

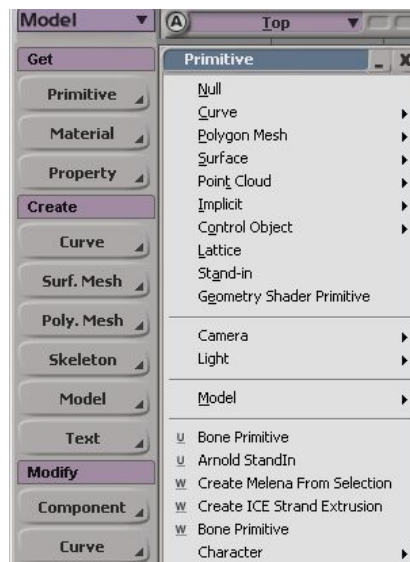
studioNEST's Melena Documentation Version 1.1

Melena is a hair styling and simulation toolset for Autodesk Softimage 2010 SP1 or higher which provides a collection of compounds for creating fur. Additionally it contains some compiled operators for connecting particles to Softimage's Hair as well as custom nodes.

Requirements: Windows XP 32 / 64 bit or later versions, Linux x86_64 (RHEL 5)
Autodesk Softimage 2010 SP1 32 / 64 bit or later.
Python 2.x

Installation: In Autodesk Softimage, choose the File -> Addon -> Install menu, choose the addon file and click on Install. Restart Softimage to complete the installation.

Usage: Generally you can create Melena hair by using the "Create Melena From Selection" in the Model -> Get -> Primitive menu. This works for selected polymeshes, nurbsurfaces, curves as well as Softimage Hair.



Melena covers two main areas relating to fur / hair: Styling and Simulation.

For the styling part we provide two main categories of compounds: Emitters and Modifiers. For the simulation we provide the categories of forces, simulation nodes and some more. All of the compounds can be found in the task area of the icetree view, under Melena. Some compound appear in more than one category, however in this document you can find them only in their primary category. (Example: Strand Fcurve is a Modifier as well as a Getter.)

Generation techniques

Melena hair can be created in a variety of ways.

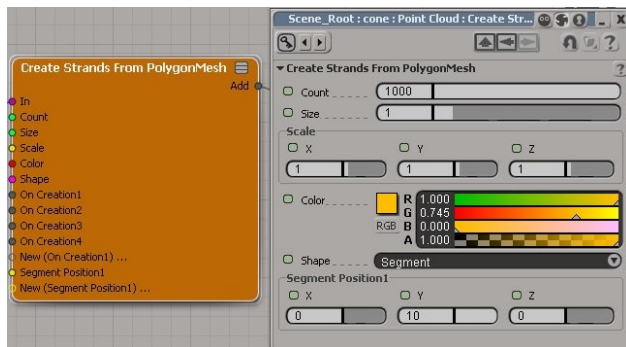
- **Melena from PolygonMeshes**
The hair is generated by a emitter compound directly on the polygonmesh object. The generated strands are especially useful for really short fur. By default you only have two segments, which results in very good performance when using loads of fur.
- **Melena from NurbsSurface**
The hair is generated using a compound as a projected subcurve of the nurbssurface. You can control the amount of curves easily and properly drive the shape of the resulting fur since it is connected live to the nurbs.
- **Melena from NurbsCurve**
The hair is also generated using a single compound. This technique is useful for creating one clump of hair with the maximum control over the shape along a single line.
- **Melena from Softimage Hair**
For this technique we are creating an intermediate curvelist to get access to the hair shapes, since ICE is not able to access hair data directly. On the curvelist you can find an operator which drives the curve generation, here you can also control if you want to only use the guide hairs or the render hairs, and, of course, how many render hairs. That way you can combine the power of the styling tools as well as the standard hair effects with the Melena compounds for styling as well.

Emitter Compounds

These compounds deal with the creation of hair, as discussed in the section above. Let's have a look at each of them.

Create Strands From PolygonMesh

This compound is used to create single-segment fur on a polygonmesh.



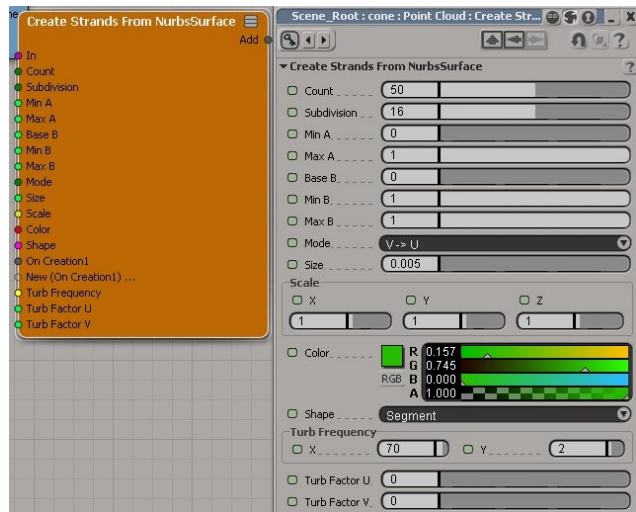
This compound is used to create strands on the surface of a polygonmesh.

- Count** Controls the amount
- Size** Controls the thickness
- Scale** The hair's scaling
- Color** The hair's color
- Shape** The shape type (should be always set to segment for rendering with MR)

Segment Position This drives the first segments position. By default you will only have one segment, but you can overdrive this by adding more segments. You can also plugin an interpolated array here to auto-sample the hair.

Create Strands From NurbsSurface

This compound is used to project strands onto a nurbssurface.



Count	Controls the amount of fur
Subdiv.	How many segments to use
Min/MaxA	The min and max for the first axis
Base B	The minimum for the second axis
Min/MaxB	Random min max second axis
Mode	Choose between two projections
Size	The width of the hair
Scale	The scaling of the hair
Color	The hair's color
Shape	The shape type for the fur. Use Segment for rendering with MR.

