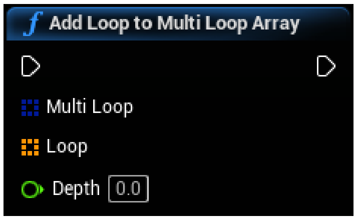
Documentation > Mesh Ops Plugin BPLibrary > *Add Loop to Multi Loop Array*

Add Loop to Multi Loop Array

Adds a Loop of points to a MultiLoop Array.



Inputs

In `Exec`	
Multi Loop `Array of Loop Structures`	
Loop `Array of Vectors`	
Depth `Float`	

Outputs

Out `Exec`	

Documentation > Mesh Ops Plugin BPLibrary > *Add to Bounds*

Add to Bounds

Adds a vertex to a Mesh Data's bounding box (Call after adding a new vertex, alternatively call CalcMeshDataBounds with Force = true after adding several)

Inputs

In `Exec`	
Data `Mesh Data Structure (by ref)`	
Vert `Vector`	

Outputs

Out `Exec`	

Documentation > Mesh Ops Plugin BPLibrary > *Apart Mesh Data*

Apart Mesh Data

Detects separate pieces in Mesh Data, and outputs them as an array.

Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Tol Float	Tolerance of vertex distance between shapes.

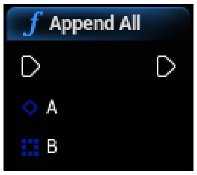
Outputs

Out Exec	
Out Arr Array of Mesh Data Structures	

Documentation > Mesh Ops Plugin BPLibrary > Append All

Append All

Appends all Mesh Datas within B to A.



Inputs

In Exec	
A Mesh Data Structure (by ref)	
B Array of Mesh Data Structures	

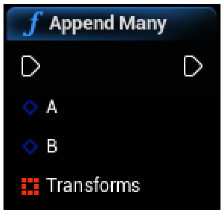
Outputs

Out Exec	

Documentation > Mesh Ops Plugin BPLibrary > Append Many

Append Many

Appends Mesh Data B to A with Transforms. Akin to how an instanced static mesh works.



Inputs

In Exec	
A Mesh Data Structure (by ref)	
B Mesh Data Structure (by ref)	

Transforms Array of Transforms	

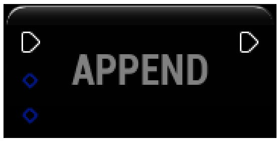
Outputs

Out Exec	

Documentation > Mesh Ops Plugin BPLibrary > *Append Mesh Data*

Append Mesh Data

Appends Mesh Data B onto A.



Inputs

In Exec	
A Mesh Data Structure (by ref)	
B Mesh Data Structure (by ref)	

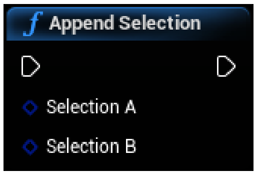
Outputs

Out Exec	

Documentation > Mesh Ops Plugin BPLibrary > *Append Selection*

Append Selection

Adds SelectionB into SelectionA, only adds unique values. Assumes selections are the same type, behavior is undefined (but should work) if not.



Inputs

In Exec	
Selection A Mesh Selection Structure (by ref)	
Selection B Mesh Selection Structure (by ref)	

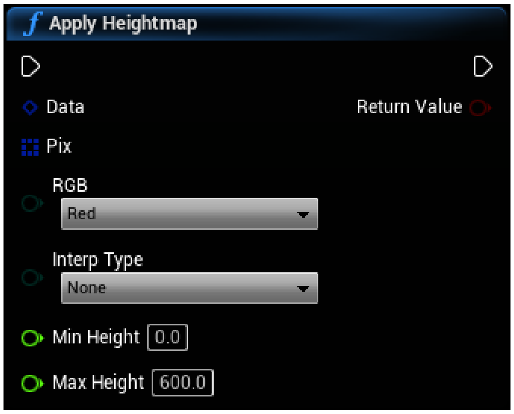
Outputs

Out Exec	

Documentation > Mesh Ops Plugin BPLibrary > *Apply Heightmap*

Apply Heightmap

Applies heights to a plane mesh based on a heightmap in the form of a linearcolor array Pix. Technically works on any mesh, a plane (or using Create Grid Mesh node) is recommended though. Assumes the Pix array is square (equal width and height).



Apply Heightmap

RGB: Red

Interp Type: None

Min Height 0.0

Max Height 600.0

Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Pix Array of Linear Color Structures	
RGB ColorChannelEnum Enum	
Interp Type InterpolationTypeEnum Enum	
Min Height Float	
Max Height Float	

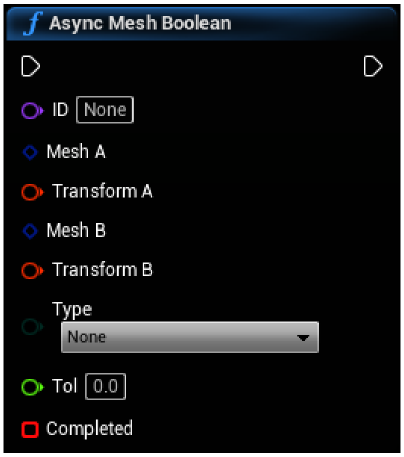
Outputs

Out Exec	
Return Value Boolean	Applies heights to a plane mesh based on a heightmap in the form of a linearcolor array Pix. Technically works on any mesh, a plane (or using Create Grid Mesh node) is recommended though. Assumes the Pix array is square (equal width and height).

Documentation > Mesh Ops Plugin BPLibrary > Async Mesh Boolean

Async Mesh Boolean

Performs Mesh Boolean Operation in a different thread. Uses "Geometry Processing" plugin included in the engine since 4.26.



Async Mesh Boolean

ID None

Mesh A

Transform A

Mesh B

Transform B

Type

None

Tol 0.0

Completed

Inputs

In `Exec`	
ID `Name`	
Mesh A `Mesh Data Structure (by ref)`	
Transform A `Transform`	
Mesh B `Mesh Data Structure (by ref)`	
Transform B `Transform`	
Type `BooleanType Enum`	
Tol `Float`	Simplification angle
Completed `Delegate`	

Outputs

Out `Exec`	

Documentation > Mesh Ops Plugin BPLibrary > *Auto UVs*

Auto UVs

Automatically projects uvs onto virtual planes based upon similar vertex normals using Tol tolerance.

Inputs

In `Exec`	
Data `Mesh Data Structure (by ref)`	
Tol `Float`	

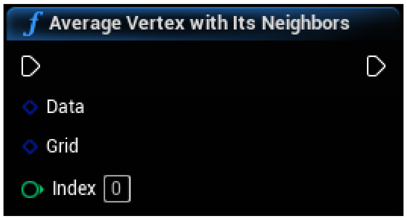
Outputs

Out `Exec`	

Documentation > Mesh Ops Plugin BPLibrary > *Average Vertex with Its Neighbors*

Average Vertex with Its Neighbors

Moves a single vert to the average point of itself and its neighbors, requires a Localized Grid from LocalizeMeshData() which doesn't need ComputeAvgs, that's something else.



Inputs

In `Exec`	
Data `Mesh Data Structure (by ref)`	
Grid `Localized Grid Structure (by ref)`	
Index `Integer`	

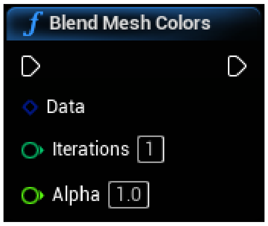
Outputs

Out `Exec`	

Documentation > Mesh Ops Plugin BPLibrary > *Blend Mesh Colors*

Blend Mesh Colors

Blends vertex colors with surrounding vertices, based on Alpha.



Inputs

In `Exec`	
Data `Mesh Data Structure (by ref)`	
Iterations `Integer`	
Alpha `Float`	

Outputs

Out `Exec`	

Documentation > Mesh Ops Plugin BPLibrary > *Bridge Loops*

Bridge Loops

Bridges between two loops (loops must be the same size and winding direction.)

Inputs

In Exec	
Loop A Array of Vectors	
Loop B Array of Vectors	
Border UVSize Float	
Segments Integer	
Closed Boolean	
Invert Result Boolean	

Outputs

Out Exec	
Return Value Mesh Data Structure	Bridges between two loops (loops must be the same size and winding direction.)

Documentation > Mesh Ops Plugin BPLibrary > *Bridge Loop to Center*

Bridge Loop to Center

Bridges all points on the Loop to the center point.

Inputs

In Exec	
Loop Array of Vectors	
Override Center Vector	
UVWidth Float	
Closed Boolean	

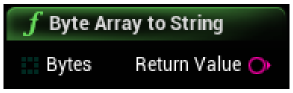
Outputs

Out Exec	
Return Value Mesh Data Structure	Bridges all points on the Loop to the center point.

Documentation > Mesh Ops Plugin BPLibrary > *Byte Array to String*

Byte Array to String

Bytes to String



Inputs

Bytes Array of Bytes	

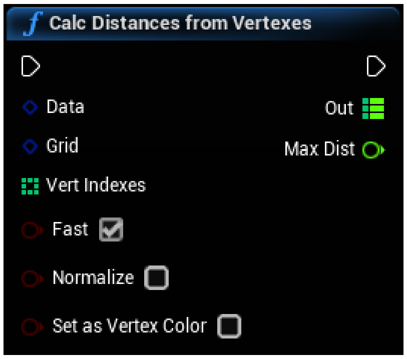
Outputs

Return Value String	Bytes to String

Documentation > Mesh Ops Plugin BPLibrary > *Calc Distances from Vertexes*

Calc Distances from Vertexes

Maps distances along the mesh surface, optionally as vertex colors (R channel) Used with PathToVertex() for pathfinding.



Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Grid Localized Grid Structure (by ref)	
Vert Indexes Array of Integers	
Fast Boolean	
Normalize Boolean	
Set as Vertex Color Boolean	

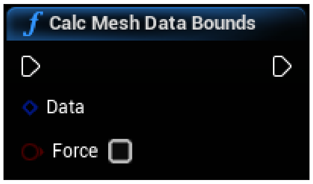
Outputs

Out Exec	
Out Map of Integers to Floats	
Max Dist Float	

Documentation > Mesh Ops Plugin BPLibrary > *Calc Mesh Data Bounds*

Calc Mesh Data Bounds

Calculates bounding box for Mesh Data, only recalculates if it has never been, or if Force is true.



Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Force Boolean	

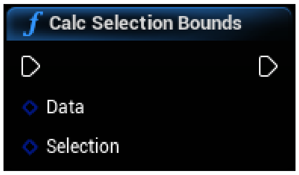
Outputs

Out Exec	

Documentation > Mesh Ops Plugin BPLibrary > *Calc Selection Bounds*

Calc Selection Bounds

Calculates Selection's bounds, used internally.



Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Selection Mesh Selection Structure (by ref)	

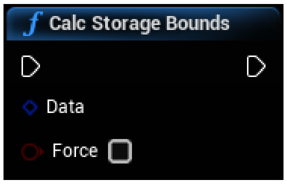
Outputs

Out Exec	

Documentation > Mesh Ops Plugin BPLibrary > *Calc Storage Bounds*

Calc Storage Bounds

Calculates bounding box for all Mesh Data in storage Array, only recalculates if it has never been, or if Force is true.



Inputs

In Exec	
Data Mesh Data Storage Structure (by ref)	
Force Boolean	

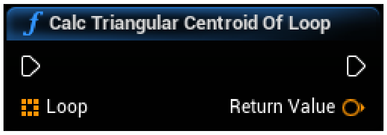
Outputs

Out Exec	

Documentation > *Mesh Ops Plugin BPLibrary* > *Calc Triangular Centroid Of Loop*

Calc Triangular Centroid Of Loop

Calculates a central point usually within the Loop. First triangulates the Loop and then averages all triangle centers.



Inputs

In Exec	
Loop Array of Vectors	

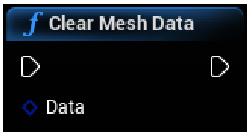
Outputs

Out Exec	
Return Value Vector	Calculates a central point usually within the Loop. First triangulates the Loop and then averages all triangle centers.

Documentation > *Mesh Ops Plugin BPLibrary* > *Clear Mesh Data*

Clear Mesh Data

Resets Mesh Data to empty values.



Inputs

In Exec	
Data	

Mesh Data Structure (by ref)	

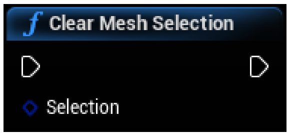
Outputs

Out Exec	

Documentation > Mesh Ops Plugin BPLibrary > *Clear Mesh Selection*

Clear Mesh Selection

Clears a Selection.



Inputs

In Exec	
Selection Mesh Selection Structure (by ref)	

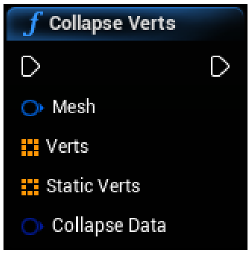
Outputs

Out Exec	

Documentation > Mesh Ops Plugin BPLibrary > *Collapse Verts*

Collapse Verts

EXPERIMENTAL



Inputs

In Exec	
Mesh Skeletal Mesh Component Object Reference	
Verts Array of Vectors	
Static Verts Array of Vectors	
Collapse Data Collapse Verts Structure	

Outputs

Out Exec	

[Documentation](#) > [Mesh Ops Plugin BPLibrary](#) > *Collision Deformation*

Collision Deformation

(WiP) Deforms Vertexes of Data where DataB's mesh collides with it. Ideally both mesh datas are stationary and TransformB holds any relative transformations between the meshes. Velocity indicates the direction mesh B was travelling in before collision.

Inputs

In `Exec`	
Data `Mesh Data Structure (by ref)`	
Data B `Mesh Data Structure (by ref)`	
Transform B `Transform`	
Velocity `Vector`	

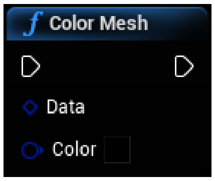
Outputs

Out `Exec`	

[Documentation](#) > [Mesh Ops Plugin BPLibrary](#) > *Color Mesh*

Color Mesh

Sets the vertex color for the entire mesh at once.



Inputs

In `Exec`	
Data `Mesh Data Structure (by ref)`	
Color `Linear Color Structure`	

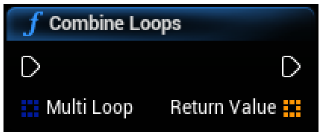
Outputs

Out `Exec`	

[Documentation](#) > [Mesh Ops Plugin BPLibrary](#) > *Combine Loops*

Combine Loops

Combines loops together, if the direction of the next loop is counter-clockwise it is cut from loop 0, or added to it if it's clockwise.



Inputs

In Exec	
Multi Loop Array of Loop Structures	

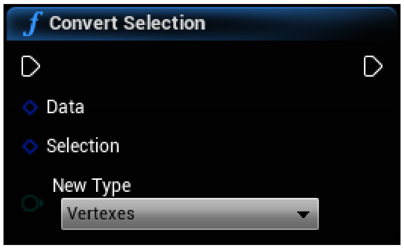
Outputs

Out Exec	
Return Value Array of Vectors	Combines loops together, if the direction of the next loop is counter-clockwise it is cut from loop 0, or added to it if it's clockwise.

Documentation > Mesh Ops Plugin BPLibrary > Convert Selection

Convert Selection

Converts a Selection struct into a different type. Conversions will still retain all data, but will generate anything that is missing for the NewType selected.



New Type: Vertexes

Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Selection Mesh Selection Structure (by ref)	
New Type SelectionType Enum	

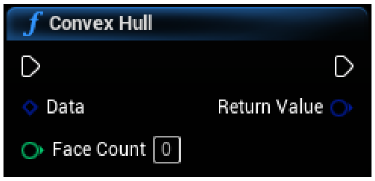
Outputs

Out Exec	

Documentation > Mesh Ops Plugin BPLibrary > Convex Hull

Convex Hull

Forms convex hull for simple collisions and simulating procedural meshes.



Face Count: 0

Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Face Count Integer	target face count, may not be respected in all cases.

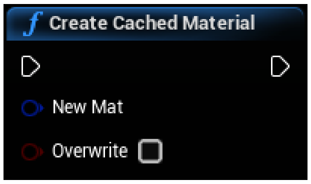
Outputs

Out Exec	
Return Value Mesh Data Structure	

Documentation > Mesh Ops Plugin BPLibrary > *Create Cached Material*

Create Cached Material

EXPERIMENTAL Creates and caches a Cached Material for use in functions that need repeated access to material and texture information.



Inputs

In Exec	
New Mat Cached Material Structure	
Overwrite Boolean	

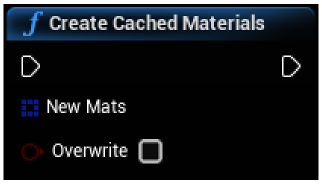
Outputs

Out Exec	

Documentation > Mesh Ops Plugin BPLibrary > *Create Cached Materials*

Create Cached Materials

EXPERIMENTAL Creates and caches Cached Materials for use in functions that need repeated access to material and texture information.



Inputs

In Exec	
New Mats	

Array of Cached Material Structures	
Overwrite Boolean	

Outputs

Out Exec	

Documentation > Mesh Ops Plugin BPLibrary > Create Empty Texture

Create Empty Texture

Creates and fills a new Texture2D.

