Line Renderer 3D

Technical Documentation
Author: Eugen Axenoi

This package is designed to create line renderers as 3D tubes similar to Unity's native flat line renderers.

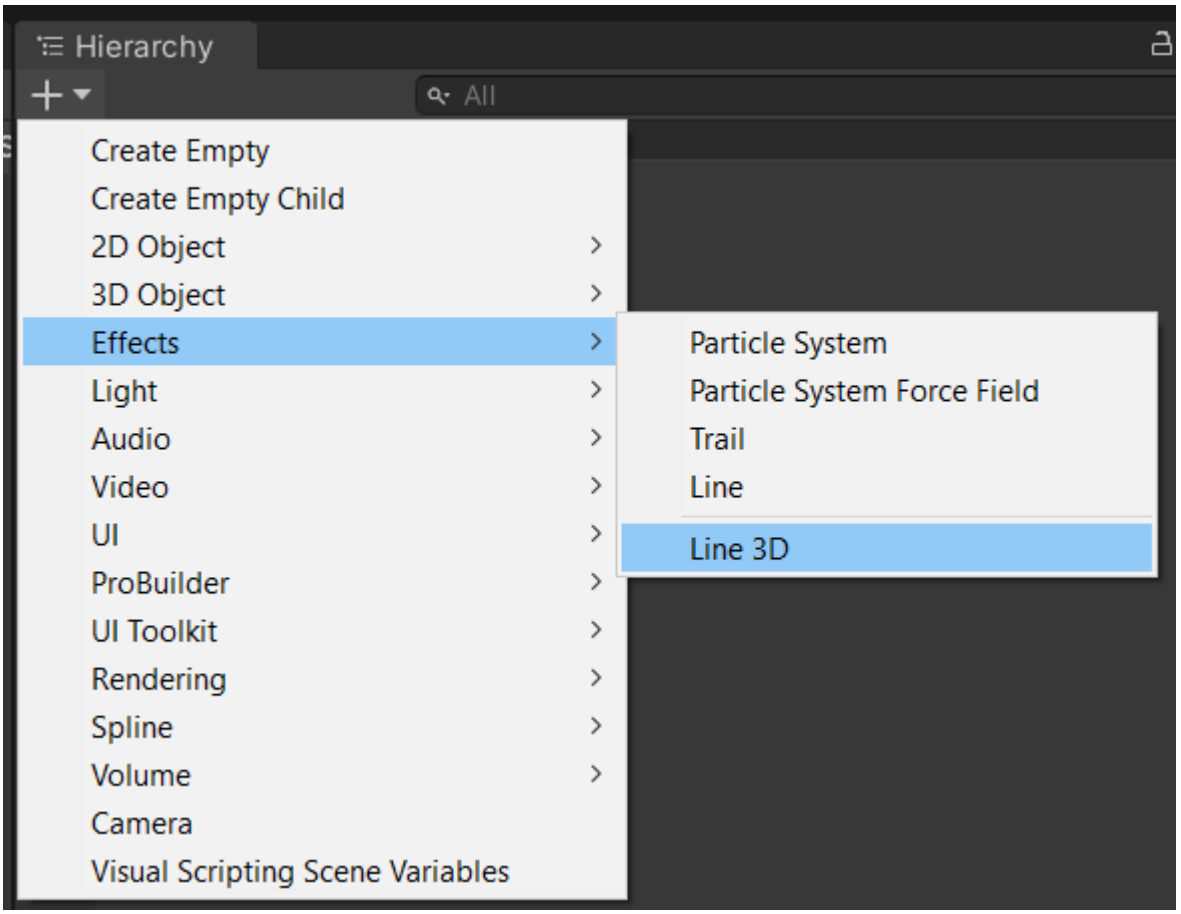
How to create instances

A line can be created:

