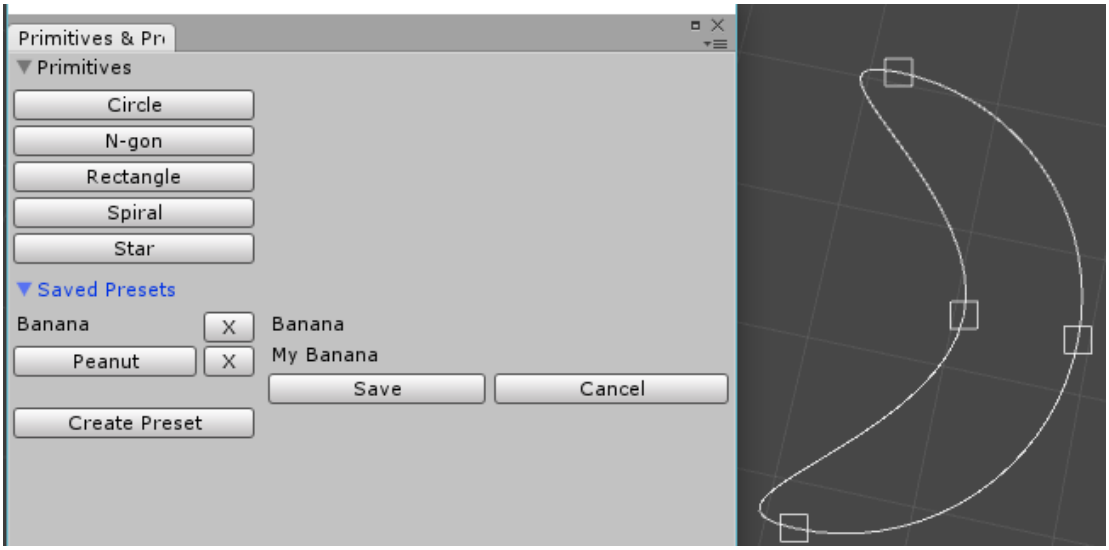
When a primitive is selected, it's menu is displayed in the right side of the Primitives & Presets window. This menu features settings for the procedural primitive and a "Save" and "Cancel" button. In the scene view, the spline is set to display the primitive.



If "Save" is pressed, the spline shape and settings is set, if "Cancel" is pressed, the spline gets reverted to it's previous state.

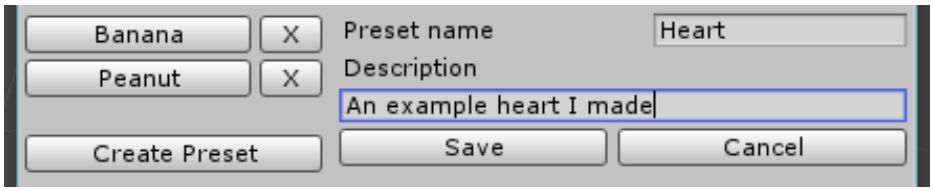
3.5.2. Using presets

When a preset is selected, it's title and description as well as "Save" and "Cancel" buttons are displayed on the right side of the Primitives & Presets window. If "Save" is pressed, the preset is applied, if "Cancel" is pressed, the spline gets reverted to it's previous state.



3.5.2.1. Creating a preset

To create a new preset, click the “Create preset” button. Fill the name and description fields that are displayed and click “Save”. This will create a new preset file in the presets folder.



If after saving, the file does not appear in the Presets folder, right click inside the folder and select “Refresh”.

Presets can be transferred between projects .

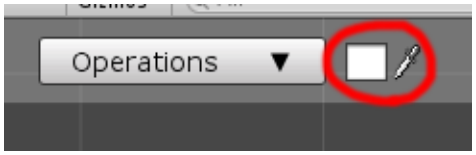
3.6. Configuring the Spline Computer editor

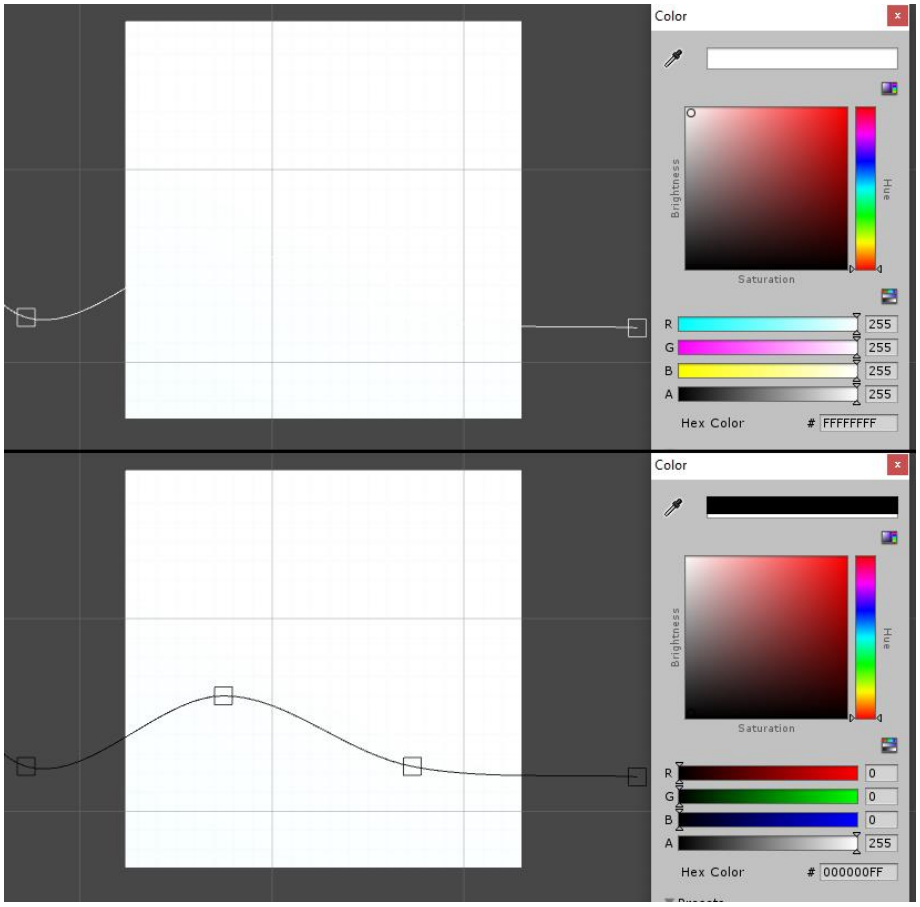
Each Spline Computer has several options which define how it is drawn in the editor.

3.6.1. Custom Color

Custom colors can be used differentiating Spline Computers when several are drawn simultaneously or to prevent blending with the background.

To choose a color for the currently selected Spline Computer use the color picker located in the right side of the scene view toolbar.



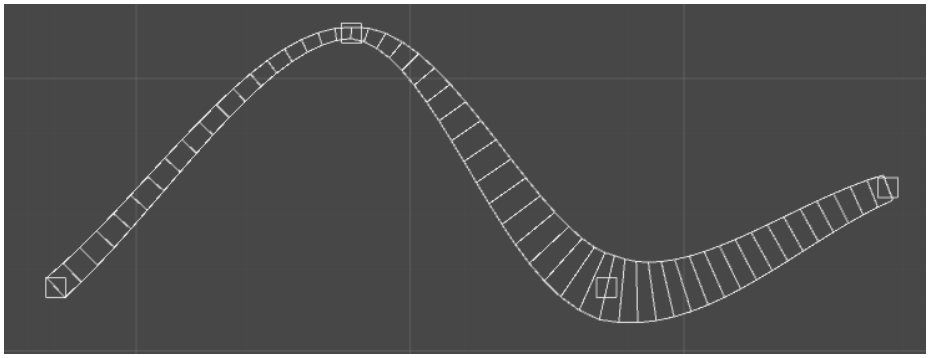


3.6.2. Always draw

Always draw can be toggled on or off from the Spline Computer's inspector. This option causes the spline to be drawn all the time even when the Spline Computer is not selected.

3.6.3. Draw thickness

Draw thickness is toggled on or off from the Spline Computer's inspector. This option causes the spline's thickness to visualize. The thickness is defined by the size of the points (which by default is 1).

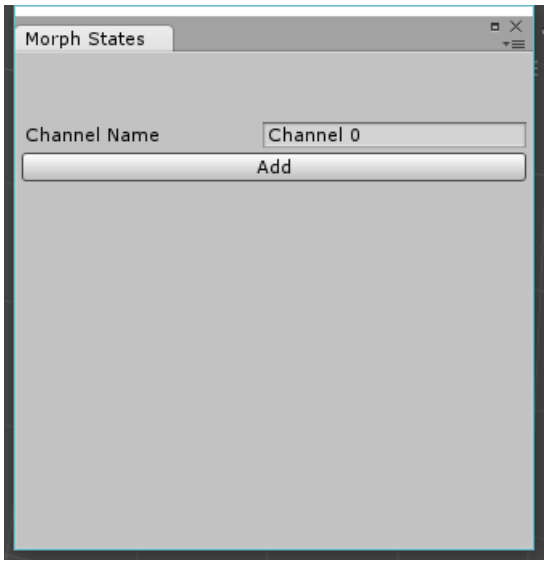


4. Morph states

Each Spline Computer has a morph state module which allows adding channels that contain different spline shapes to the Spline Computer. These channels can then be blended using weights.