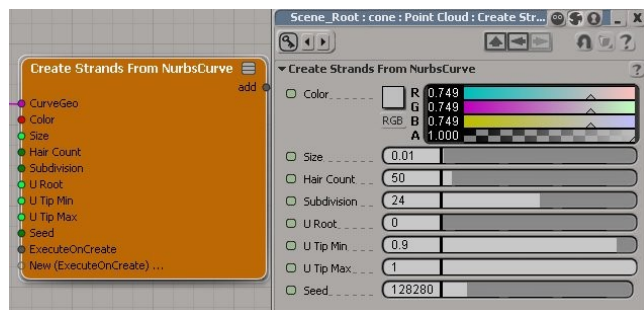
Turb Freq The frequency for an additional distribution turbulence.

Turb Fact. U The factor of the distribution turbulence for the first axis.

Turb Fact. V The factor of the distribution turbulence for the second axis.

Create Strands From NurbsCurve

This compound is used to create strands from a single nurbscurve.

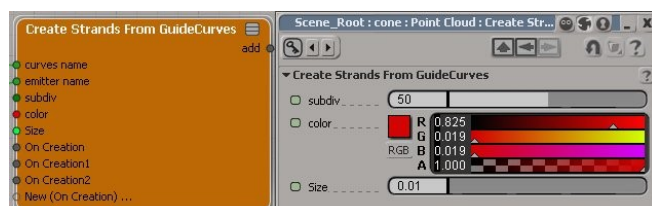


Color	The color of the fur
Size	The width of the strands
Hair Count	The amount of fur to create
Subdiv.	The number of segments per hair
U Root	The U value of the root
Tip MinMax	The random value min max of the tip of the fur

Seed This controls the base number for the random number generator.

Create Strands From GuideCurves

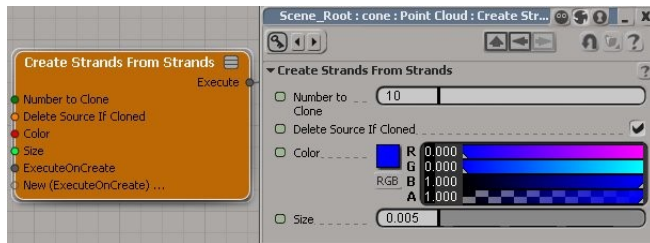
This compound is used to create strands from a nurbscurve containing guide curves.



Subdiv.	The number of segments per hair
Color	The fur's color
Size	The width for the fur.

Create Strands From Strands

This compound is used to create strands from strands, for example for creating fur clumps.



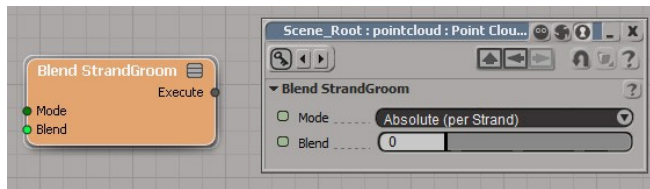
- Nb 2 Clone** The amount of clone to create
- Delete...** If checked we remove the original
- Color** The color of the fur
- Size** The width of the fur

Modifier Compounds

These compounds are used to drive the styling of the fur. Nearly all of them are dependant on the “Orient Strands” compound, as they require a strandorientation to be set. Let's have a look at each one.

Blend StrandGroom

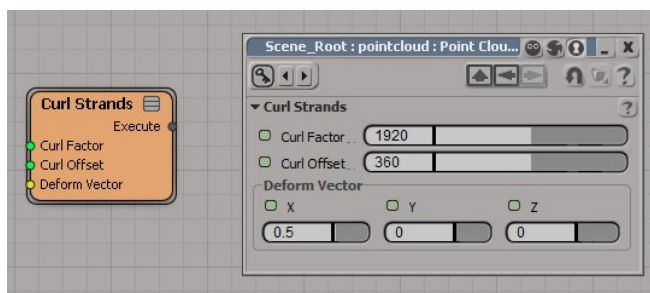
This compound blends a stored groom with the current shape of the strand. A groom can be stored with the “Store StrandGroom” compound. See the “Setters” section of the document. For a more detailed description of the modes, also see the “Store StrandGroom”'s description.



- Mode** You can choose between relative and absolute modes.
- Blend** The amount of blend to use.

Curl Strands

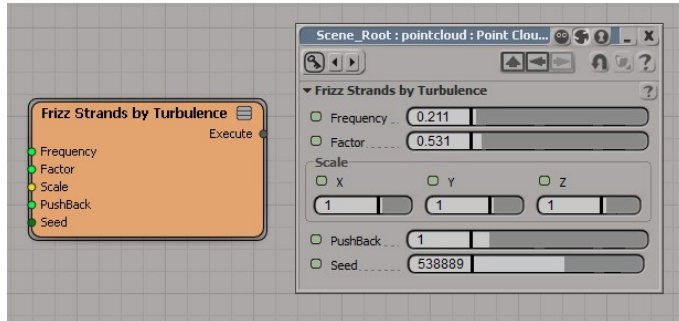
This compound creates curls around the current strands. You need to have the StrandOrientation set for this to work properly.



- C. Factor** The degrees to turn the strand
- C. Offset** The amount of displacement
- D.Vector** The vector to use for displacem.

Frizz Strands

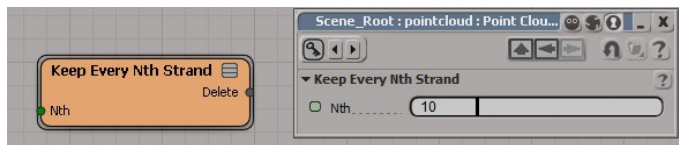
This compound is used to create some turbulence based displacement along the strand for soft and slightly curly hair for example for a long beard.. This compound also requires the StrandOrientation to be set.