Inputs

In Exec	
Fill Color Linear Color Structure	
Width Integer	
Height Integer	

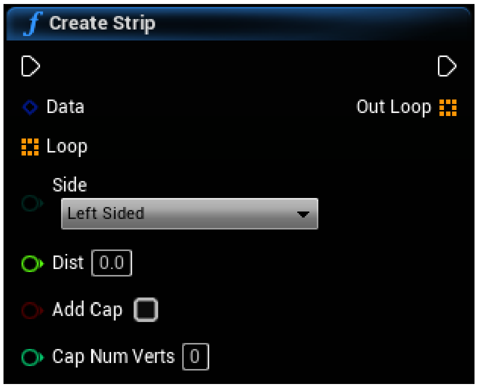
Outputs

Out Exec	
Return Value Texture 2D Object Reference	Creates and fills a new Texture2D.

Documentation > Mesh Ops Plugin BPLibrary > Create Strip

Create Strip

Creates a strip of triangles along an open or closed Loop.



Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Loop Array of Vectors	

Side StripSide Enum	
Dist Float	
Add Cap Boolean	
Cap Num Verts Integer	

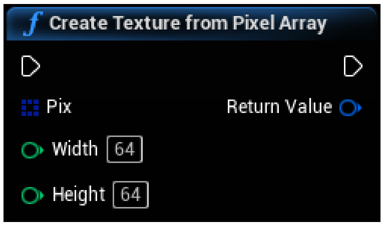
Outputs

Out Exec	
Out Loop Array of Vectors	

Documentation > Mesh Ops Plugin BPLibrary > Create Texture from Pixel Array

Create Texture from Pixel Array

Uses memcpy to fill a new texture2D.



Inputs

In Exec	
Pix Array of Linear Color Structures	
Width Integer	
Height Integer	

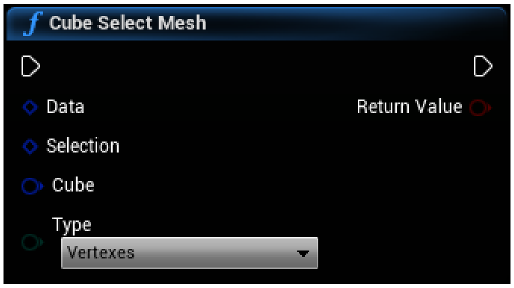
Outputs

Out Exec	
Return Value Texture 2D Object Reference	Uses memcpy to fill a new texture2D.

Documentation > Mesh Ops Plugin BPLibrary > Cube Select Mesh

Cube Select Mesh

Selects vertexes of a mesh in a cube for use in other functions, like TransformSelection().



Inputs

In `Exec`	
Data `Mesh Data Structure (by ref)`	
Selection `Mesh Selection Structure (by ref)`	
Cube `Cube Brush Structure`	
Type `SelectionType Enum`	

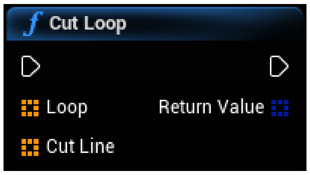
Outputs

Out `Exec`	
Return Value `Boolean`	Selects vertexes of a mesh in a cube for use in other functions, like TransformSelection().

Documentation > Mesh Ops Plugin BPLibrary > Cut Loop

Cut Loop

Cuts a Loop into two pieces along Cutline, must have one entry and one exit intersections.



Inputs

In `Exec`	
Loop `Array of Vectors`	
Cut Line `Array of Vectors`	

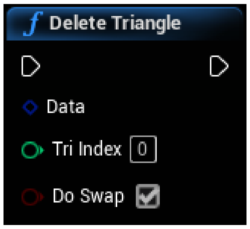
Outputs

Out `Exec`	
Return Value `Array of Loop Structures`	Cuts a Loop into two pieces along Cutline, must have one entry and one exit intersections.

Documentation > Mesh Ops Plugin BPLibrary > Delete Triangle

Delete Triangle

Deletes a triangle from a mesh starting at TriIndex, leaves any verts that were attached untouched. DoSwap should be much, much faster for large meshes, but will reorder the final triangle to fill the gap in the array. All other indexes will be the same still. Without swapping, the entire array after the deleted triangle has to be shifted down 3 times.



Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Tri Index Integer	
Do Swap Boolean	

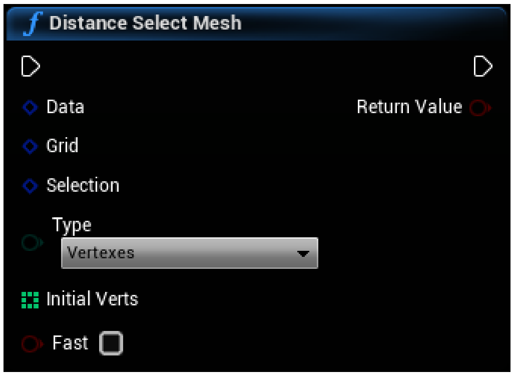
Outputs

Out Exec	

Documentation > Mesh Ops Plugin BPLibrary > Distance Select Mesh

Distance Select Mesh

Selects verts with Alpha as the normalized distance away from InitialVerts.



Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Grid Localized Grid Structure (by ref)	
Selection Mesh Selection Structure (by ref)	
Type SelectionType Enum	
Initial Verts Array of Integers	

Fast Boolean	

Outputs

Out Exec	
Return Value Boolean	Selects verts with Alpha as the normalized distance away from InitialVerts.

[Documentation](#) > [Mesh Ops Plugin BPLibrary](#) > Distance to Mesh Surface

Distance to Mesh Surface

Gets the distance to the approximate closest point on the mesh. Returns early if distance is within AcceptWithin.

Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Point Vector	
Accept Within Float	

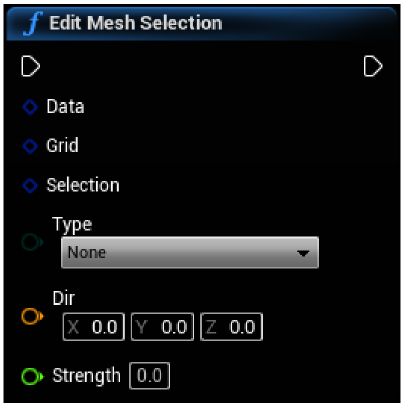
Outputs

Out Exec	
Out Point Vector	
Return Value Float	Gets the distance to the approximate closest point on the mesh. Returns early if distance is within AcceptWithin.

[Documentation](#) > [Mesh Ops Plugin BPLibrary](#) > Edit Mesh Selection

Edit Mesh Selection

Edits a part of the mesh within the Selection. Smooth requires a Localized Grid from LocalizeMeshData(). Dir only has relevance for certain edit types, and Strength can be negative for certain types.



Type: None

Dir: X 0.0 Y 0.0 Z 0.0

Strength: 0.0

Inputs

In Exec	

Data Mesh Data Structure (by ref)	
Grid Localized Grid Structure (by ref)	
Selection Mesh Selection Structure (by ref)	
Type BrushEditEnum Enum	
Dir Vector	
Strength Float	

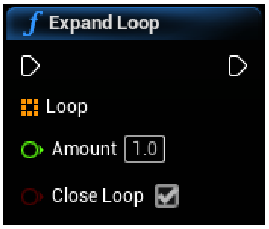
Outputs

Out Exec	

Documentation > Mesh Ops Plugin BPLibrary > *Expand Loop*

Expand Loop

Expands a Loop along its normals. Its 'normals' are calculated as the direction perpendicular to the segment from x to x+1.



Inputs

In Exec	
Loop Array of Vectors	
Amount Float	
Close Loop Boolean	

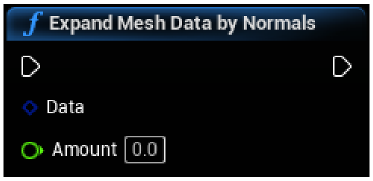
Outputs

Out Exec	

Documentation > Mesh Ops Plugin BPLibrary > *Expand Mesh Data by Normals*

Expand Mesh Data by Normals

Inflates Mesh Data along normals.



Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Amount Float	

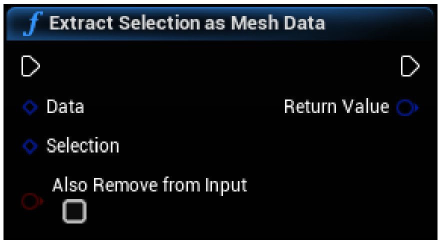
Outputs

Out Exec	

Documentation > *Mesh Ops Plugin BPLibrary* > *Extract Selection as Mesh Data*

Extract Selection as Mesh Data

Extracts Selection into a new Mesh Data and optionally removes it from Data.



Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Selection Mesh Selection Structure (by ref)	
Also Remove from Input Boolean	

Outputs

Out Exec	
Return Value Mesh Data Structure	Extracts Selection into a new Mesh Data and optionally removes it from Data.

Documentation > *Mesh Ops Plugin BPLibrary* > *Extrude Mesh Data*

Extrude Mesh Data

Extrudes a flatish mesh in Dir direction. Requires a Border array, which is an unclosed loop.

Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Border	

Array of Vectors	
Dir Vector	
Dist Float	
Separate Border Boolean	
Border UVSize Float	
Flip Backside UVs Boolean	

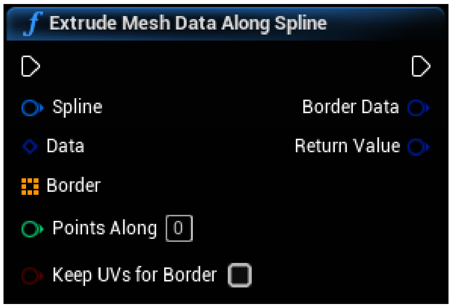
Outputs

Out Exec	
Border Data Mesh Data Structure	
Return Value Mesh Data Structure	Extrudes a flatish mesh in Dir direction. Requires a Border array, which is an unclosed loop.

Documentation > Mesh Ops Plugin BPLibrary > Extrude Mesh Data Along Spline

Extrude Mesh Data Along Spline

EXPERIMENTAL



Inputs

In Exec	
Spline Spline Component Object Reference	
Data Mesh Data Structure (by ref)	
Border Array of Vectors	
Points Along Integer	
Keep UVs for Border Boolean	

Outputs

Out Exec	
Border Data Mesh Data Structure	
Return Value Mesh Data Structure	EXPERIMENTAL

Documentation > Mesh Ops Plugin BPLibrary > *Extrude Mesh Data Inflated*

Extrude Mesh Data Inflated

Extrudes a flatish mesh in Dir direction. Requires a Border array, which is an unclosed loop. Inflates border verts.

Inputs

In `Exec`	
Data `Mesh Data Structure (by ref)`	
Border `Array of Vectors`	
Dir `Vector`	
Dist `Float`	
Points Along `Integer`	
Inflation `Float`	
Multiply `Float`	
Pow `Float`	
Keep UVs for Border `Boolean`	

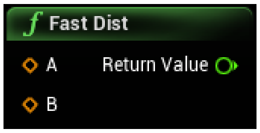
Outputs

Out `Exec`	
Border Data `Mesh Data Structure`	
Return Value `Mesh Data Structure`	Extrudes a flatish mesh in Dir direction. Requires a Border array, which is an unclosed loop. Inflates border verts.

Documentation > Mesh Ops Plugin BPLibrary > *Fast Dist*

Fast Dist

Distance squared from vector A to vector B. Use (distance*distance) for comparisons - i.e. (FastDist(A,B)<=(dist*dist)), or compare two FastDist results directly.



Inputs

A `Vector (by ref)`	
B `Vector (by ref)`	

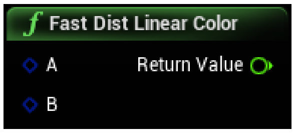
Outputs

Return Value Float	Distance squared from vector A to vector B. Use (distance*distance) for comparisons - i.e. (FastDist(A,B)<=(dist*dist)), or compare two FastDist results directly.

Documentation > *Mesh Ops Plugin BPLibrary* > *Fast Dist Linear Color*

Fast Dist Linear Color

Distance squared from LinearColor A to LinearColor B. Use (distance*distance) for comparisons - i.e. (FastDistLinearColor(A,B)<=(dist*dist)), or compare two FastDistLinearColor results directly.



Inputs

A Linear Color Structure (by ref)	
B Linear Color Structure (by ref)	

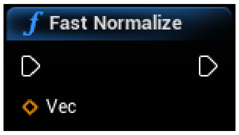
Outputs

Return Value Float	Distance squared from LinearColor A to LinearColor B. Use (distance*distance) for comparisons - i.e. (FastDistLinearColor(A,B)<=(dist*dist)), or compare two FastDistLinearColor results directly.

Documentation > *Mesh Ops Plugin BPLibrary* > *Fast Normalize*

Fast Normalize

Normalizes a vector using FMath::InvSqrtEst() from unreal code. Used internally.



Inputs

In Exec	
Vec Vector (by ref)	

Outputs

Out Exec	

Documentation > *Mesh Ops Plugin BPLibrary* > *Fill Hole Loop*

Fill Hole Loop

Fills loop with polygons. Uses "Geometry Processing" plugin included in the engine since 4.26.

Inputs

In Exec	
Data Mesh Data Structure (by ref)	Only uses an unclosed loop in Verts array.

UVWidth Float	
Fan Limit Integer	
In Place Boolean	

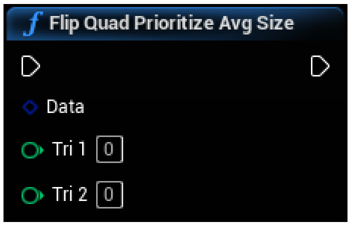
Outputs

Out Exec	
Return Value Mesh Data Structure	

[Documentation](#) > [Mesh Ops Plugin BPLibrary](#) > *Flip Quad Prioritize Avg Size*

Flip Quad Prioritize Avg Size

Flips two Triangles starting at Tri1 and Tri2 by their shared edge if the result makes the new triangles closer to the same size. A way to visualize this is form a rectangle with the two triangles, take their shared edge, and turn it to become two different triangles within the same quad.



Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Tri 1 Integer	
Tri 2 Integer	

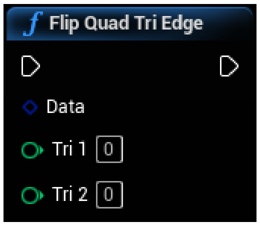
Outputs

Out Exec	

[Documentation](#) > [Mesh Ops Plugin BPLibrary](#) > *Flip Quad Tri Edge*

Flip Quad Tri Edge

Flips two Triangles starting at Tri1 and Tri2 by their shared edge. A way to visualize this is form a rectangle with the two triangles, take their shared edge, and turn it to become two different triangles within the same quad.



Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Tri 1 Integer	
Tri 2 Integer	

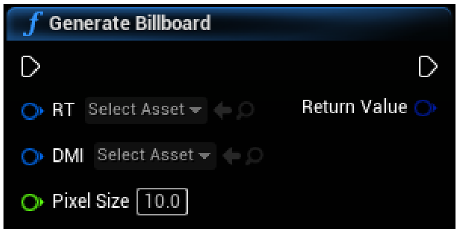
Outputs

Out Exec	

Documentation > Mesh Ops Plugin BPLibrary > Generate Billboard

Generate Billboard

Generates a plane with vertex colors of the image.



Inputs

In Exec	
RT Texture Render Target 2D Object Reference	
DMI Material Instance Dynamic Object Reference	
Pixel Size Float	

Outputs

Out Exec	
Return Value Mesh Data Structure	Generates a plane with vertex colors of the image.

Documentation > Mesh Ops Plugin BPLibrary > Generate Circle Loop

Generate Circle Loop

Generates a Loop of vertexes in a circle.

Inputs

In Exec	
NPoints Integer	
Radius	

Float	

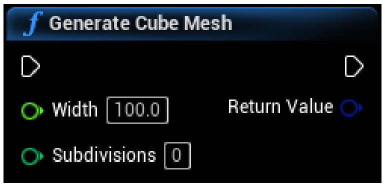
Outputs

Out Exec	
Return Value Array of Vectors	Generates a Loop of vertexes in a circle.

Documentation > Mesh Ops Plugin BPLibrary > Generate Cube Mesh

Generate Cube Mesh

Generates a cube Width wide, and subdivided Subdivisions number of times.



Inputs

In Exec	
Width Float	
Subdivisions Integer	

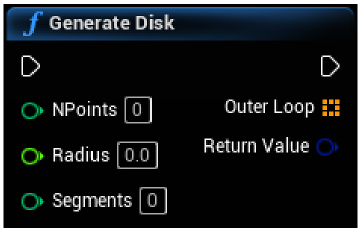
Outputs

Out Exec	
Return Value Mesh Data Structure	Generates a cube Width wide, and subdivided Subdivisions number of times.

Documentation > Mesh Ops Plugin BPLibrary > Generate Disk

Generate Disk

Creates a disk with NPoints number of points and Radius radius. If Segment is 1, you can make a loop by copying the verts and removing index 0.



Inputs

In Exec	
NPoints Integer	
Radius Float	
Segments Integer	

Outputs

Out `Exec`	
Outer Loop `Array of Vectors`	
Return Value `Mesh Data Structure`	Creates a disk with NPoints number of points and Radius radius. If Segment is 1, you can make a loop by copying the verts and removing index 0.

Documentation > Mesh Ops Plugin BPLibrary > Generate Grid Mesh

Generate Grid Mesh

Generates a plane segmented into quads in a grid. Can be optionally welded.

Inputs

In `Exec`	
XNum `Integer`	
YNum `Integer`	
Grid Size `Float`	
Welded `Boolean`	

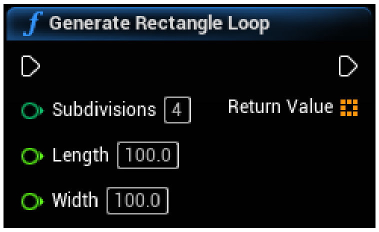
Outputs

Out `Exec`	
Return Value `Mesh Data Structure`	Generates a plane segmented into quads in a grid. Can be optionally welded.

Documentation > Mesh Ops Plugin BPLibrary > Generate Rectangle Loop

Generate Rectangle Loop

Generates a Loop of vertexes in a circle.



