

LeanTween Editor - Documentation

- [Installation](#)
- [Bezier Paths](#)
- [Visual Editor](#)

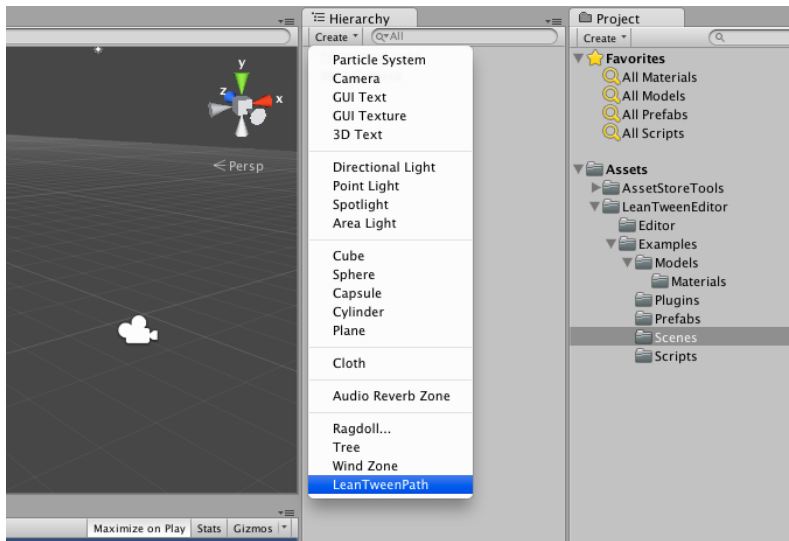
Installation

The first step you need to take is to move the Plugins folder into the root of your Assets folder. If you already have a Plugins folder just move all of the contents under LeanTweenEditor/Plugins/* into this folder. This should get rid of any errors you may see after importing the LeanTween Editor.

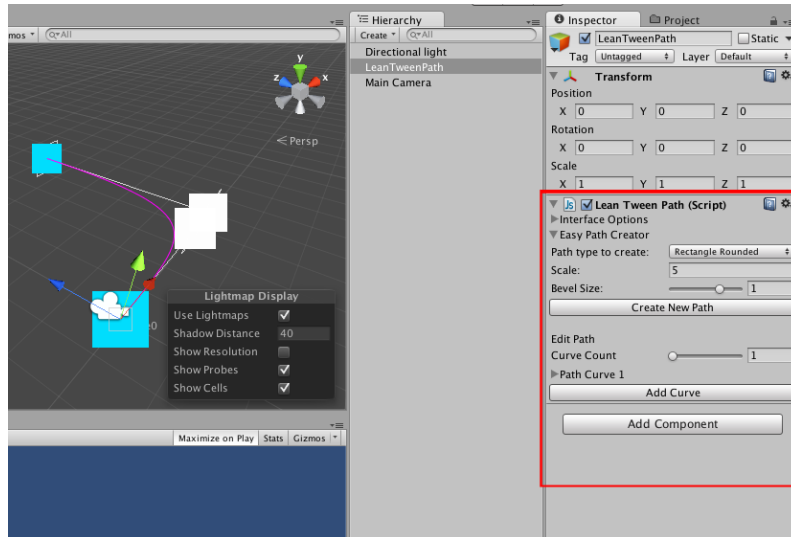
Bezier Paths

[LeanTween](#) now supports animation along Bezier curves, and this editor gives you a visual system in which to create them! Create one Bezier curve or a complex set of Bezier curves.

To create a path simply select the Create drop-down menu from the hierarchy window and select LeanTweenPath.



To edit this path click on the object and you should see a LeanTween Path Editor show up in your components window.