- Freq.** The frequency for the turbulence
- Factor** The amount of displacement
- Scale** The factor separate per axis
- PushB.** The amount of reverse displacem.
- Seed** The random base number

Keep Every Nth Strand

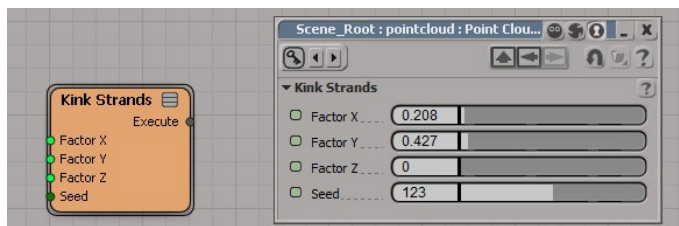
This compound is used to delete regularly strands and keep only every nth one. This way you can do simulation for example on less strands without changing the base emitters.



- Nth** The base number for the strands to be kept.

Kink Strands

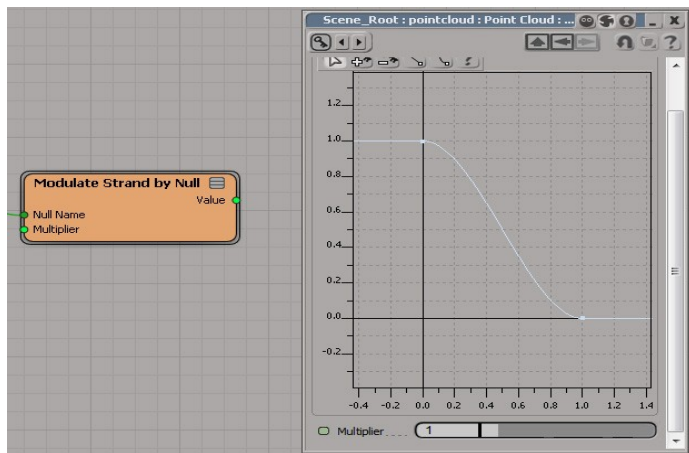
This compound is used to randomly displace the strands using true random number. This can be used to emulate electricity along the hair. Thie compound requires the StrandOrientation to be set.



- FactorX** The amount of displacement for X
- FactorX** The amount of displacement for Y
- FactorX** The amount of displacement for Z
- Seed** The base random number

Modulate Strand by Null

This strand corresponds to the factory “Modulate By Null” compound to receive a Scalar value based on the distance to a null object. We just added this one so it also works with strands.

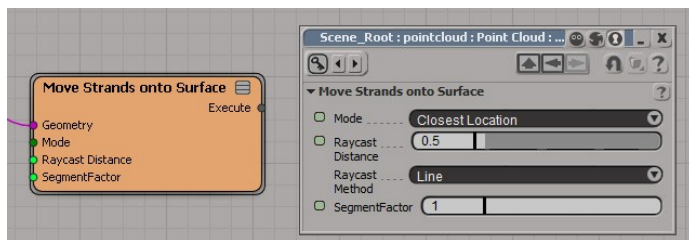


Fcurve This profile can be used to define a smooth step along the distance.

Multi. This slider can be used to control the strength of the output value.

Move Strands onto Surface

In some situations the strands (including the root) can be sitting in open space, disconnected from the emitting surface. With this compound you can move the strand back. Also consider connecting a StrandFcurve to the “segment factor” port to keep the tip in place, but gradually move the hair to connect it to the surface.



Mode You can choose several projection methods.

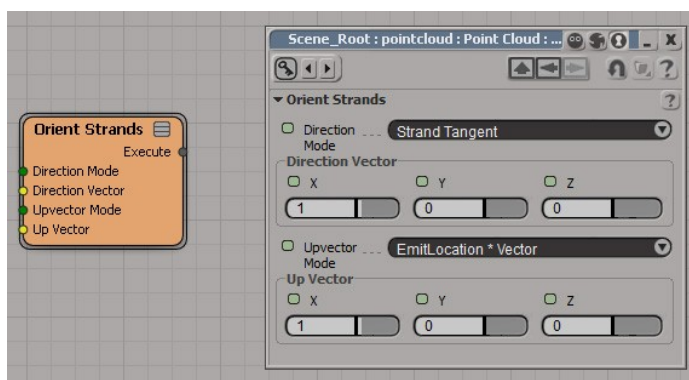
RayDist. Controls the distance of the used ray, in case you use ray mode.

RayMeth. Controls the raycasting method, in case you are using ray mode.

SegFac. Controls how much the segment is moved.

Orient Strands

This compound is one of the most important modifiers. It sets the StrandOrientation based on several settings. The default settings should work however in most situations.



