

# **SPLINE POWER!**

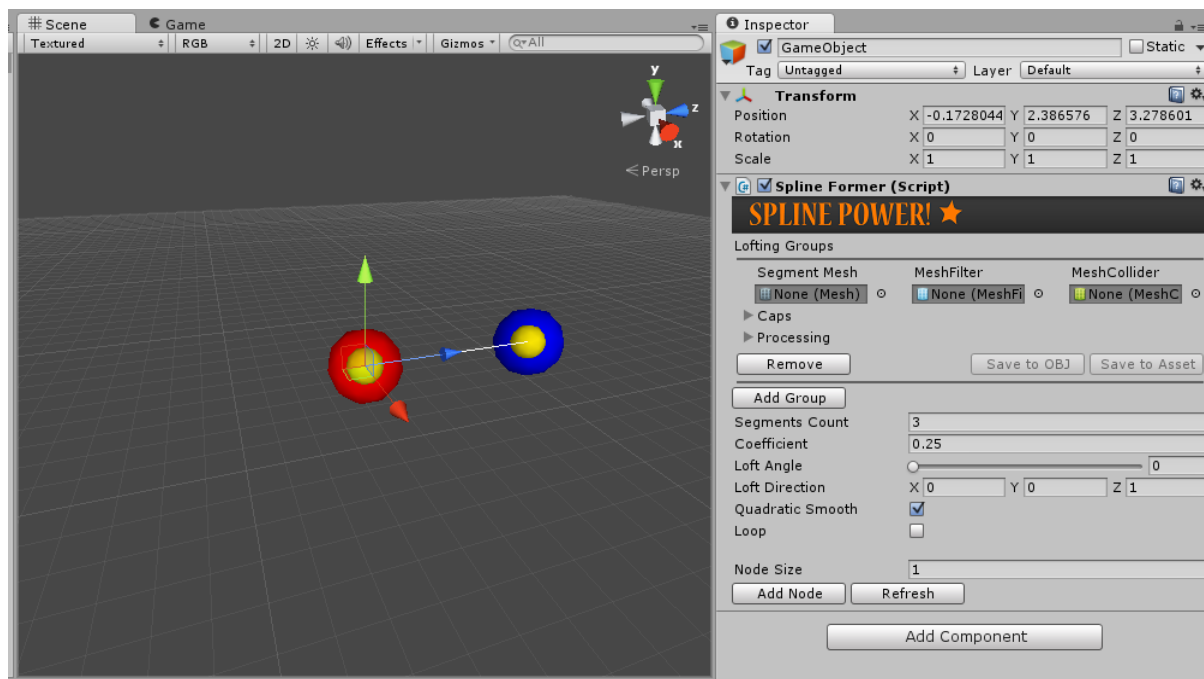
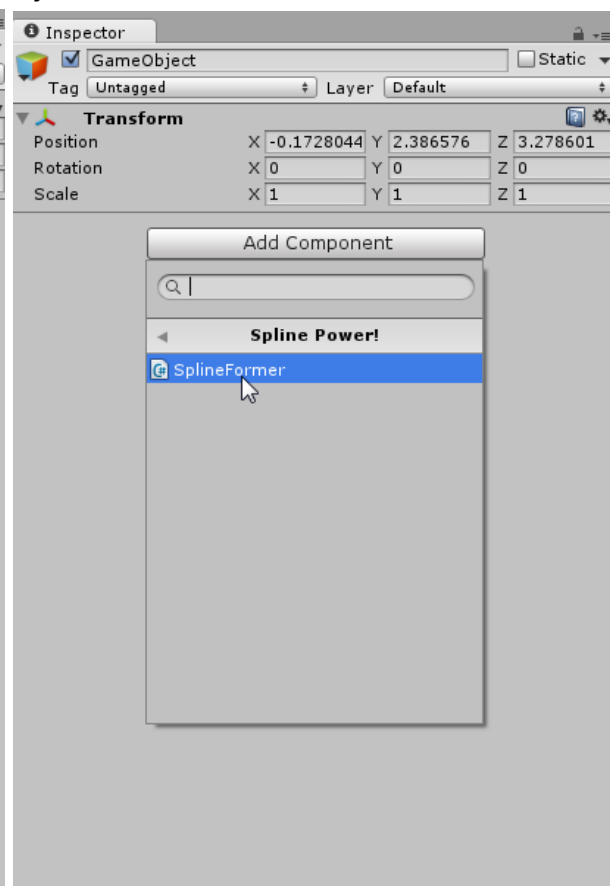
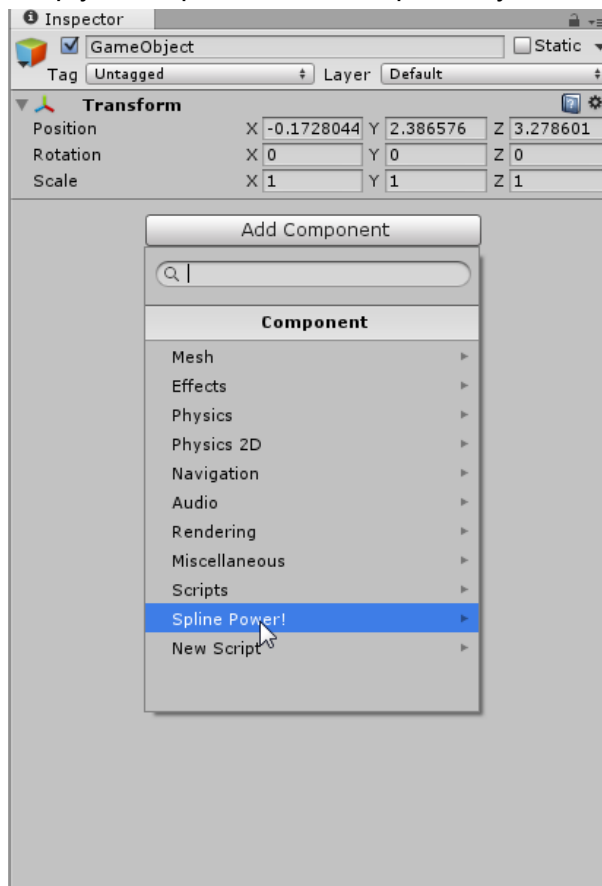
**asset documentation**

