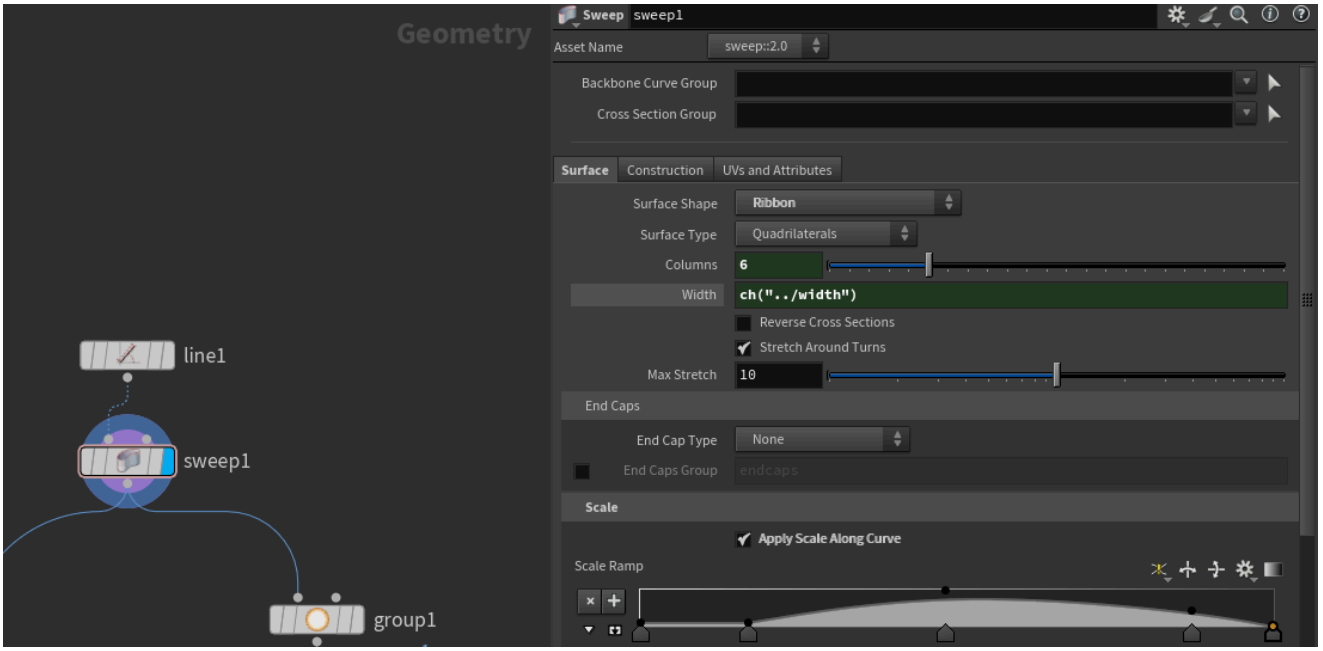
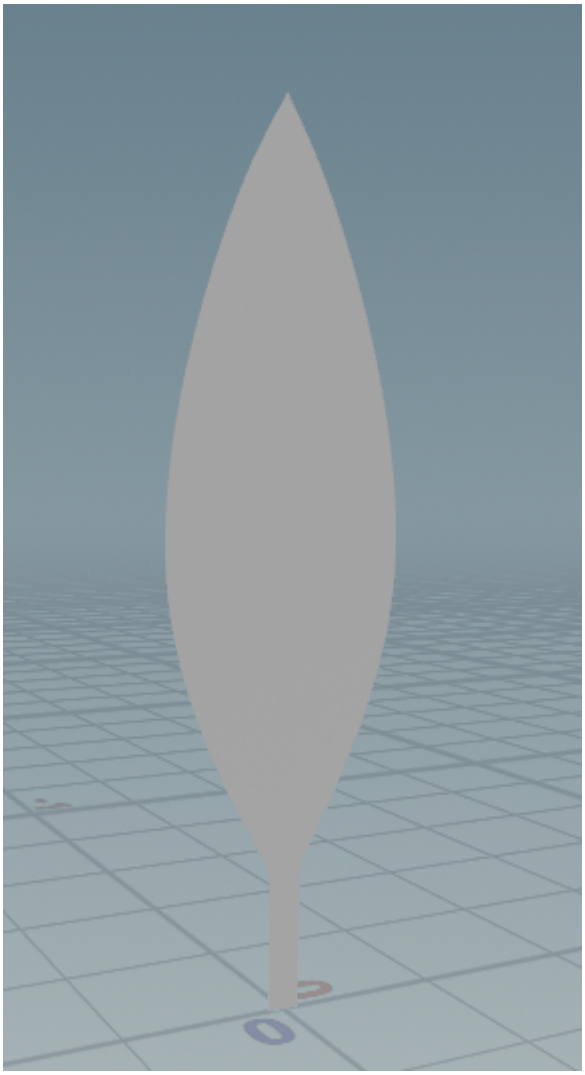
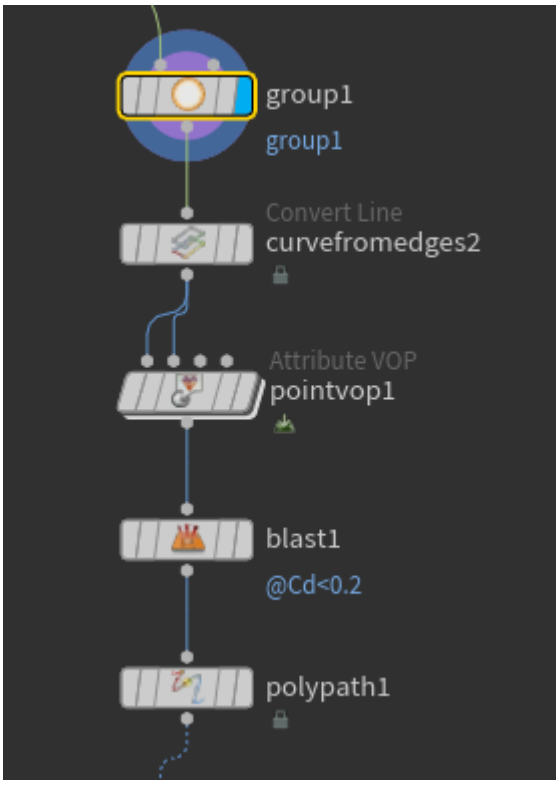
Foliage Maker

