

- [skip to content](#)



User Tools

- [Log In](#)

Site Tools

 Search
Tools ▼ >

Trace: • [mesh_fronds](#)

Table of Contents

- [Overview](#)
- [Difference from Standard Fronds](#)
- [Working with 3D Mesh Frond Assets](#)
- [Working with Flat Mesh Frond Assets](#)

Mesh Fronds

A “mesh frond” can be thought of as any sort of custom mesh put in place of the standard planar frond geometry. Frond meshes inherit the shape of standard fronds, allowing for a great deal of variation.

Overview

A “mesh frond” can be thought of as any sort of custom mesh put in place of standard planar frond geometry. Flowers, fruit, pine cones, hand-carved trunks and branches, two-sided leaves, simple texture cutouts and more can be easier to model with mesh fronds rather than with leaf meshes. Take a look at the following examples. Do you see how each banana leaf has a different shape? Those are fronds meshes. The original mesh was modeled relatively flat down the Y axis. The shape of each node is derived from the shape of the underlying spine and the replaced “standard” frond.



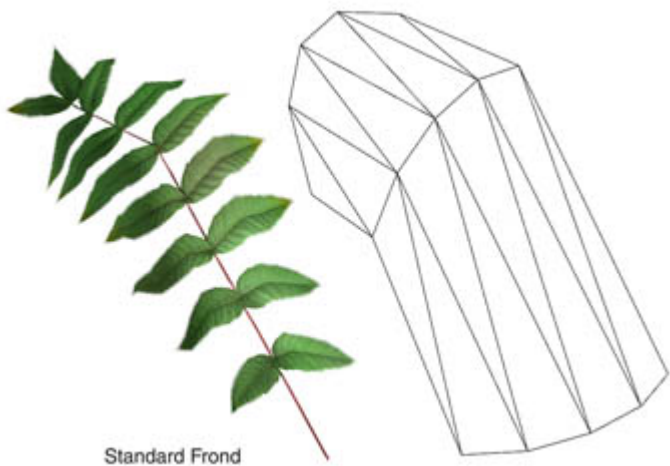
Various examples of mesh fronds in action (all visible geometry besides major branches).

Any geometry that can be exported from third party 3D software as 'OBJ', 'FBX', or 'STM' (SpeedTree Mesh) files can be used for a frond mesh. Additionally, frond mesh assets can be generated natively by SpeedTree and then incorporated back into your trees using built-in tools.

Difference from Standard Fronds

In most ways, a mesh frond behaves exactly the same as a standard frond. However there are a few basic differences, most of which are due to the arbitrary and unpredictable nature of externally created assets. Please keep the following points in mind:

Technical Differences



- **The number of vertices in a frond mesh is determined solely by the source mesh.** This means that changing the segment count on the spine generator will not affect the number of frond vertices. It will affect the distortion of the shape of the mesh, as described in detail here. Changing the number of vertices in the mesh itself must be accomplished externally. A side effect of this is that the position of break points along the length of a mesh frond may not coincide with the break points of the actual spine. As such, it is better to determine the resolution of (and distribution of) spine segments before creating mesh frond assets so that the mesh can be matched during creation.
- **The number of segments in a spine DOES affect the shape of a mesh frond.** Even though the segmentation of the spine doesn't affect the number of vertices in a mesh frond, it does affect its shape. Make sure to allocate an appropriate amount of segments for the intended mesh.
- **The properties in the Frond: Shape group contribute to the shape of the mesh.** Width, height, and the contour profile curve have an affect on the shape of mesh fronds. See the “Working with 3D mesh frond assets” section for details on their relationship.
- **Mesh fronds can only have one material assigned, and it probably shouldn't tile.** There is only a single material slot per frond mesh.
- **Bifurcated spines result in mesh fronds being pushed out past the Bifurcation spot.** Unlike standard fronds, mesh fronds cannot be arbitrarily sliced at the point of bifurcation, since their geometry is unknown and may be very complex. However, mesh fronds can still exist in a bifurcated scenario, but it is recommended to still enable the branch geometry so that

“floating mesh fronds” don't occur.