To open the Morph States window click the “Morph States” button in the Spline Computer inspector.

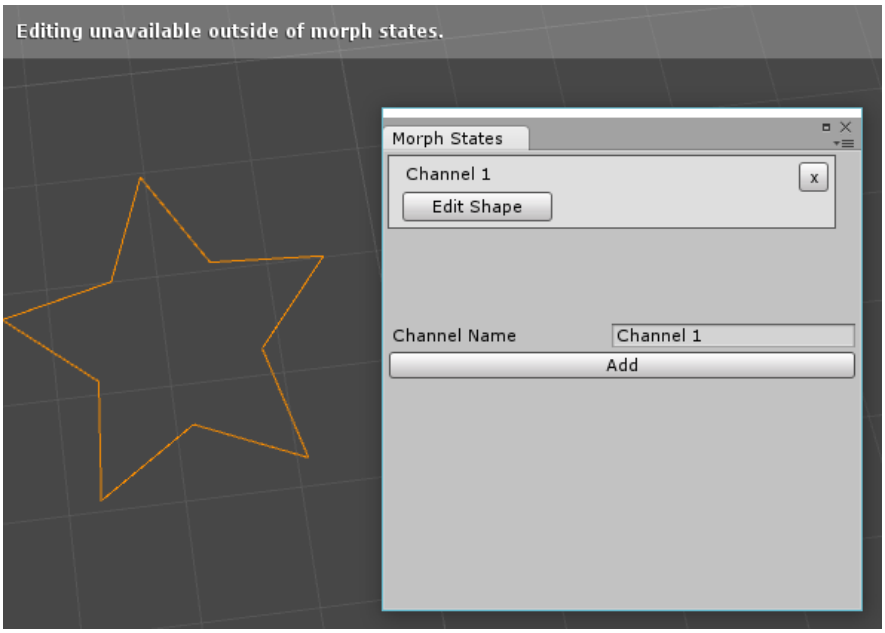
This will open a new empty window with the option to add a channel.



4.1. Creating morph channels

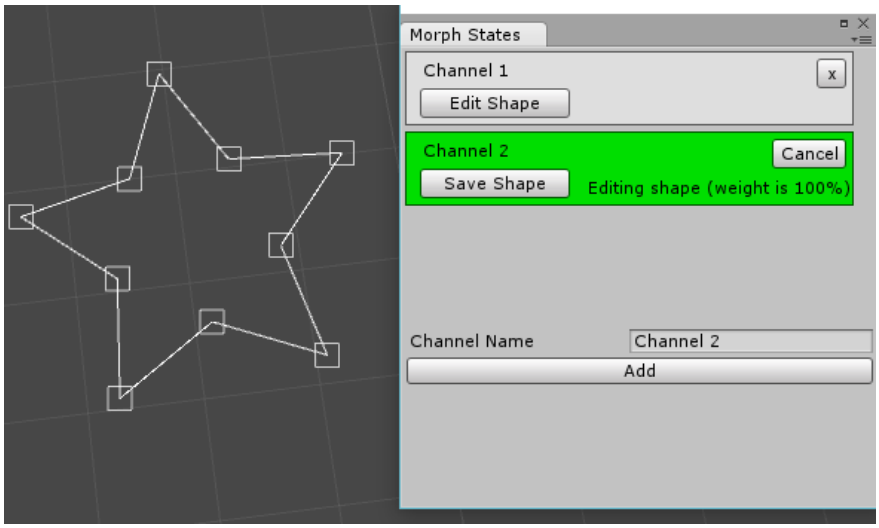
Clicking on the “Add” button in the Morph States window will create a new base channel which has 100% weight.

After the first channel is created, it will appear in the window above the “Add” button. The spline will turn orange and editing will be disabled. From this moment on, editing will be only enabled inside channels.



4.2. Editing morph channels

To edit the shape of a morph channel, go to the Morph States menu and click the “Edit Shape” button of the morph channel which shape must be edited. This will highlight the morph channel in green. It will also enable the spline points and editing tools.

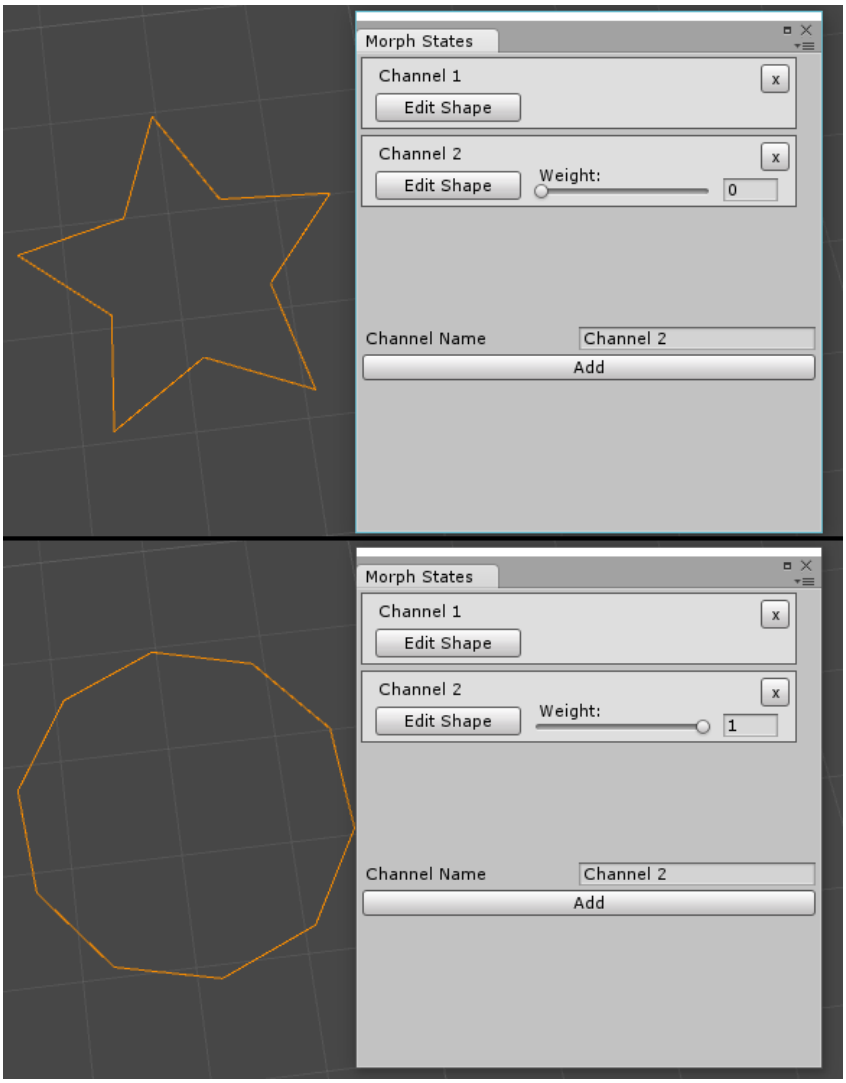


When editing the shape of a morph channel, the morph channel is weighted at 100%. After editing is complete, “Save Shape” should be pressed. This will write the edited shape to the channel and exit edit mode.

IMPORTANT! Each channel must contain exactly the same amount of points as the other channels. Therefore deleting and creation of points are disabled.

4.3. Blending between shapes

Once there are two channels with different shapes they can be blended. To blend a channel, drag it's weight slider to the right.



During runtime, channel weights are set via code or UnityEvents.

5. Spline Users

The Spline Computer all by itself cannot do much more than provide sampling date from it's spline. To make use of the Spline Computer, Dreamteck Spline implements the Spline User. This is a base class, derived from MonoBeaviour. It's purpose is to sample the Spline Computer and use the sampled data for various purposes.

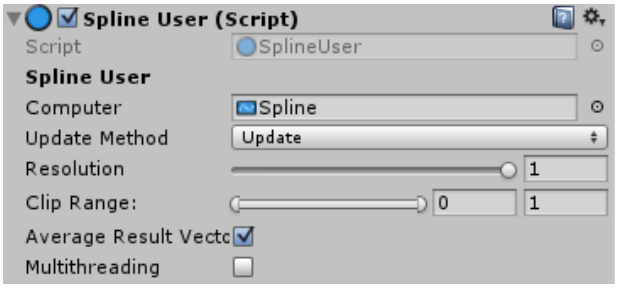
When a Spline User is selected in the editor, it's Spline Computer is automatically drawn.

The Spline Users only sample Spline Computers when there has been a change in the spline, therefore do not have a constant overhead if no changes are made.

The Spline User class supports multithreading so sampling and calculations can be done in a separate thread, not affecting the game performance.