Leaves

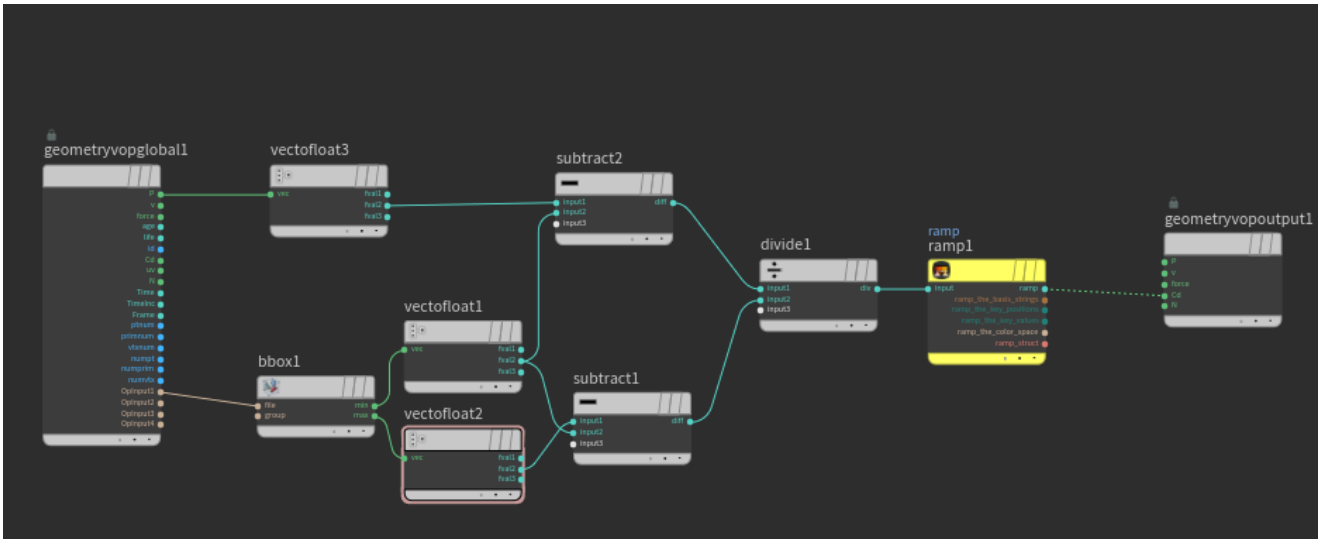




利用line 和 sweep 做出leaves的大型。ps：利用sweep里的scale ramp做leaves。

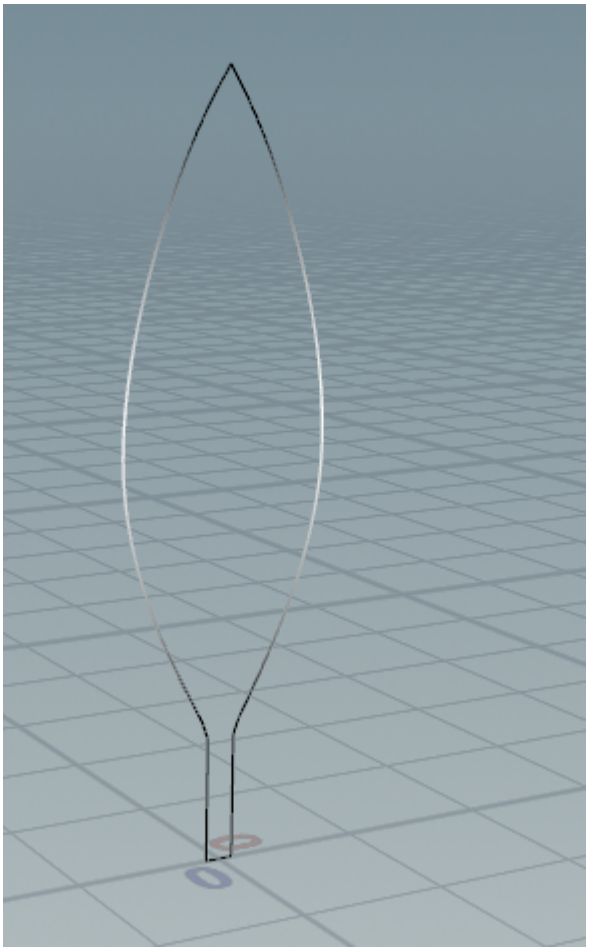


用group点提取出叶子的外形的线段，在利用curvefromedges节点将线段转为curve状态。



$$P - min/(max - min)$$

在vop里面对Y轴进行归一化，然后ramp进行操控输出到cd，画一个mask出来。



利用blast筛选出需要的区域

```
@Cd <0.2
```

PolyPath geometry node
PolyPath 几何节点

Cleans up topology of polygon curves.