- **Mesh fronds will not respect a Count value greater than “1”.** Only a single mesh frond is legal per node (standard fronds can have multiple blades along the same spine), with the exception of bifurcated spines which will have two mesh frond instances, one on each end after the split.
- **Mesh fronds should have a Frond: Roll value of '1.0' to match branch alignment.** If you have used the Alignment spine property to make the tops of branches and fronds point towards the sky, you'll need to ensure that the frond:roll property is set to “1”, which aligns the mesh to the same up vector as the branch.
- **Mesh fronds are centered horizontally unless Centered surface is the mapping type.** Make sure to use “centered surface” as the mapping type if there is a clearly defined centerline to your mesh (such as a custom palm frond). Otherwise, the shape will be centered horizontally and it will be difficult to ensure that the mesh extends equidistantly on both sides.
- **Mesh fronds can have flipped shape, but not flipped mapping.** Any mesh asset could potentially rely very specific, unwrapped texture coordinates. Flipping the map would result in mismatched texture coordinates. However, the entire mesh can still be flipped over the Y axis.

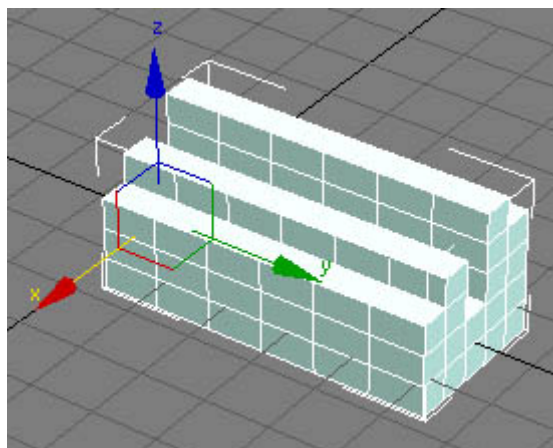
While this list of precautions may seem daunting, don't fret! We just want you to be aware of what can and can't be done with this type of geometry. Mesh fronds are still very powerful despite these few caveats – keep reading for in depth information on actually working with mesh frond assets.

Working with 3D Mesh Frond Assets

Mesh fronds come with many options to ensure that the user can achieve the desired look. A working knowledge of how frond meshes are constructed is warranted before maximizing the potential of this powerful modeling tool. Below is a brief set of notable properties and techniques that you, the user should be cognizant of when working with fully 3D mesh fronds.

1. Align source material down the 'Y' axis

The Modeler assumes that a frond mesh is aligned down the 'Y' axis. If your model does not lie properly on the spine, change the orientation of the mesh asset on the 'Mesh Assets' bar.

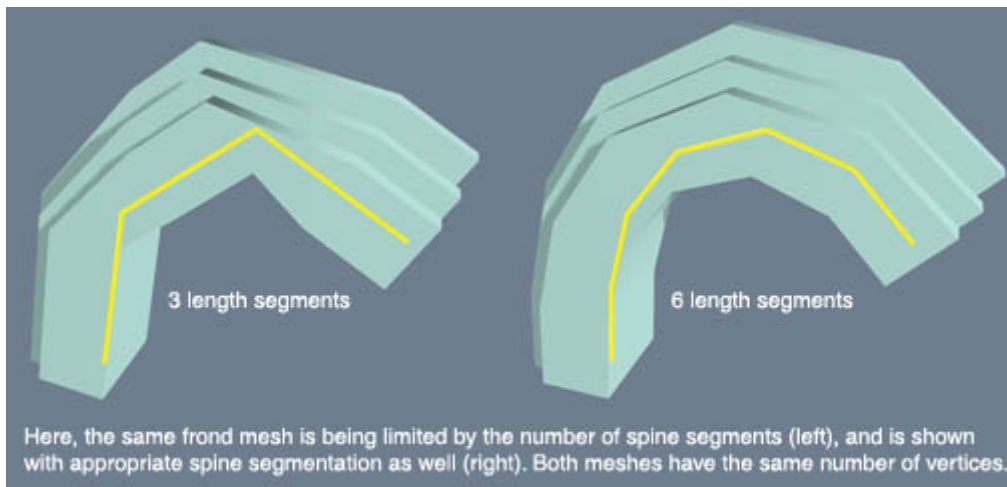


This mesh asset is correctly pointing down the 'Y' axis.