Every Spline User has several key properties:

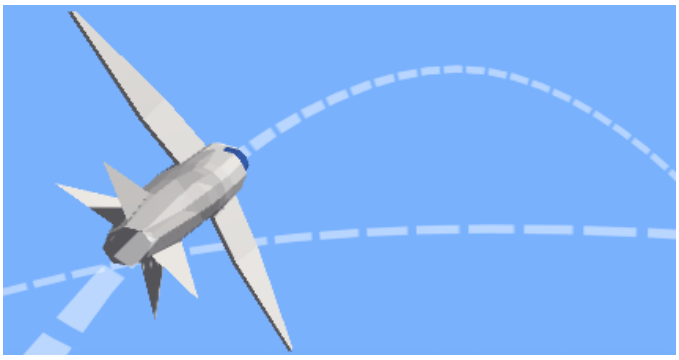
- Computer – The Spline Computer reference this user is using.
- Update Method – When is the user updated? Update/FixedUpdate/LateUpdate
- Resolution – (0-1) acts like a multiplier to the Spline Computer's precision
- Clip Range – Two values – clipFrom and clipTo which define the range of the spline that will be used
- Average result vectors – Whether or not to average each sample's direction and normal (overall gives a good result when turned on)
- Multithreading – Whether or not to use multithreading



Dreamteck Splines comes with a collection of SplineUser-derived classes which are ready to be used in the editor and during runtime.

5.1. Spline Follower

The Spline Follower component makes the object it's attached to follow a spline with a uniform speed. If autoFollow is enabled, the follower will automatically follow the spline with the follow speed during runtime. If autoFollow is disabled, the follower can be caused to follow by calling it's public method **Move(float distance).**



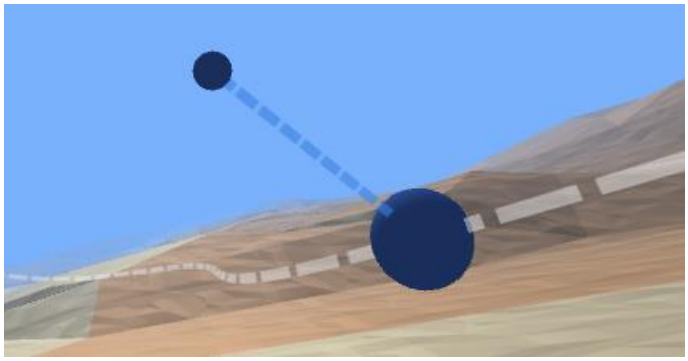
The Spline Follower can follow the spline forward or in reverse. Select the direction it should go from the Direction dropdown.

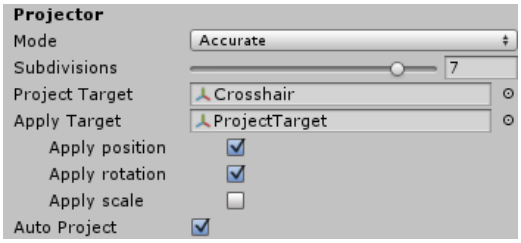
The Spline Follower has three wrap modes:

- Default – when the end of the spline is reached, the follower will stop
- Loop – When the end of the spline is reached, the follower will start again from the beginning.
- PingPong – When the end of the spline is reached, the follower will change it's direction

5.2. Spline Projector

The Spline Projector projects a Game Object from the scene on a spline. It will find the closest point of the spline to the object's position.





The Spline Projector needs two Transform components as targets – The target that is projected on the spline (Project Target) and the target that is positioned at the Project Target's projection. If Project Target is not set, then the Spline Projector's Transform is used by default. If Apply Target is not set, then Transform is set to the projection's result. The projection result can be accessed via code using the **projectResult** property.

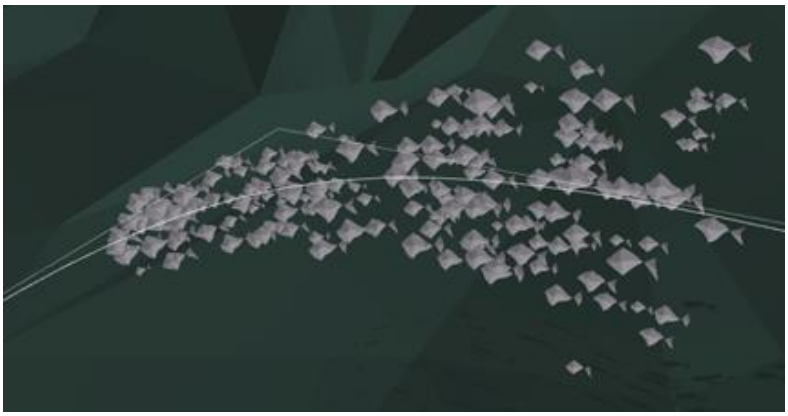
The Spline Projector has two modes: Accurate and Cached.

In accurate mode, the resolution slider of the Spline User is disabled. This tries to project the point on the spline as accurately as possible using subdivisions. The subdivisions are set to 4 by default. This in overall is a good value to produce an accurate result but there might be cases when it isn't enough.

In cached mode, the projector will find the two closest sampled results and project the target on the line between them. This is mode is designed to be a lot lighter than the Accurate mode but it will not give smooth results with low resolution samples.

5.3. Particle Controller

The Particle Controller uses a Shuriken particle system and places it's particles along a spline.



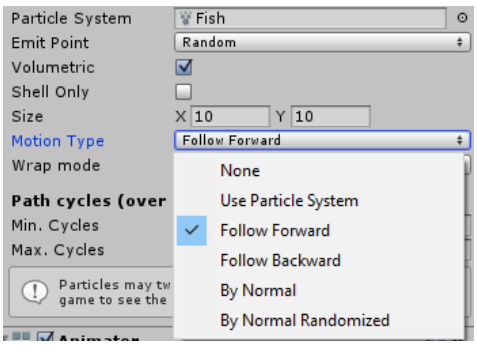
This component offers a preview right in the editor if it's attached to the particle system that it controls.

The Particle Controller has an Emit point which controls where new particles will be spawned. The modes are:

- Beginning – Emits particles from the beginning of the spline
- Ending – Emits particles from the final point of the spline
- Ordered – Emits particles from the beginning of the spline towards it's end based on particle index
- Random – Emits particles between the beginning and the end of the spline on a random basis

Another very important option of the Particle Controller is the Motion type. This controls the particle behavior over it's lifetime. The available motion types are:

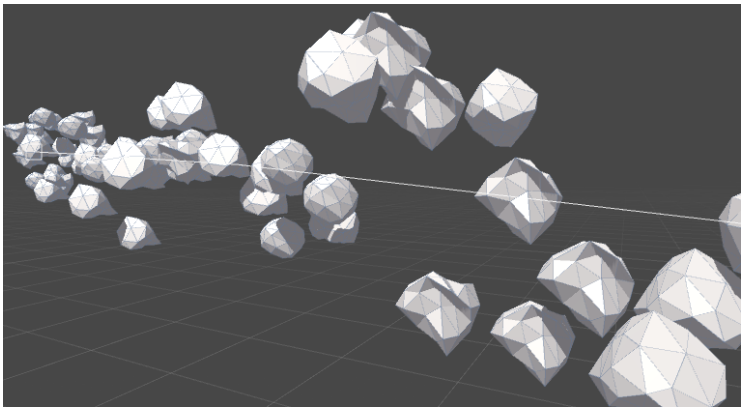
- Use Particle System – Uses the motion, defined by the Particle System component



- Follow Forward – Particles move forward along the spline based on their lifetime
- Follow Backward – Particles move backward along the spline based on their lifetime
- By Normal – Particles get their initial velocity set in the direction of the spline sample's normal
- By Normal Randomized – Same as above but the direction vector is randomized

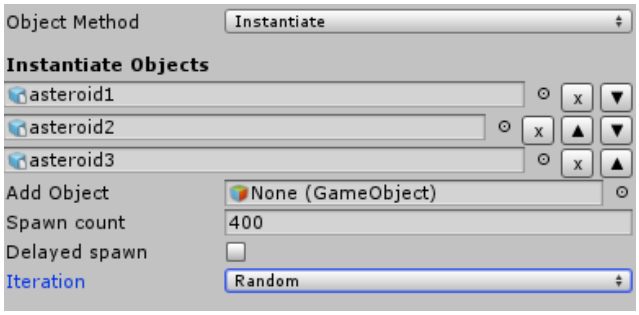
5.4. Object Controller

The Object Controller component instantiates and positions Game Objects along a spline. This could be done both during runtime and in the editor.



The Object Controller has two Object Methods which define how the objects it controls are created.

- Instantiate – This will instantiate new objects
- Get Children – This will not instantiate new objects but instead will take the existing children of the Object Controller's Transform and use them.



In Instantiate mode, a Game Object array is presented. For Instantiate to work there must be at least one added Game Object reference.

A Spawn Count property is presented. Modifying it will create or destroy objects.

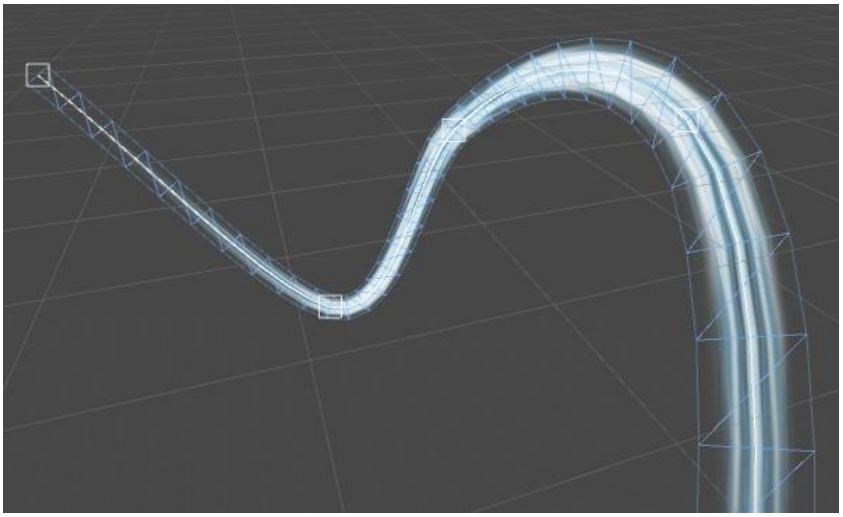
The Iteration dropdown defines in what order the Game Objects are instantiated. If Ordered is selected, the Game Objects will be spawned in the order they are added. To change this order, use the up and down arrows next to the objects. In Random mode, objects will be instantiated in a random order.