Programming

To use this path in your project create a public variable in your script such as:

```
public var ItPath:LeanTweenPath; // Javascript
```

```
public LeanTweenPath ItPath; // C#
```

Then you can animate over this path by passing it to LeanTween like:

```
// Javascript
LeanTween.move(It, ItPath.vec3,
4.0).setOrientToPath(true).setDelay(1f).setEase(LeanTweenType.easeInOutQuad).setOnComplete(loopAgain);
```

```
// C#
LeanTween.move(It, ItPath.vec3,
4.0f).setOrientToPath(true).setDelay(1f).setEase(LeanTweenType.easeInOutQuad).setOnComplete(loopAgain);
```

Or iterate over it yourself manually such as:

```
// Javascript
public var ItPath:LeanTweenPath;
private var ItBezierPath:LTBezierPath;
private var iter:float;

function Start () {
    ItBezierPath = new LTBezierPath(ItPath.vec3);
}
function Update () {
    transform.position = ItBezierPath.point( iter );
    iter += Time.deltaTime;
    if(iter>1.0)
        iter = 0.0;
}
```

```
// C#
using UnityEngine;
```

```

public class LTPathExampleSimpleCSharp : MonoBehaviour {
    public LeanTweenPath ltPath;
    private LTBezierPath ltBezierPath;
    private float iter;

    void Start () {
        ltBezierPath = new LTBezierPath(ltPath.path);
    }

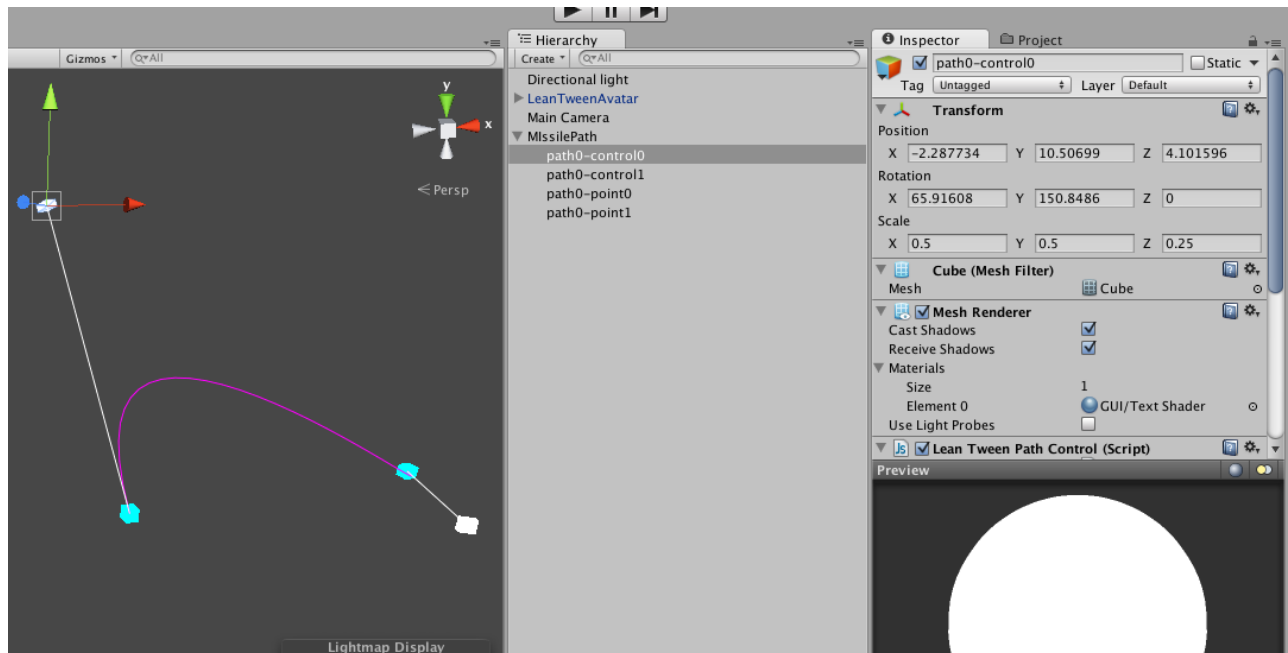
    void Update () {
        transform.position = ltBezierPath.point( iter );
        iter += Time.deltaTime;
        if(iter>1.0f)
            iter = 0.0f;
    }
}

```

See the included examples for more detailed information!

Editing Created Path

You can edit the created path by manipulating the the objects in the scene window or by drilling down into the child nodes of the path and adjusting the positions by changing their value in the transform component inspector.