2. Mesh fronds are at the mercy of the segmentation you provide it

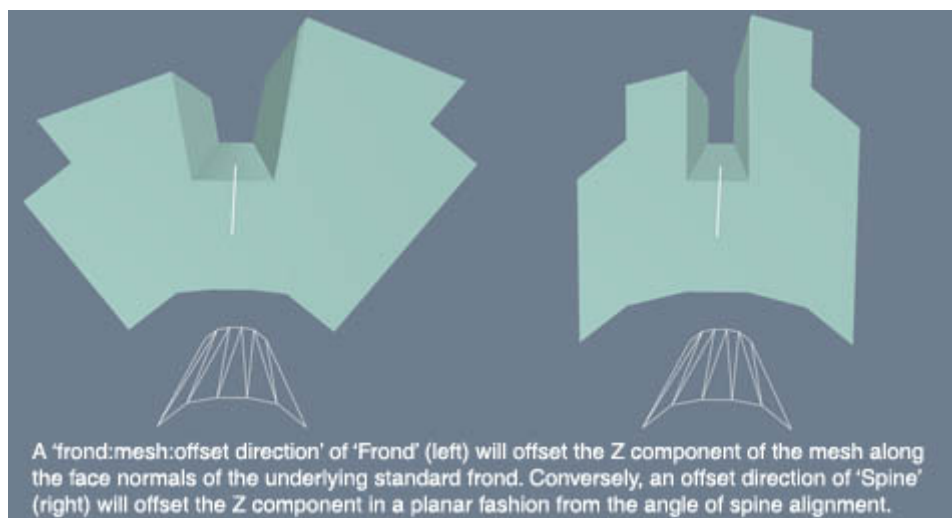
When using a mesh in place of a standard frond, the shape of the standard frond is still inherited. In order for your mesh to come across faithfully, the underlying spine and frond components need to have a serviceable amount of resolution. This means that the 'Accuracy', the number of 'Length segments', and both the 'Optimization' and 'Frond' optimization properties should be adjusted accordingly.

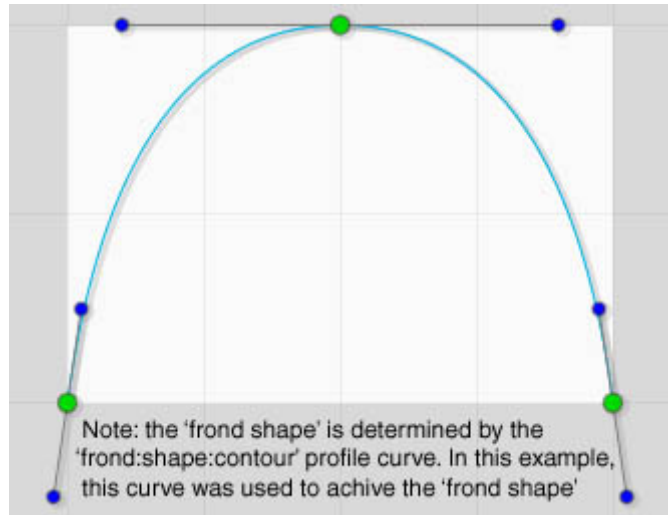
For instance, the mesh depicted above has 6 lengthwise sectional divides. In order for this mesh to utilize all 6 of these break points, the spine in the Modeler needs at least an 'Accuracy' and 'Length segment' absolute value of 6, and 'Optimization' brought down to zero.