1.3

# SPLINE POWER! ★

## Adding Component

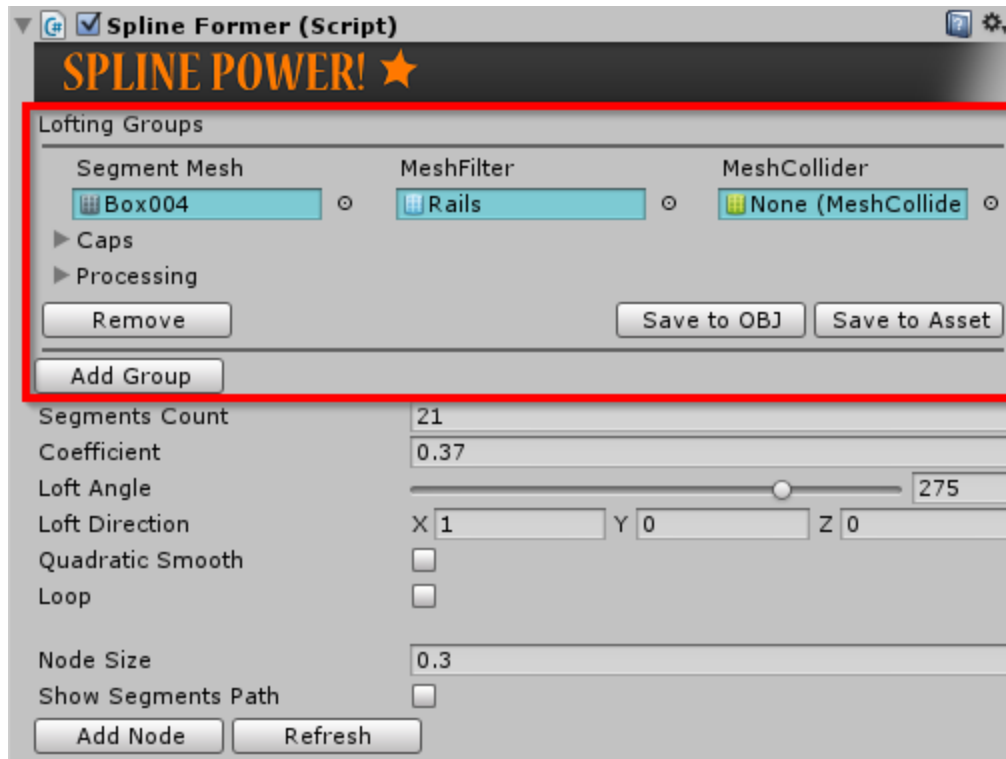
Simply add SplineFormer script to any GameObject.