Inputs

In `Exec`	
Subdivisions `Integer`	
Length `Float`	
Width `Float`	

Outputs

Out `Exec`	
Return Value `Array of Vectors`	Generates a Loop of vertexes in a circle.

Documentation > Mesh Ops Plugin BPLibrary > *Generate Skin Weights*

Generate Skin Weights

EXPERIMENTAL Generates Skin Weights for a mesh before skinning it, uses Skeleton's StaticBones, influences are determined by the skeleton's Radii and bone positions in world units. Factor contributes to the falloff. Falls back to closest bone when a vert is outside of all bone radii.

Inputs

In `Exec`	
Data `Mesh Data Structure (by ref)`	
Skeleton `Mesh Skeleton Structure (by ref)`	
Factor `Float`	

Outputs

Out `Exec`	
Skin Weights `Array of Skin Weight Structures`	
Success `Boolean`	

Documentation > Mesh Ops Plugin BPLibrary > *Generate Sphere Mesh*

Generate Sphere Mesh

Generates a sphere with Segments number of verts in both directions (up/down & around).

Inputs

In `Exec`	
Segments `Integer`	
Radius `Float`	

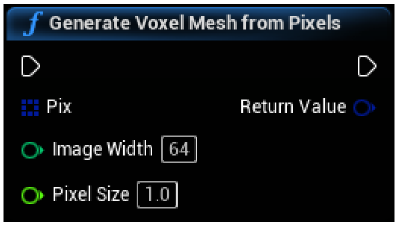
Outputs

Out `Exec`	
Return Value `Mesh Data Structure`	Generates a sphere with Segments number of verts in both directions (up/down & around).

Documentation > Mesh Ops Plugin BPLibrary > *Generate Voxel Mesh from Pixels*

Generate Voxel Mesh from Pixels

EXPERIMENTAL Generates a 2D blocky voxel mesh from pixels. Internal faces are removed. Vertex Colors store the color information.



Inputs

In `Exec`	
Pix `Array of Linear Color Structures`	
Image Width `Integer`	
Pixel Size `Float`	

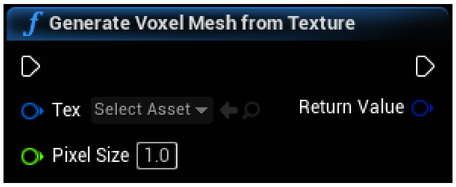
Outputs

Out `Exec`	
Return Value `Mesh Data Structure`	EXPERIMENTAL Generates a 2D blocky voxel mesh from pixels. Internal faces are removed. Vertex Colors store the color information.

Documentation > Mesh Ops Plugin BPLibrary > *Generate Voxel Mesh from Texture*

Generate Voxel Mesh from Texture

EXPERIMENTAL Generates a 2D blocky voxel mesh from a texture. Internal faces are removed. Vertex Colors store the color information.



Inputs

In `Exec`	
Tex `Texture 2D Object Reference`	
Pixel Size `Float`	

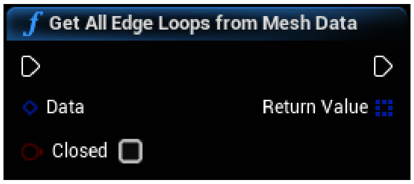
Outputs

Out `Exec`	
Return Value `Mesh Data Structure`	EXPERIMENTAL Generates a 2D blocky voxel mesh from a texture. Internal faces are removed. Vertex Colors store the color information.

Documentation > Mesh Ops Plugin BPLibrary > *Get All Edge Loops from Mesh Data*

Get All Edge Loops from Mesh Data

Gets all exposed edge loops from the mesh.



Inputs

In `Exec`	
Data `Mesh Data Structure (by ref)`	
Closed `Boolean`	

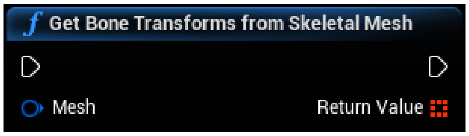
Outputs

Out `Exec`	
Return Value `Array of Loop Structures`	Gets all exposed edge loops from the mesh.

Documentation > Mesh Ops Plugin BPLibrary > Get Bone Transforms from Skeletal Mesh

Get Bone Transforms from Skeletal Mesh

EXPERIMENTAL Gets transforms of all bones in a Skeletal Mesh.



Inputs

In `Exec`	
Mesh `Skeletal Mesh Component Object Reference`	

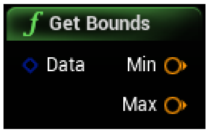
Outputs

Out `Exec`	
Return Value `Array of Transforms`	EXPERIMENTAL Gets transforms of all bones in a Skeletal Mesh.

Documentation > Mesh Ops Plugin BPLibrary > Get Bounds

Get Bounds

Gets Min Max Bounds of a Mesh Data



Inputs

Data Mesh Data Structure (by ref)	

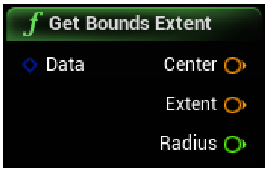
Outputs

Min Vector	
Max Vector	

Documentation > Mesh Ops Plugin BPLibrary > *Get Bounds Extent*

Get Bounds Extent

Gets Center, Extent, Radius of Mesh Data



Inputs

Data Mesh Data Structure (by ref)	

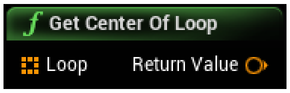
Outputs

Center Vector	
Extent Vector	
Radius Float	

Documentation > Mesh Ops Plugin BPLibrary > *Get Center Of Loop*

Get Center Of Loop

Gets the center of the Loop. It's a pure function for ease of use, still iterates all points.



Inputs

Loop Array of Vectors	

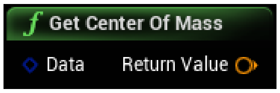
Outputs

Return Value Vector	Gets the center of the Loop. It's a pure function for ease of use, still iterates all points.

Documentation > Mesh Ops Plugin BPLibrary > *Get Center Of Mass*

Get Center Of Mass

Calculates center of mass for the mesh, assuming even mass distribution. Assumes the mesh is closed.



Inputs

Data `Mesh Data Structure (by ref)`	

Outputs

Return Value `Vector`	Calculates center of mass for the mesh, assuming even mass distribution. Assumes the mesh is closed.

Documentation > Mesh Ops Plugin BPLibrary > Get Closest Bone

Get Closest Bone

EXPERIMENTAL

Inputs

Mesh `Skinned Mesh Component Object Reference`	
Ignore `Array of Names`	
Loc `Vector`	
World Loc `Boolean`	

Outputs

Out `Vector`	
Return Value `Name`	EXPERIMENTAL

Documentation > Mesh Ops Plugin BPLibrary > Get Closest Point in Vector Array

Get Closest Point in Vector Array

Gets the index of the closest point in an array of vectors. Optionally stops early if within CloseEnough.

Inputs

In `Exec`	
Points `Array of Vectors`	
Check `Vector`	
Ignore Indexes `Array of Integers`	
Close Enough `Float`	

Outputs

Out Exec	
Return Value Integer	Gets the index of the closest point in an array of vectors. Optionally stops early if within CloseEnough.

Documentation > Mesh Ops Plugin BPLibrary > *Get Closest Point on Loop*

Get Closest Point on Loop

Gets the closest point on a Loop, also returns the index of the point that starts the segment that the new point is on.

Inputs

In Exec	
Loop Array of Vectors	
Check Vector	
Closed Boolean	

Outputs

Out Exec	
Out Index Integer	
Return Value Vector	Gets the closest point on a Loop, also returns the index of the point that starts the segment that the new point is on.

Documentation > Mesh Ops Plugin BPLibrary > *Get Closest Vert*

Get Closest Vert

Finds closest vertex to point Loc. Stops early if it found one within GoodEnough range.

Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Loc Vector	
Good Enough Float	

Outputs

Out Exec	
Return Value Integer	Finds closest vertex to point Loc. Stops early if it found one within GoodEnough range.

Documentation > Mesh Ops Plugin BPLibrary > *Get Cross Section Edge Loop*

Get Cross Section Edge Loop

EXPERIMENTAL Gets an edge loop where a mesh intersects a plane.

Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Plane Origin Vector	
Plane Up Vector	
Try Reorder Boolean	
Reverse if Not Clockwise Boolean	

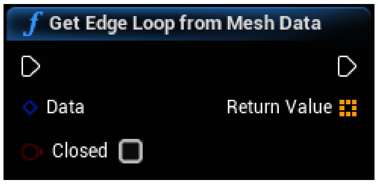
Outputs

Out Exec	
Return Value Array of Vectors	EXPERIMENTAL Gets an edge loop where a mesh intersects a plane.

Documentation > *Mesh Ops Plugin BPLibrary* > *Get Edge Loop from Mesh Data*

Get Edge Loop from Mesh Data

Generates an Edge Loop from Mesh Data, will likely fail if there aren't any exposed edges (only one triangle adjacent to it). Works best with flat meshes.



Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Closed Boolean	

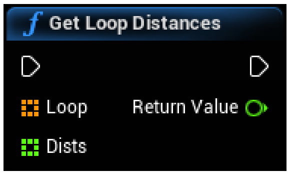
Outputs

Out Exec	
Return Value Array of Vectors	Generates an Edge Loop from Mesh Data, will likely fail if there aren't any exposed edges (only one triangle adjacent to it). Works best with flat meshes.

Documentation > *Mesh Ops Plugin BPLibrary* > *Get Loop Distances*

Get Loop Distances

Gets total length of a Loop, stores unit distance between each point.



Inputs

In `Exec`	
Loop `Array of Vectors`	
Dists `Array of Floats`	

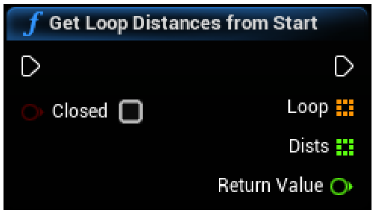
Outputs

Out `Exec`	
Return Value `Float`	Gets total length of a Loop, stores unit distance between each point.

Documentation > Mesh Ops Plugin BPLibrary > Get Loop Distances from Start

Get Loop Distances from Start

Gets total length of a Loop, stores unit distance from start for each point.



Inputs

In `Exec`	
Closed `Boolean`	

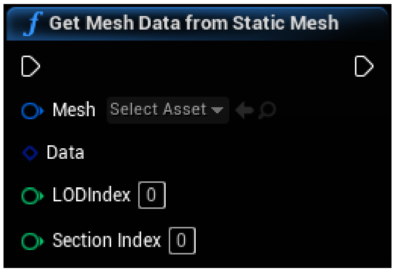
Outputs

Out `Exec`	
Loop `Array of Vectors`	
Dists `Array of Floats`	
Return Value `Float`	Gets total length of a Loop, stores unit distance from start for each point.

Documentation > Mesh Ops Plugin BPLibrary > Get Mesh Data from Static Mesh

Get Mesh Data from Static Mesh

Gets a Mesh Data from a static mesh. Had to adapt the built-in code to support vertex colors as well.



Inputs

In Exec	
Mesh Static Mesh Object Reference	
Data Mesh Data Structure (by ref)	
LODIndex Integer	
Section Index Integer	

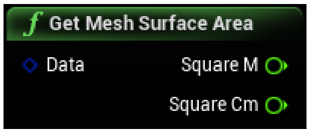
Outputs

Out Exec	

Documentation > Mesh Ops Plugin BPLibrary > Get Mesh Surface Area

Get Mesh Surface Area

Gets entire Surface area of a mesh. Loops through all triangles. Pure node for convenience.



Inputs

Data Mesh Data Structure (by ref)	

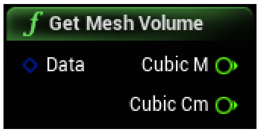
Outputs

Square M Float	
Square Cm Float	

Documentation > Mesh Ops Plugin BPLibrary > Get Mesh Volume

Get Mesh Volume

Calculates entire mesh volume, in both cubic M and Cm, assumes the mesh is closed. Loops through all triangles. Pure node for convenience.



Inputs

Data Mesh Data Structure (by ref)	

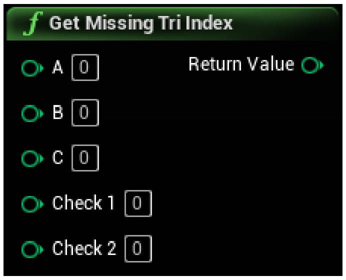
Outputs

Cubic M Float	
Cubic Cm Float	

[Documentation](#) > [Mesh Ops Plugin BPLibrary](#) > *Get Missing Tri Index*

Get Missing Tri Index

Gets the Triangle index (A, B, or C) that is not equal to Check1 or Check2, silently fails returning A if no result. Used in FlipQuadTriEdge.



Inputs

A Integer	
B Integer	
C Integer	
Check 1 Integer	
Check 2 Integer	

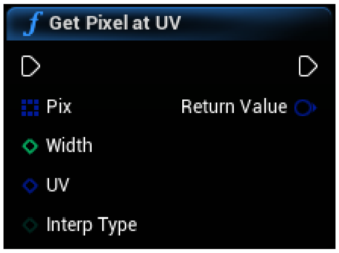
Outputs

Return Value Integer	Gets the Triangle index (A, B, or C) that is not equal to Check1 or Check2, silently fails returning A if no result. Used in FlipQuadTriEdge.

[Documentation](#) > [Mesh Ops Plugin BPLibrary](#) > *Get Pixel at UV*

Get Pixel at UV

Gets a pixel at the UV coordinate. Assumes Pix is square.



Inputs

In Exec	
Pix Array of Linear Color Structures	
Width Integer (by ref)	
UV Vector 2D Structure (by ref)	
Interp Type InterpolationTypeEnum Enum (by ref)	

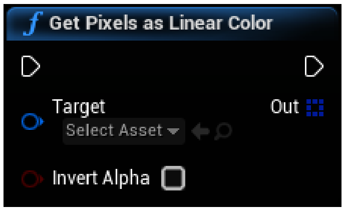
Outputs

Out Exec	
Return Value Linear Color Structure	Gets a pixel at the UV coordinate. Assumes Pix is square.

Documentation > Mesh Ops Plugin BPLibrary > Get Pixels as Linear Color

Get Pixels as Linear Color

Gets pixels Array from a Render Target. SLOW, use with care.



Inputs

In Exec	
Target Texture Render Target 2D Object Reference	
Invert Alpha Boolean	

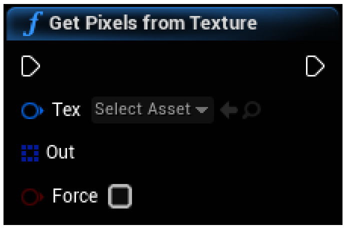
Outputs

Out Exec	
Out Array of Linear Color Structures	

Documentation > Mesh Ops Plugin BPLibrary > Get Pixels from Texture

Get Pixels from Texture

Gets pixels from a texture and caches the result, returns from cache if Force is false.



Inputs

In `Exec`	
Tex `Texture 2D Object Reference`	
Out `Array of Linear Color Structures`	
Force `Boolean`	

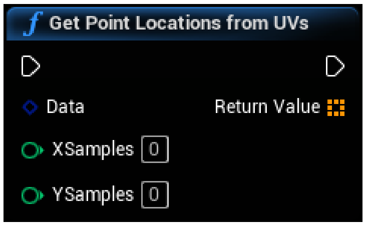
Outputs

Out `Exec`	

Documentation > Mesh Ops Plugin BPLibrary > Get Point Locations from UVs

Get Point Locations from UVs

Gets all points on the mesh surface that coorespond to UV points, sampling XSamples times on the x axis and YSamples on the y axis. UV overlap should not matter.



Inputs

In `Exec`	
Data `Mesh Data Structure (by ref)`	
XSamples `Integer`	
YSamples `Integer`	

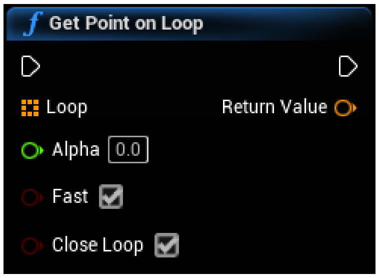
Outputs

Out `Exec`	
Return Value `Array of Vectors`	Gets all points on the mesh surface that coorespond to UV points, sampling XSamples times on the x axis and YSamples on the y axis. UV overlap should not matter.

Documentation > Mesh Ops Plugin BPLibrary > Get Point on Loop

Get Point on Loop

Gets a point on a loop, using Alpha (0.0-1.0), basically lerping along a set of points. Fast mode doesn't need to iterate at all. Slow mode is accurate for nonuniform distances, but iterates through every segment with sqrt distance calculated.



Inputs

In Exec	
Loop Array of Vectors	
Alpha Float	
Fast Boolean	
Close Loop Boolean	

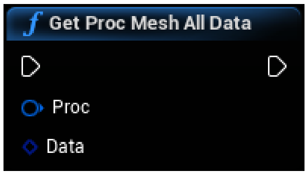
Outputs

Out Exec	
Return Value Vector	Gets a point on a loop, using Alpha (0.0-1.0), basically lerping along a set of points. Fast mode doesn't need to iterate at all. Slow mode is accurate for nonuniform distances, but iterates through every segment with sqrt distance calculated.

Documentation > Mesh Ops Plugin BPLibrary > Get Proc Mesh All Data

Get Proc Mesh All Data

Pulls raw mesh info from the procedural mesh as a Mesh Data Storage.