- From script

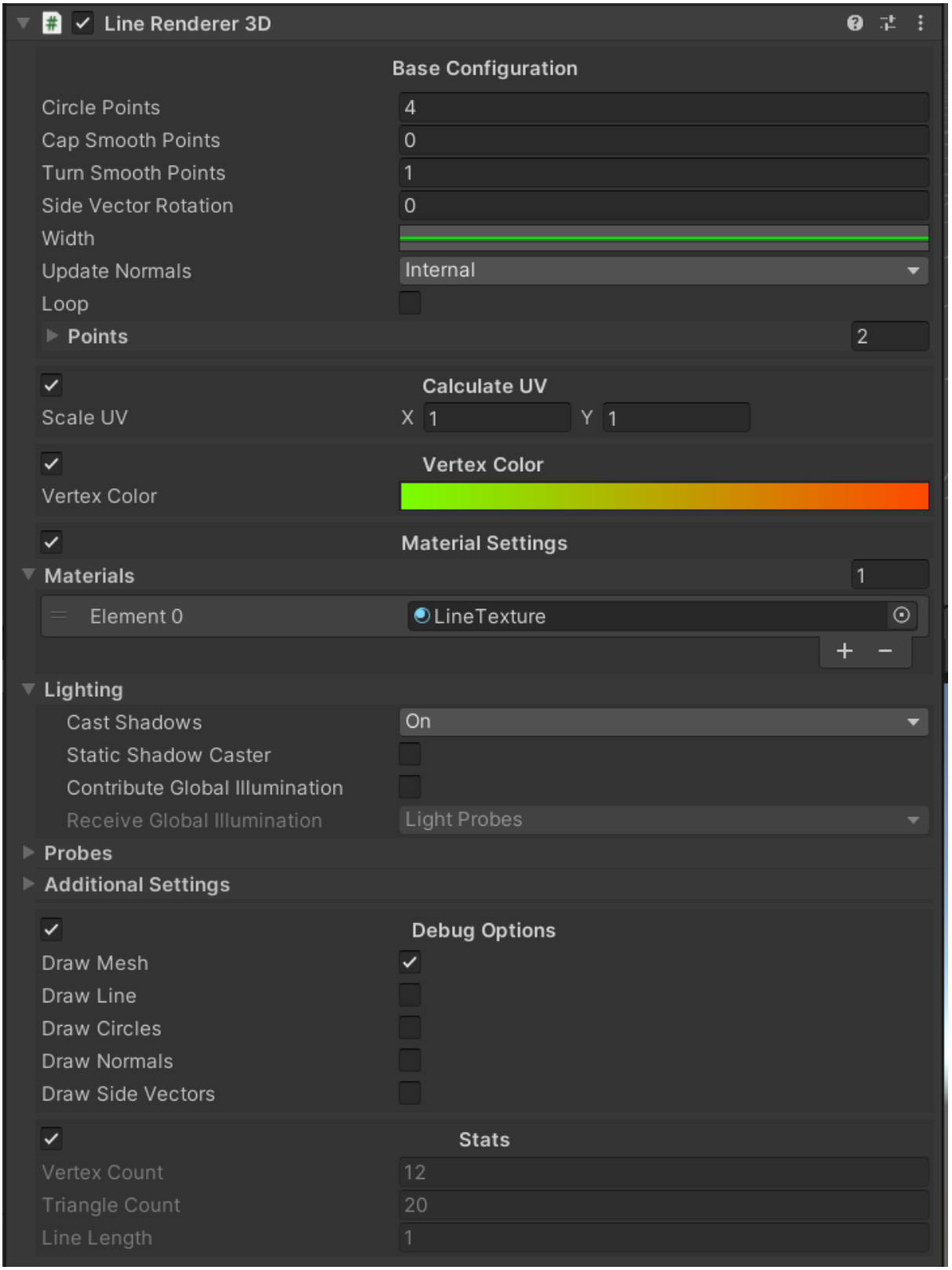
```
GameObject gameObject = new GameObject("Line Example");
var instance = gameObject.AddComponent<LineRenderer3D>();

Vector3[] points = new Vector3[] { Vector3.down, Vector3.up, Vector3.right };

instance.SetPoints(points);
instance.Loop = true;
```

- From scene Hierarchy



The line instance can be edited from script or in inspector.

Instance Control



Configuration Fields

Circle Points

This parameter indicates how many vertices will be created for the line slice. The minimum amount is 3.

Cap Smooth Points

This parameter indicates how smooth the starting and ending caps of a line should be if it is not in loop mode. If it is 0 then there is a single centre point. If it is 1 then it is still a single point but bumped out and forms a cone.

Turn Smooth Points

This parameter indicates how smooth are the turns in the line. The minimum value is 1.