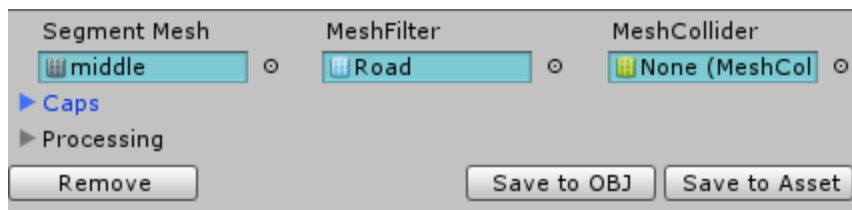
## Interface

### Spline Former:



**Lofting group** - Everything you need for one result mesh. It includes source meshes and options of the generating process.

**Add Group** button - Adds one empty Lofting group.  
**Remove** button - Removes the corresponding Lofting group.



**Segment Mesh** - The mesh from which the segments of resulting mesh will be made. Consider this as middle or the main source mesh.

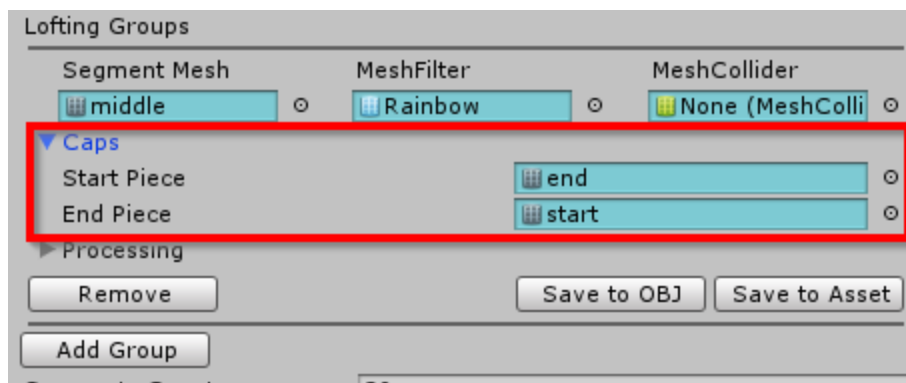
**MeshFilter** *[Optional]* - MeshFilter component to which the resulted mesh would be applied.

**MeshCollider** *[Optional]* - MeshCollider component to which the resulted mesh would be applied.

**Save to OBJ** - Saves the result mesh as OBJ. Button is disabled if there is no result yet.

**Save to Asset** - Saves the result mesh as Asset. Button is disabled if there is no result yet.

Caps:



**Start Piece** *[Optional]* - Source mesh which represent start cap of result mesh.

**End Piece** *[Optional]* - Source mesh which represent end cap of result mesh.

## Processing:



**Process Origin Normals** - Enables the processing for normals where the source mesh normals considered as a source for the result normals. In most cases normals are important for the result look, but sometimes the source meshes just don't have correct normals, or they should be ignored. And the MeshCollider can definitely live without normals at all.

**Process Origin Tangents** - Enables the processing for tangents where the source mesh tangents considered as a source for result tangents. Tangents are heavy values, that usually are precalculated, mostly used for the normal mapping. A lot of shaders don't need tangents. Consider disabling this option for better performance.