Inputs

In Exec	
Proc Procedural Mesh Component Object Reference	
Data Mesh Data Storage Structure (by ref)	

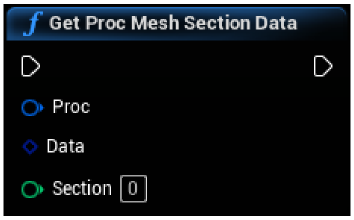
Outputs

Out Exec	

Documentation > Mesh Ops Plugin BPLibrary > Get Proc Mesh Section Data

Get Proc Mesh Section Data

Pulls raw mesh info from the procedural mesh as a Mesh Data.



Inputs

In Exec	
Proc Procedural Mesh Component Object Reference	
Data Mesh Data Structure (by ref)	
Section Integer	

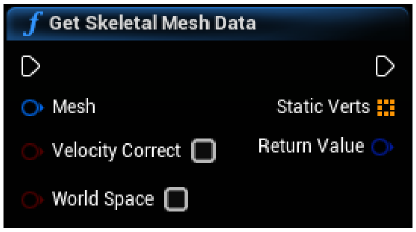
Outputs

Out Exec	

Documentation > *Mesh Ops Plugin BPLibrary* > *Get Skeletal Mesh Data*

Get Skeletal Mesh Data

EXPERIMENTAL



Inputs

In Exec	
Mesh Skeletal Mesh Component Object Reference	
Velocity Correct Boolean	
World Space Boolean	

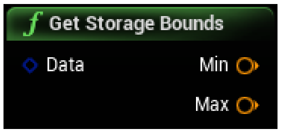
Outputs

Out Exec	
Static Verts Array of Vectors	
Return Value Mesh Data Structure	EXPERIMENTAL

Documentation > *Mesh Ops Plugin BPLibrary* > *Get Storage Bounds*

Get Storage Bounds

Gets Min Max Bounds of a Mesh Data Storage



Inputs

Data Mesh Data Storage Structure (by ref)	

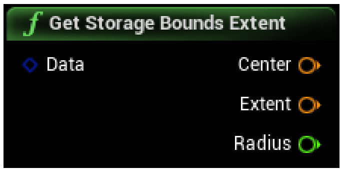
Outputs

Min Vector	
Max Vector	

Documentation > Mesh Ops Plugin BPLibrary > Get Storage Bounds Extent

Get Storage Bounds Extent

Gets Center, Extent, Radius of Mesh Data Storage



Inputs

Data Mesh Data Storage Structure (by ref)	

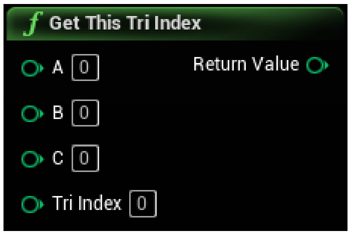
Outputs

Center Vector	
Extent Vector	
Radius Float	

Documentation > Mesh Ops Plugin BPLibrary > Get This Tri Index

Get This Tri Index

Gets the 0,1,2 index corresponding to A,B,C respectively that matches TriIndex. Returns -1 if no result. Used in FlipQuadTriEdge.



Inputs

A `Integer`	
B `Integer`	
C `Integer`	
Tri Index `Integer`	

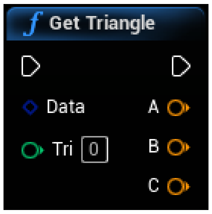
Outputs

Return Value `Integer`	Gets the 0,1,2 index corresponding to A,B,C respectively that matches TriIndex. Returns -1 if no result. Used in FlipQuadTriEdge.

Documentation > *Mesh Ops Plugin BPLibrary* > *Get Triangle*

Get Triangle

Get points of a triangle from a Mesh Data.



Inputs

In `Exec`	
Data `Mesh Data Structure (by ref)`	
Tri `Integer`	

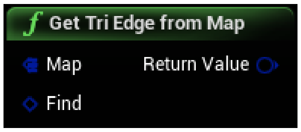
Outputs

Out `Exec`	
A `Vector`	
B `Vector`	
C `Vector`	

Documentation > *Mesh Ops Plugin BPLibrary* > *Get Tri Edge from Map*

Get Tri Edge from Map

Gets a result from the Edge Map, regardless of the A vs B order of the key edge. Flips the edge if needed.



Inputs

Map Map of Tri Edge Structures to Tri Edge Structures	
Find Tri Edge Structure (by ref)	

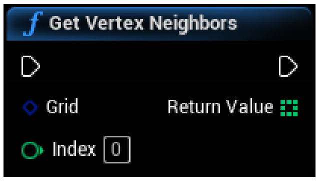
Outputs

Return Value Tri Edge Structure	Gets a result from the Edge Map, regardless of the A vs B order of the key edge. Flips the edge if needed.

Documentation > *Mesh Ops Plugin BPLibrary* > *Get Vertex Neighbors*

Get Vertex Neighbors

Gets neighboring vertex indexes, requires a Localized Grid from LocalizeMeshData()



Inputs

In Exec	
Grid Localized Grid Structure (by ref)	
Index Integer	

Outputs

Out Exec	
Return Value Array of Integers	Gets neighboring vertex indexes, requires a Localized Grid from LocalizeMeshData()

Documentation > *Mesh Ops Plugin BPLibrary* > *Gradient Color Mesh*

Gradient Color Mesh

Sets the vertex color for the entire mesh at once based on a gradient exending from PointA to PointB.

Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Point A Vector	
Point B Vector	
Color A Linear Color Structure	
Color B Linear Color Structure	

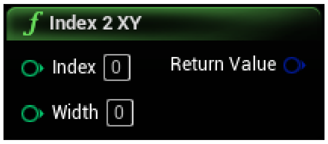
Outputs

Out Exec	

[Documentation](#) > [Mesh Ops Plugin BPLibrary](#) > *Index 2 XY*

Index 2 XY

Pixel index to 2D Coord



Inputs

Index Integer	pixel index
Width Integer	image width for pixel array, if using a Render Target, get Size X variable from it.

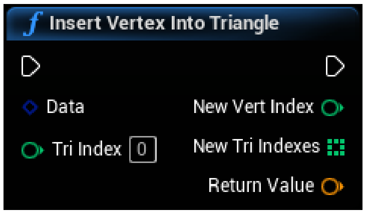
Outputs

Return Value Coord Structure	

[Documentation](#) > [Mesh Ops Plugin BPLibrary](#) > *Insert Vertex Into Triangle*

Insert Vertex Into Triangle

Adds a new vertex to a triangle, splitting it into 3 new triangles.



Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Tri Index Integer	

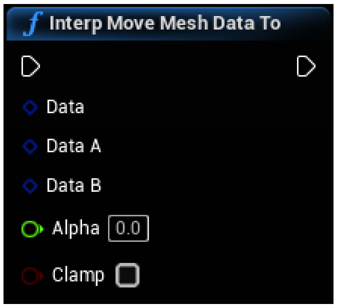
Outputs

Out Exec	
New Vert Index Integer	
New Tri Indexes Array of Integers	
Return Value Vector	Adds a new vertex to a triangle, splitting it into 3 new triangles.

[Documentation](#) > [Mesh Ops Plugin BPLibrary](#) > *Interp Move Mesh Data To*

Interp Move Mesh Data To

Moves all verts in Data from DataA toward DataB, with Aplha ranging from 0.0 to 1.0. All three Datas should be the same mesh with DataB transformed in some way and Data starting out equal to DataA.



Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Data A Mesh Data Structure (by ref)	
Data B Mesh Data Structure (by ref)	
Alpha Float	
Clamp Boolean	

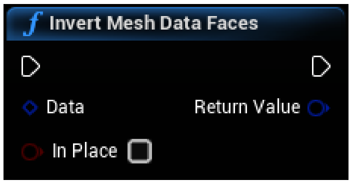
Outputs

Out Exec	

Documentation > Mesh Ops Plugin BPLibrary > Invert Mesh Data Faces

Invert Mesh Data Faces

Inverts Triangles/Faces in a Mesh Data



Inputs

In Exec	
Data Mesh Data Structure (by ref)	
In Place Boolean	

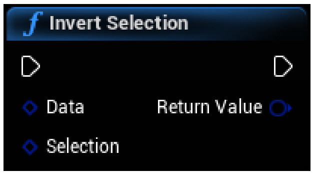
Outputs

Out Exec	

Return Value `Mesh Data Structure`	Inverts Triangles/Faces in a Mesh Data

Invert Selection

Inverts a Selection. (Selects everything except Selection)



Inputs

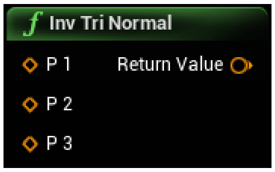
In `Exec`	
Data `Mesh Data Structure (by ref)`	
Selection `Mesh Selection Structure (by ref)`	

Outputs

Out `Exec`	
Return Value `Mesh Selection Structure`	Inverts a Selection. (Selects everything except Selection)

Inv Tri Normal

Normal direction of a triangle, used internally. Gives inverted (*-1) result.



Inputs

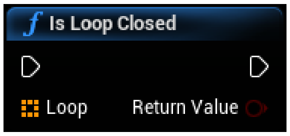
P 1 `Vector (by ref)`	
P 2 `Vector (by ref)`	
P 3 `Vector (by ref)`	

Outputs

Return Value `Vector`	Normal direction of a triangle, used internally. Gives inverted (*-1) result.

Is Loop Closed

Checks if a loop has identical Loop[0] and Loop[LastIndex]



Inputs

In Exec	
Loop Array of Vectors	

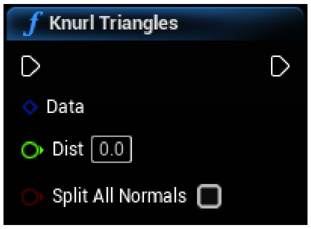
Outputs

Out Exec	
Return Value Boolean	Checks if a loop has identical Loop[0] and Loop[LastIndex]

Documentation > Mesh Ops Plugin BPLibrary > Knurl Triangles

Knurl Triangles

Adds 'knurling' to all triangles in the mesh. (Adds a raised center point to each triangle and splits the new point's normals)



Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Dist Float	
Split All Normals Boolean	

Outputs

Out Exec	

Documentation > Mesh Ops Plugin BPLibrary > Line Intersect 2DPoly

Line Intersect 2DPoly

Checks if a line intersects a loop of verts, returns on hit.

Inputs

Loop Array of Vectors	
A Vector	

B Vector	

Outputs

Return Value Boolean	Checks if a line intersects a loop of verts, returns on hit.

Documentation > *Mesh Ops Plugin BPLibrary* > *Line Intersect 3D*

Line Intersect 3D

Finds closest point on line A, if these two lines actually cross it is their intersection, which is profoundly unlikely in 3D space in general - hence the closest point approximation.

Inputs

A1 Vector	
A2 Vector	
B1 Vector	
B2 Vector	
Clamp to Segment Boolean	

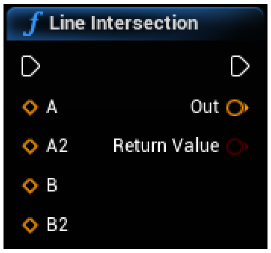
Outputs

Intersected Boolean	
Return Value Vector	Finds closest point on line A, if these two lines actually cross it is their intersection, which is profoundly unlikely in 3D space in general - hence the closest point approximation.

Documentation > *Mesh Ops Plugin BPLibrary* > *Line Intersection*

Line Intersection

Line segment intersection



Inputs

In Exec	
A Vector (by ref)	
A2 Vector (by ref)	
B Vector (by ref)	

B2 `Vector (by ref)`	

Outputs

Out `Exec`	
Out `Vector`	
Return Value `Boolean`	Line segment intersection

Documentation > Mesh Ops Plugin BPLibrary > Line Plane Intersection

Line Plane Intersection

Gets the intersection point of a line segment and a plane, in 3D.

Inputs

In `Exec`	
Start `Vector`	
End `Vector`	
Plane Origin `Vector`	
Plane Up `Vector`	

Outputs

Out `Exec`	
Out Point `Vector`	
Return Value `Boolean`	Gets the intersection point of a line segment and a plane, in 3D.

Documentation > Mesh Ops Plugin BPLibrary > Line Trace Mesh Data

Line Trace Mesh Data

Does an infinite line trace from Start in direction Dir, against Data. Searches every triangle.

Inputs

In `Exec`	
Data `Mesh Data Structure (by ref)`	
Start `Vector`	
Dir `Vector`	

Outputs

Out Exec	
Out Boolean	
Return Value Tri Hit Result Structure	Does an infinite line trace from Start in direction Dir, against Data. Searches every triangle.

Documentation > Mesh Ops Plugin BPLibrary > Line Trace Triangle

Line Trace Triangle

Does an infinite line trace from Start in direction Dir, against the triangle starting at TriIndex.

Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Start Vector	
Dir Vector	
Tri Index Integer	

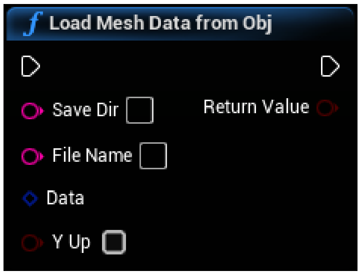
Outputs

Out Exec	
Out Boolean	
Return Value Tri Hit Result Structure	Does an infinite line trace from Start in direction Dir, against the triangle starting at TriIndex.

Documentation > Mesh Ops Plugin BPLibrary > Load Mesh Data from Obj

Load Mesh Data from Obj

Loads a mesh from an obj file. Not guaranteed to work with all variations of .obj formatting.



Inputs

In Exec	
Save Dir String	
File Name String	
Data Mesh Data Structure (by ref)	

Y Up Boolean	

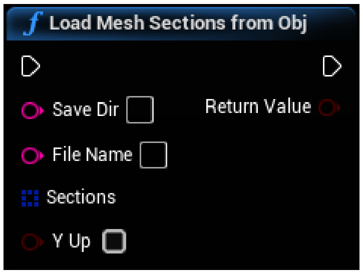
Outputs

Out Exec	
Return Value Boolean	Loads a mesh from an obj file. Not guaranteed to work with all variations of .obj formatting.

Documentation > Mesh Ops Plugin BPLibrary > Load Mesh Sections from Obj

Load Mesh Sections from Obj

Loads all sections of a mesh from an obj file. Not guaranteed to work with all variations of .obj formatting.



Inputs

In Exec	
Save Dir String	
File Name String	
Sections Array of Mesh Data Structures	
Y Up Boolean	

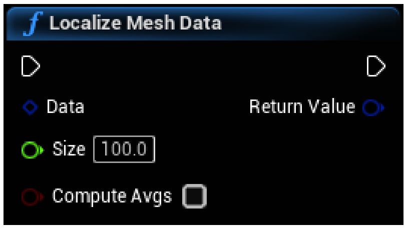
Outputs

Out Exec	
Return Value Boolean	Loads all sections of a mesh from an obj file. Not guaranteed to work with all variations of .obj formatting.

Documentation > Mesh Ops Plugin BPLibrary > Localize Mesh Data

Localize Mesh Data

Groups vertices into grid-aligned boxes, makes localized editing much faster because nodes that use it don't need to sample the entire mesh at once. ComputeAvgs computes averages for each group, which isn't currently used.



Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Size Float	
Compute Avgs Boolean	

Outputs

Out Exec	
Return Value Localized Grid Structure	Groups vertices into grid-aligned boxes, makes localized editing much faster because nodes that use it don't need to sample the entire mesh at once. ComputeAvgs computes averages for each group, which isn't currently used.

Documentation > Mesh Ops Plugin BPLibrary > *Loop Is Clockwise*

Loop Is Clockwise

Checks if a Loop is Clockwise (true) or counter-clockwise (false).

Inputs

Loop Array of Vectors	

Outputs

Return Value Boolean	Checks if a Loop is Clockwise (true) or counter-clockwise (false).

Documentation > Mesh Ops Plugin BPLibrary > *Loop to Loop Struct*

Loop to Loop Struct

Changes an Array of vectors Loop into a Loop Structure.

Inputs

In Exec	
Loop Array of Vectors	
Depth Float	

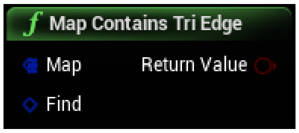
Outputs

Out Exec	
Return Value Loop Structure	Changes an Array of vectors Loop into a Loop Structure.

Documentation > Mesh Ops Plugin BPLibrary > *Map Contains Tri Edge*

Map Contains Tri Edge

Calls Contains from the Edge Map, regardless of the A vs B order of the key edge. Flips the edge if needed.



Inputs

Map `Map of Tri Edge Structures to Tri Edge Structures`	
Find `Tri Edge Structure (by ref)`	

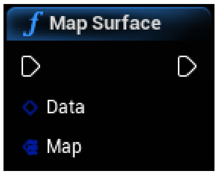
Outputs

Return Value `Boolean`	Calls Contains from the Edge Map, regardless of the A vs B order of the key edge. Flips the edge if needed.

Documentation > *Mesh Ops Plugin BPLibrary* > *Map Surface*

Map Surface

Builds Map of edges to connected Tris



Inputs

In `Exec`	
Data `Mesh Data Structure (by ref)`	
Map `Map of Tri Edge Structures to Tri Edge Structures`	

Outputs

Out `Exec`	

Documentation > *Mesh Ops Plugin BPLibrary* > *Merge 2Spheres*

Merge 2Spheres

EXPERIMENTAL Merges two spheres, finding a sphere that encompasses both and has a new center between them. Used internally, this is not a mesh operation.