If Delayed spawn is turned on, the objects will spawn over a certain period of time. This is mostly used to prevent spike lags caused by spawning many objects at once. Delayed spawn does not work in the editor.

In Get Children mode, the child objects of the Object Controller's Transform will be used. Adding or removing objects from the hierarchy will include or exclude them from the controller.

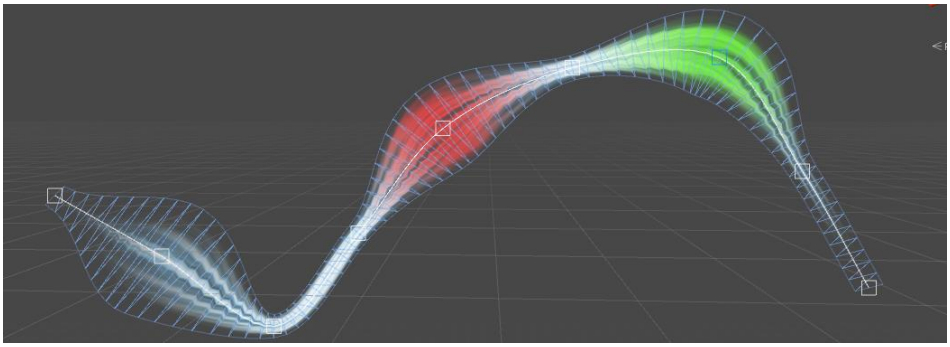
5.5. Spline Renderer

The Spline Renderer is an analogue of Unity's Built-in Line renderer. It uses a spline to visualize instead of a set of points. It's the first of a set of behaviors, derived from a special Spline User class called SplineMeshGeneration. This class is used to provide basic functionality for creating procedural geometry using a spline.

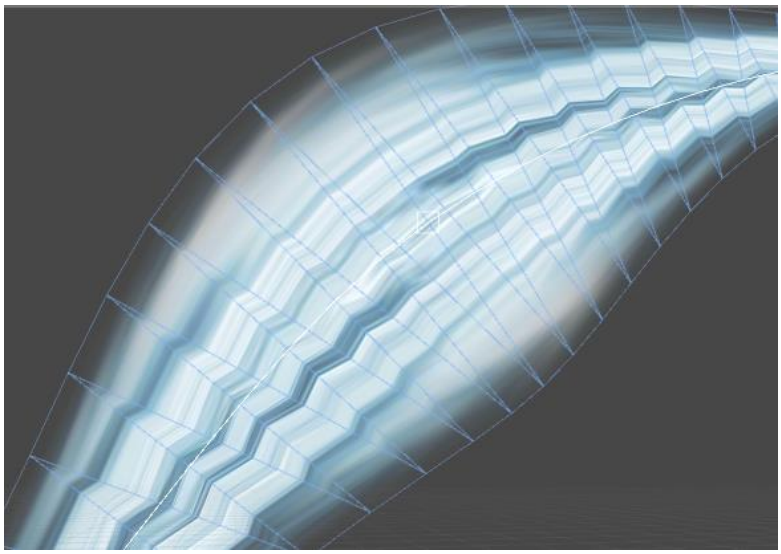


Editing the SplineRenderer and anything else, derived from SplineMeshGeneration is as easy as editing the Spline Computer's spline.

Each SplineMeshGeneration behavior is affected by the spline points' colors and most of them are affected by the points' sizes.

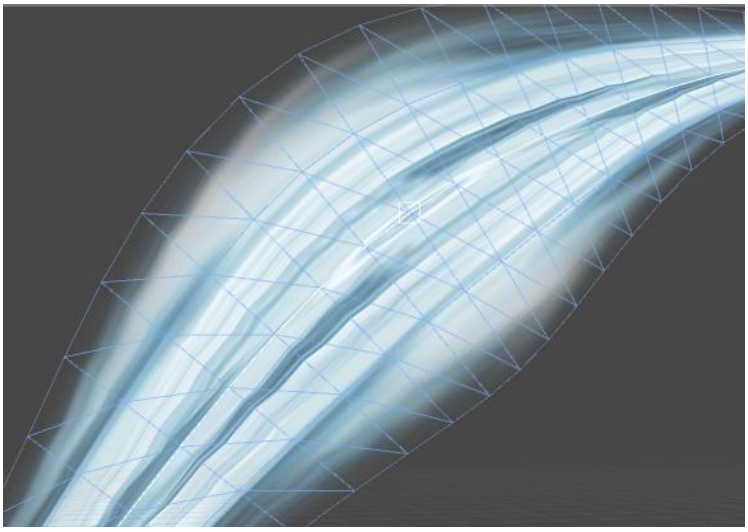


Because of the triangular nature of the mesh, in some cases the texture coordinates of the Spline Renderer might appear jagged.



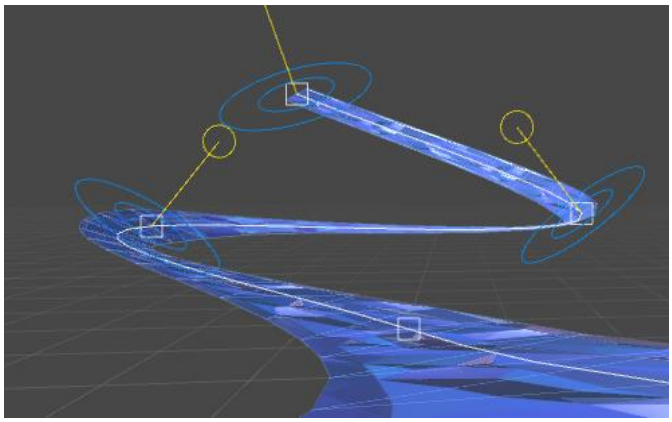
This is most often observed when there are spline points with different sizes and the solution to this is to add more edge loops. To do so go to the inspector and increase the number of Slices:

This will reduce the jagged artefacts.

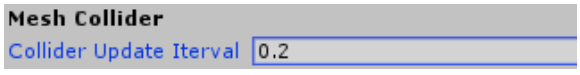


5.6. Path Generator

The Path Generator is very similar to the Spline Renderer with a few exceptions. First, unlike the Spline Renderer, the Path Generator does not orientate the generated geometry to face the camera. Instead the geometry is orientated in the direction of the spline normal.

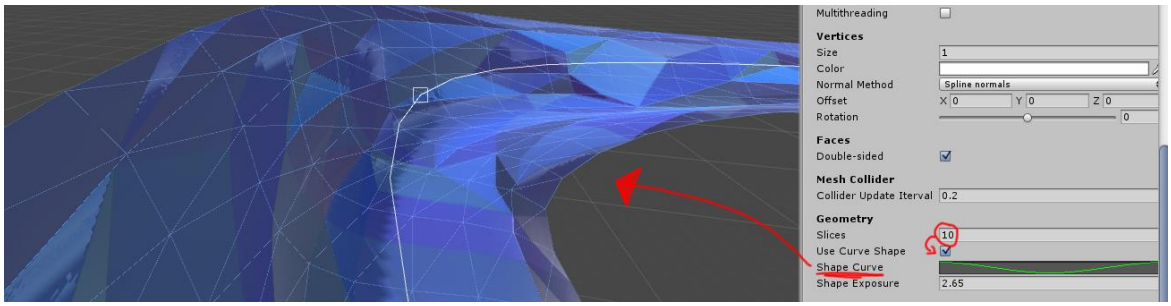


The Path Generator also supports colliders. Adding a Mesh Collider component to the object that has a Path Generator will automatically update the Mesh Collider to fit the generated mesh. Since updating mesh colliders is a heavy task, when a Mesh Collider is present an additional value will be exposed in the inspector. This value is called “Collider Update Interval” and defines how frequently the mesh collider will be updated.



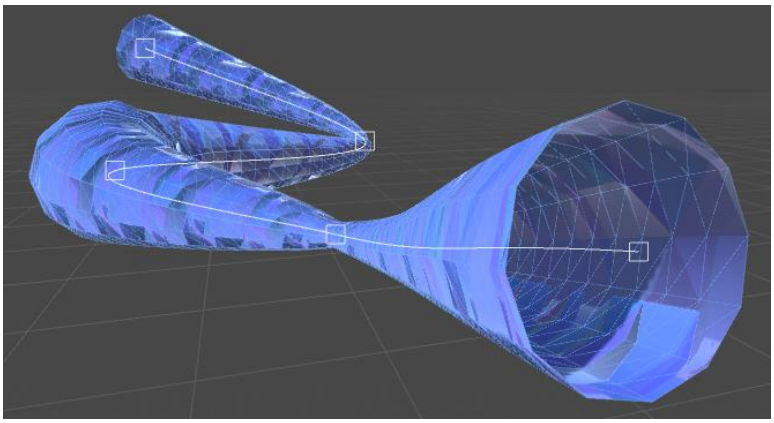
Setting this value to 0 will cause the Mesh Collider to update as soon as there is a change in the mesh. Note that the Mesh Collider will not update if there isn’t a change in the mesh.