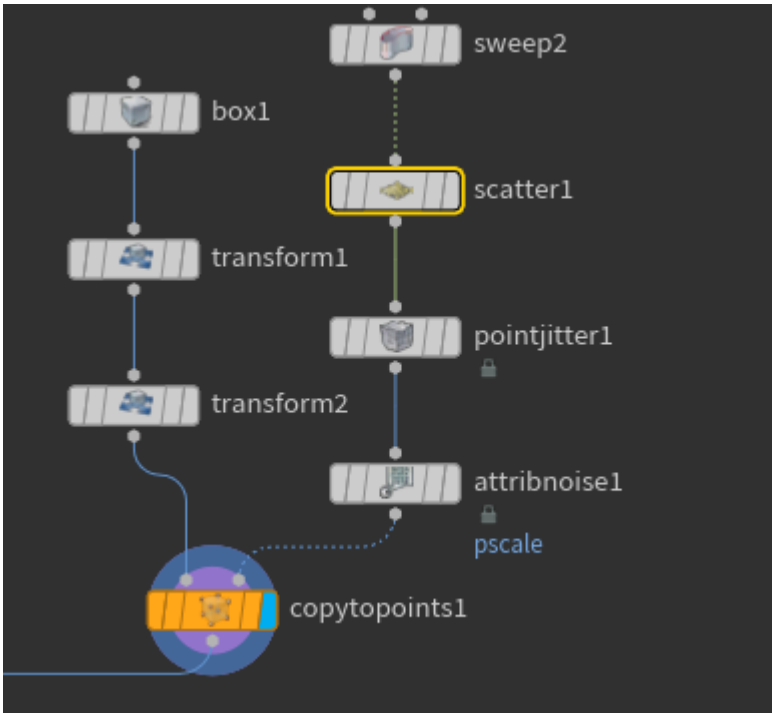
清理多边形曲线的拓扑。

On this page **本页内容**	• Overview 概述 • Parameters 参数
Since 自从	16.0

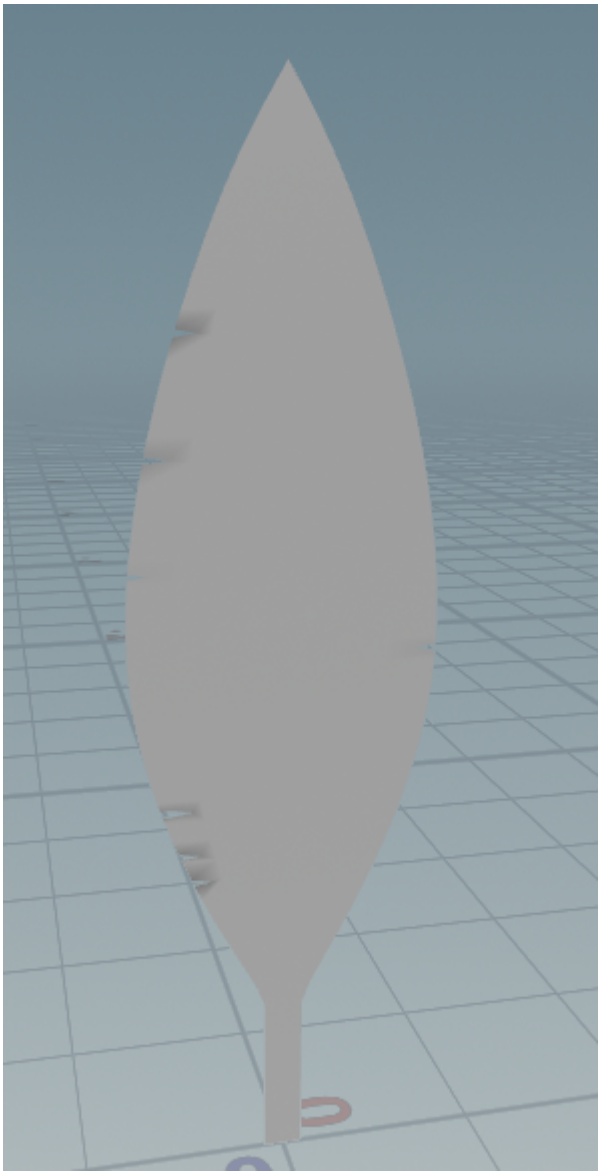
Overview 概述

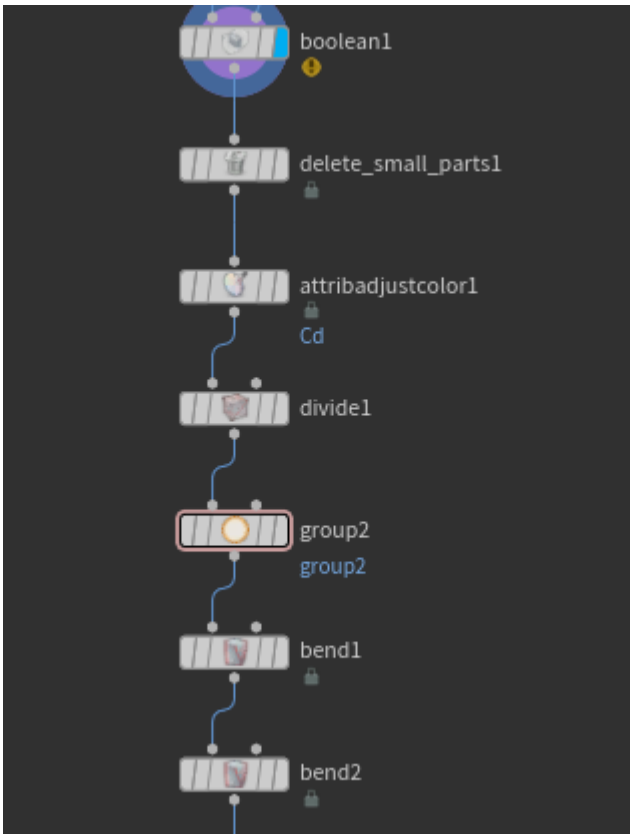
This node replaces all edges with polygon curves that continue until any point connected to three or more other points is reached. By default, this will reduce any edges that occur more than once to a single edge, and will remove any edges from a point to itself. It also has various options for cleaning up topology in other ways.

此节点将所有边替换为多边形曲线，直至到达任何与三个或更多其他点相连的点。默认情况下，这会将出现多次的边简化为一条边，并移除从某个点到自身的所有边。它还提供各种选项，用于以其他方式清理拓扑。