DirMode Choose between several modes for calculating the direction

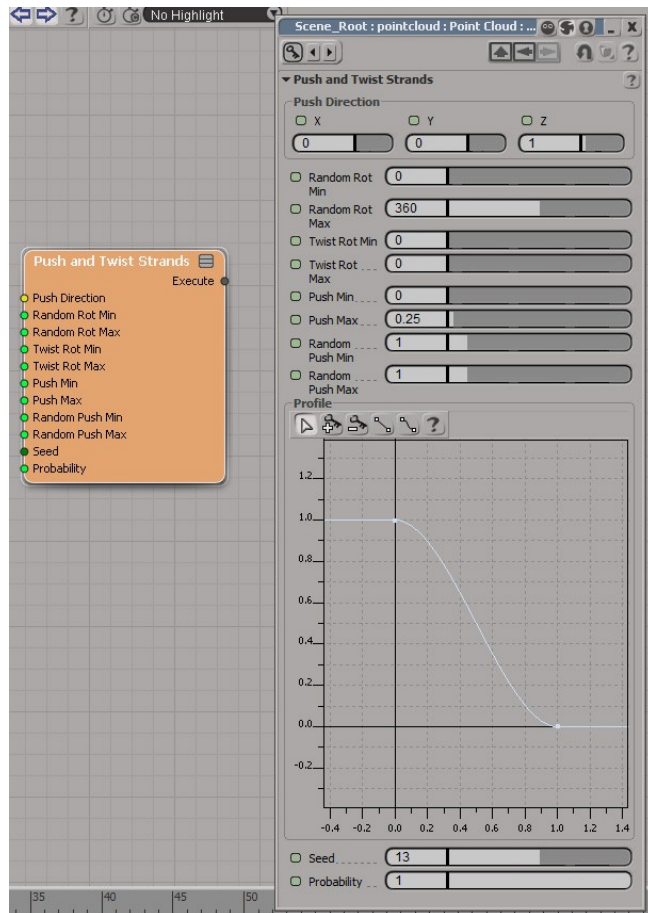
DirVec. The vector used in some modes

UpMode Choose between different modes for defining the up vector

UpVec. The vector used in some modes

Push and Twist Strands

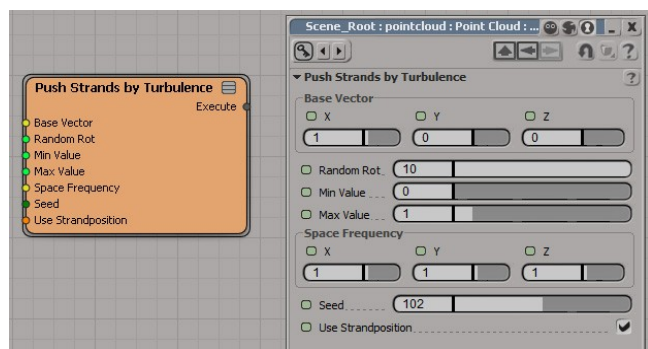
This compound requires the StrandOrientation to be set. The strand is pushed away along one local axis and can also be twisted in the same effect. This is very good for typical clumping setups after creating Melena fur from a NurbsCurve.



- PushDir.** The local direction for the push.
- R.Rot Min** Minimum random rotation
- R.Rot Max** The relating maximum rotation
- Twist Min** The minimum rotation for the twist
- Twist Max** The max rotation for twisting
- Push Min** The minimum displacement
- Push Max** The max displacement to use
- R.PushMin** The min. for a random multiplier
- R.PushMax** The max. for a random multiplier
- Profile** The profile for the push along the strand
- Seed** The base for the random number
- Probabil.** The probability for the effect to be applied

Push Strands by Turbulence

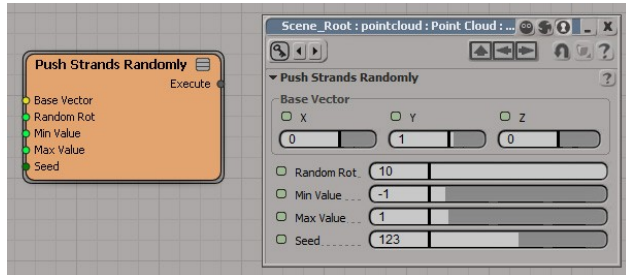
This compound also requires the StrandOrientation. The strand is pushed along local axes based on the strength of a turbulence field.



- BaseVec.** The local vector to use for push
- Rand. Rot** Random rotation for base vector.
- Min Value** The min. value for the rand. Push
- Max Value** The max. value for the rand. Push
- SpaceFreq.** The frequency of the turbulence
- Seed** The base for random numbers
- StrandPos.** If checked use 3D pos. (not 2D)

Push Strands Randomly

This compound is used to create volume inside of a clump. You can randomly push the strands inside of their own orientation reference. Therefore the compound requires the StrandOrientation to be set.



Base Vector The direction of the push effect

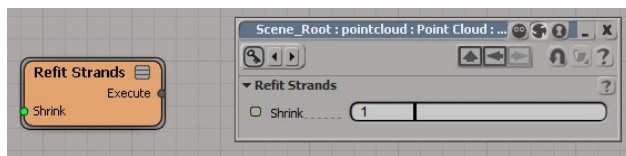
Random Rot The amount of random rotation

MinMax Val. The minimum and maximum of the random push effect

Seed The base random number

Refit Strands

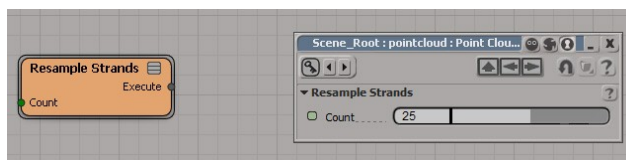
This compound is used to project the curve by changing its length while keeping the shape. This is very similar to the “Fit On Curve” curve generator.



Shrink The maximum projection facot. If set to value below 1, the curve will be shrunk (shorter).

Resample

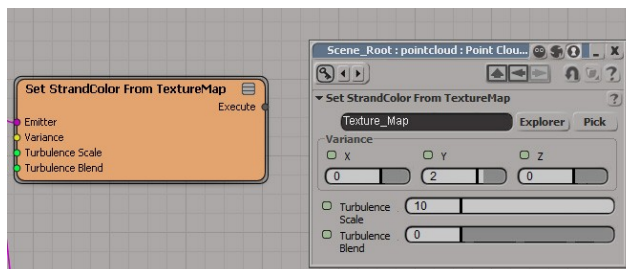
This compound is used to change the number of segments per strands while keeping the shape. This is another feature that is similar to the “Fit On Curve” curve generator.



Count The new number of segments

Set StrandColor from Texturemap

With this compound you can project a texturemap's colors on the strands in a simple fashion.