3. Frond shape segments affect the shape of the mesh too

Just the same as 'Length segmentation', the number of 'Frond shape segments' will affect the shape of your mesh. To illustrate, see the diagram below (the wireframe image below the mesh is the standard frond before the mesh is applied):

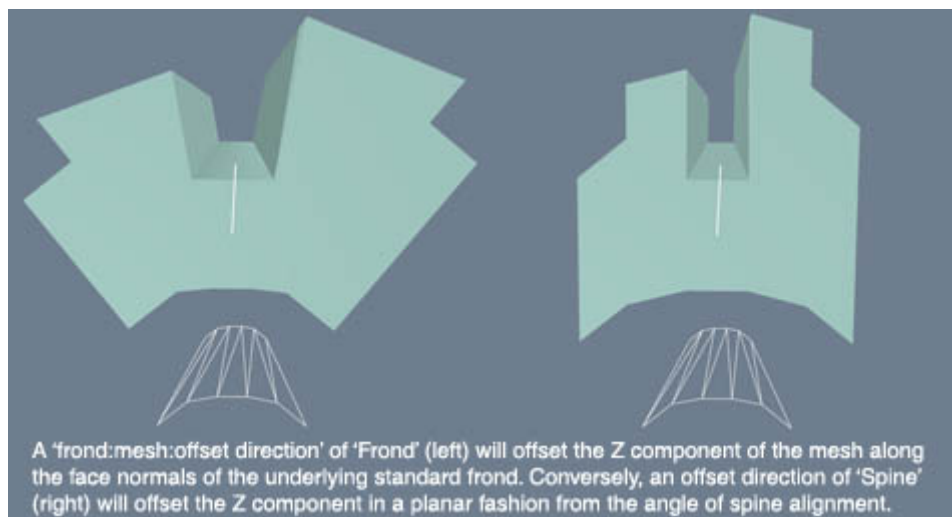




Using a 'FronD shape segment' value of “1” keeps the mesh from distorting across its width, regardless of the *FronD:shape:contour* profile curve. For many cases, this is ideal since the shape as you modeled it remains relatively intact and free of distortion artifacts. However, setting this profile curve to the “max” preset will achieve the same effect, since the number of segments is irrelevant if the profile shape is flat.

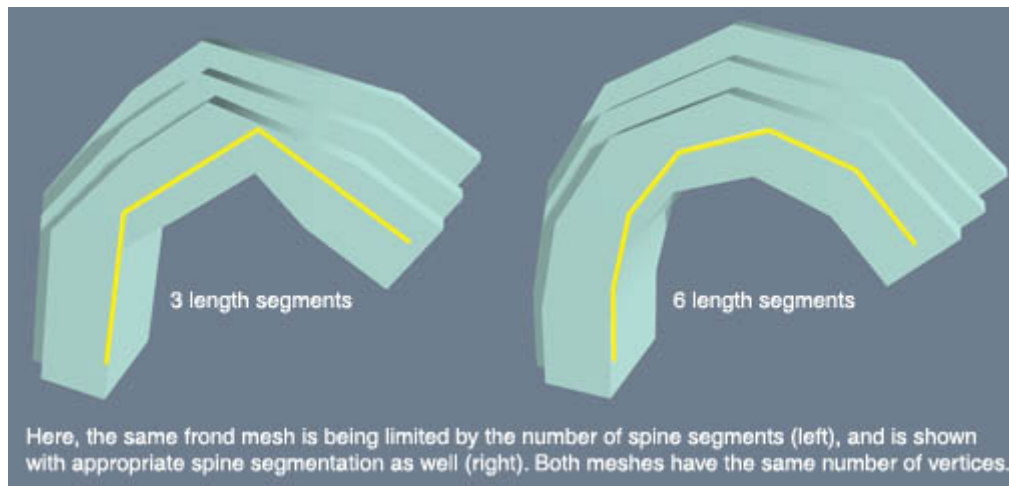
4. “Offset direction” affects the Z component of the mesh