Inputs

In `Exec`	
Sphere 1	

Vector (by ref)	
Radius 1 Float	
Sphere 2 Vector (by ref)	
Radius 2 Float	

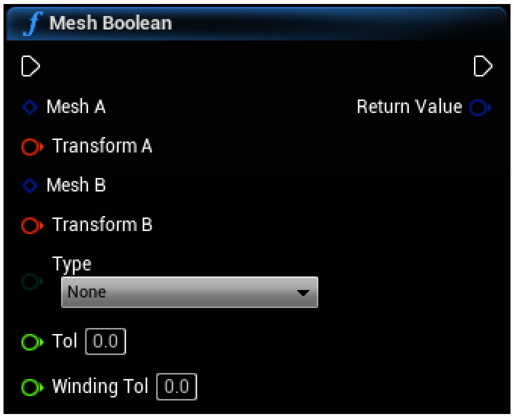
Outputs

Out Exec	
New Radius Float	
Return Value Vector	EXPERIMENTAL Merges two spheres, finding a sphere that encompasses both and has a new center between them. Used internally, this is not a mesh operation.

Documentation > *Mesh Ops Plugin BPLibrary* > *Mesh Boolean*

Mesh Boolean

Performs Mesh Boolean Operation. Uses "Geometry Processing" plugin included in the engine since 4.26.



Type None

Tol 0.0

Winding Tol 0.0

Inputs

In Exec	
Mesh A Mesh Data Structure (by ref)	
Transform A Transform	
Mesh B Mesh Data Structure (by ref)	
Transform B Transform	
Type BooleanType Enum	
Tol Float	Simplification angle
Winding Tol Float	

Outputs

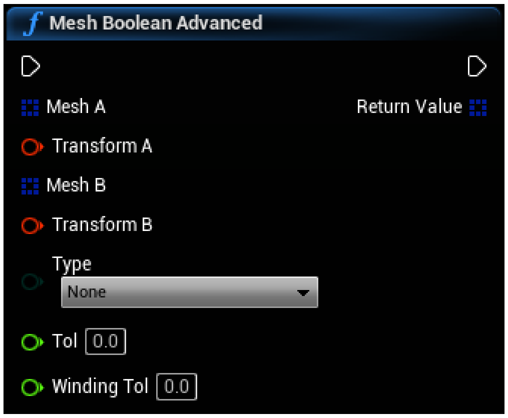
Out Exec	
Return Value	

Mesh Data Structure	

[Documentation](#) > [Mesh Ops Plugin BPLibrary](#) > *Mesh Boolean Advanced*

Mesh Boolean Advanced

Performs Mesh Boolean Operation. Uses "Geometry Processing" plugin included in the engine since 4.26.



Inputs

In Exec	
Mesh A Array of Mesh Data Structures	
Transform A Transform	
Mesh B Array of Mesh Data Structures	
Transform B Transform	
Type BooleanType Enum	
Tol Float	Simplification angle
Winding Tol Float	

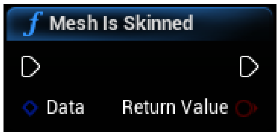
Outputs

Out Exec	
Return Value Array of Mesh Data Structures	

[Documentation](#) > [Mesh Ops Plugin BPLibrary](#) > *Mesh Is Skinned*

Mesh Is Skinned

EXPERIMENTAL Checks whether a mesh has been skinned.



Inputs

In	

Exec	
Data `Mesh Data Structure (by ref)`	

Outputs

Out `Exec`	
Return Value `Boolean`	EXPERIMENTAL Checks whether a mesh has been skinned.

[Documentation](#) > [Mesh Ops Plugin BPLibrary](#) > *Mirror Cut Mesh Data*

Mirror Cut Mesh Data

Slices a mesh along a plane, and then mirrors across the plane.

Inputs

In `Exec`	
Data `Mesh Data Structure (by ref)`	
Center `Vector`	
Normal `Vector`	
In Place `Boolean`	

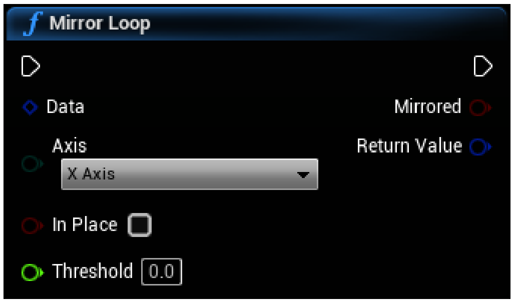
Outputs

Out `Exec`	
Return Value `Mesh Data Structure`	Slices a mesh along a plane, and then mirrors across the plane.

[Documentation](#) > [Mesh Ops Plugin BPLibrary](#) > *Mirror Loop*

Mirror Loop

EXPERIMENTAL



Axis X Axis

In Place ☐

Threshold 0.0

Inputs

In `Exec`	
Data `Mesh Data Structure (by ref)`	
Axis	

AxisEnum Enum	
In Place Boolean	
Threshold Float	

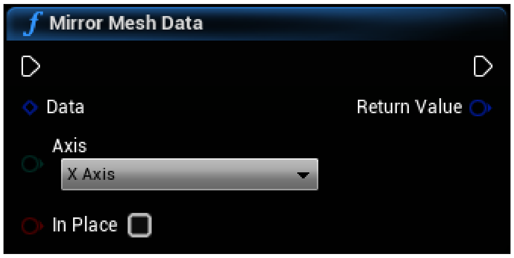
Outputs

Out Exec	
Mirrored Boolean	
Return Value Mesh Data Structure	EXPERIMENTAL

Documentation > Mesh Ops Plugin BPLibrary > Mirror Mesh Data

Mirror Mesh Data

EXPERIMENTAL



Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Axis AxisEnum Enum	
In Place Boolean	

Outputs

Out Exec	
Return Value Mesh Data Structure	EXPERIMENTAL

Documentation > Mesh Ops Plugin BPLibrary > Multi Loop Extrude

Multi Loop Extrude

Fills and extrudes multiple loop structs upward and puts it into Data. Counter-clockwise loops are subtracted from the result. Can be used for buildings from mapping data.

Inputs

In Exec	
Data Mesh Data Structure (by ref)	

Multi Loop Array of Loop Structures	
Border UVSize Float	

Outputs

Out Exec	

Documentation > Mesh Ops Plugin BPLibrary > *Next Triangle*

Next Triangle

Gets index of triangle Point+Dir is in, based off of the mapped surface. Returns same triangle if it remains inside. Use MapSurface as Map input.

Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Map Map of Tri Edge Structures to Tri Edge Structures	
Tri Integer	
Point Vector	
Dir Vector	

Outputs

Out Exec	
Out Vector	
Out Dir Vector	
Return Value Integer	Gets index of triangle Point+Dir is in, based off of the mapped surface. Returns same triangle if it remains inside. Use MapSurface as Map input.

Documentation > Mesh Ops Plugin BPLibrary > *Now*

Now

Current system time

Inputs

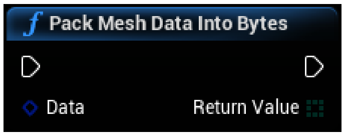
Outputs

Return Value Float	Current system time

Documentation > Mesh Ops Plugin BPLibrary > *Pack Mesh Data Into Bytes*

Pack Mesh Data Into Bytes

Serializes Mesh Data into Bytes



Inputs

In Exec	
Data Mesh Data Structure (by ref)	

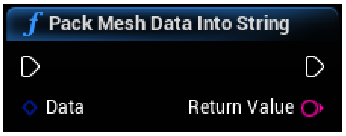
Outputs

Out Exec	
Return Value Array of Bytes	Serializes Mesh Data into Bytes

Documentation > Mesh Ops Plugin BPLibrary > *Pack Mesh Data Into String*

Pack Mesh Data Into String

Serializes Mesh Data into a string.



Inputs

In Exec	
Data Mesh Data Structure (by ref)	

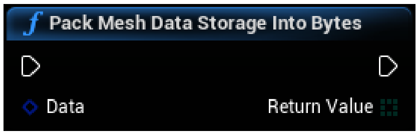
Outputs

Out Exec	
Return Value String	Serializes Mesh Data into a string.

Documentation > Mesh Ops Plugin BPLibrary > *Pack Mesh Data Storage Into Bytes*

Pack Mesh Data Storage Into Bytes

Serializes Mesh Data Storage into Bytes



Inputs

In `Exec`	
Data `Mesh Data Storage Structure (by ref)`	

Outputs

Out `Exec`	
Return Value `Array of Bytes`	Serializes Mesh Data Storage into Bytes

Documentation > Mesh Ops Plugin BPLibrary > Paint Mesh Color

Paint Mesh Color

Sets the vertex color on verts within a sphere at SphereLoc and radius Radius. Opacity determines the amount of blending with the color that's already there.

Inputs

In `Exec`	
Data `Mesh Data Structure (by ref)`	
Sphere Loc `Vector`	
Radius `Float`	
Color `Linear Color Structure`	
Opacity `Float`	

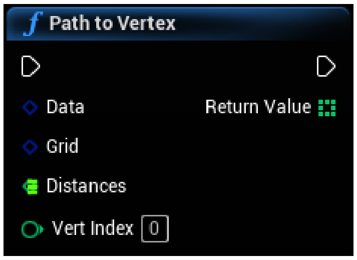
Outputs

Out `Exec`	

Documentation > Mesh Ops Plugin BPLibrary > Path to Vertex

Path to Vertex

Uses data from CalcDistancesFromVertexes() for pathfinding.



Inputs

In `Exec`	
Data `Mesh Data Structure (by ref)`	

Grid Localized Grid Structure (by ref)	
Distances Map of Integers to Floats	
Vert Index Integer	

Outputs

Out Exec	
Return Value Array of Integers	Uses data from CalcDistancesFromVertexes() for pathfinding.

Documentation > Mesh Ops Plugin BPLibrary > *Planar Project UVs*

Planar Project UVs

Generates UVs based on a virtual plane crossing through the mesh. MaxWidth should be roughly the maximum dimension of the mesh, which will be calculated automatically if left at 0.0

Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Max Width Float	
Transform Transform	

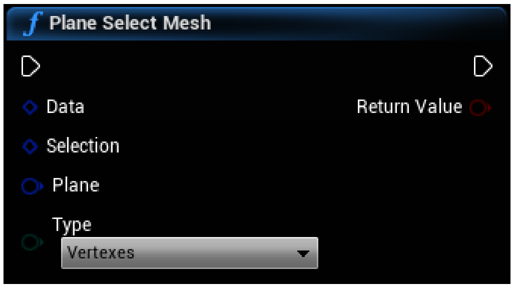
Outputs

Out Exec	

Documentation > Mesh Ops Plugin BPLibrary > *Plane Select Mesh*

Plane Select Mesh

Selects vertexes of a mesh under a plane for use in other functions, like TransformSelection().



Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Selection	

Mesh Selection Structure (by ref)	
Plane Plane Brush Structure	
Type SelectionType Enum	

Outputs

Out Exec	
Return Value Boolean	Selects vertexes of a mesh under a plane for use in other functions, like TransformSelection().

Documentation > Mesh Ops Plugin BPLibrary > *Point Inside 2DPoly*

Point Inside 2DPoly

Checks if a point is inside a 2D loop, uses raycast counting method.

Inputs

In Exec	
Loop Array of Vectors	
Px Float	
Py Float	

Outputs

Out Exec	
Return Value Boolean	Checks if a point is inside a 2D loop, uses raycast counting method.

Documentation > Mesh Ops Plugin BPLibrary > *Point Inside 2DTriangle*

Point Inside 2DTriangle

Checks if point pt is within triangle (v1,v2,v3).

Inputs

Pt Vector 2D Structure (by ref)	
V 1 Vector 2D Structure (by ref)	
V 2 Vector 2D Structure (by ref)	
V 3 Vector 2D Structure (by ref)	
Tolerance Float	

Outputs

Return Value Boolean	Checks if point pt is within triangle (v1,v2,v3).

[Documentation](#) > [Mesh Ops Plugin BPLibrary](#) > *Point Inside Mesh*

Point Inside Mesh

Checks if a point is inside Data. Slow operation, checks every triangle.

Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Point Vector	
Trace Dir Vector	

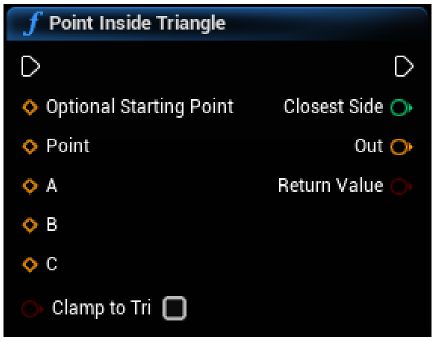
Outputs

Out Exec	
Return Value Boolean	Checks if a point is inside Data. Slow operation, checks every triangle.

[Documentation](#) > [Mesh Ops Plugin BPLibrary](#) > *Point Inside Triangle*

Point Inside Triangle

Checks if a point is inside a triangle and gets the closest edge index as well.



Inputs

In Exec	
Optional Starting Point Vector (by ref)	
Point Vector (by ref)	
A Vector (by ref)	
B Vector (by ref)	
C Vector (by ref)	
Clamp to Tri Boolean	

Outputs

Out Exec	
Closest Side Integer	
Out Vector	
Return Value Boolean	Checks if a point is inside a triangle and gets the closest edge index as well.

Documentation > Mesh Ops Plugin BPLibrary > Point Inside Tri Prism

Point Inside Tri Prism

Check if Point is in front of a triangle, i.e. check if Point is inside an infinitely long triangular prism starting at the triangle, and extending in the normal's direction.

Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Point Vector	
Tri Integer	
Tol Float	

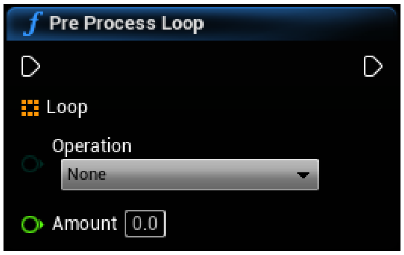
Outputs

Out Exec	
Return Value Boolean	Check if Point is in front of a triangle, i.e. check if Point is inside an infinitely long triangular prism starting at the triangle, and extending in the normal's direction.

Documentation > Mesh Ops Plugin BPLibrary > Pre Process Loop

Pre Process Loop

Pre-Process a Loop, performing various operations before performing something else, such as Filling.



Inputs

In Exec	
Loop Array of Vectors	
Operation LoopPreProcessType Enum	

Amount Float	

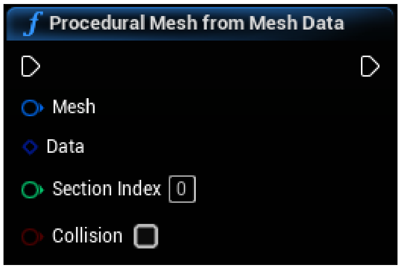
Outputs

Out Exec	

Documentation > Mesh Ops Plugin BPLibrary > Procedural Mesh from Mesh Data

Procedural Mesh from Mesh Data

Creates a procedural mesh section from a Mesh Data



Inputs

In Exec	
Mesh Procedural Mesh Component Object Reference	
Data Mesh Data Structure (by ref)	
Section Index Integer	
Collision Boolean	

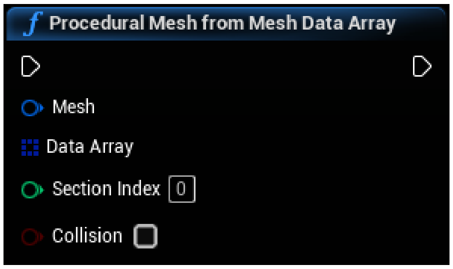
Outputs

Out Exec	

Documentation > Mesh Ops Plugin BPLibrary > Procedural Mesh from Mesh Data Array

Procedural Mesh from Mesh Data Array

Creates all procedural mesh sections from a Mesh Data Storage



Inputs

In Exec	
Mesh	

Procedural Mesh Component Object Reference	
Data Array Array of Mesh Data Structures	
Section Index Integer	
Collision Boolean	

Outputs

Out Exec	

[Documentation](#) > [Mesh Ops Plugin BPLibrary](#) > Project Point to Tri

Project Point to Tri

Projects a Point onto a triangle who's index starts at Tri, clamps its position to the edge if it's not on the triangle.

Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Point Vector	
Tri Integer	

Outputs

Out Exec	
Return Value Vector	Projects a Point onto a triangle who's index starts at Tri, clamps its position to the edge if it's not on the triangle.

[Documentation](#) > [Mesh Ops Plugin BPLibrary](#) > Project Vector to Tri

Project Vector to Tri

Projects a direction vector onto a Triangle based on its normal.

Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Dir Vector	
Tri Integer	

Outputs

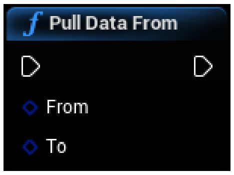
Out Exec	

Return Value Vector	Projects a direction vector onto a Triangle based on its normal.

Documentation > Mesh Ops Plugin BPLibrary > *Pull Data From*

Pull Data From

Pulls mesh data from the From input according to To's Tris. It only pulls what it needs. Useful for extracting triangles, but you can use ExtractSelectionAsMeshData() as well.



Inputs

In Exec	
From Mesh Data Structure (by ref)	
To Mesh Data Structure (by ref)	

Outputs

Out Exec	

Documentation > Mesh Ops Plugin BPLibrary > *Random Point on Surface*

Random Point on Surface

Gets a random point on a surface, taking into consideration the relative sizes of all the triangles. If Map isn't generated yet, it does it for you.

Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Map Map of Integers to Integers	
Seed Integer	
Adjust Float	
Bias Vector	

Outputs

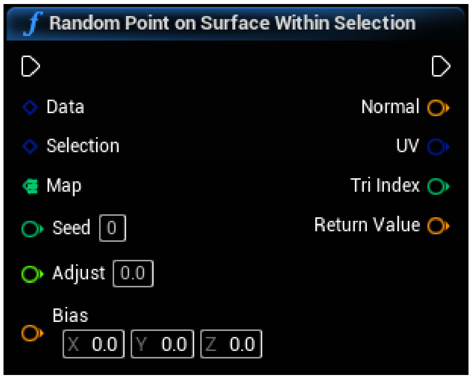
Out Exec	
Normal Vector	
UV Vector 2D Structure	

Tri Index `Integer`	
Return Value `Vector`	Gets a random point on a surface, taking into consideration the relative sizes of all the triangles. If Map isn't generated yet, it does it for you.