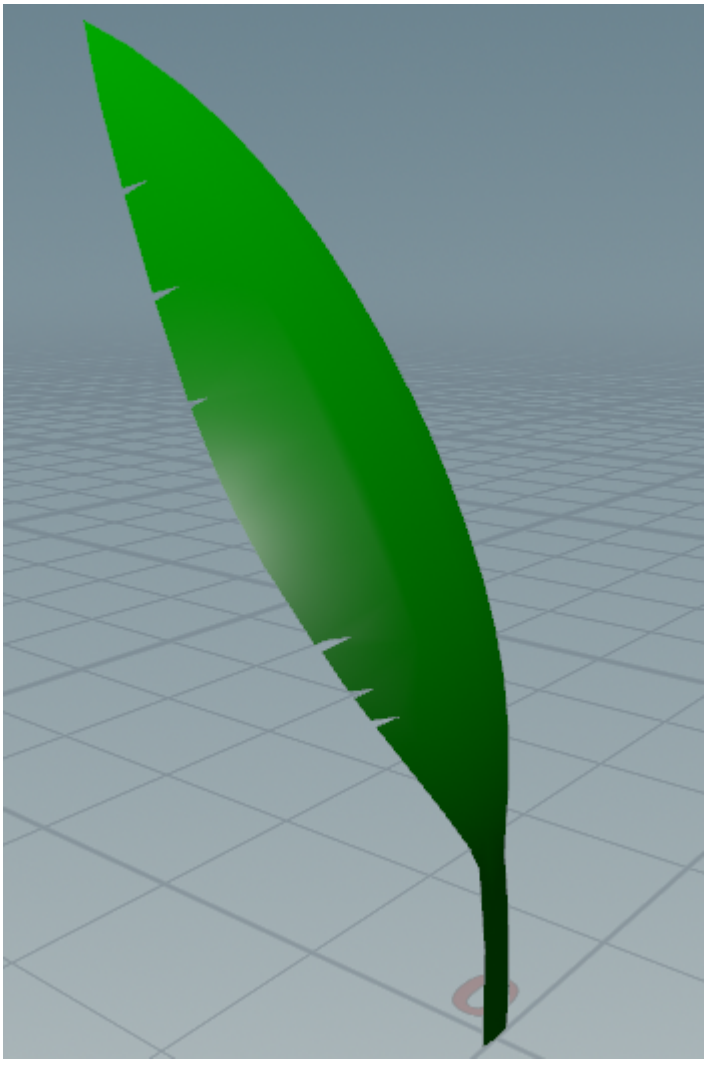


sweep后进行scatter，然后利用pointjitter对scatter出来的点进行rand，然后对这些点进行pscale的noise操作，然后捏一个叶子破损的box，然后copy to point出来，在跟原本的leaves进行boolean

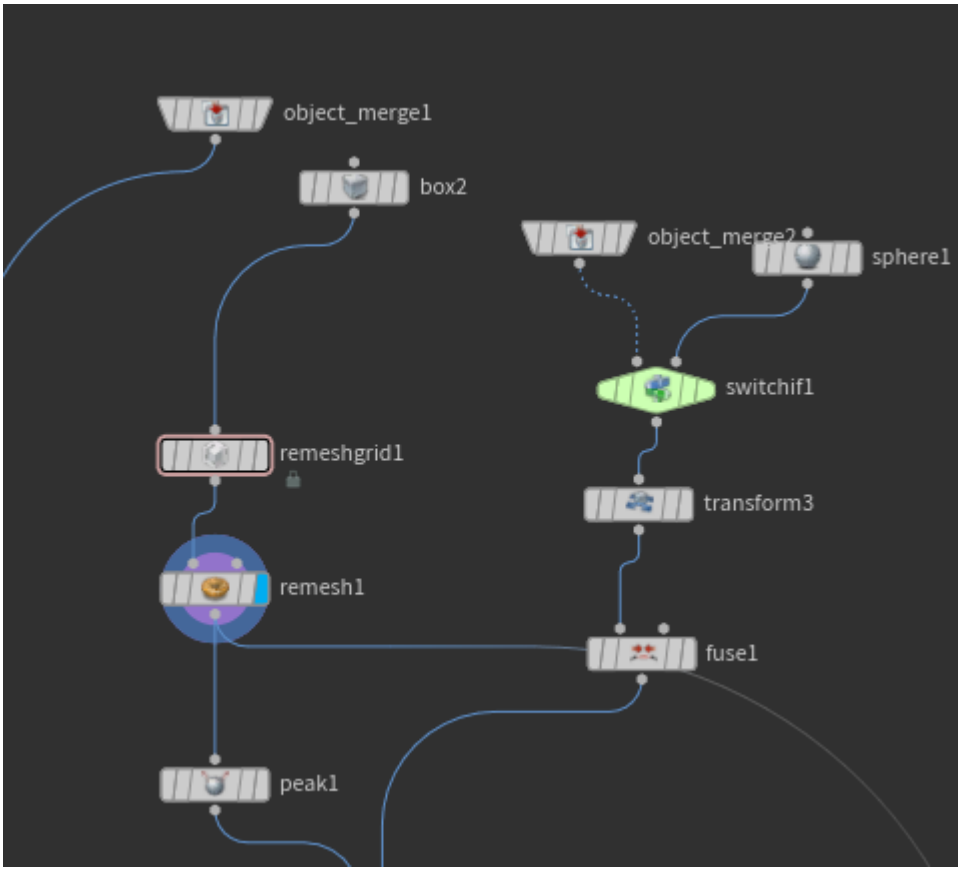




后面就是简单的上个色，利用bend节点对leaves做弯曲。在计算一下normal以及polyreduce。



Ivy



这边用box暂时替代objectmerge进行展示效果，首先voxel mesh 一下，目的就是清理input进来的模型

Remesh to Grid geometry node
重新划分网格 几何节点

Rebuilds polygonal topology to straighten edges, close small holes, and remove interior geometry.

重建多边形拓扑以拉直边缘、关闭小孔并移除内部几何体。

On this page **本页内容**

- Overview 概述
- Parameters 参数
- Examples 示例

Since 自从 19.0

Overview
概述

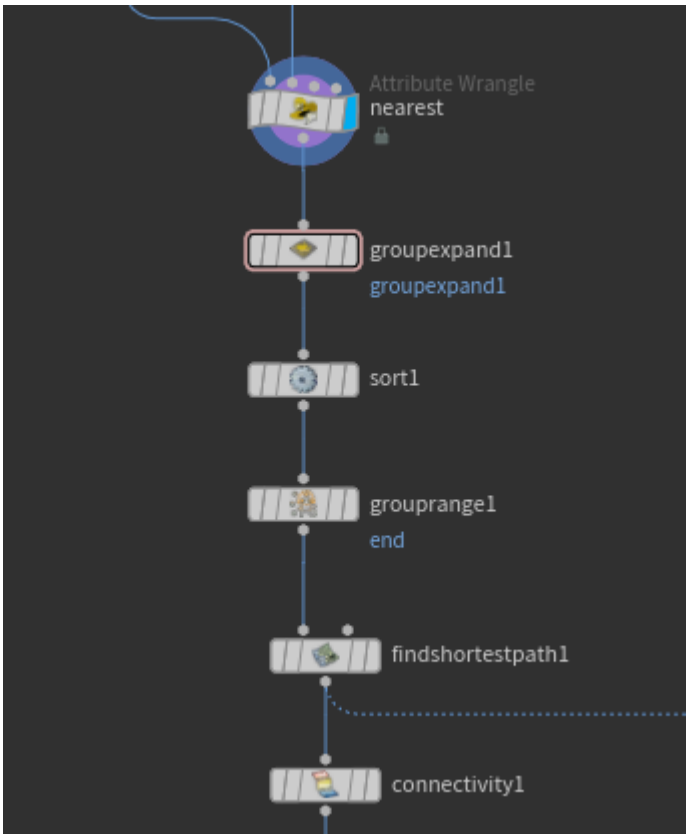
This node converts polygonal geometry into a volume (VDB), then back into polygons, as a way to recompute the polygon layout. This can make the polygonal structure more regular, as well as more or less dense. It also eliminates any interior polygons, and can close small holes.