Reference The name of the texturemap

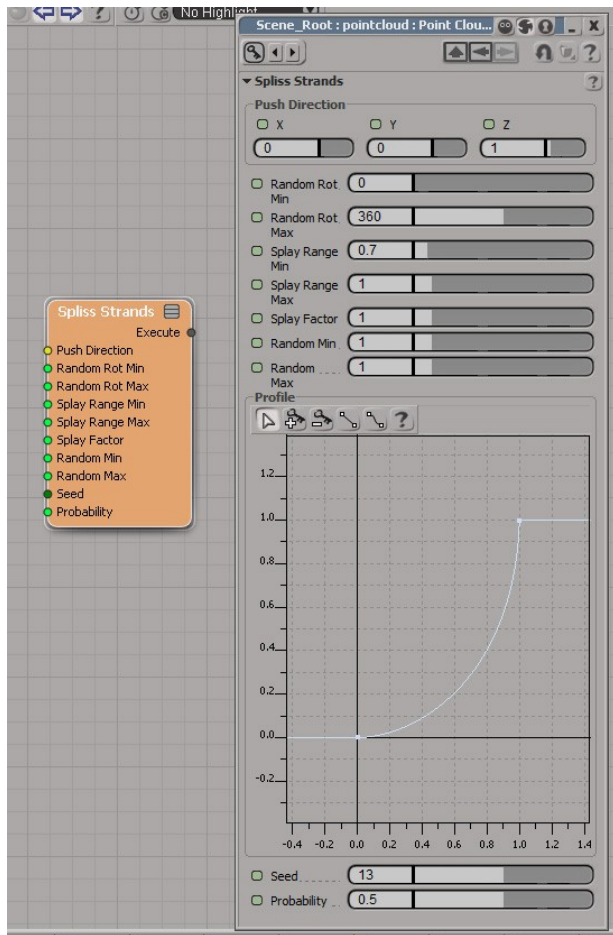
Variance A randomness for color lookup

Turb.Scale The frequency for the turbulence

Turb.Blend The amount of turbulence to use

Splice Strands

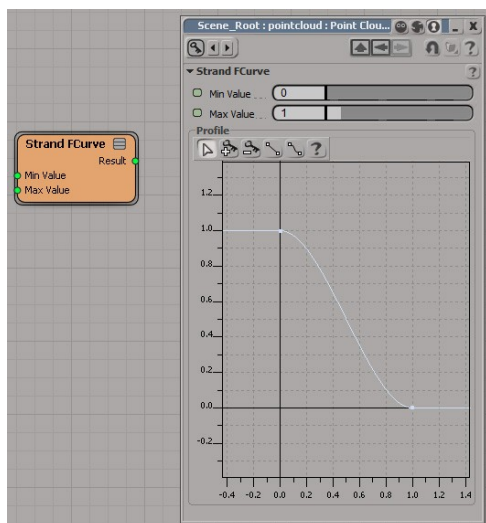
This compound allows so emulate the phenomena of spliss when styling fur. Basically it creates single hairs that are disconnected from the clump due to electricity.



- Push Dir.** The direction of the push effect
- Rnd. Rot** The min and max for rotating the push direction randomly
- Splay Range** The min and max for the U value along the strand to start the displacement / push
- Splay Factor** The amount of displacement/push
- Rnd. Min/Max** A multiplier for the displacement, defined by a min and max value
- Profile** A fcurve multiplier for the spliss displacement effect
- Seed** The base random number
- Probability** Drives how many hairs are effected by the spliss compound.

Strand Fcurve

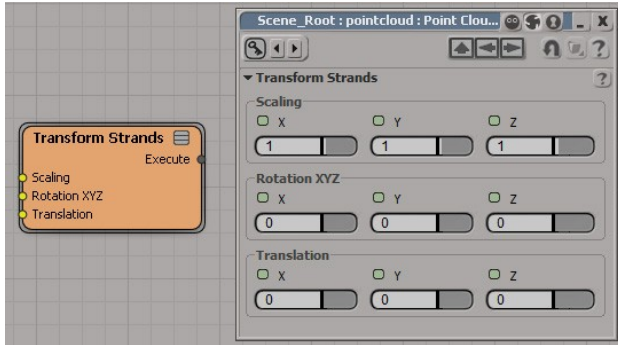
Using this compound you can drive other compounds' attribute along the strand, for example the factor of one of the Push compounds.



- Min Value** The value on the left side of the profile
- Max Value** The value on the right side of the profile
- Profile** The profile along the strand, left being the root of the strand, right representing the tip

Transform Strands

This compound allows to move strands, rotate strands, or scale strands as if they had a real transform attached. This is extremely useful for doing post transformation of strands after they have been styled already.



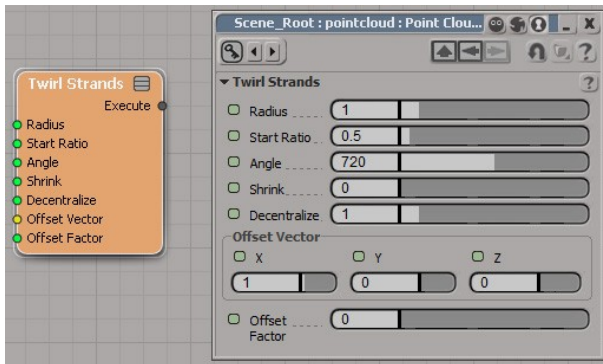
Scaling The amount of scaling of the XF

Rotation The given of rotation of the XF

Translation The translation of the XF

Twirl Strands

This compound can be used to create a twirl effect, specifically for non realistic stylized fur. It is similar to curl, but while curl is a twist around the strand itself, twirl is simply a circular twist along a given local axis.