[Documentation](#) > [Mesh Ops Plugin BPLibrary](#) > *Random Point on Surface Within Selection*

Random Point on Surface Within Selection

Gets a random point on a surface, taking into consideration the relative sizes of all the triangles. If Map isn't generated yet, it does it for you.



Inputs

In `Exec`	
Data `Mesh Data Structure (by ref)`	
Selection `Mesh Selection Structure (by ref)`	
Map `Map of Integers to Integers`	
Seed `Integer`	
Adjust `Float`	
Bias `Vector`	

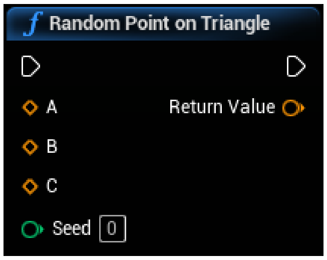
Outputs

Out `Exec`	
Normal `Vector`	
UV `Vector 2D Structure`	
Tri Index `Integer`	
Return Value `Vector`	Gets a random point on a surface, taking into consideration the relative sizes of all the triangles. If Map isn't generated yet, it does it for you.

[Documentation](#) > [Mesh Ops Plugin BPLibrary](#) > *Random Point on Triangle*

Random Point on Triangle

Gets a random point on a triangle with points A, B, and C



Inputs

In `Exec`	
A `Vector (by ref)`	
B `Vector (by ref)`	
C `Vector (by ref)`	
Seed `Integer`	

Outputs

Out `Exec`	
Return Value `Vector`	Gets a random point on a triangle with points A, B, and C

[Documentation](#) > [Mesh Ops Plugin BPLibrary](#) > *Random Shift Mesh Data Verts*

Random Shift Mesh Data Verts

Randomly shifts vertexes along normals, Keeps faces together using Tol tolerance.

Inputs

In `Exec`	
Data `Mesh Data Structure (by ref)`	
Min Dist `Float`	
Max Dist `Float`	
Tol `Float`	

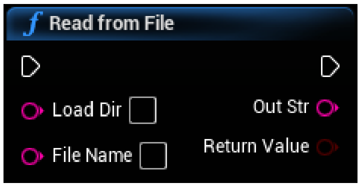
Outputs

Out `Exec`	
Return Value `Mesh Data Structure`	Randomly shifts vertexes along normals, Keeps faces together using Tol tolerance.

[Documentation](#) > [Mesh Ops Plugin BPLibrary](#) > *Read from File*

Read from File

Reads file as a String



Inputs

In Exec	
Load Dir String	
File Name String	

Outputs

Out Exec	
Out Str String	
Return Value Boolean	Reads file as a String

Documentation > Mesh Ops Plugin BPLibrary > Rediscover Verts

Rediscover Verts

Rediscovers vertexes after an operation that may reorder vertexes. Any vertexes in GoodEnough range will be considered the same vertex.

Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Verts Array of Vectors	
Good Enough Float	
Reverse Boolean	swaps Remap to be Verts indexes to Data indexes.
Find Closest Boolean	

Outputs

Out Exec	
Remap Map of Integers to Integers	Data indexes to Verts indexes.

Documentation > Mesh Ops Plugin BPLibrary > Reduce Mesh Data

Reduce Mesh Data

Simplifies/Reduces a Mesh Data, using anglular Deviation of face normals. Stitches mesh together via StitchDist distance, 0.0 skips stitching

Inputs

In `Exec`	
Data `Mesh Data Structure (by ref)`	
Deviation `Float`	
Stitch Dist `Float`	

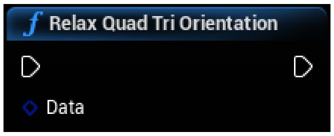
Outputs

Out `Exec`	
Return Value `Mesh Data Structure`	Simplifies/Reduces a Mesh Data, using anglular Deviation of face normals. Stitches mesh together via StitchDist distance, 0.0 skips stitching

[Documentation](#) > [Mesh Ops Plugin BPLibrary](#) > Relax Quad Tri Orientation

Relax Quad Tri Orientation

Flips all triangle orientations depending on how each quad is stretched, useful to reduce obvious diagonal lines on a heightmapped mesh. Only works correctly with quad-type meshes where every 2 tris are a quad, like grid planes. e.g. Tris = {quad0_tri0_0,quad0_tri0_1,quad0_tri0_2,quad0_tri1_0,quad0_tri1_1,quad0_tri1_2, quad1_tri0_0,quad1_tri0_1,quad1_tri0_2,quad1_tri1_0,quad1_tri1_1,quad1_tri1_2}



Inputs

In `Exec`	
Data `Mesh Data Structure (by ref)`	

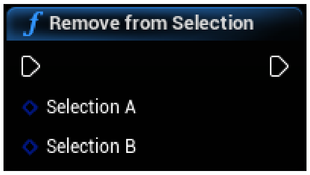
Outputs

Out `Exec`	

[Documentation](#) > [Mesh Ops Plugin BPLibrary](#) > Remove from Selection

Remove from Selection

Removes SelectionB from SelectionA. Assumes selections are the same type, behavior is undefined (but should work) if not.



Inputs

In `Exec`	
Selection A	

`Mesh Selection Structure (by ref)`	
Selection B `Mesh Selection Structure (by ref)`	

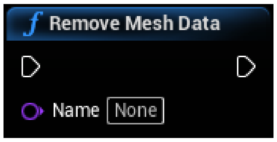
Outputs

Out `Exec`	

Documentation > Mesh Ops Plugin BPLibrary > Remove Mesh Data

Remove Mesh Data

Removes temporarily stored Mesh Data.



Inputs

In `Exec`	
Name `Name`	

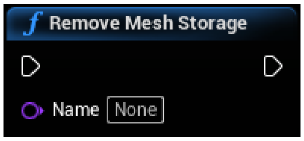
Outputs

Out `Exec`	

Documentation > Mesh Ops Plugin BPLibrary > Remove Mesh Storage

Remove Mesh Storage

Removes temporarily stored Mesh Data Storage.



Inputs

In `Exec`	
Name `Name`	

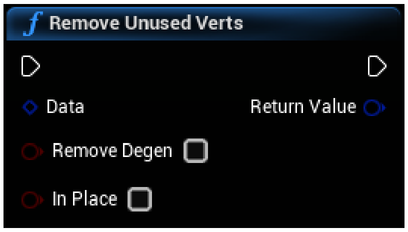
Outputs

Out `Exec`	

Documentation > Mesh Ops Plugin BPLibrary > Remove Unused Verts

Remove Unused Verts

Removes all unused vertexes as well as normals, uvs, and colors.



Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Remove Degen Boolean	
In Place Boolean	

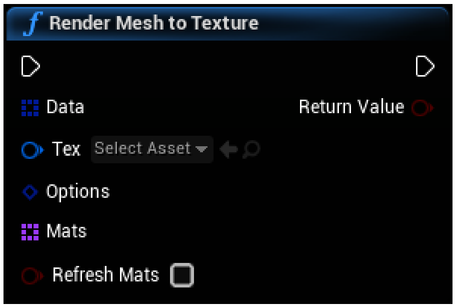
Outputs

Out Exec	
Return Value Mesh Data Structure	Removes all unused vertexes as well as normals, uvs, and colors.

Documentation > Mesh Ops Plugin BPLibrary > Render Mesh to Texture

Render Mesh to Texture

EXPERIMENTAL Renders a 3D mesh in 2D. Requires a Texture2D, you can create one from CreateEmptyTexture().



Inputs

In Exec	
Data Array of Mesh Data Structures	
Tex Texture 2D Object Reference	
Options Mesh Render Options Structure (by ref)	
Mats Array of Names	
Refresh Mats Boolean	

Outputs

Out	

Exec	
Return Value Boolean	EXPERIMENTAL Renders a 3D mesh in 2D. Requires a Texture2D, you can create one from CreateEmptyTexture().

Documentation > Mesh Ops Plugin BPLibrary > Render Normals to Texture

Render Normals to Texture

EXPERIMENTAL Renders mesh normals to a new texture2D using the mesh's UVs.

Inputs

In Exec	
Data Mesh Data Structure (by ref)	
XSamples Integer	
YSamples Integer	
Tolerance Float	

Outputs

Out Exec	
Return Value Texture 2D Object Reference	EXPERIMENTAL Renders mesh normals to a new texture2D using the mesh's UVs.

Documentation > Mesh Ops Plugin BPLibrary > Retrieve Mesh Data

Retrieve Mesh Data

Retrieves temporarily stored Mesh Data.

Inputs

In Exec	
Name Name	

Outputs

Out Exec	
Found Boolean	
Return Value Mesh Data Structure	Retrieves temporarily stored Mesh Data.

Documentation > Mesh Ops Plugin BPLibrary > Retrieve Mesh Storage

Retrieve Mesh Storage

Retrieves temporarily stored Mesh Data Storage.

Inputs

In `Exec`	
Name `Name`	

Outputs

Out `Exec`	
Found `Boolean`	
Return Value `Mesh Data Storage Structure`	Retrieves temporarily stored Mesh Data Storage.

Documentation > Mesh Ops Plugin BPLibrary > Revolve Loop

Revolve Loop

Revolves a Loop to form a solid mesh, forms donut-like shapes.

Inputs

In `Exec`	
Loop `Array of Vectors`	
UVSize `Float`	
Axis `Vector`	
Pivot `Vector`	
Iterations `Integer`	

Outputs

Out `Exec`	
Return Value `Mesh Data Structure`	Revolves a Loop to form a solid mesh, forms donut-like shapes.

Documentation > Mesh Ops Plugin BPLibrary > Rotate Loop Around Axis

Rotate Loop Around Axis

Rotates a Loop around an Axis using an Angle (0-360), optionally writes directly to the loop with InPlace.

Inputs

In `Exec`	
Loop `Array of Vectors`	
Axis `Vector`	

Angle Float	
In Place Boolean	
Pivot Vector	

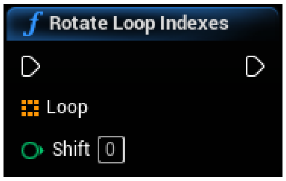
Outputs

Out Exec	
Return Value Array of Vectors	Rotates a Loop around an Axis using an Angle (0-360), optionally writes directly to the loop with InPlace.

Documentation > Mesh Ops Plugin BPLibrary > *Rotate Loop Indexes*

Rotate Loop Indexes

Revolves Loop indexes Shift amount, forward (positive shift) or backward (negative shift)



Inputs

In Exec	
Loop Array of Vectors	
Shift Integer	

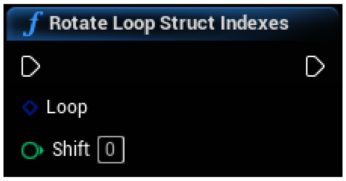
Outputs

Out Exec	

Documentation > Mesh Ops Plugin BPLibrary > *Rotate Loop Struct Indexes*

Rotate Loop Struct Indexes

Revolves Loop indexes Shift amount, forward (positive shift) or backward (negative shift)



Inputs

In Exec	
Loop Loop Structure (by ref)	
Shift Integer	

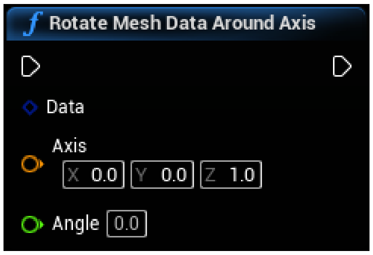
Outputs

Out `Exec`	

Documentation > Mesh Ops Plugin BPLibrary > *Rotate Mesh Data Around Axis*

Rotate Mesh Data Around Axis

Rotates a mesh around the specified axis.



Inputs

In `Exec`	
Data `Mesh Data Structure (by ref)`	
Axis `Vector`	
Angle `Float`	

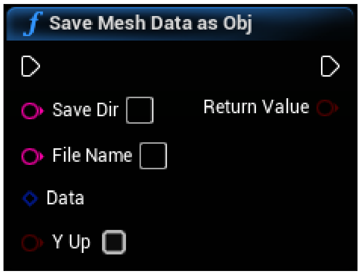
Outputs

Out `Exec`	

Documentation > Mesh Ops Plugin BPLibrary > *Save Mesh Data as Obj*

Save Mesh Data as Obj

Saves a mesh as an obj file. Tested with Blender, Maya, 3D Viewer, and Paint 3D. There may be issues with face vertices being detached, you can either weld them beforehand or edit in the software. e.g. in Maya using the merge vertex tool.



Inputs

In `Exec`	
Save Dir `String`	
File Name `String`	
Data `Mesh Data Structure (by ref)`	

Y Up `Boolean`	

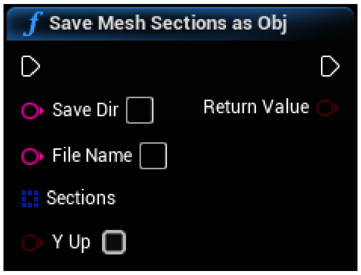
Outputs

Out `Exec`	
Return Value `Boolean`	Saves a mesh as an obj file. Tested with Blender, Maya, 3D Viewer, and Paint 3D. There may be issues with face vertices being detached, you can either weld them beforehand or edit in the software. e.g. in Maya using the merge vertex tool.

[Documentation](#) > [Mesh Ops Plugin BPLibrary](#) > *Save Mesh Sections as Obj*

Save Mesh Sections as Obj

Saves all sections of a mesh as an obj file. Tested with Blender, Maya, 3D Viewer, and Paint 3D. There may be issues with face vertices being detached, you can either weld them beforehand or edit in the software. e.g. in Maya using the merge vertex tool.



Inputs

In `Exec`	
Save Dir `String`	
File Name `String`	
Sections `Array of Mesh Data Structures`	
Y Up `Boolean`	

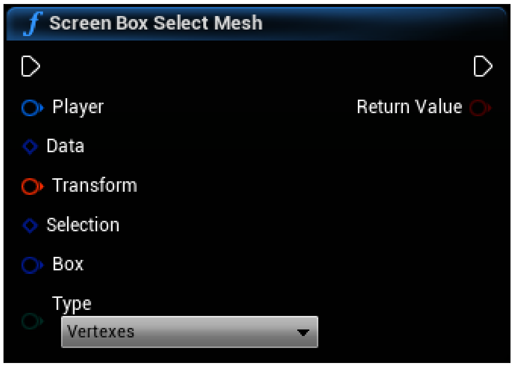
Outputs

Out `Exec`	
Return Value `Boolean`	Saves all sections of a mesh as an obj file. Tested with Blender, Maya, 3D Viewer, and Paint 3D. There may be issues with face vertices being detached, you can either weld them beforehand or edit in the software. e.g. in Maya using the merge vertex tool.

[Documentation](#) > [Mesh Ops Plugin BPLibrary](#) > *Screen Box Select Mesh*

Screen Box Select Mesh

Makes a Selection within a box in screen space. Returns false if no selection is made.



Inputs

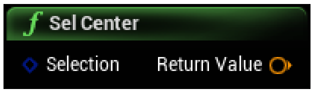
In Exec	
Player Player Controller Object Reference	
Data Mesh Data Structure (by ref)	
Transform Transform	
Selection Mesh Selection Structure (by ref)	
Box Box 2D Structure	
Type SelectionType Enum	

Outputs

Out Exec	
Return Value Boolean	Makes a Selection within a box in screen space. Returns false if no selection is made.

Sel Center

Simple accessor for the center of the selection.



Inputs

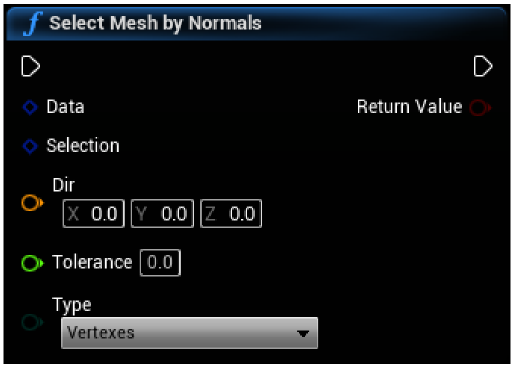
Selection Mesh Selection Structure (by ref)	

Outputs

Return Value Vector	Simple accessor for the center of the selection.

Select Mesh by Normals

Selects vertexes of a mesh by their normal for use in other functions, like TransformSelection().



Dir X 0.0 Y 0.0 Z 0.0

Tolerance 0.0

Type Vertexes

Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Selection Mesh Selection Structure (by ref)	
Dir Vector	
Tolerance Float	
Type SelectionType Enum	

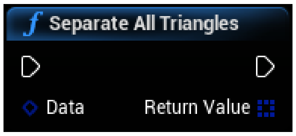
Outputs

Out Exec	
Return Value Boolean	Selects vertexes of a mesh by their normal for use in other functions, like TransformSelection().

Documentation > Mesh Ops Plugin BPLibrary > *Separate All Triangles*

Separate All Triangles

Gets all triangles and then turns each of them into a new mesh data.



Inputs

In Exec	
Data Mesh Data Structure (by ref)	

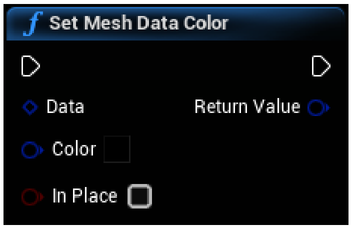
Outputs

Out Exec	
Return Value Array of Mesh Data Structures	Gets all triangles and then turns each of them into a new mesh data.