此节点将多边形几何体转换为体积 (VDB)，然后再转换为多边形，以此重新计算多边形布局。这可以使多边形结构更加规则，密度也更高或更低。它还可以消除任何内部多边形，并封闭小孔。

This is often useful to clean up irregular polgyons created by photogrammetry.

这通常有助于清理摄影测量产生的不规则多面体。

另一边input进来的sphere就是用来控制ivy的朝向。



Wrangle(nearest)

```
vector pos2 = point(1, "P", 0);

int closetP = nearpoint(0 , pos2);

if( closetP == @ptnum)
{
   @Cd = 0;
   setpointgroup(0 , "Begin", @ptnum, 1, "set");
}
```

这边首先先读取input进来的sphere的第0个点的position赋值给pos2。

`point` 观点

Reads a point attribute value from a geometry.

从几何图形中读取点属性值。

```
                  <geometry>   string           int
<type> point(geometry, attribute_name, pointnumber)
                    <geometry>   string           int
<type>[] point(geometry, attribute_name, pointnumber)
```

Show/hide arguments 显示/隐藏参数

`<geometry>`

When running in the context of a node (such as a wrangle SOP), this argument can be an integer representing the input number (starting at 0) to read the geometry from.

当在节点（例如 wrangle SOP）的上下文中运行时，此参数可以是表示从中读取几何的输入数字（从 0 开始）的整数。

Alternatively, the argument can be a string specifying a geometry file (for example, a `.bgeo`) to read from. When running inside Houdini, this can be an `op:/path/to/sop` reference.

或者，参数可以是指定要读取的几何文件（例如 `.bgeo`）的字符串。在 Houdini 中运行时，可以是 `op:/path/to/sop` 引用。

`attribute_name`

The name of the attribute (or intrinsic) to read.

要读取的属性（或内在属性）的名称。

`pointnumber`

The point number to read the attribute on.

读取属性的点编号。

然后利用nearpoint 表达式寻找距离pos2最近的input进来的第一个box的点。

```
                  <geometry>   vector
int  nearpoint(geometry, pt)
                  <geometry>   vectorfloat
int  nearpoint(geometry, pt, maxdist)
                  <geometry>   string      vector
int  nearpoint(geometry, ptgroup, pt)
                  <geometry>   string      vectorfloat
int  nearpoint(geometry, ptgroup, pt, maxdist)
```

Returns the number of the closest point on the geometry. This will only search against points, not the surface locations of the geometry. Use `xyzdist` to find the closest point on surfaces or curves.

返回几何体上最近点的编号。此方法仅针对点进行搜索，而不针对几何体的表面位置。使用 `xyzdist` 查找曲面或曲线上的最近点。

-1 is returned if no point is found in the search distance.

如果在搜索距离内未找到点，则返回 -1。

`<geometry>`

When running in the context of a node (such as a wrangle SOP), this argument can be an integer representing the input number (starting at 0) to read the geometry from.

当在节点（例如 wrangle SOP）的上下文中运行时，此参数可以是表示从中读取几何的输入数字（从 0 开始）的整数。

Alternatively, the argument can be a string specifying a geometry file (for example, a `.bgeo`) to read from. When running inside Houdini, this can be an `op:/path/to/sop` reference.

或者，参数可以是指定要读取的几何文件（例如 `.bgeo`）的字符串。在 Houdini 中运行时，可以是 `op:/path/to/sop` 引用。

`ptgroup`

A point group pattern to limit the search to. Can be a SOP-style group pattern such as `0-10` or `@Cd.x>0.5`. An empty string will match all points.

用于限制搜索范围的点组模式。可以是 SOP 样式的组模式，例如 `0-10` 或 `@Cd.x>0.5`。空字符串表示匹配所有点。

`pt`

The position in space to find the closest point on the geometry to.

在空间中寻找几何体上最近点的位置。

`maxdist`

The maximum distance to search. The operation can be sped up if it is allowed to quit early.

搜索的最大距离。如果允许提前退出，则可以加快操作速度。

然后在做个if判断一下这个点是否存在，然后给他标记上个色，并且把他加入到一个begin的点组里面。

@ptnum

The point number of the current point, when the snippet is looping over points. If looping over vertices, this is the point that the vertex is wired to. If looping over primitives, this is the point of the 0th vertex on the primitive.

当代码片段循环遍历点时，当前点的点编号。如果循环遍历顶点，则此点为顶点所连接的点。如果循环遍历图元，则此点为图元上第 0 个顶点的点。

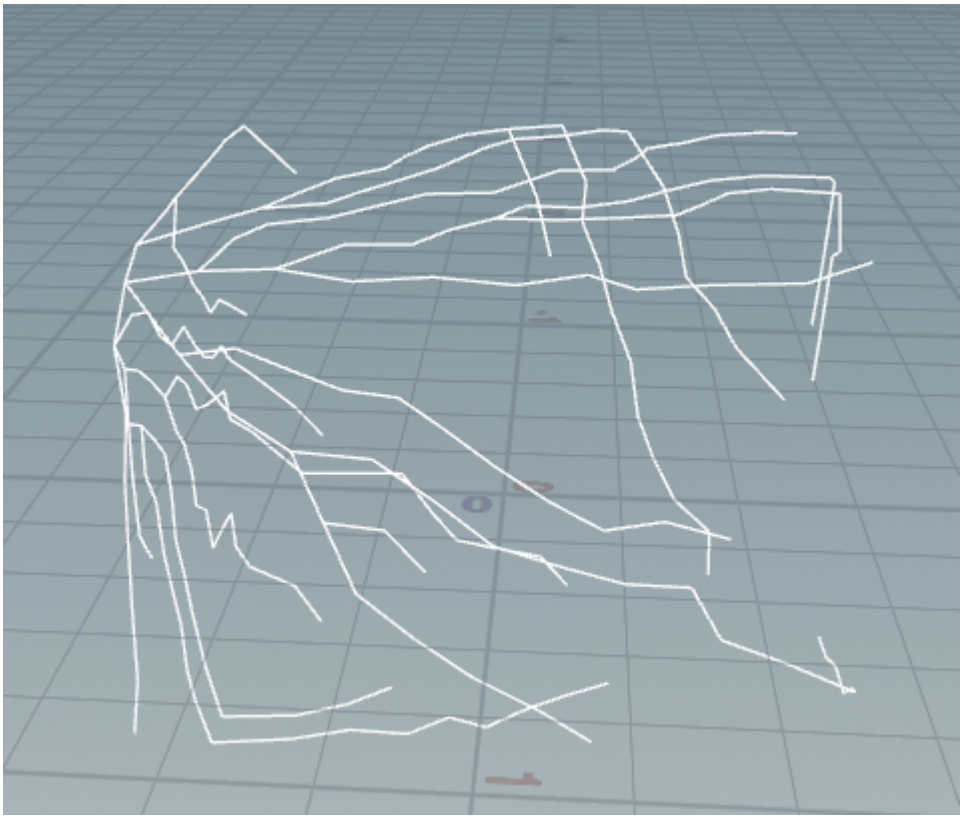
然后利用groupexpand这个节点对begin这个点组进行扩展(expand)，并用sort节点将他们打乱顺序。打乱之后我们再用grouprange就能filter出来随机的end点了。