Like the Spline Renderer, the Path Generator has the option for multiple edge loops (called Slices) to prevent jagged texture coordinates. However, the Path Generator’s slices also serve a function of deforming the generated path. Turning on the “Use Shape Curve” option will expose a curve editor. If the slices of the Path Generator are set to a value bigger than 1, then the path can be deformed using the curve editor.

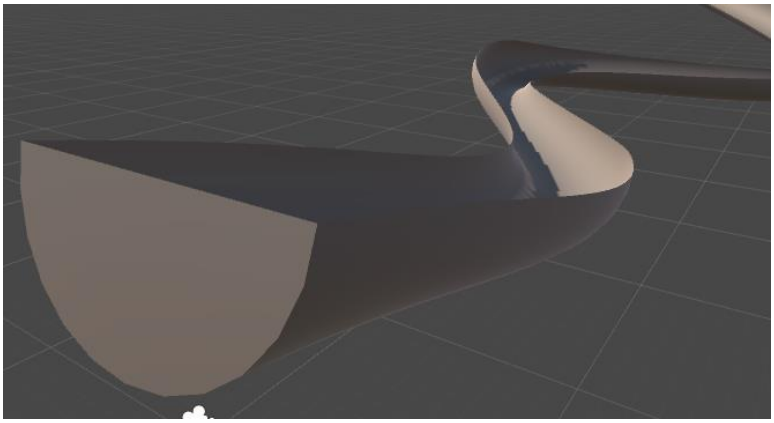


5.7. Tube Generator

The Tube Generator is another Mesh Generator, derived from the SplineMeshGeneration class. It Creates a tube along a spline and is affected by the same things the Path Generator and the Spline Renderer are.

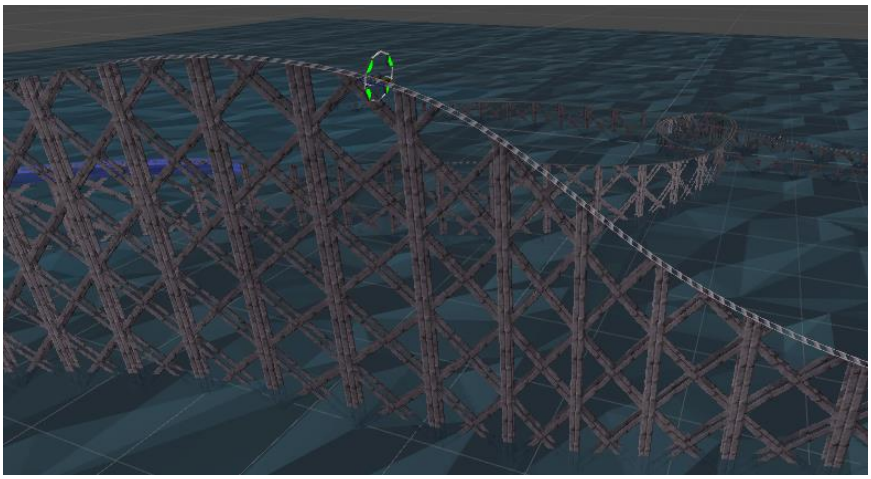


It's ends can have caps and it can also represent a half-pipe instead of a whole tube by reducing it's Integrity value.



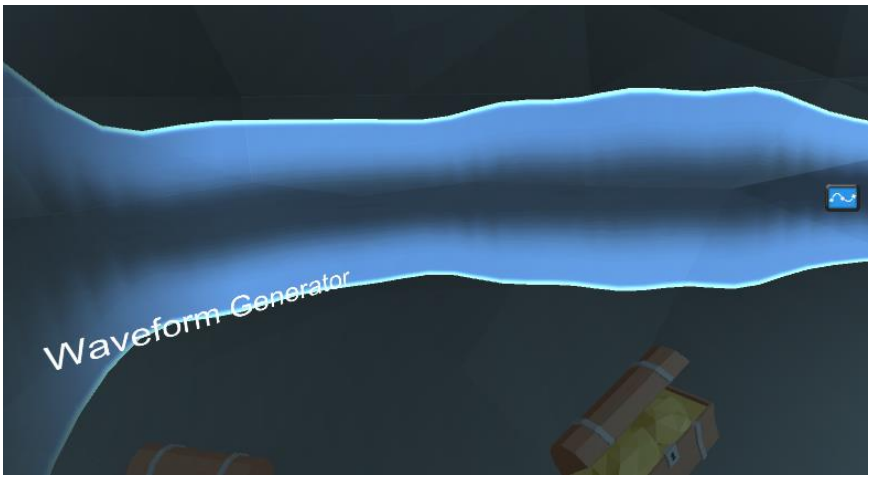
5.8. Waveform Generator

The Waveform Generator creates curved surfaces that resemble a waveform. It can be used for audio visualization, 2D terrains, fences and other structures. In the Roller Coaster example, Waveform Generators are used to make the railway supports.



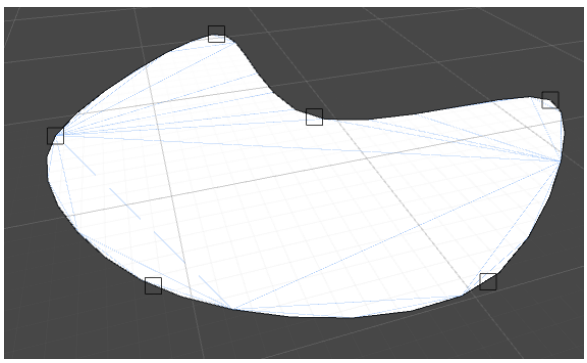
The Waveform Generator does not use the spline points' sizes for the mesh generation unlike the other three components described above.

There is a symmetry option which causes the waveform's shape to mirror along the axis it's created on.

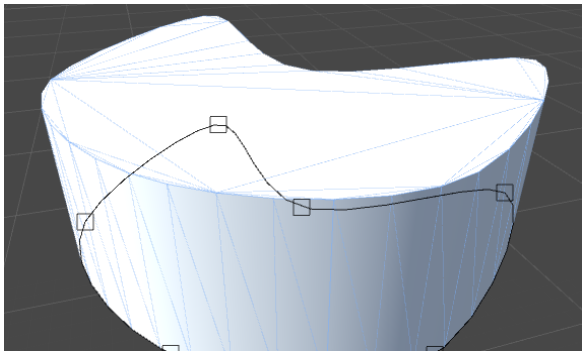


5.9. Surface Generator

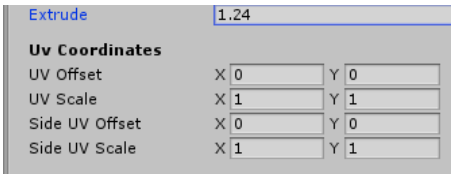
The Surface Generator creates a 3D surface from a spline. These surfaces can be used for creating different kinds of platforms, natural water pools, shape animations etc. They don't require the spline to be closed to work.



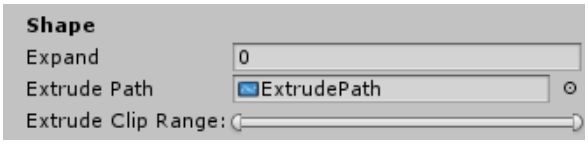
By default, the generated surfaces don't have thickness which may not look good in some situations. To add thickness to the surface, increase the "Extrude" value in the inspector.



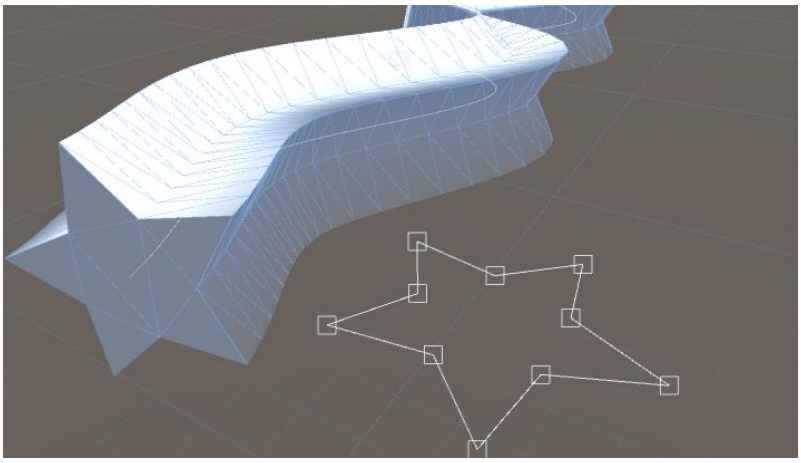
If Extrude is different than zero, a secondary set of UV coordinates which control the side UVs will be exposed in the inspector.