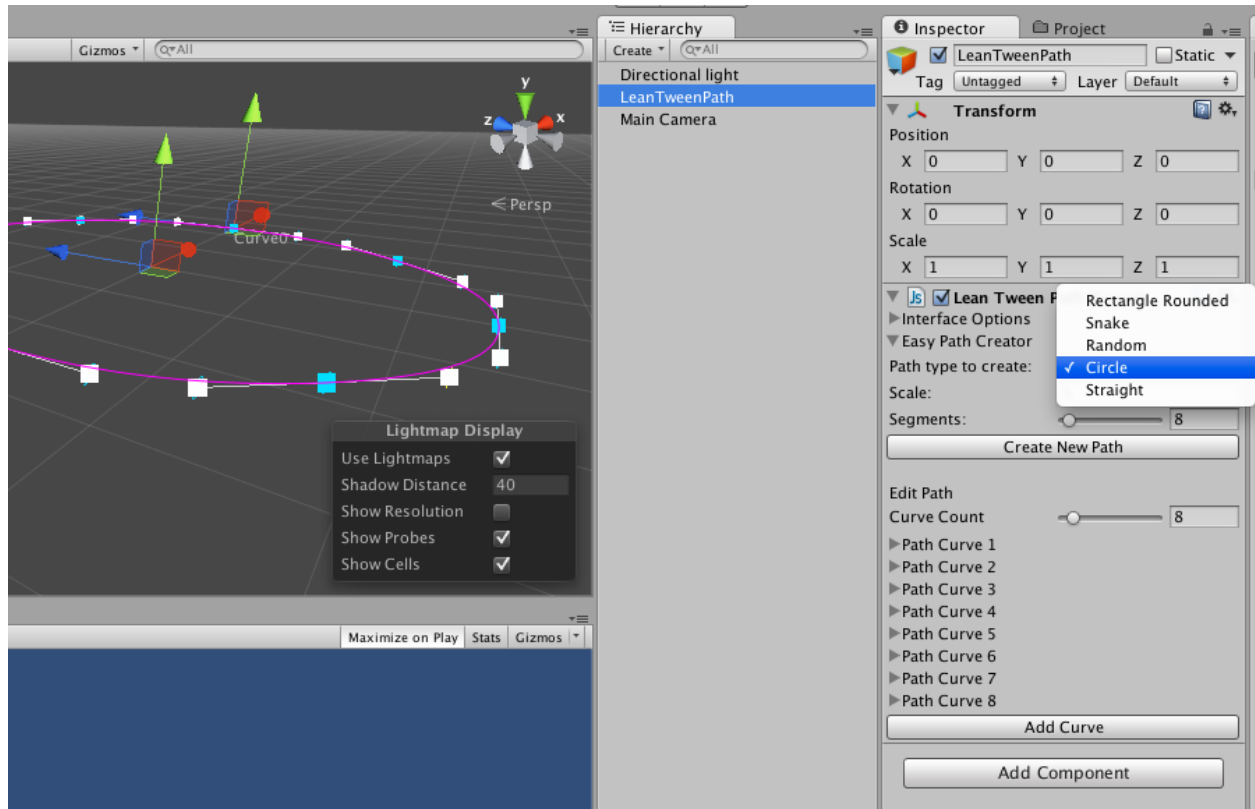


Add additional curves, be pressing the add curve button (or increasing the curve count slider). Delete curves by changing the curve count with the slider or by pressing the delete button.

Path Advanced Features

Easy Path creation:

Create paths such as rounded rectangle, circle, snake and random from the easy path creation tool. Fine tune these paths by selecting different options, then once you are done press create (warning this will erase any path that is currently there).