Random **随机的**

Scrambles the elements into a random order.

将元素打乱为随机顺序。

使用findshortestpath节点，input进去我们的begin和end点组，然后就能得到多个线段，这就是想要的形状。



Find Shortest Path geometry node
查找最短路径 几何节点

Finds the shortest paths from start points to end points, following the edges of a surface.

沿着表面的边缘，找到从起点到终点的最短路径。

On this page **本页内容**
- Overview 概述
- ⊞ Parameters 参数
- Locals 当地人
- Examples 示例

Since 自从 13.0

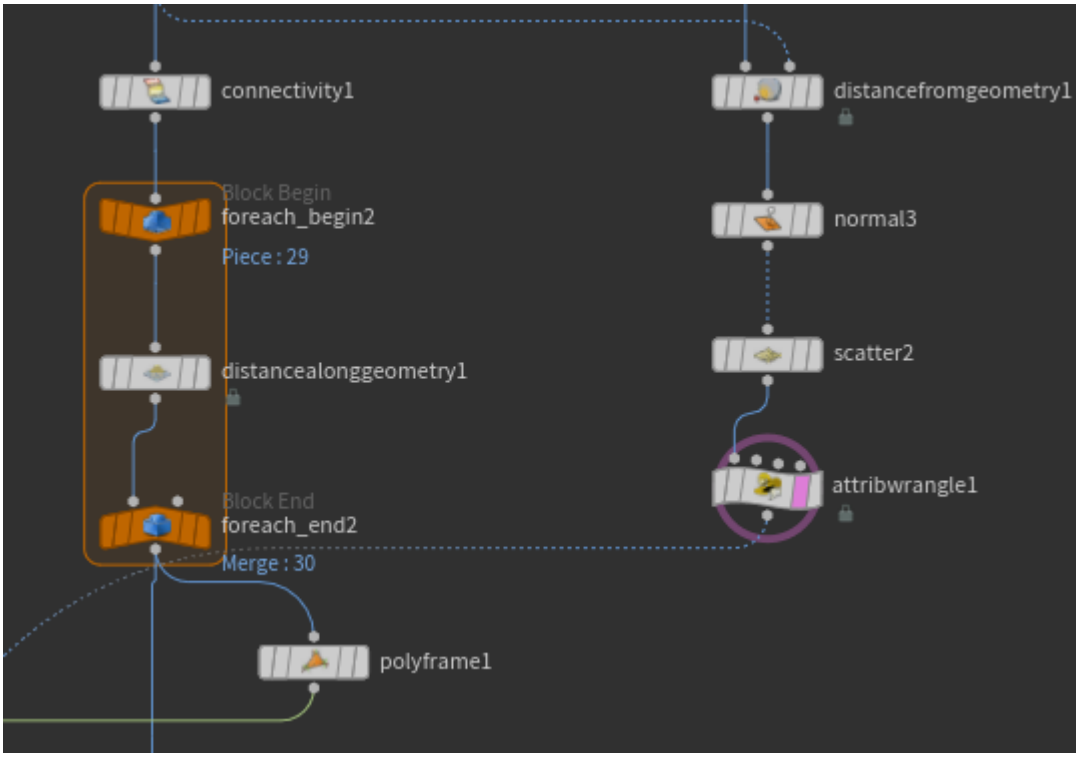
Overview 概述

This node finds the shortest paths through edges of the input surface geometry, between all pairs of start and end points, creating polygon curves along those paths.

该节点在所有起点和终点对之间找到通过输入曲面几何边缘的最短路径，并沿这些路径创建多边形曲线。

There are many options for specifying path costs other than just the length of the path, for example, costs for visiting certain points. If a pair of start and end points cannot be joined by a path, or if the cost would be infinite, no polygon is created for that pair.

除了路径长度之外，还有很多选项可以指定路径成本，例如，访问某些点的成本。如果一对起点和终点无法通过路径连接，或者成本无限大，则不会为该对创建面。



这边算出path后，用foreach遍历每一个点，然后用distancealonggeometry节点，做一个关于pscale的mask，这样不管是树枝还是树叶，在之后都会随着距离begin点越远逐渐缩小。然后在用distancefromgeometry节点算出path里面的点与参考几何（也就是一开始的box）的距离根据这个距离output一个mask出来，然后将其重新计算一下normal，再然后根据这个maskscatter参考点出来，在将这些点的法线复制到up属性并将其传给要生成树叶的点，这样树叶的朝向就正确了。

Houdini 21.0 节点几何节点 ›

Distance along Geometry geometry node
沿几何的距离 几何节点

Measures the distance of the shortest path along the geometry's edges or surfaces from each start point.

测量从每个起点沿几何体边缘或表面的最短路径的距离。

On this page **本页内容**
- Parameters 参数
- Examples 示例

Since 自从 18.0

This node can be used to measure the distance along a surface, which is useful when masking operations based on distance.

该节点可用于测量沿表面的距离，这在基于距离进行屏蔽操作时很有用。

Houdini 20.5 节点几何节点 ›

Distance from Geometry geometry node
与几何体的距离 几何节点

Measures distance between each point and a reference geometry.