Documentation > Mesh Ops Plugin BPLibrary > *Set Mesh Data Color*

Set Mesh Data Color

Sets Vertex Colors to Color of entire Mesh Data.



Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Color Linear Color Structure	
In Place Boolean	

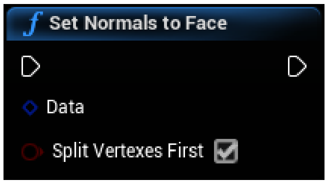
Outputs

Out Exec	
Return Value Mesh Data Structure	Sets Vertex Colors to Color of entire Mesh Data.

Documentation > Mesh Ops Plugin BPLibrary > *Set Normals to Face*

Set Normals to Face

Sets all normals to their face direction. Creates hard edges or low poly look, use SoftenNormals after to fix this up.



Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Split Vertexes First Boolean	

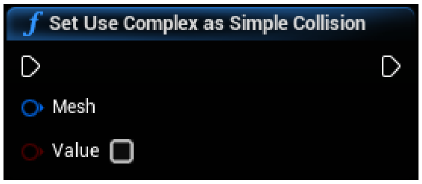
Outputs

Out Exec	

Documentation > Mesh Ops Plugin BPLibrary > *Set Use Complex as Simple Collision*

Set Use Complex as Simple Collision

Sets bUseComplexAsSimpleCollision in a procedural mesh



Inputs

In Exec	
Mesh Procedural Mesh Component Object Reference	
Value Boolean	

Outputs

Out Exec	

Documentation > Mesh Ops Plugin BPLibrary > *Simplify Mesh Data*

Simplify Mesh Data

Simplifies/Reduces a Mesh Data, targeting TriCount number of triangles. Stitches mesh together via StitchDist distance, 0.0 skips stitching

Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Tri Count Integer	
Stitch Dist Float	

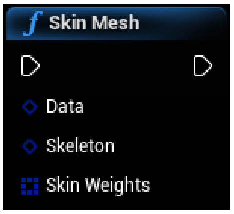
Outputs

Out Exec	
Return Value Mesh Data Structure	Simplifies/Reduces a Mesh Data, targeting TriCount number of triangles. Stitches mesh together via StitchDist distance, 0.0 skips stitching

Documentation > Mesh Ops Plugin BPLibrary > *Skin Mesh*

Skin Mesh

EXPERIMENTAL Adds a skeleton to the MeshData. This merely assigns skinning data to Data, actual skinning calculations have to be done beforehand, e.g. with GenerateSkinWeights().



Inputs

In `Exec`	
Data `Mesh Data Structure (by ref)`	
Skeleton `Mesh Skeleton Structure (by ref)`	
Skin Weights `Array of Skin Weight Structures`	

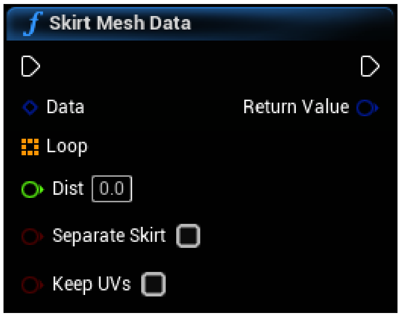
Outputs

Out `Exec`	

Documentation > Mesh Ops Plugin BPLibrary > *Skirt Mesh Data*

Skirt Mesh Data

Generates a 'skirt' around a Mesh Data, skirt is not welded to the mesh. (creates new vertices) Will generate a loop from exposed edges if Loop is left empty.



Inputs

In `Exec`	
Data `Mesh Data Structure (by ref)`	
Loop `Array of Vectors`	
Dist `Float`	
Separate Skirt `Boolean`	
Keep UVs `Boolean`	

Outputs

Out `Exec`	
Return Value `Mesh Data Structure`	Generates a 'skirt' around a Mesh Data, skirt is not welded to the mesh. (creates new vertices) Will generate a loop from exposed edges if Loop is left empty.

Documentation > Mesh Ops Plugin BPLibrary > *Slice Loop*

Slice Loop

Cuts a Loop into two pieces along the line formed from Start to End, must have one entry and one exit intersections.

Inputs

In Exec	
Loop Array of Vectors	
Start Vector	
End Vector	

Outputs

Out Exec	
Return Value Array of Loop Structures	Cuts a Loop into two pieces along the line formed from Start to End, must have one entry and one exit intersections.

Documentation > Mesh Ops Plugin BPLibrary > Slice Mesh Data

Slice Mesh Data

Slice Mesh Data with a plane, code adapted from Slice Procedural Mesh code in KismetProceduralMeshLibrary from Unreal.

Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Center Vector	
Normal Vector	
Create Other Half Boolean	
Append Caps Boolean	

Outputs

Out Exec	
Other Half Mesh Data Structure	

Documentation > Mesh Ops Plugin BPLibrary > Slice Mesh Data Storage

Slice Mesh Data Storage

Slice Mesh Data with a plane, code adapted from Slice Procedural Mesh code in KismetProceduralMeshLibrary from Unreal.

Inputs

In Exec	

Data Mesh Data Storage Structure (by ref)	
Center Vector	
Normal Vector	
Create Other Half Boolean	
Append Caps Boolean	

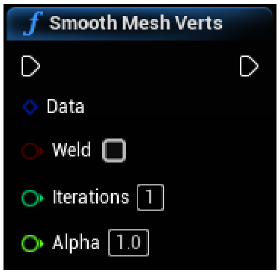
Outputs

Out Exec	
Other Half Mesh Data Storage Structure	

Documentation > Mesh Ops Plugin BPLibrary > *Smooth Mesh Verts*

Smooth Mesh Verts

Smooths out vertices across mesh, optionally welds beforehand if Weld is true.



Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Weld Boolean	
Iterations Integer	
Alpha Float	

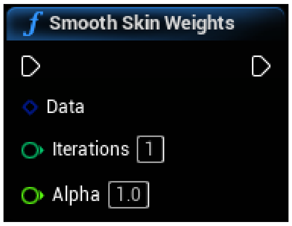
Outputs

Out Exec	

Documentation > Mesh Ops Plugin BPLibrary > *Smooth Skin Weights*

Smooth Skin Weights

EXPERIMENTAL Smooth skin weights for all verts with their neighbors.



Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Iterations Integer	
Alpha Float	

Outputs

Out Exec	

Documentation > Mesh Ops Plugin BPLibrary > Snap Point to Mesh

Snap Point to Mesh

Projects a Point onto a Mesh Data, which iterates over every triangle until a suitable position is found within Tol distance. If Tol is not set, the closest point is returned.

Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Point Vector	
Fast Boolean	
Tol Float	

Outputs

Out Exec	
Success Boolean	
Tri Index Integer	
Return Value Vector	Projects a Point onto a Mesh Data, which iterates over every triangle until a suitable position is found within Tol distance. If Tol is not set, the closest point is returned.

Documentation > Mesh Ops Plugin BPLibrary > Soften Normals

Soften Normals

Softens any hard edges, doesn't weld. Alpha lerps to the new value. Any split normal vertices should all point the same direction after, when using an Alpha of 1. AvgAlpha blends normals with neighboring verts.

Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Alpha Float	
Avg Alpha Float	

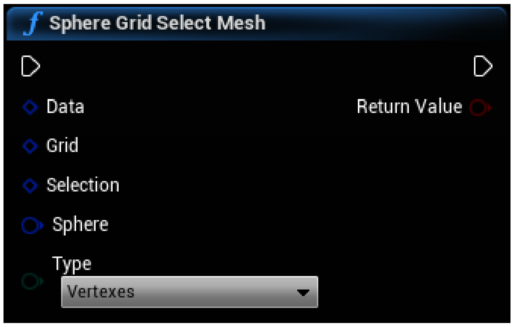
Outputs

Out Exec	

[Documentation](#) > [Mesh Ops Plugin BPLibrary](#) > *Sphere Grid Select Mesh*

Sphere Grid Select Mesh

Selects vertexes of a mesh in a sphere for use in other functions, like TransformSelection(). Utilizes a Localized Grid for high poly meshes.



Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Grid Localized Grid Structure (by ref)	
Selection Mesh Selection Structure (by ref)	
Sphere Sphere Brush Structure	
Type SelectionType Enum	

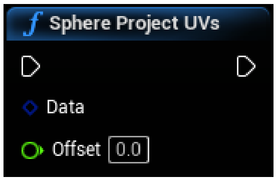
Outputs

Out Exec	
Return Value Boolean	Selects vertexes of a mesh in a sphere for use in other functions, like TransformSelection(). Utilizes a Localized Grid for high poly meshes.

[Documentation](#) > [Mesh Ops Plugin BPLibrary](#) > *Sphere Project UVs*

Sphere Project UVs

Projects UVs as a sphere onto a mesh.



Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Offset Float	

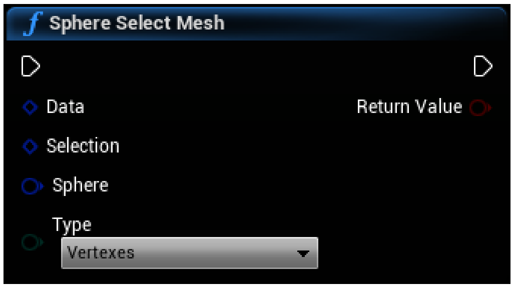
Outputs

Out Exec	

Documentation > Mesh Ops Plugin BPLibrary > *Sphere Select Mesh*

Sphere Select Mesh

Selects vertexes of a mesh in a sphere for use in other functions, like TransformSelection().



Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Selection Mesh Selection Structure (by ref)	
Sphere Sphere Brush Structure	
Type SelectionType Enum	

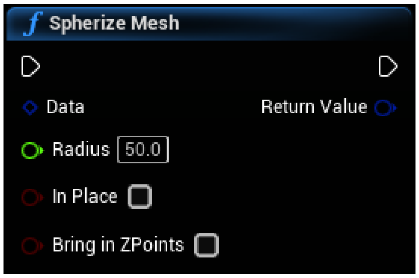
Outputs

Out Exec	
Return Value Boolean	Selects vertexes of a mesh in a sphere for use in other functions, like TransformSelection().

Documentation > Mesh Ops Plugin BPLibrary > *Spherize Mesh*

Spherize Mesh

Turns a mesh into a sphere with its center at (0.0,0.0,0.0), should only be used for convex meshes.



Inputs

In `Exec`	
Data `Mesh Data Structure (by ref)`	
Radius `Float`	
In Place `Boolean`	
Bring in ZPoints `Boolean`	

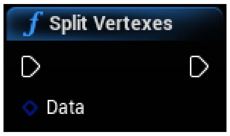
Outputs

Out `Exec`	
Return Value `Mesh Data Structure`	Turns a mesh into a sphere with its center at (0.0,0.0,0.0), should only be used for convex meshes.

Documentation > Mesh Ops Plugin BPLibrary > Split Vertexes

Split Vertexes

Adds vertices at each shared vertex, so that each triangle has its own unique vertices/normals. Used in SetNormalsToFace



Inputs

In `Exec`	
Data `Mesh Data Structure (by ref)`	

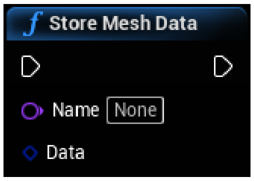
Outputs

Out `Exec`	

Documentation > Mesh Ops Plugin BPLibrary > Store Mesh Data

Store Mesh Data

Stores Mesh Data temporarily. Gets cleared when the engine/game closes.



Inputs

In Exec	
Name Name	
Data Mesh Data Structure (by ref)	

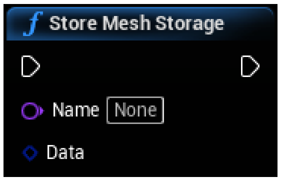
Outputs

Out Exec	

Documentation > Mesh Ops Plugin BPLibrary > *Store Mesh Storage*

Store Mesh Storage

Stores Mesh Data Storage temporarily. Gets cleared when the engine/game closes.



Inputs

In Exec	
Name Name	
Data Mesh Data Storage Structure (by ref)	

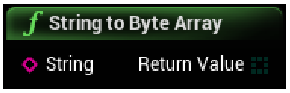
Outputs

Out Exec	

Documentation > Mesh Ops Plugin BPLibrary > *String to Byte Array*

String to Byte Array

String to Bytes



Inputs

String String (by ref)	

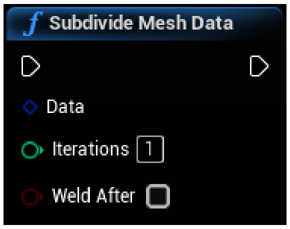
Outputs

Return Value `Array of Bytes`	String to Bytes

Documentation > Mesh Ops Plugin BPLibrary > *Subdivide Mesh Data*

Subdivide Mesh Data

Subdivides a mesh. Due to the technique used, welding may be needed afterward.



Inputs

In `Exec`	
Data `Mesh Data Structure (by ref)`	
Iterations `Integer`	
Weld After `Boolean`	

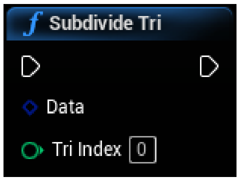
Outputs

Out `Exec`	

Documentation > Mesh Ops Plugin BPLibrary > *Subdivide Tri*

Subdivide Tri

Subdivides a single triangle starting at TriIndex into 4 triangles.



Inputs

In `Exec`	
Data `Mesh Data Structure (by ref)`	
Tri Index `Integer`	

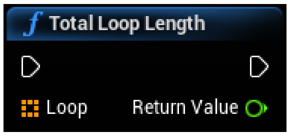
Outputs

Out `Exec`	

Documentation > Mesh Ops Plugin BPLibrary > *Total Loop Length*

Total Loop Length

Gets total length of a Loop, i.e. total unit distance between each point.



Inputs

In Exec	
Loop Array of Vectors	

Outputs

Out Exec	
Return Value Float	Gets total length of a Loop, i.e. total unit distance between each point.

Documentation > Mesh Ops Plugin BPLibrary > Transform Loop

Transform Loop

Transforms every point in a Loop.

Inputs

In Exec	
Loop Array of Vectors	
Transform Transform	
In Place Boolean	
Pivot Vector	

Outputs

Out Exec	
Return Value Array of Vectors	Transforms every point in a Loop.

Documentation > Mesh Ops Plugin BPLibrary > Transform Loop Struct

Transform Loop Struct

Transforms every point in a Loop Structure.

Inputs

In Exec	

Loop Loop Structure (by ref)	
Transform Transform	
In Place Boolean	
Pivot Vector	

Outputs

Out Exec	
Return Value Loop Structure	Transforms every point in a Loop Structure.

Documentation > *Mesh Ops Plugin BPLibrary* > *Transform Mesh Data*

Transform Mesh Data

Transforms Mesh Data.

Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Transform Transform	
In Place Boolean	edits input Data instead of the output.
Pivot Vector	

Outputs

Out Exec	
Return Value Mesh Data Structure	

Documentation > *Mesh Ops Plugin BPLibrary* > *Transform Selection*

Transform Selection

Transforms vertexes within Selection arount Pivot (which should usually be Selection.Center)

Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Selection Mesh Selection Structure (by ref)	
Transform	

Transform	
Pivot Vector	

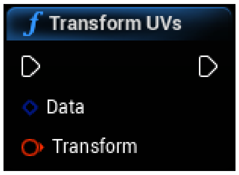
Outputs

Out Exec	

Documentation > Mesh Ops Plugin BPLibrary > Transform UVs

Transform UVs

Transforms uvs for the entire mesh, similar to Maya's UV Editor behavior. You can Translate, Scale, and Rotate.



Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Transform Transform	

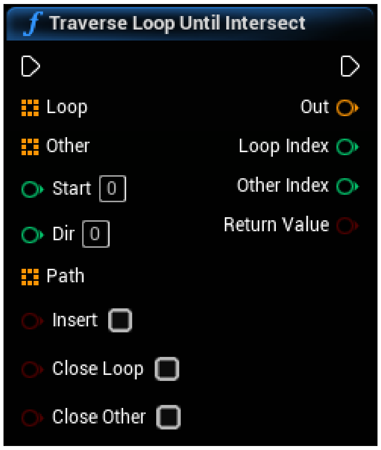
Outputs

Out Exec	

Documentation > Mesh Ops Plugin BPLibrary > Traverse Loop Until Intersect

Traverse Loop Until Intersect

Goes along Loop from Start going forward if Dir is one, and backward if negative one. It stops when it has found an intersection with Other. Optionally inserts the new point into Other at OtherIndex.



Inputs

In Exec	
Loop Array of Vectors	

Other `Array of Vectors`	
Start `Integer`	
Dir `Integer`	
Path `Array of Vectors`	
Insert `Boolean`	
Close Loop `Boolean`	
Close Other `Boolean`	

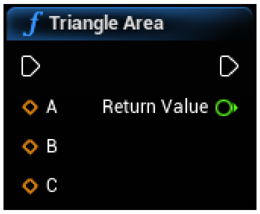
Outputs

Out `Exec`	
Out `Vector`	
Loop Index `Integer`	
Other Index `Integer`	
Return Value `Boolean`	Goes along Loop from Start going forward if Dir is one, and backward if negative one. It stops when it has found an intersection with Other. Optionally inserts the new point into Other at OtherIndex.

Documentation > *Mesh Ops Plugin BPLibrary* > *Triangle Area*

Triangle Area

Gets an estimate of a triangle's size, useful for relative comparisons with other triangles.



Inputs

In `Exec`	
A `Vector (by ref)`	
B `Vector (by ref)`	
C `Vector (by ref)`	

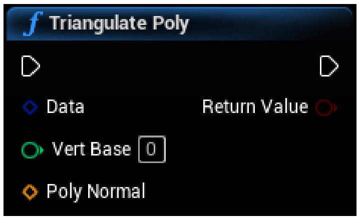
Outputs

Out `Exec`	
Return Value `Float`	Gets an estimate of a triangle's size, useful for relative comparisons with other triangles.