Version 1.0.3 introduces extrusion along another Spline. To extrude the surface along a spline, drag the SplineComputer component of the extrusion spline in the "Extrude Path" field in the Inspector:

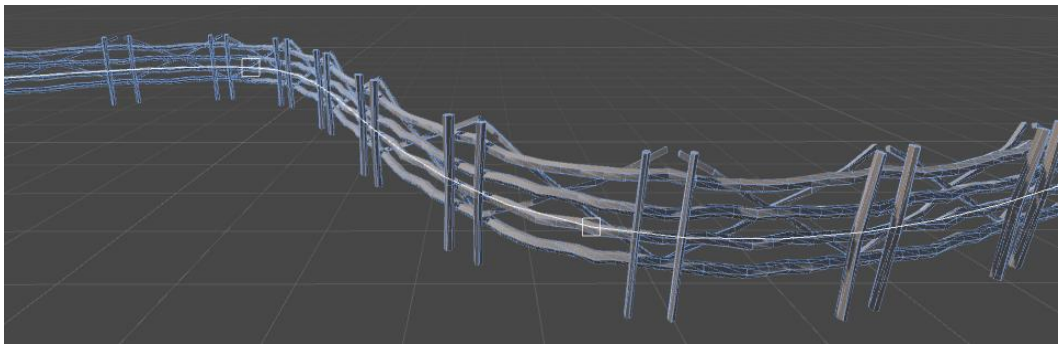


Doing so will automatically extrude the generated shape along the extrusion path.

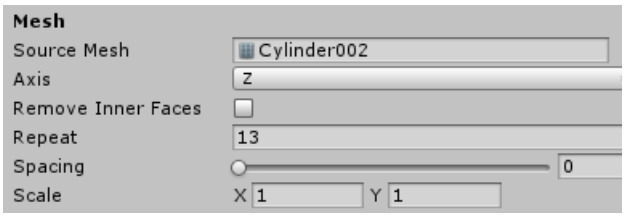


5.10. Extrude Mesh

The Extrude Mesh component extrudes a mesh along a spline.

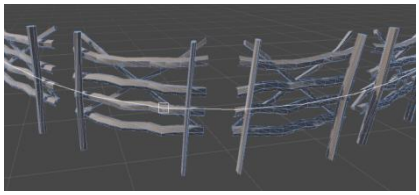


When an Extrude Mesh component is added it will require an input mesh to work. The mesh is selected using the Object field “Source Mesh”. The object field accepts Game Objects and Renderers too. If a Game Objects is passed, it will try to find a Mesh Renderer in the object. If a Mesh Renderer is found, then the Mesh Renderer’s mesh and materials will be used for the Extrude Mesh object.

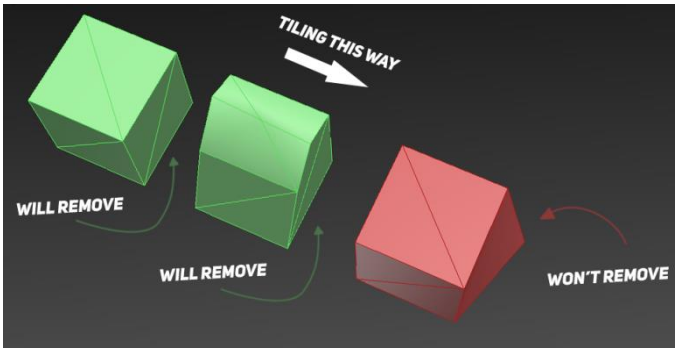


Once a mesh is set, it will automatically get extruded along the spline. The Axis dropdown menu changes the axis along which the mesh will be extruded. Repeat controls how many times the mesh will be repeated.

Increasing the Spacing will add space between each repeat. The Spacing actually shrinks the mesh along the extrude axis. A spacing of 1 will result in an infinitely thin mesh while a spacing of 0 will result in stretched meshes without space in between.



Remove Inner Faces will attempt to remove the triangles of the mesh which have become invisible due to the tiling (when repeat is greater than 1). For this function to work, the end faces of the mesh have to be completely flat and vertical (like the walls of a cube). Even the slightest vertex offset may cause the function to fail.

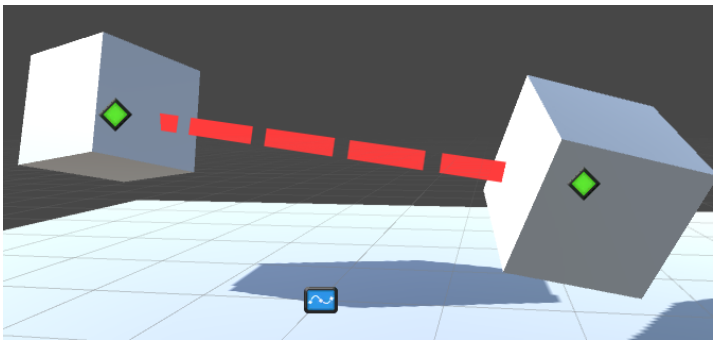


5.11. Length Calculator

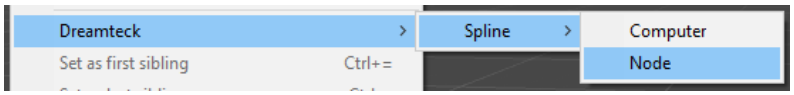
A basic component which calculates the length of the spline using the approximation rate defined by the Spline Computer's precision and the Resolution of the Length Calculator component.

6. Nodes

Nodes are a key feature and serve two very important purposes - to bind points of Spline Computers to other scene objects and to create junctions.

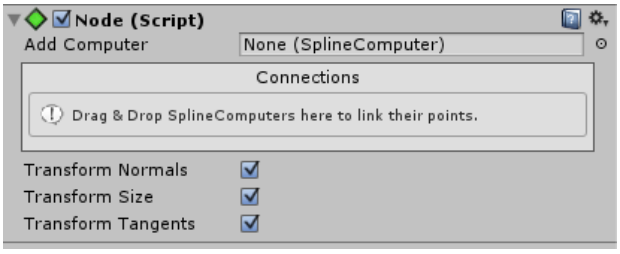


To create a Node, go to Game Object->Dreamteck->Spline->Node. This will create a new Game Object with a Node component in the scene.

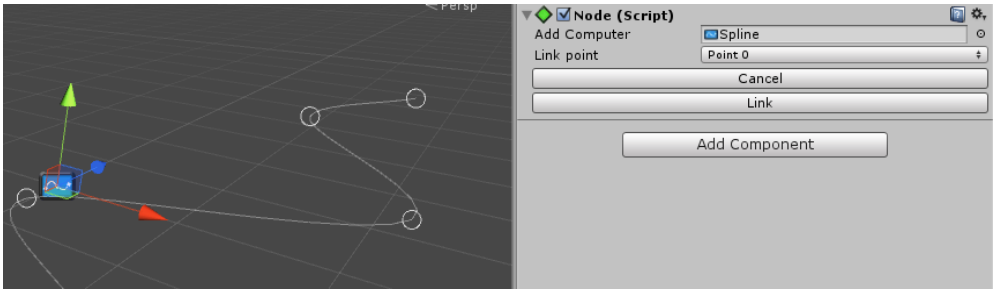


Another way to create a Node is just by adding a Node component to an existing object.

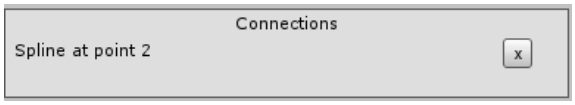
The newly created node will be empty and will require a connection to be added. To add a connection, drag a Game Object with a Spline Computer from the scene into the Connections box or select it through the object field above the connections box.



When a Spline Computer is added, it's points will visualize as circles and a point dropdown menu will be shown in the inspector.

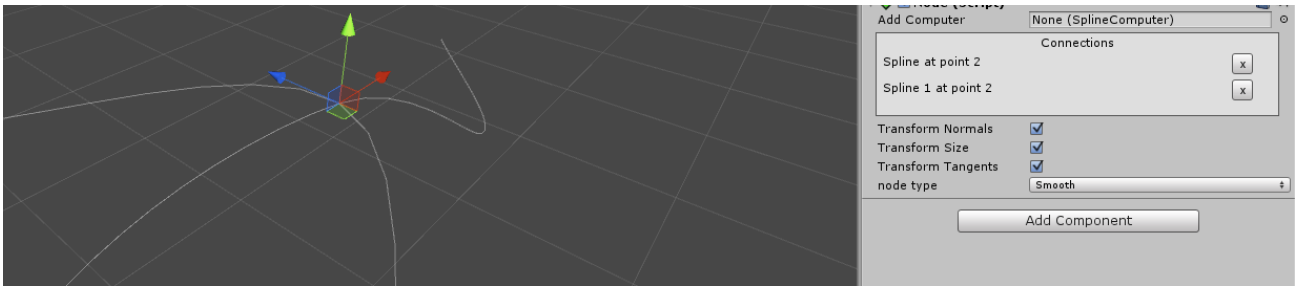


The Node will now require a point from the Spline Computer to be selected. To select a point either select it from the dropdown menu and click “Link” or click on a point in the scene view. After a point is selected, a new connection will be added in the Node. The new connection will be listed in the Connections box.



After a connection is added, the spline point of the added Spline Computer will snap to the Node’s transform. The Node’s Transform can then be animated or a Rigidbody can be attached and when the Node moves, the connected point will also be updated.