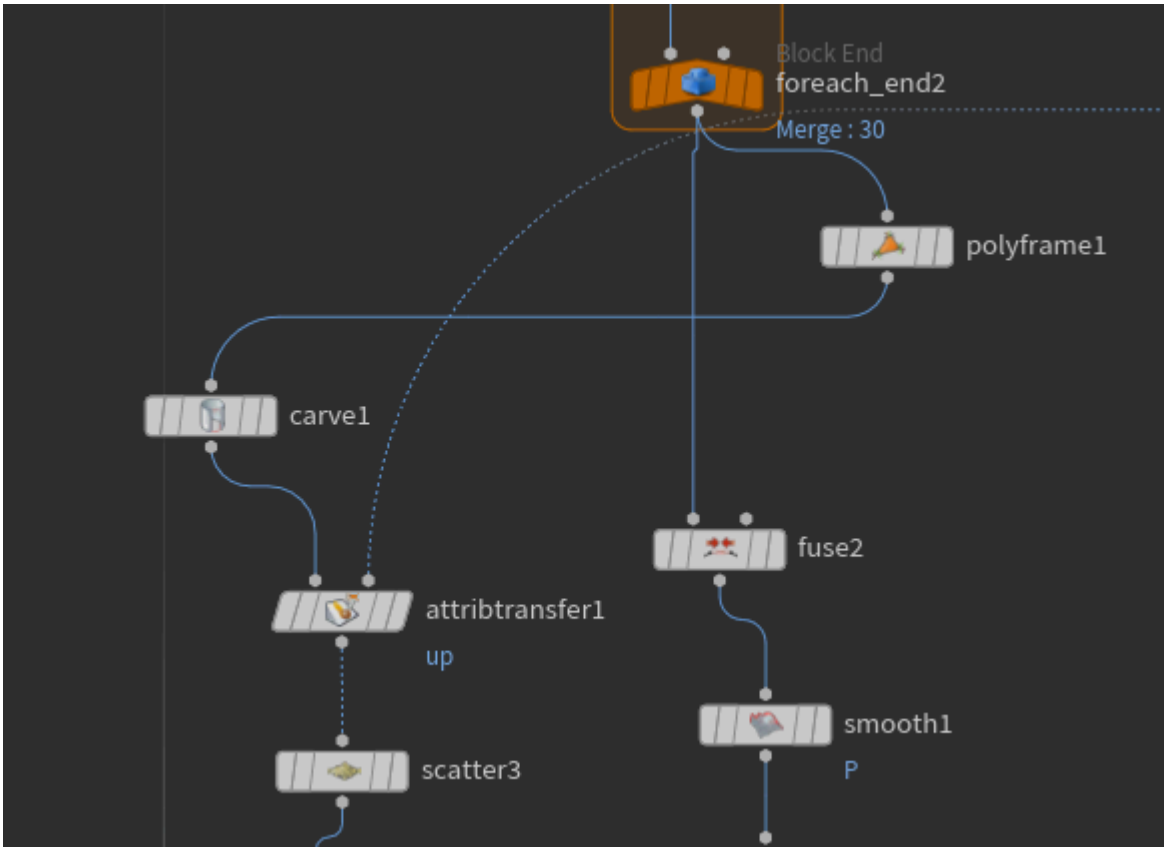
测量每个点与参考几何之间的距离。

On this page **本页内容**
- Parameters 参数
- Examples 示例

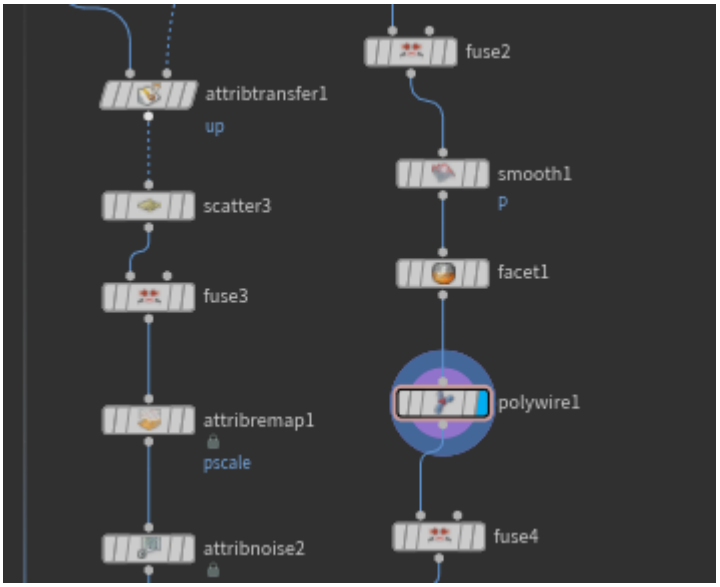
Since 自从 18.0

This node can be used to measure the distance to a reference geometry, which is useful when masking operations based on distance.

该节点可用于测量到参考几何体的距离，这在基于距离进行屏蔽操作时很有用。



这个carve就是裁切一下线段，因为树叶是不会生长在根的最尾部的，所以要裁切一下。然后在根据这个线段scatter出来，这边的scatter scale锁生成数来的点就是树叶的密度了。



fuse节点就是合并点，清理多余点。

Houdini 20.5 › Nodes › Geometry nodes

Houdini 20.5 节点几何节点 ›

Fuse 2.0 geometry node
保险丝 2.0 几何节点

Merges or snaps points.

合并或捕捉点。

On this page **本页内容**

- Fuse shelf tool
  保险丝架工具
- Snap shelf tool
  卡扣式搁架工具
- ⊞ Parameters 参数
- Examples 示例

Since 自从 17.5

The Fuse SOP is used to snap points together or snap points to a 3D grid, and optionally fusing points after the snap.

Fuse SOP 用于将点捕捉在一起或将点捕捉到 3D 网格，并可选择在捕捉后融合点。

To restrict the points that can fuse and be fused to, points can be query points, or target points, or both. With only one input to the node, both query and target points are from the single input. However, with a second input to the node, only points in the first input can be query points and only points in the second input can be target points.

为了限制可以融合和被融合的点，点可以是查询点、目标点，或两者兼而有之。当节点只有一个输入时，查询点和目标点都来自该输入。但是，当节点有第二个输入时，只有第一个输入中的点可以是查询点，只有第二个输入中的点可以是目标点。

先讲枝干的部分，线段smooth完在用facet清理一下，然后利用polywire生成树枝部分，整体的wire radius就要依赖我们之前计算的随着距离逐渐衰减的pscale值，这边0.1是自定义的值，然后在引入一个HDA的值进行控制。

```
0.1 * @pscale * ch("../branch_radius")
```

Houdini 20.5 › Nodes › Geometry nodes

Houdini 20.5 节点几何节点 ›

PolyWire geometry node
聚线 几何节点

Constructs polygonal tubes around polylines, creating renderable geometry with smooth bends and intersections.