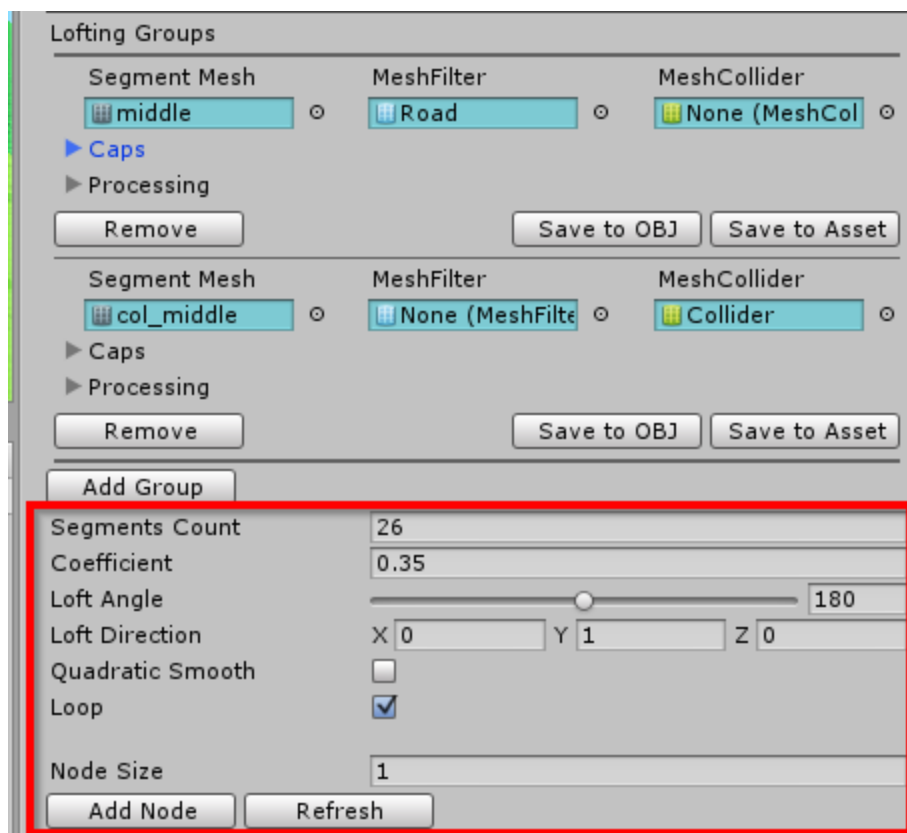
**Weld Close Vertices** - Welds close vertices for eliminating annoying artifacts on the seams that caused by inaccuracy of float calculation and poor source meshes. Makes MeshRender look neat and MeshCollider work stable.

**Welding Distance** - Minimum distance between result vertices. All vertices that closer than welding distance snap together.

**Recalculate Normals** - Uses Unity function to recalculate result normals based on the real mesh geometry. Useful when the source meshes don't have own normals, or they are incorrect.

**Smooth Normals** - Enables simple surface smoothing. Usually the result of Recalculate Normals is pretty harsh. With Smooth Normals on, the mesh generator recalculates normals for the close vertices to make the surface look smoother, but preserves sharp edges.

# SPLINE POWER! ★



**Segments Count** - Current count of the result mesh segment including the start and the end pieces. The segments are uniformly distributed along the entire distance of the lofting path.

**Coefficient** - The curvature coefficient. Choose that value purely by eye. Low values make spline look more straight and sharp. High values may do weird things.

**Loft Angle** - The pitch angle of the result mesh along the lofting path. Makes the source mesh that looks like floor to look like wall and such things.

**Loft Direction** - The direction along which the lofting will be performed. Basically it's a direction of the hole in the bead. Note that the lofting path will go through the source mesh pivot. Therefore, if you wish your caps fit perfectly, the floor will be floor, and won't any wrong lofting artifacts, consider the positions of the meshes pivots.

**Quadratic Smooth** - Adds quadratic curvature smoothing. Sometimes it works better with the sharp angles. Use to your taste.

**Loop** - Makes the result mesh looping seamlessly. Perfect for road loops.

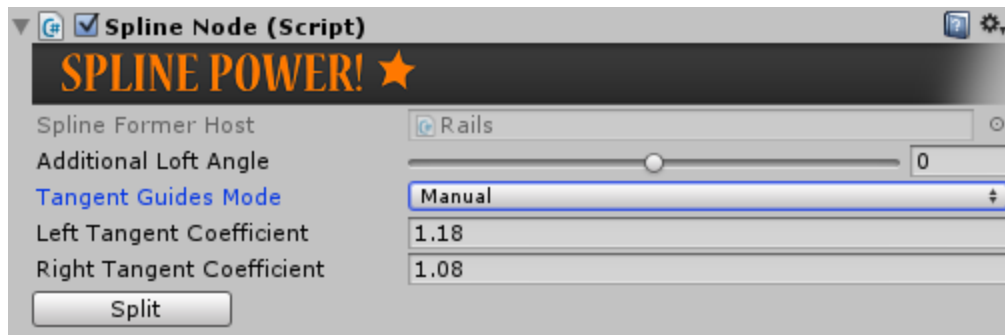
**Node Size** - Sets the size of the nodes gizmos.