Radius The radius for the twist effect

Start Ratio The U value where to start

Angle The amount of rotation for twist

Shrink Use this to create spirals, by shrinking the strand additionally

Decentralize Push away from the center

Offset Vector A secondary offset vector

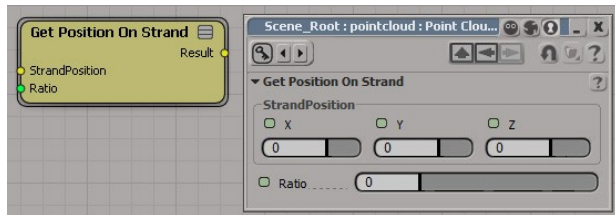
Offset factor The amount of secondary offset

Getter Compounds

These compounds are used to receive some data from an existing styling inside Melena.

Get Position On StrandColor

This compound retrieves an interpolated position on a given strand defined but the U value.



StrandPos.

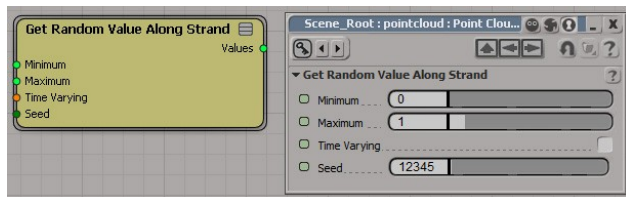
The given strand

Ratio

The U value to interpolate to

Get Random Value Along Strand

This is a clone of the “Get Random Value” compound used to retrieve random values which are unique along the strand for every segment.



Minimum

The minimum value

Maximum

The maximum value

Time Var.

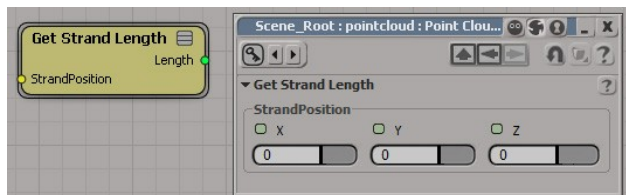
If checked value change over time

Seed

The base random number

Get Strand Length

This compound returns the length of the strand.



StrandPos.

The given strand to calculate the length for.

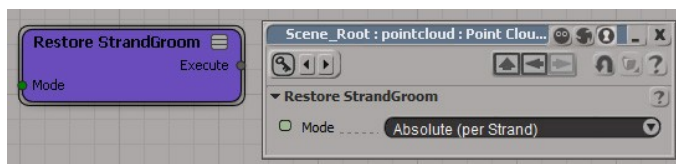
Get Strand Tangents

This one calculates the tangent for all strand segments and returns an array of these vectors.



Restore StrandGroom

This compound restores a stored groom, which can be stored with “Store StrandGroom”.



Mode

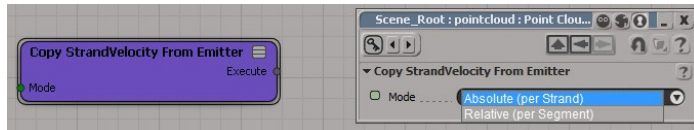
You can choose between relative and absolute modes.

Utility Compounds / Setter Compounds

This compounds are very low-level helpers, that didn't fit in any other category.

Copy StrandVelocity from Emitter

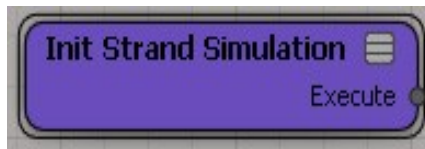
This compound is used to simply set the velocity of the strands to the one of the emitter, in case they strands don't have a velocity themselves due to a missing simulation stack.



Mode You can choose between absolute and relative mode.

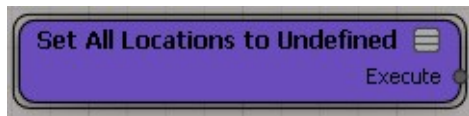
Init Strand Simulation

This node is part of all of the Emitter compounds, however in certain scenarios you will want to add it yourself. It takes care of setting all required attributes for simulation later on.



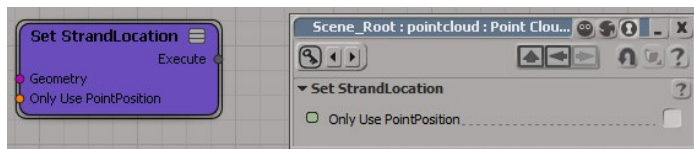
Set All Locations to Undefined

When caching or freezing out a pointcloud sometimes you want to first get rid of all of the pointlocator data. This compound overwrites all of this data with an undefined location.



Set StrandLocation

This compound is able to set the strandlocation quickly again using a closest location node.



Only... If this is checked, the position to do the lookup will be the using only of the point, which literally speeds up the process by the number of segments used.

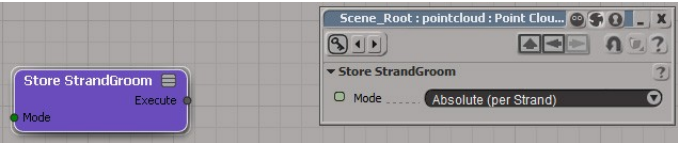
Store StrandLength

This compound can be used to store the strandlength to a custom attribute. The length then can be used later in some of the Simulation compounds. Be aware that this compound is part of all Emitter anyway, and that you only need to use it in very custom situations.



Store StrandGroom

This compound is used to store and remember a given shape / groom of the strand in the current situation. There are two modes: Absolute and Relative. Absolute stores the shape in the reference frame (S R T) of the particle, where as Relative uses every segments' reference frame. When used together with the “Blend StrandGroom”, “Restore StrandGroom” or the “StrandGroom Force”, the modes have to match otherwise it will lead to unexpected results.