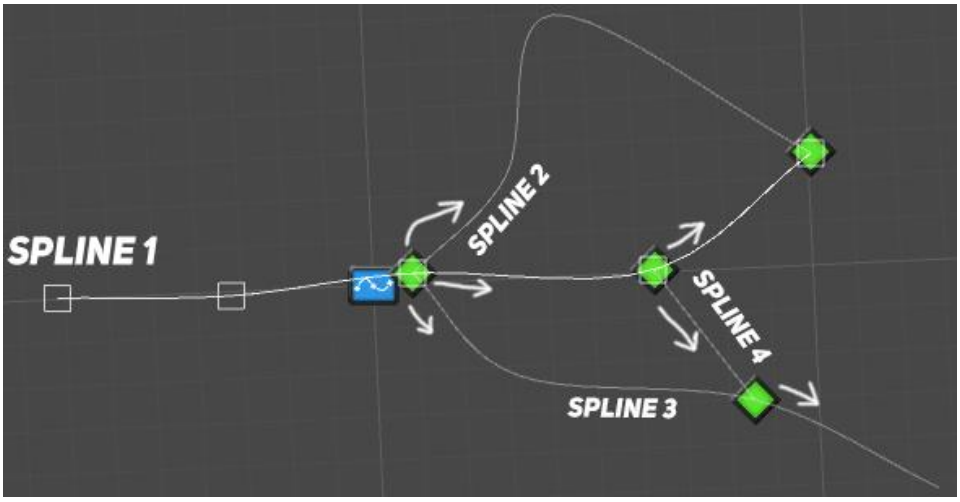
A Node can have more than one Spline Computer connected to it. Repeat the procedure with another Spline Computer to add a new connection.



When editing a point that is connected to a Node in the Spline Editor, the node’s position will also be updated and therefore all the other Spline Computers connected to it.

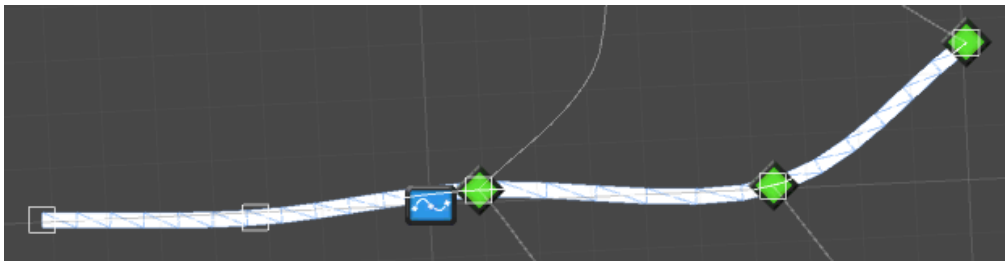
6.1. Junctions

When two Spline Computers are connected to the same Node, this is considered a junction. Junctions do not affect the shapes of the splines. When junctions are created nothing changes in the Spline Computer component except that it receives information about the junction.

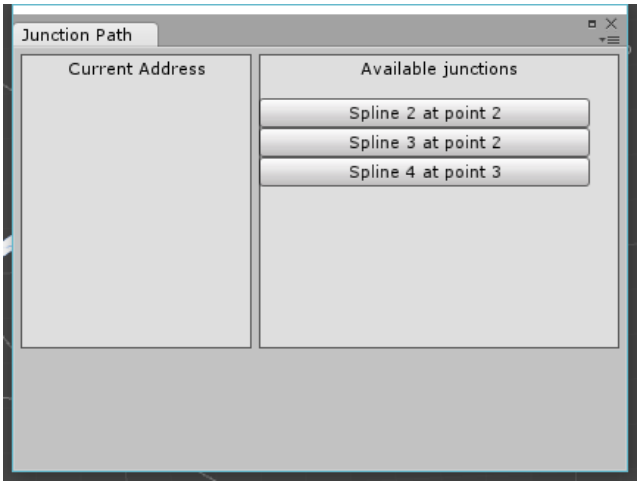


To make use of the junctions, a junction path must be set in the Spline User components. When a Spline User uses a Spline Computer with junctions, a new button called “Edit junction path” is shown in the Spline User inspector.

Using the junction example above, a new Spline Renderer is added to Spline 1. This will result in geometry being created along Spline 1

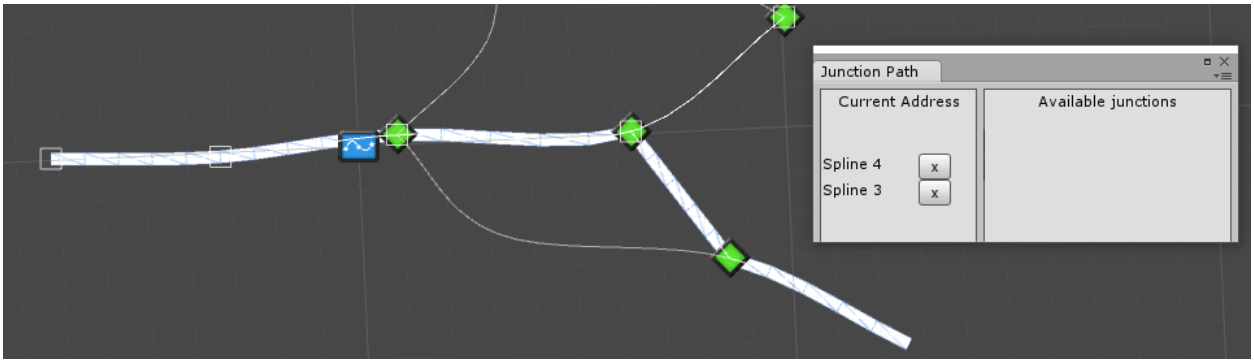


Clicking the “Edit junction path” will open the Junction Path window.



The Junction Path window contains two panels. On the left is the Current Address panel. This panel shows the currently set junction path. For example: Spline 4, Spline 3.

On the right are the available junctions. The available junctions are all of the junctions in the current spline. In the above case there are three junctions along Spline 1 and they are listed in the panel. Clicking on a junction will add it to the current address.

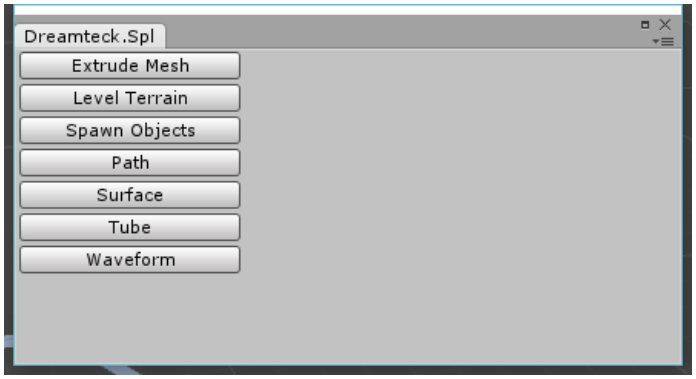


To remove a junction from the current address click the cross button next to it in the Current Address panel. Removing a junction from the current address will remove all other junctions that have been added after it.

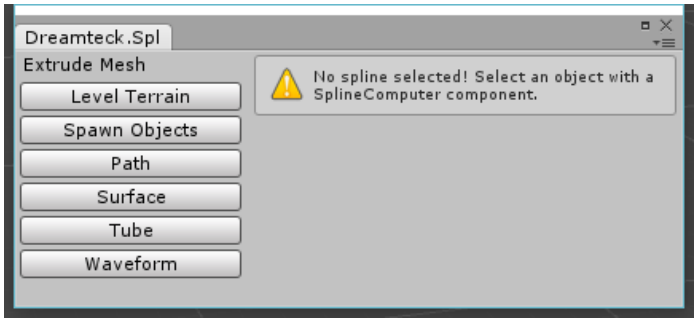
7. Editor Tools

The Spline User is a component that is mostly meant to work in runtime although it can be configured in the editor. Sometimes, however, level designers just need a tool to help them in the work process. For pure level design purposes, Dreamteck Splines offers the Spline Tools window which is located in **Window->Dreamteck->Spline Tools**

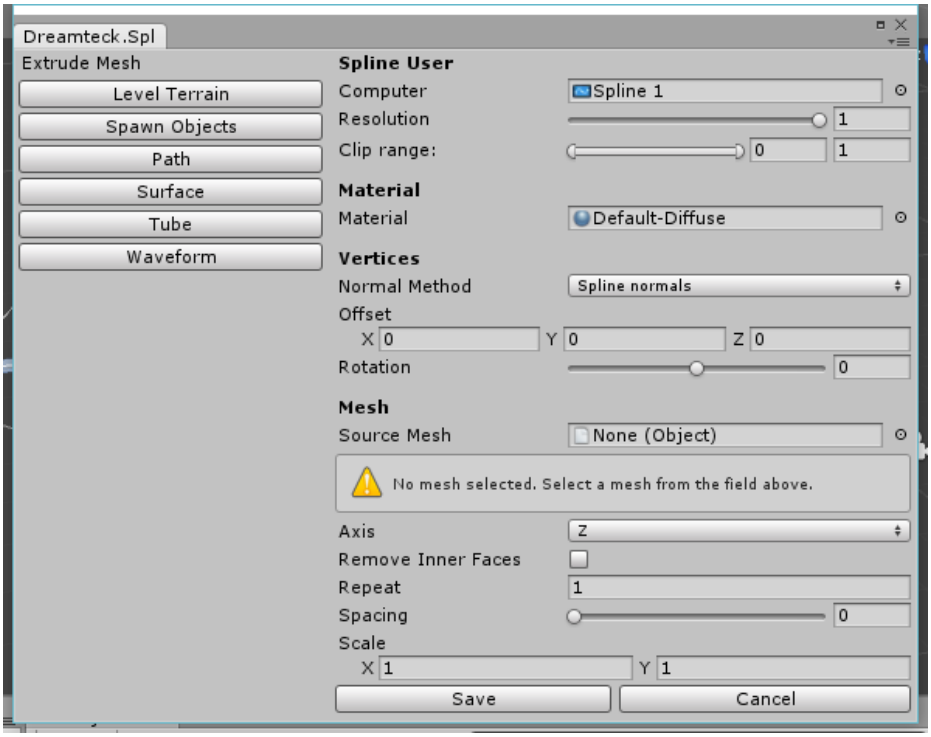
This will open a window with a set of tools for level editing:



Each tool can be selected by clicking on it's button. All tools require an object with a Spline Computer to be selected in the editor in order to work. Otherwise a warning will be displayed:



Select an object with a Spline Computer component from the scene and focus the window again. The warning will disappear and the tool UI will open.