Documentation > *Mesh Ops Plugin BPLibrary* > *Triangulate Poly*

Triangulate Poly

Triangulate polygonal vertices, code adapted from Slice Procedural Mesh code in KismetProceduralMeshLibrary from Unreal.



Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Vert Base Integer	
Poly Normal Vector (by ref)	

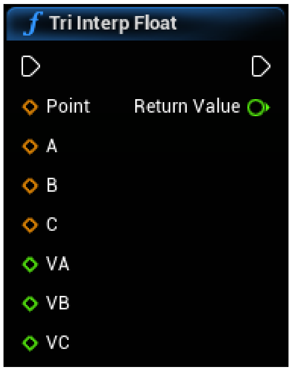
Outputs

Out Exec	
Return Value Boolean	Triangulate polygonal vertices, code adapted from Slice Procedural Mesh code in KismetProceduralMeshLibrary from Unreal.

Documentation > *Mesh Ops Plugin BPLibrary* > *Tri Interp Float*

Tri Interp Float

Gets a float value based on position within a triangle.



Inputs

In Exec	
Point Vector (by ref)	
A Vector (by ref)	
B Vector (by ref)	
C Vector (by ref)	
VA	

Float (by ref)	
VB Float (by ref)	
VC Float (by ref)	

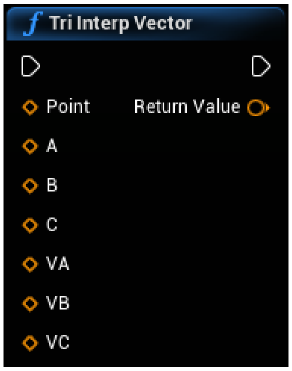
Outputs

Out Exec	
Return Value Float	Gets a float value based on position within a triangle.

[Documentation](#) > [Mesh Ops Plugin BPLibrary](#) > *Tri Interp Vector*

Tri Interp Vector

Gets a vector value based on position within a triangle.



Inputs

In Exec	
Point Vector (by ref)	
A Vector (by ref)	
B Vector (by ref)	
C Vector (by ref)	
VA Vector (by ref)	
VB Vector (by ref)	
VC Vector (by ref)	

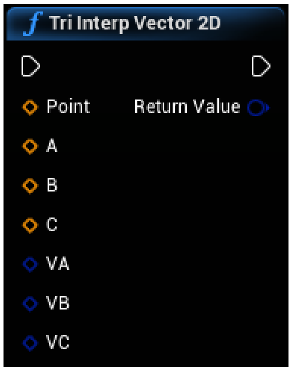
Outputs

Out Exec	
Return Value Vector	Gets a vector value based on position within a triangle.

[Documentation](#) > [Mesh Ops Plugin BPLibrary](#) > *Tri Interp Vector 2D*

Tri Interp Vector 2D

Gets a vector2D value based on position within a triangle.



Inputs

In Exec	
Point Vector (by ref)	
A Vector (by ref)	
B Vector (by ref)	
C Vector (by ref)	
VA Vector 2D Structure (by ref)	
VB Vector 2D Structure (by ref)	
VC Vector 2D Structure (by ref)	

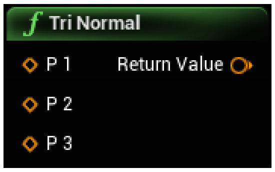
Outputs

Out Exec	
Return Value Vector 2D Structure	Gets a vector2D value based on position within a triangle.

Documentation > *Mesh Ops Plugin BPLibrary* > *Tri Normal*

Tri Normal

Normal direction of a triangle, used internally.



Inputs

P 1 Vector (by ref)	
P 2 Vector (by ref)	
P 3 Vector (by ref)	

Outputs

Return Value Vector	Normal direction of a triangle, used internally.

Documentation > Mesh Ops Plugin BPLibrary > *Tri UVTo Bary*

Tri UVTo Bary

Gets barycentric coordinates from a UV coordinate on the triangle starting at TriIndex.

Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Tri Index Integer	
UV Vector 2D Structure	

Outputs

Out Exec	
Return Value Vector	Gets barycentric coordinates from a UV coordinate on the triangle starting at TriIndex.

Documentation > Mesh Ops Plugin BPLibrary > *Tri UVTo Location*

Tri UVTo Location

Gets the point at a UV location. This should be valid for any UV coordinate within the triangle. Triangle starts at TriIndex.

Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Tri Index Integer	
UV Vector 2D Structure	

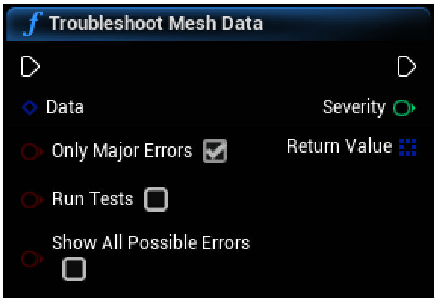
Outputs

Out Exec	
Return Value Vector	Gets the point at a UV location. This should be valid for any UV coordinate within the triangle. Triangle starts at TriIndex.

Documentation > Mesh Ops Plugin BPLibrary > *Troubleshoot Mesh Data*

Troubleshoot Mesh Data

Outputs Errors and Notes about problems a MeshData may have, such as missing tris, normals, uvs, etc. ShowAllPossibleErrors outputs everything, regardless if it is relevant, for debugging.



Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Only Major Errors Boolean	
Run Tests Boolean	
Show All Possible Errors Boolean	

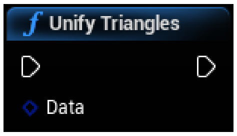
Outputs

Out Exec	
Severity Integer	
Return Value Array of Mesh Ops Error Structures	Outputs Errors and Notes about problems a MeshData may have, such as missing tris, normals, uvs, etc. ShowAllPossibleErrors outputs everything, regardless if it is relevant, for debugging.

Documentation > Mesh Ops Plugin BPLibrary > Unify Triangles

Unify Triangles

Orders Tris Array in a nice linear fashion. e.g. Tris = {0,1,2,3,4,5,6,7,8} All vertexes are split to do this. Used to correctly interp between different meshes with InterpMoveMeshDataTo().



Inputs

In Exec	
Data Mesh Data Structure (by ref)	

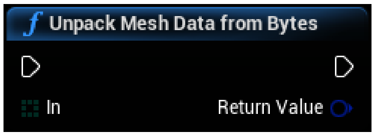
Outputs

Out Exec	

Documentation > Mesh Ops Plugin BPLibrary > Unpack Mesh Data from Bytes

Unpack Mesh Data from Bytes

De-Serializes Bytes into Mesh Data



Inputs

In `Exec`	
In `Array of Bytes`	

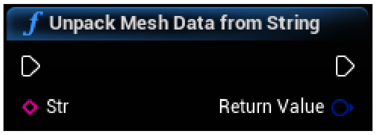
Outputs

Out `Exec`	
Return Value `Mesh Data Structure`	De-Serializes Bytes into Mesh Data

[Documentation](#) > [Mesh Ops Plugin BPLibrary](#) > *Unpack Mesh Data from String*

Unpack Mesh Data from String

De-Serializes Serialized Mesh Data string into Mesh Data



Inputs

In `Exec`	
Str `String (by ref)`	

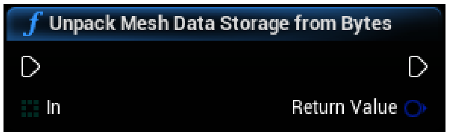
Outputs

Out `Exec`	
Return Value `Mesh Data Structure`	De-Serializes Serialized Mesh Data string into Mesh Data

[Documentation](#) > [Mesh Ops Plugin BPLibrary](#) > *Unpack Mesh Data Storage from Bytes*

Unpack Mesh Data Storage from Bytes

De-Serializes Bytes into Mesh Data Storage



Inputs

In `Exec`	
In `Array of Bytes`	

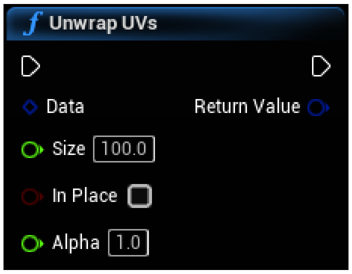
Outputs

Out `Exec`	
Return Value `Mesh Data Storage Structure`	De-Serializes Bytes into Mesh Data Storage

Documentation > Mesh Ops Plugin BPLibrary > Unwrap UVs

Unwrap UVs

Unwraps mesh verts to be as they are in UV space, it can be animated using Alpha.



Inputs

In `Exec`	
Data `Mesh Data Structure (by ref)`	
Size `Float`	
In Place `Boolean`	
Alpha `Float`	

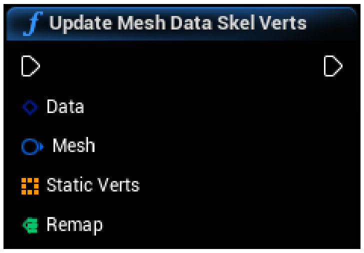
Outputs

Out `Exec`	
Return Value `Mesh Data Structure`	Unwraps mesh verts to be as they are in UV space, it can be animated using Alpha.

Documentation > Mesh Ops Plugin BPLibrary > Update Mesh Data Skel Verts

Update Mesh Data Skel Verts

EXPERIMENTAL



Inputs

In `Exec`	
Data	

Mesh Data Structure (by ref)	
Mesh Skeletal Mesh Component Object Reference	
Static Verts Array of Vectors	
Remap Map of Integers to Integers	

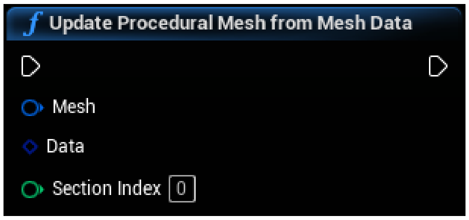
Outputs

Out Exec	

Documentation > Mesh Ops Plugin BPLibrary > Update Procedural Mesh from Mesh Data

Update Procedural Mesh from Mesh Data

Updates a Procedural mesh from Mesh Data, updating is usually faster and does not allow modifying the Tris array.



Inputs

In Exec	
Mesh Procedural Mesh Component Object Reference	
Data Mesh Data Structure (by ref)	
Section Index Integer	

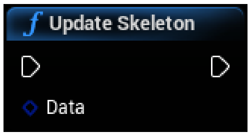
Outputs

Out Exec	

Documentation > Mesh Ops Plugin BPLibrary > Update Skeleton

Update Skeleton

EXPERIMENTAL Updates vertexes and normals in the mesh according to how its skeleton has transformed.



Inputs

In Exec	
Data Mesh Data Structure (by ref)	

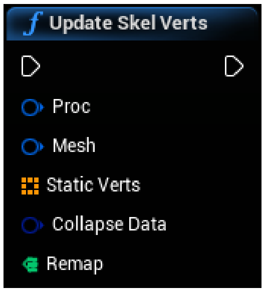
Outputs

Out Exec	

[Documentation](#) > [Mesh Ops Plugin BPLibrary](#) > *Update Skel Verts*

Update Skel Verts

EXPERIMENTAL



Inputs

In Exec	
Proc Procedural Mesh Component Object Reference	
Mesh Skeletal Mesh Component Object Reference	
Static Verts Array of Vectors	
Collapse Data Collapse Verts Structure	
Remap Map of Integers to Integers	

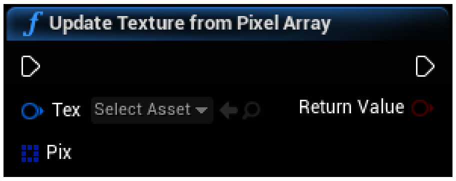
Outputs

Out Exec	

[Documentation](#) > [Mesh Ops Plugin BPLibrary](#) > *Update Texture from Pixel Array*

Update Texture from Pixel Array

Uses memcpy to fill a texture2D from CreateTextureFromPixelArray().



Inputs

In Exec	
Tex Texture 2D Object Reference	
Pix Array of Linear Color Structures	

Outputs

Out Exec	
Return Value Boolean	Uses memcpy to fill a texture2D from CreateTextureFromPixelArray().

Documentation > *Mesh Ops Plugin BPLibrary* > *Voxelize Mesh Data*

Voxelize Mesh Data

EXPERIMENTAL Creates Voxel Data from Mesh Data.

Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Cell Size Float	

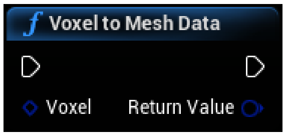
Outputs

Out Exec	
Return Value Voxel Data Structure	EXPERIMENTAL Creates Voxel Data from Mesh Data.

Documentation > *Mesh Ops Plugin BPLibrary* > *Voxel to Mesh Data*

Voxel to Mesh Data

EXPERIMENTAL Creates Mesh Data from Voxel Data.



Inputs

In Exec	
Voxel Voxel Data Structure (by ref)	

Outputs

Out Exec	
Return Value Mesh Data Structure	EXPERIMENTAL Creates Mesh Data from Voxel Data.

Documentation > *Mesh Ops Plugin BPLibrary* > *Walk Pixels*

Walk Pixels

Walks along non-transparent pixels in a transparent image, and cuts out the first complete shape as a loop of vertexes.

Inputs

In `Exec`	
Pix `Array of Linear Color Structures`	
Width `Integer`	Image width, if using a Render Target, get Size X variable from it.
W Width `Float`	Walk Width, world size of generated loop. If shape is 50px and Width is 100px, with a wWidth of 100.0, the shape loop will be 50.0 units wide.
Out Data `Mesh Data Structure (by ref)`	
Min Dist `Float`	

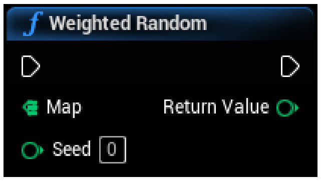
Outputs

Out `Exec`	
Return Value `Boolean`	

[Documentation](#) > [Mesh Ops Plugin BPLibrary](#) > Weighted Random

Weighted Random

Gets a weighted random value from a Map. The result is a key from the map which should represent the chosen result as an index in an array.



Inputs

In `Exec`	
Map `Map of Integers to Integers`	
Seed `Integer`	

Outputs

Out `Exec`	
Return Value `Integer`	Gets a weighted random value from a Map. The result is a key from the map which should represent the chosen result as an index in an array.

[Documentation](#) > [Mesh Ops Plugin BPLibrary](#) > Weighted Tri Map

Weighted Tri Map

Generates a Map used to get a weighted random triangle index. The key of the map represents indexes in the Tris array, and the value of the map at each key represents its weight by relative size.

Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Map Map of Integers to Integers	
Force Boolean	
Adjust Float	
Bias Vector	

Outputs

Out Exec	
Return Value Boolean	Generates a Map used to get a weighted random triangle index. The key of the map represents indexes in the Tris array, and the value of the map at each key represents its weight by relative size.

Documentation > Mesh Ops Plugin BPLibrary > *Weighted Tri Map from Selection*

Weighted Tri Map from Selection

Generates a Map from Selection used to get a weighted random triangle index. The key of the map represents indexes in the Tris array, and the value of the map at each key represents its weight by relative size.

Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Selection Mesh Selection Structure (by ref)	
Map Map of Integers to Integers	
Force Boolean	
Adjust Float	
Bias Vector	

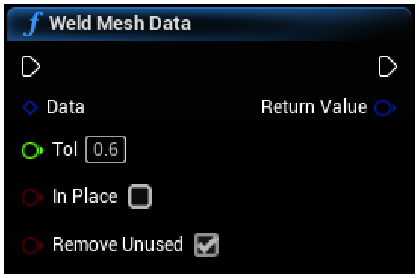
Outputs

Out Exec	
Return Value Boolean	Generates a Map from Selection used to get a weighted random triangle index. The key of the map represents indexes in the Tris array, and the value of the map at each key represents its weight by relative size.

Documentation > Mesh Ops Plugin BPLibrary > *Weld Mesh Data*

Weld Mesh Data

Welds vertices together i.e. removes hard edges, split vertices/normals, etc. RemoveUnused makes sure there's no extra unused vertex data left over.



Inputs

In Exec	
Data Mesh Data Structure (by ref)	
Tol Float	
In Place Boolean	
Remove Unused Boolean	

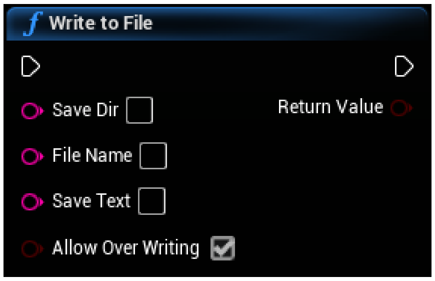
Outputs

Out Exec	
Return Value Mesh Data Structure	Welds vertices together i.e. removes hard edges, split vertices/normals, etc. RemoveUnused makes sure there's no extra unused vertex data left over.

Documentation > Mesh Ops Plugin BPLibrary > Write to File

Write to File

Writes String to file.



Inputs

In Exec	
Save Dir String	
File Name String	
Save Text String	
Allow Over Writing Boolean	

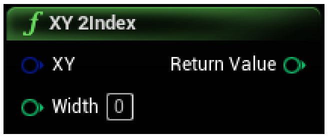
Outputs

Out Exec	
Return Value Boolean	Writes String to file.

Documentation > Mesh Ops Plugin BPLibrary > *XY 2Index*

XY 2Index

2D Coord to pixel index.



Inputs

XY Coord Structure	coord with only X and Y
Width Integer	image width for pixel array, if using a Render Target, get Size X variable from it.

Outputs

Return Value Integer	