You may have wondered why the previous few diagrams feature a type of distortion where the mesh appears to “bow out” as the width resolution increases. This behavior is controlled with the Offset direction property. Depending on whether this choice-box is set to “FronD” or “Spine”, the offset of the third dimension of your mesh may change substantially, as depicted below:



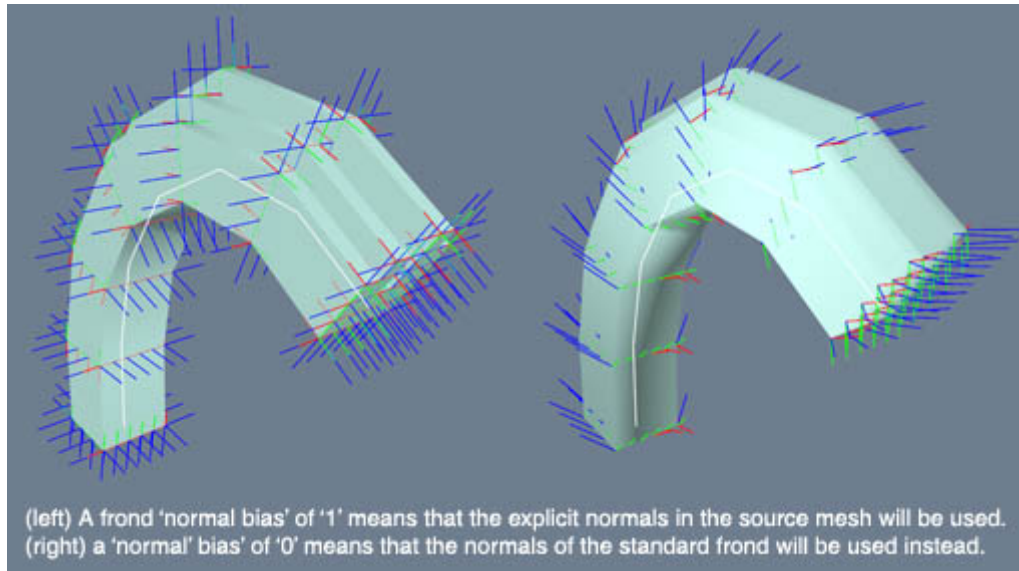
5. Don't confuse “Height” with “Z scale”

The *FronD:shape:height* property controls only the height of the underlying standard frond shape, not that of the assigned mesh frond. To scale the height of the mesh frond, use the 'Z' scale property instead (note that 'Z' scale is affected by the state of the Lock aspect ratio checkbox).

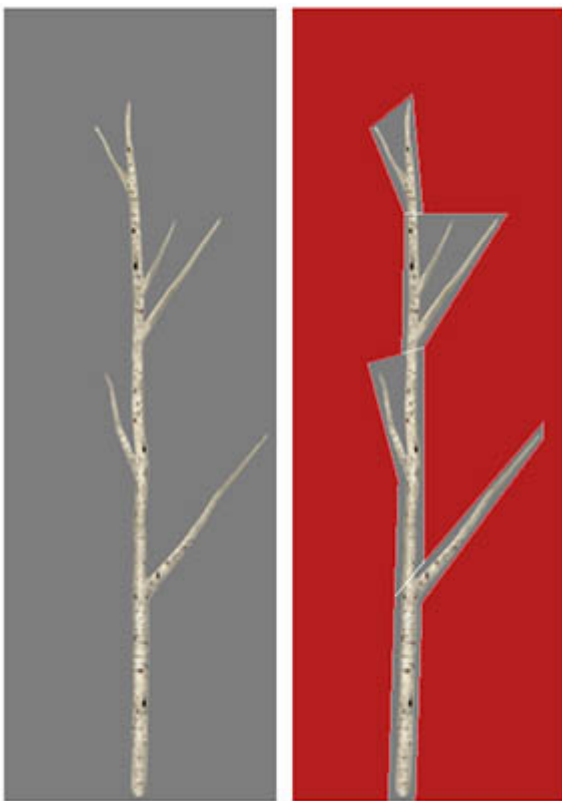


6. The normals of the mesh can be altered with “Normal bias”

The 'Normal bias' property will modulate between the normals set in the source mesh (1.0), and the normals that would have occurred had a standard frond been used (0.0).