**Add node** - Adds another node at the end of the spline.



**Refresh** - Forces the mesh generator to rebuild all the lofting groups immediately.

### Spline Node:



**Additional Loft Angle** - Additional pitch angle for current node. This angle is interpolated between adjacent nodes.

### Tangent Guides Mode:

**Auto** - position of tangent nodes sets automatically considering value of **Coefficient** set in host **Spline Former**

**Manual** - position of tangent nodes can be set manually.

**Split** button - Splits the current node on two.

# SPLINE POWER! ★

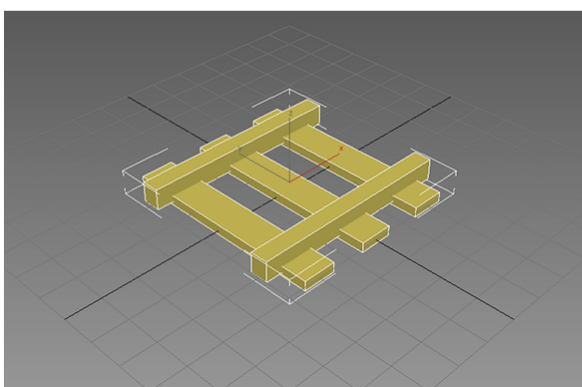
## Tips

- ☆ **Do not change parameters unnecessarily.**

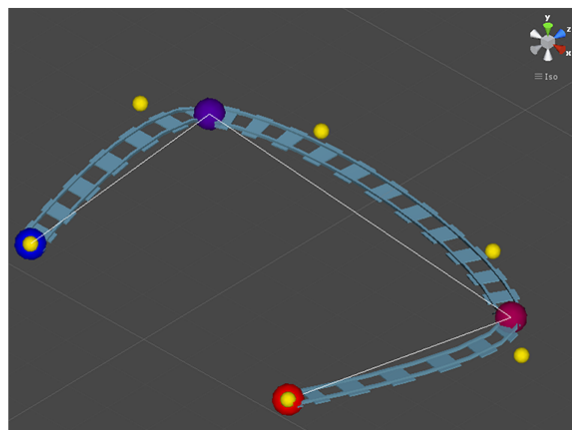
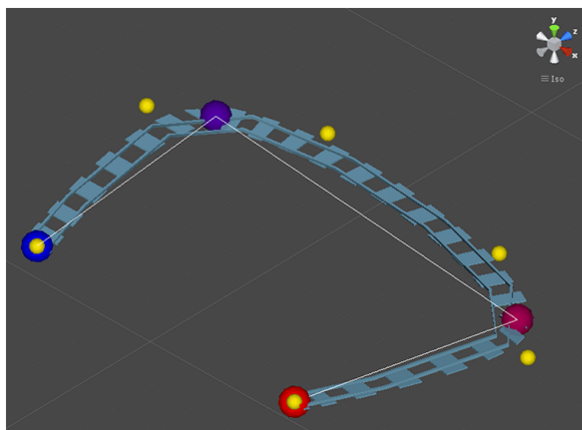
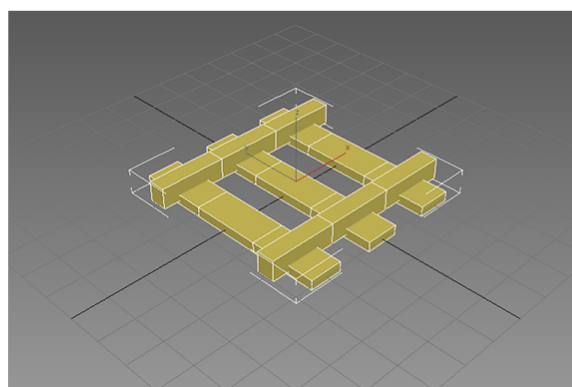
It's not a skinned mesh, every time you refresh parameters it leads to full mesh regeneration. Mesh regeneration is not terribly slow, but it's costly. I would not recommend to regenerate it in every frame.

- ☆ **Every vertex of the source mesh is bending independently**, therefore if you want your result look more smooth, it doesn't mean that you need to increase the segments count. You can just add more polygons to your source meshes.