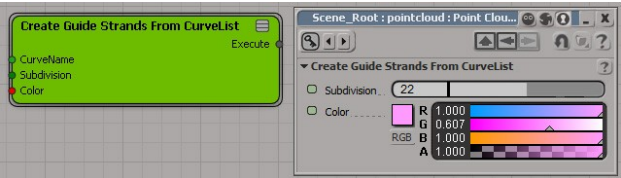
Mode The mode to use, as discussed above: Absolute or Relative

Guide System Compounds

These compounds allow for a layered workflow, similar to Autodesk Maya's fur / hair approach. You can create curve to be used as guides, and then create a more high res melena setup which follows these guides.

Create Guide Strands From Curvelist

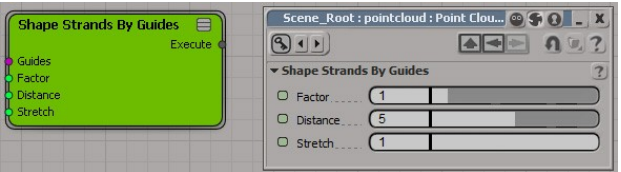
With this compound you can create a single guide curve based on a given nurbscurve.



Subdiv. The number of segments
Color The color for the guide strand

Shape Strands by Guides

This compound can also be found in the Modifiers category, and can be used to shape a strand by using a given guide pointcloud.



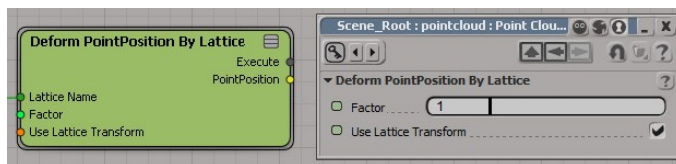
Factor The strength of the shaping
Distance The max distance for the effect
Stretch The amount of allowed stretching

Lattice System Compounds

Melena comes with a very strong custom node to calculate lattice deform through ICE. This allows to use this for deform also, but more interestingly, for simulation. To use the node however we need to first store some values per strand, to initiate the process for later speed improvements.

Deform PointPosition by Lattice

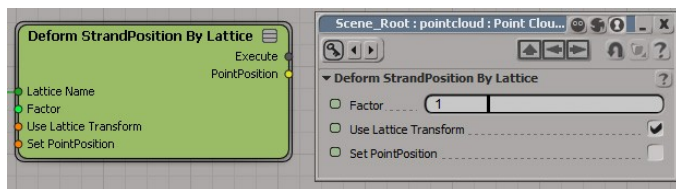
This compound does a normal deformation of the pointposition by a lattice. This can be used for any mesh, curve or particles.



Factor The amount of deformation
Use XF If checked, the lattice transform will be part of the calculation.

Deform StrandPosition by Lattice

This compound does the same as the one for pointpositions, however it works for strands. That way you can deform a complete hair styling done in Melena by a lattice.



Factor The amount of deformation
Use XF If checked, we also use the lattice's transform for deform
Set Pnt. If checked, the Point deforms too

Get Lattice Values

For the tech-savy ICE users, this compound gives access to all of the values returned from a lattice, including the current positions, the reference positions etc.. This compound can also be found in the Getters category.



Init Strand Lattices

This compound is necessary to be added when using the lattice force later on in strand simulation. It stores some intermediate values for later performance improvements. This compound can also be found in the Utilities category.



Lattice StrandForce

This compound calculates a force which represents the deformation vector of a strand to reach the final lattice position. This is very useful for creating directable simulations. This compound needs the “Init Strand Lattices” compound to be added prior in the ICE tree / stack. This compound can also be found in the Forces category.



Factor

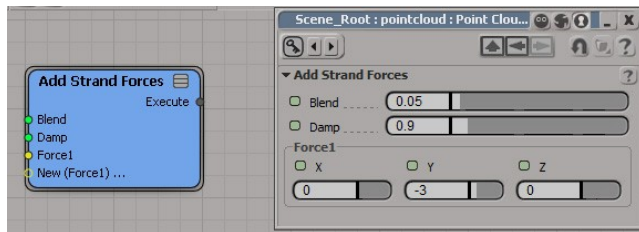
The amount of deformation

Strand Force Compounds

These compounds are forces for the strand simulation part of Melena. Some are very similar to the existing ones for particles, however they work fairly differently. Let's have a look at all of them.

Add Strand Forces

This compound is the hub for connecting all forces together.



Blend

The amount of force interpolation per frame.

Damp

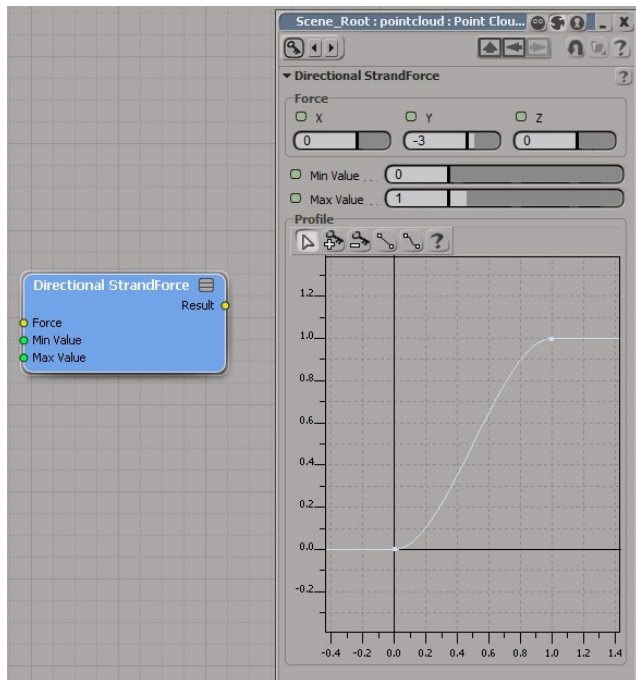
A cut down multiplier to slow down the simulation per frame.

ForceN

The force to be connected here.

Directional StrandForce

This force can be used to represent gravity, or wind. Simply a directional vector force. The force also incorporates a strandcurve, since we found it is the best practice.