Working with Flat Mesh Frond Assets



The transparent area shown in red was eliminated by using a mesh frond cutout instead of a standard frond

Many of the practices described in the previous section apply to all frond meshes (segmentation, Y axis orientation, etc). However, if your mesh is “flat” (the vertices are, or are nearly coplanar), some of the suggestions above aren't as applicable. With a flat mesh, the Frond:shape:height property is the only mechanism in place for scaling the “thickness” of the mesh, since it has no actual difference in 'Z' scale.

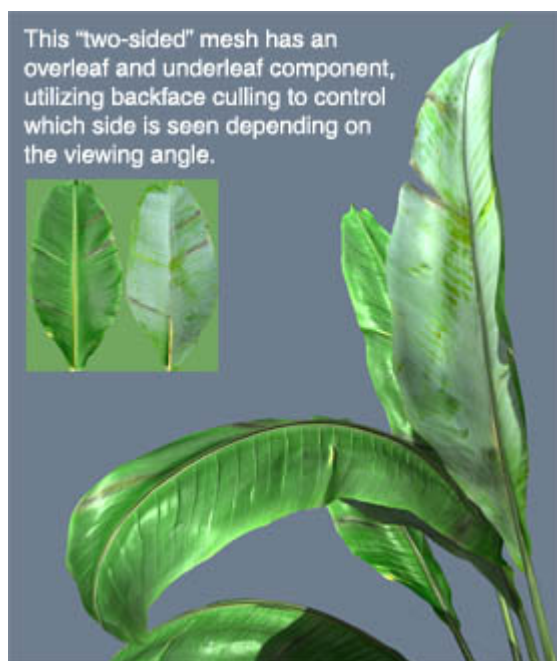
So what would a flat frond mesh be used for? Here are a couple of common examples:

Image-Based branch cutouts

One drawback of standard fronds is that they always create rectangular geometry. For “image-based branches”, which are really frond blades that pose as a group of 3D branches, the ratio of transparent pixels to opaque pixels can be very lopsided since most branch images don't utilize a rectangular texture space very well. In this case, it is much more reasonable to create flat “cutout” of the texture in question, and then to exchange the cutout shape in place of the standard frond. While this approach will inevitably result in more vertices, the real-time performance will be much improved.

Branch cutouts can be created directly in the SpeedTree Modeler using the “Generate Mesh” tool.

Two-Sided frond meshes



Another good use of a flat frond mesh is as a two-sided frond. Standard fronds project the same texture on both sides, and cannot be backface culled. As such, they suffer from only lighting correctly from the “top” side, since the normals have to favor one side. A mesh frond can be substituted instead, allowing for both overleaf and underleaf textures, as well as the ability to light correctly from both sides.

To create a two-sided frond mesh, you will first need to edit the texture coordinates so that each side corresponds to use half of the texture space. Then when imported into the Modeler, make sure backface culling is enabled on the assigned color set.

If using the “Generate Mesh” Tool, two-sided meshes can be created by simply enabling the “double-sided” property.

When modeling a two-sided frond mesh, it is important to offset the top and bottom halves by a very small increment (about 0.001 units) so that they do not occupy the exact same location in space. Many offline renders (such as Mental Ray®) may introduce artifacts caused by two faces that occupy the same space when rendered in conjunction with backface culling.

[Read our blog >>](#)

- [Home](#)

- [Company](#)
- [3D Animation Software](#)
- [3D Tree/Plant Library](#)
- [Accolades](#)
- [Documentation](#)
- [Contact](#)
- [Privacy Policy](#)
- [Terms & Conditions](#)
- [Site Map](#)

- ©2017 IDV, Inc. All Rights Reserved.
- [Questions?](#)