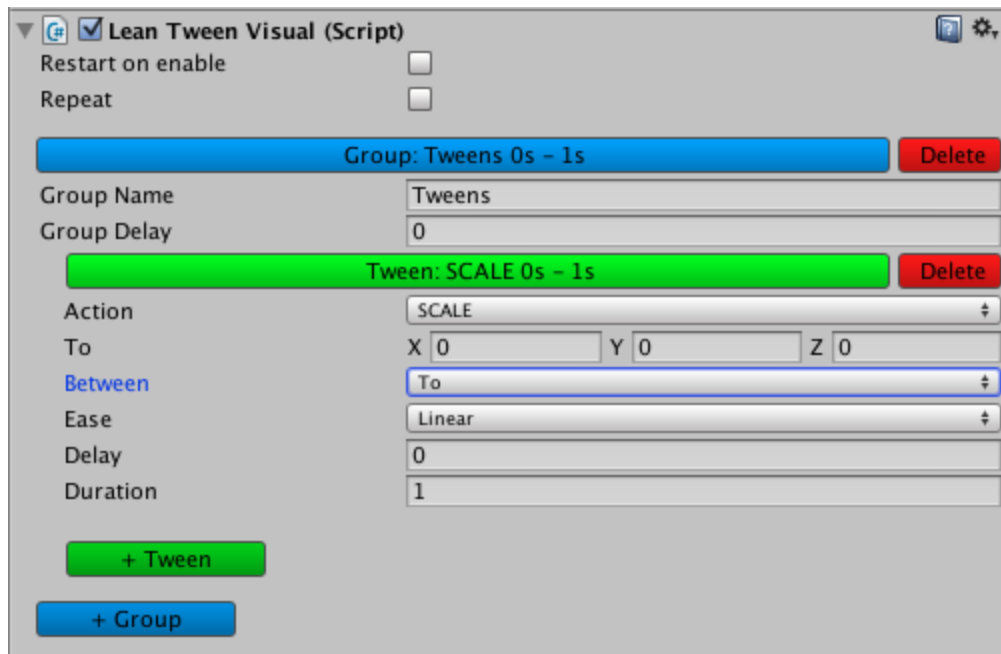
Visual Editor

Now it is easy to create, LeanTween animations without having to write any code! Of course LeanTween is still primarily is a tool designed for programmers, but there are occasions where it is just handier to create your animations in the editor and not have to bother with programming.

To add a LeanTween animation to an object, select your object in the scene and click “Add Component” then Type in “LeanTweenVisual” to add the visual editor.

Next add a Group (groups are just convenient buckets to organize a set of tweens, you can also do convenient things like repeat a certain group, or have it restart on enable). Next add your

tween. If you are regular LeanTween user, you should recognize all of the tween actions from the drop-down.

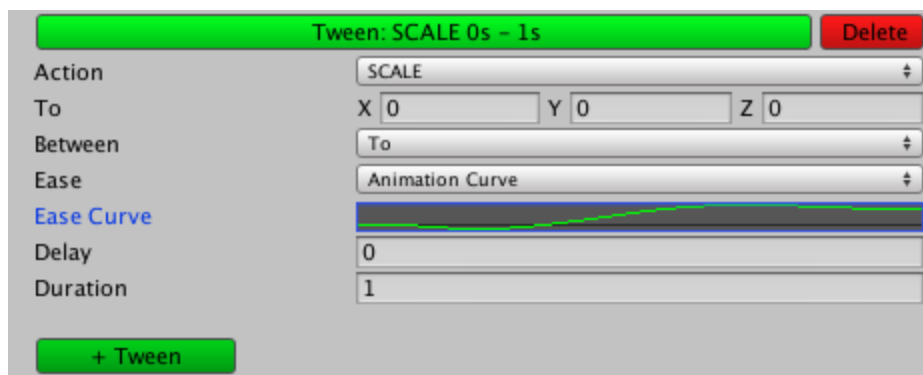


Between

The “Between” attribute defines whether or not you would like to have the object tween from it’s current value (“To”), or if you would like it to start from an arbitrary value (“From To”).

Animation Curve

If you select the “Animation Curve” ease-type, it will allow you to define the custom-ease directly in the editor.



Bezier Path

Define a path for an object to follow, by selecting either MOVE_CURVE, or MOVE_CURVE_LOCAL. The “Orient to Path” toggle defines whether the object looks along the path the gameobject is moving along.

Tween: MOVE_CURVED 0s - 10s

Delete

Action	MOVE_CURVED
CameraPath	
Orient to Path	<input checked="" type="checkbox"/>
To	0
Between	To
Ease	Linear
Delay	0
Duration	10