After setting up the object, it can be saved with the “Save” button.

7.1. Generating meshes using Spline Tools

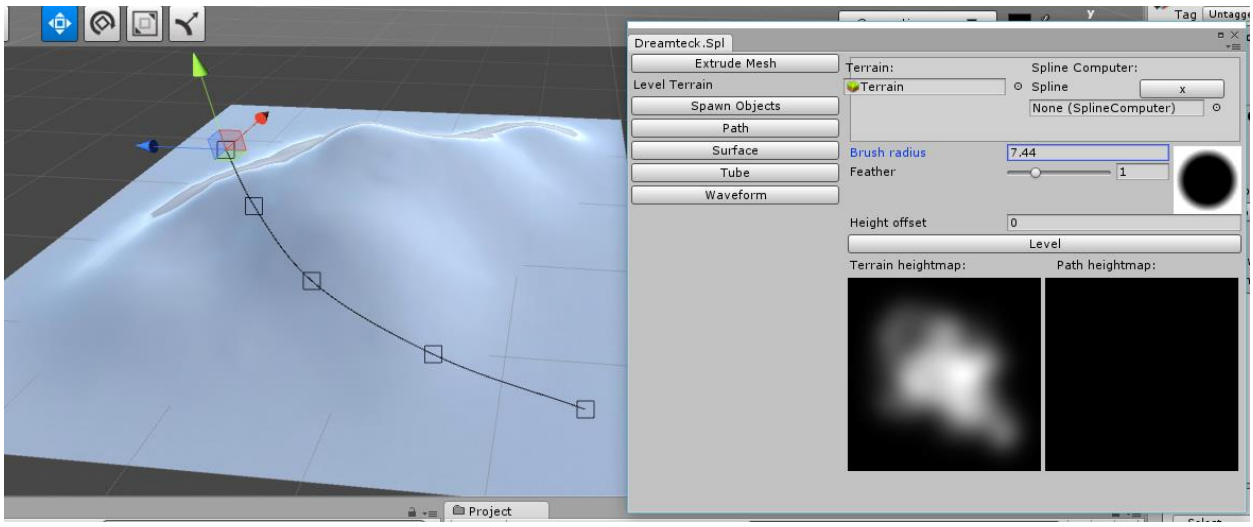
Most of the tools from the Spline Tools window are mesh generators it’s important to know how to save a mesh, generated with a Spline Tool.

When “Save” is clicked on a mesh tool, it will ask whether or not Lightmap UVs should be created for the new mesh. Click “Yes” if the mesh will be lightmapped. After the first dialog, a second one will appear. This one is very important because it asks whether or not the newly created mesh should be saved as an asset in the Assets folder. Click “Yes” if the generated object will be used in multiple scenes or prefabs will be made from it. If the generated object is meant only for the current scene, then saving it as an asset is not necessary.

7.2. Leveling terrain

The Level Terrain tool is an editor-only tool meant for sculpting terrains using splines.

For the Level Terrain tool to work, the scene needs to have at least one terrain and a Spline Computer needs to be selected. If the scene has more than one terrain, the terrain that should be leveled should also be selected along with the Spline Computer.

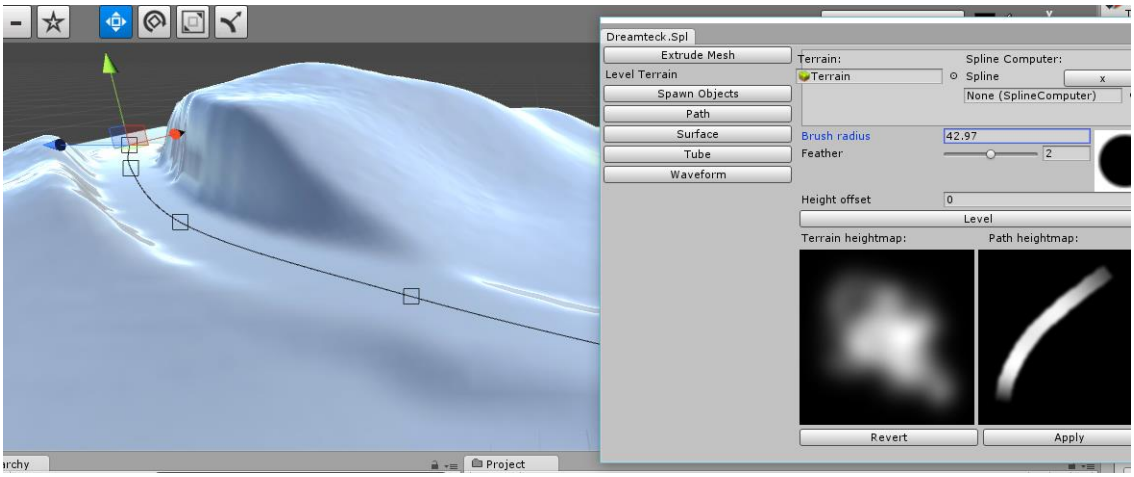


Once open, the Level Terrain tool will display a preview of the terrain’s heightmap, the path heightmap which will be black and a preview of the brush feather.

The tool has only three settings:

- Brush Radius – the radius of the brush in world units
- Feather – the feather of the brush
- Height Offset – height offset

To level the terrain using the selected spline(s), click the “Level” button. The path heightmap will be drawn and the terrain will be sculpted.



An “Apply” and “Revert” buttons will appear under the heightmap previews. If the result is not satisfying, edit the spline and the settings and click Level again – the previous level will get replaced with the new one. To save the terrain, click “Apply” – this will write the level data to the terrain permanently.

The brush radius is affected by the sizes of the points.

8. Writing a custom SplineUser class

Dreamteck Splines’ functionality can be expanded easily by creating new SplineUser components. For a SplineUser class to be created it needs to derive from the SplineUser base class. Before deriving the class from SplineUser, the following using directive should be added to the new script:

```
using Dreamteck.Splines;
```

A basic SplineUser example is provided in the Components folder. It can be used as a template for creating SplineUsers.

8.1. Protected and virtual Methods

Each SplineUser class inherits a set of methods which are automatically called. These methods are automatically called and should be overridden in order to create custom functionality.

SplineUser classes are not supposed to have Update, FixedUpdate and LateUpdate methods. They can have a Start method and an Awake method through overriding:

```
protected override void Awake()
{
base.Awake();
//Code here
}
```

8.1.1. Protected virtual void Run()

Run is called automatically on Update, FixedUpdate or LateUpdate based on the Update Method setting of the SplineUser. It's mainly used for game logic that will run on each update cycle.

8.1.2. Protected virtual void Build()

Build is called when there has been a change in the spline user. For example if the spline has changed or a property of the SplineUser has been set. Build can either be called from the main thread or from a separate one if multithreading is enabled. Therefore it is advised that there should be only thread-safe code inside. Make call calculations inside Build and then apply them in PostBuild.

8.1.3. Protected virtual void PostBuild()

PostBuild is always called on the main thread after Build has finished even if multithreading is enabled. It's used to apply the results, calculated in Build(). For example if the custom SplineUser will position objects along path. The objects' positions should be calculated in Build and then applied to the objects' transforms in PostBuild().

8.2. Rebuilding

The SplineUser class has the Rebuild(bool sampleComputer) method which forces the user to recalculate on the next update cycle.

SplineUsers rebuild automatically when their spline is modified or their properties are changed. That however does not happen when the properties of custom SplineUser classes are modified. In order to make the SplineUser class rebuild, custom setters should be implemented that call the Rebuild method. Example:

```
private bool _state = false;
public bool state
{
    get { return _state; }
    set
    {
        if (value != _state) //Only rebuild if the value is different
        {
            _state = value;
            Rebuild(false); //Rebuild but don't resample the spline
        }
    }
 }
```

Rebuild is a public method so it can also be called from another object. Rebuilding happens once per update cycle no matter how many times it's called prior to that.

8.2.1. RebuildImmediate

Rebuild waits for the next update cycle so calling it will not yield any changes immediately. In some cases, however, rebuild should happen on demand and that's what RebuildImmediate(bool sampleComputer) is for. RebuildImmediate calls he rebuild sequence regardless of update cycles and will yield results immediately.

Calling RebuildImmediately multiple times per update cycle will cause the SplineUser to rebuild as many times as RebuildImmediately is called.