LOW



HIGH





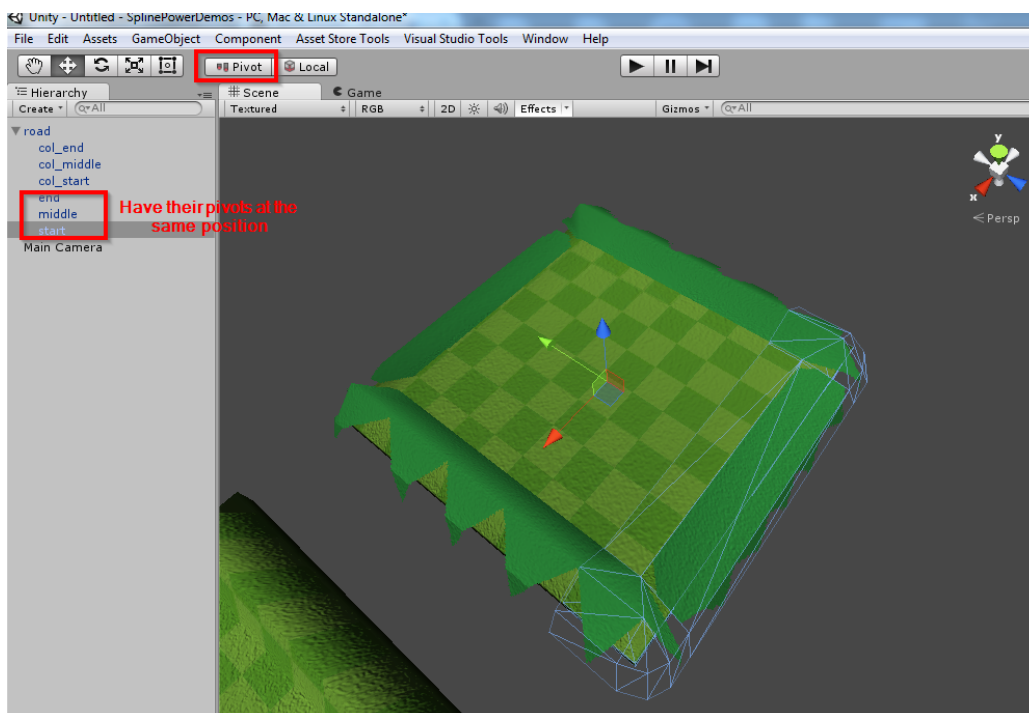
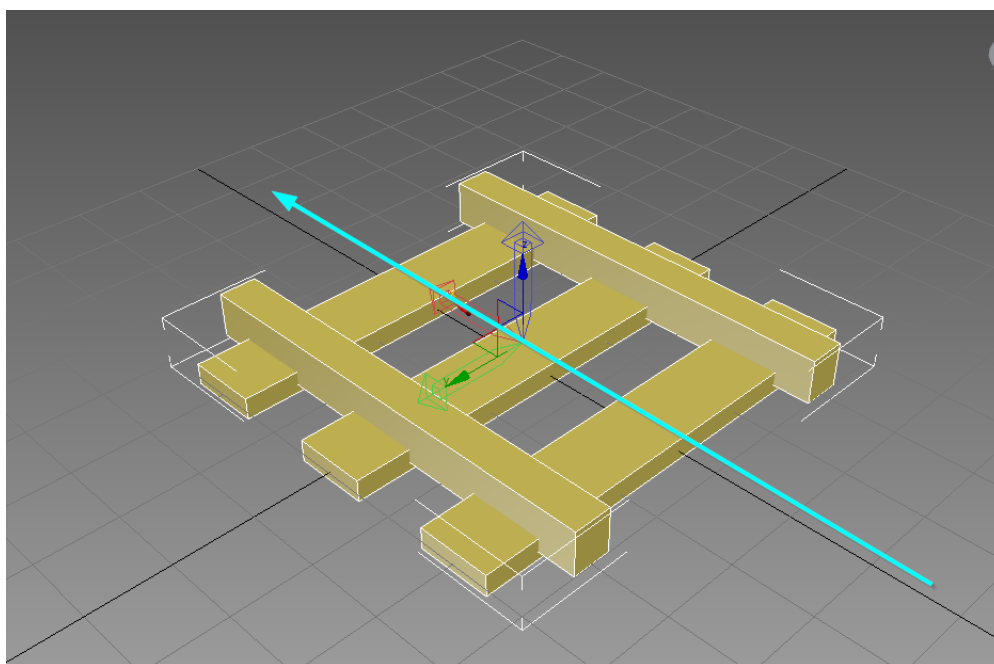
# SPLINE POWER! ★

☆ **Note that the lofting path will go through the source mesh pivot.**

Therefore, if you wish your caps fit perfectly, the floor will be floor, and won't any wrong lofting artifacts, consider the positions of the meshes pivots.

Considering path as a thread and source mesh as a bead, the pivot is a center of the hole in in the bead.

For example, the collision mesh in **road.FBX** consists of three elements, their pivots are at the same position, so they snap together fine.



# SPLINE POWER! ★

☆ Lower curvature coefficient leads to sharper edges.