On this page
本页内容

- Parameters 参数
- Locals 当地人
- Examples 示例

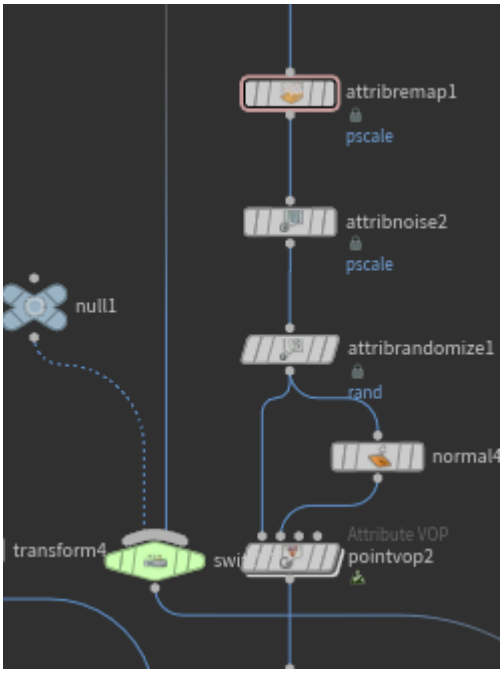
This node works like the Wireframe node, but this node creates more complex tube geometry from curves, with smoother bends and intersections than Wireframe, especially for L-systems. (The Wireframe node creates transition surfaces where the original curve branches.)

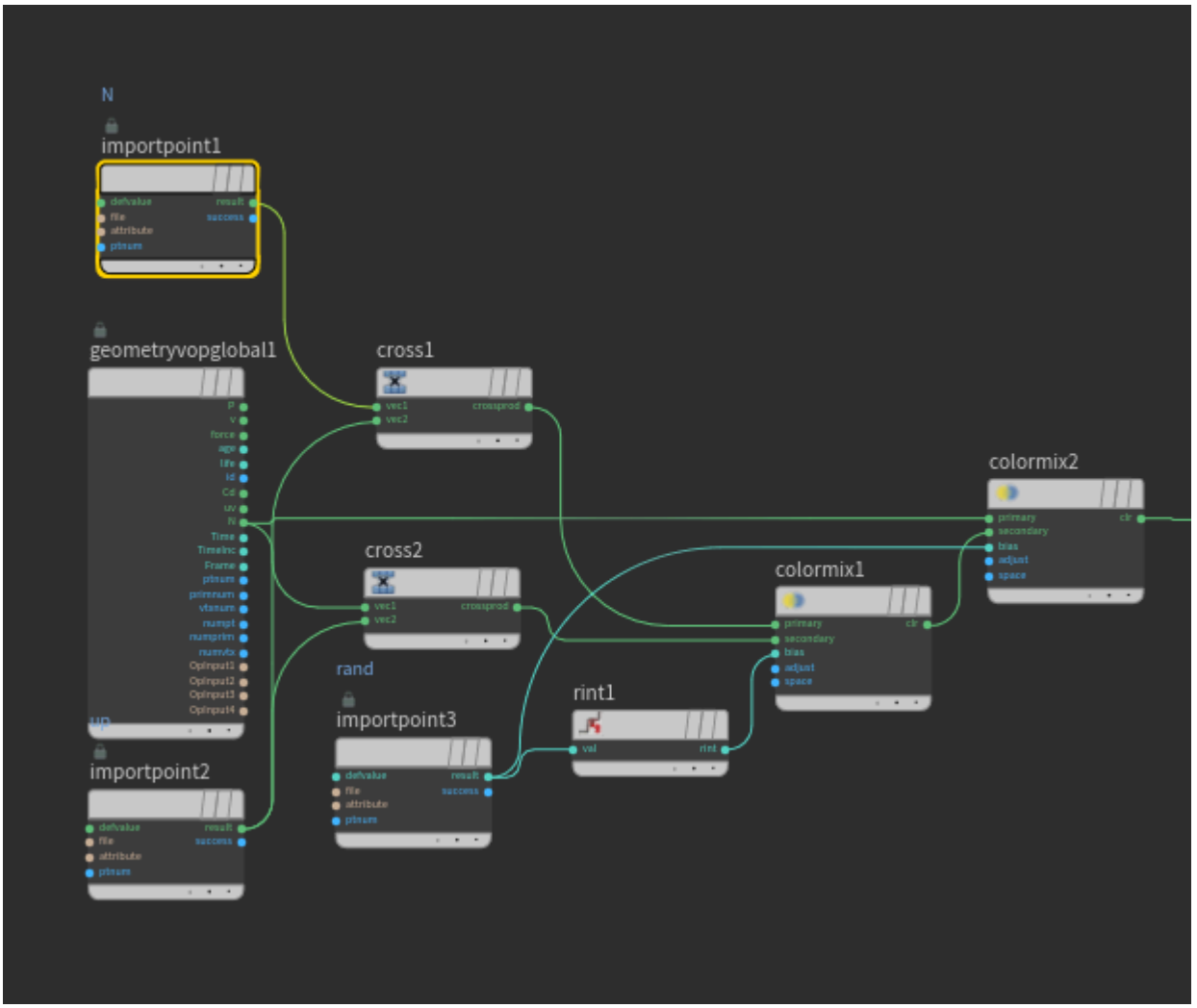
该节点的工作原理类似于线框节点， 但这个节点通过曲线创建了更复杂的管状几何形状， 比线框具有更平滑的弯曲和交叉，特别是对于 L 系统。（线框节点在原始曲线分支处创建过渡曲面。）

The parameters support local variables but it it recommended that the attribute overloads are used for controlling the parameters.

参数支持局部变量，但建议使用属性重载来控制参数。

树叶部分则是，前面已经设好点了，主要是对pscale还有朝向做进一步调整





将up向量与原本以及反转过法线的同时cross计算，然后blend(colormix)两次，一次是原本法线与反转法线的混合，一次是这个混合的值与原本法线的混合，bias是取整的rand值,最后传出混合法线方向。

最终效果