Side Vector Rotation

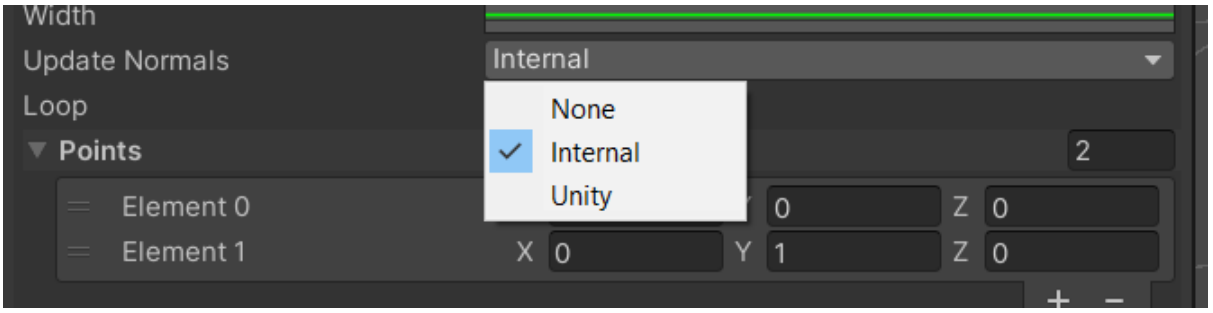
This parameter allows it to rotate the line along itself. Can be used to create some kind of an effect. The rotation is in units where 0 is 0 degree turned and 1 is 360 degrees turned. It can be looped by simply increasing the number.

Width

This animation curve is used to control line thickness along the line. Curve time of 0 is the starting point of the line and curve time of 1 is the ending point of the line.

Update Normals

This parameter allows control of how the generated mesh normals are calculated.



There are 3 variants for now:

- **None** - the normals are not calculated at all
- **Internal** - calculated by the algorithm of the line creation
- **Internal Flat Faces** - same as above but generates flat faces along the line geometry.

Note: This mode generates double the amount of vertices.

- **Unity** - is calculated by calling **Mesh.RecalculateNormals()** on the line mesh.

Loop

This flag is used to make the line loop.

Points

The list of points used to form up the line.

Calculate UV

This flag enables calculations for texture UVs.

Scale UV

This parameter makes sense only if the **CalculateUV** flag is enabled and allows to scale the texture on the line similar to how texture tiling on materials works.

Calculate Vertex Color

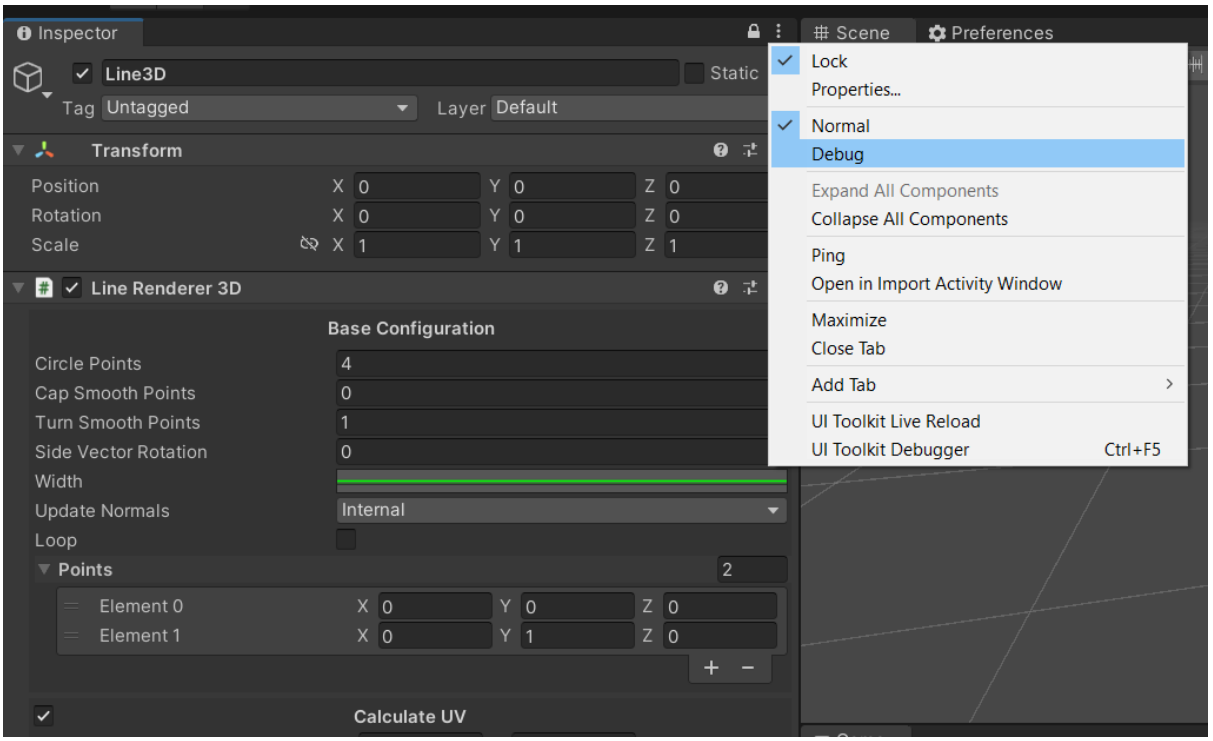
This flag enables vertex color calculations.

Vertex Color Gradient

This gradient param allows to color the line along its length.

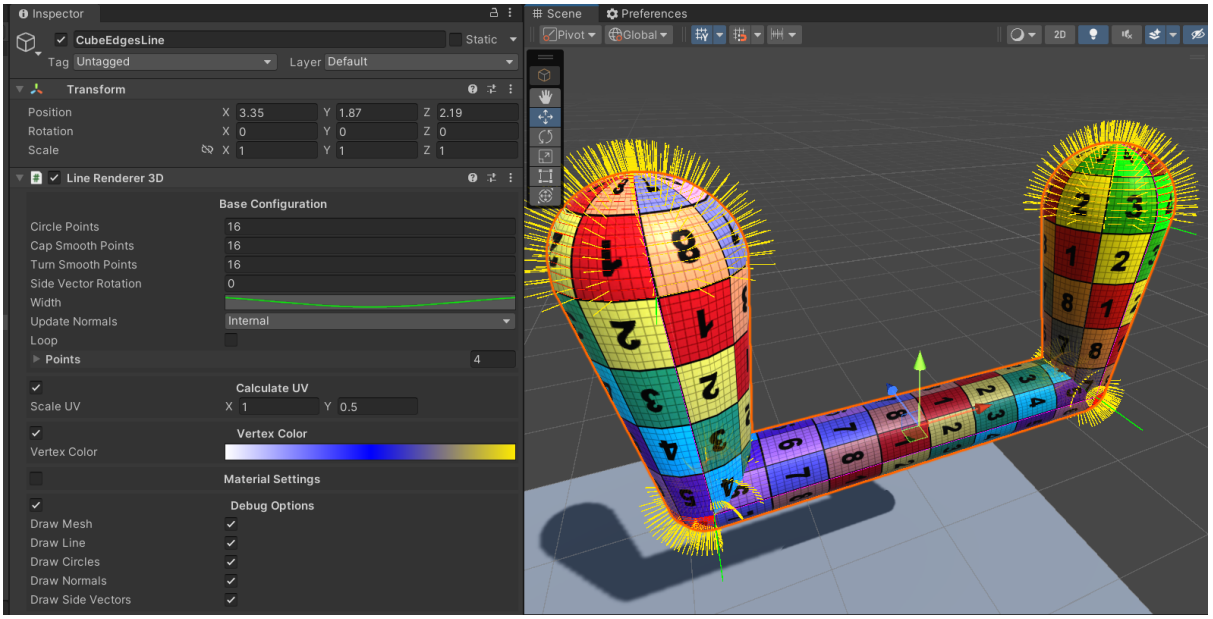
Material Settings

This group contains MeshRenderer's fields to configure material and other lighting params. By Default the MeshRenderer and MeshFilter are hidden in the **Inspector** and can be displayed by accessing the lock icon on the **Inspector** tab.



Debug Options

This group of settings enables some gizmos drawing which helps to understand how the line is created and inspect for bugs.



NOTE: An additional debug tool is an preprocessor define which allows to update line mesh enabling all default checking and error warnings. It is **LINE3D_DEV_MODE**

Stats

This small group displays a couple of simple numbers about the line like vertex count, indices count and line length.