Force

The direction vector for the force

Min Value

The value at the root of the hair.

Max Value

The value at the tip of the hair.

Profile

The profile defining the falloff.

Groom StrandForce

This force can be used to force the hair to retain shape during simulation. The force requires to store a strandgroom before being able to use in the simulation stack. See the "Store StrandGroom" compound for further details.



Factor

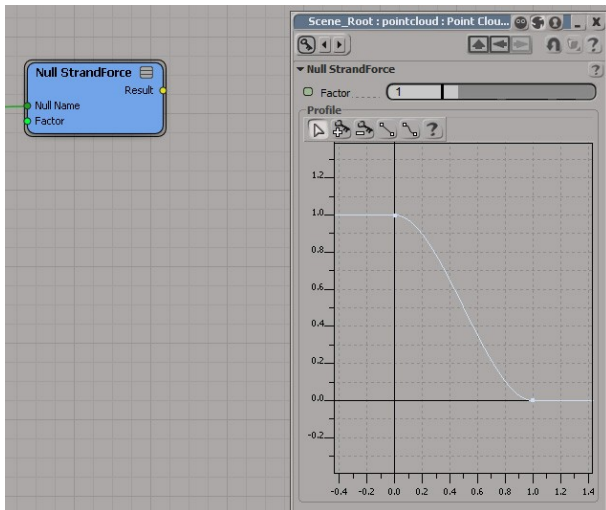
The amount of force to calculate

Mode

The mode of the groom, absolute or relative

Null StrandForce

This compound is a clone, more or less, of the “Null Force” compound used for particle simulation. It uses the same features however it works for strands.



Factor

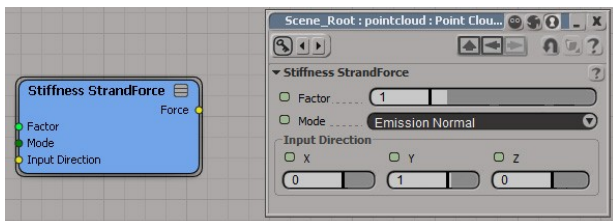
The amount of force to calculate

Profile

The curve to use for the falloff.

Stiffness StrandForce

This compound calculates a force along the strand to straighten it out. This is interesting to emulate stiffness on hair. Be aware that it can produce similar results like the Groom StrandForce, so don't use them in conjunction.



Factor

The strength of the force

Mode

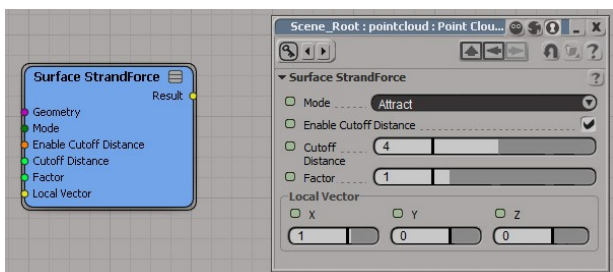
Choose between different modes for the stiffness calculation

Input Dir.

The direction used for one of the modes of the compound

Surface StrandForce

This compound can be used for pushing hair away from a surface, or have it follow a nurbscurve for example. It is fairly interesting to combine this force with stiffness to create a directable but “live” looking simulation.



Mode

You can use attract, repel, pointnormal push or local vector force.

Cutoff Dist.

Define the distance for cutoff

Factor

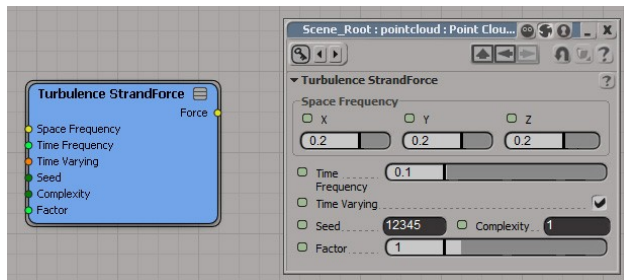
The strength of the force

Local Vector

The vector used by the last mode.

Turbulence StrandForce

This force is similar to the particle turbulence force, however it works with strands.



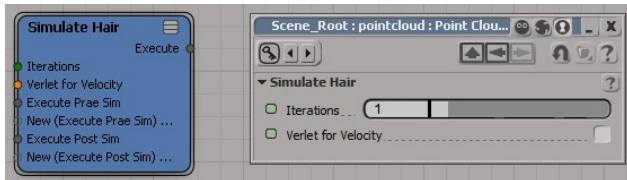
- Space Freq.*** The frequency for the turbulence
- Time Freq.*** The time speed for the turbulence
- Time Var.*** If checked changes over time
- Seed*** The base random number
- Complexity*** The complexity of the turbulence
- Factor*** The strength of the force

Strand Simulation Compounds

These compounds do the heavy lifting for computing Melena's strand simulation. There are various options, two custom nodes for processing and a simple verlet based approach.

Simulate Hair

This compound uses nest's hair solver custom node. The characteristic is slightly different from the other simulation compounds, so try it to get a better idea yourself. This compound is able to simulate strands with different segment lengths.



Iterations

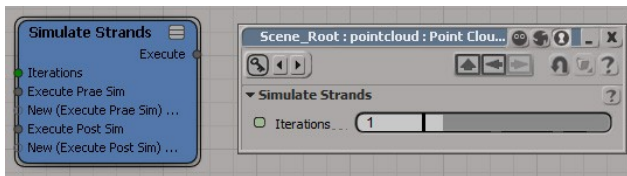
The number of sub frame computations

Verlet

If checked, the velocity is corrected after post sim processes have been applied.

Simulate Strands

This compound simulates the strands without any solver. It simply adds the StrandVelocity to the StrandPosition per frame. This can be interesting if you don't care about stretching but want to simulate a volume of hair / a full styling as is, for example with a lattice.

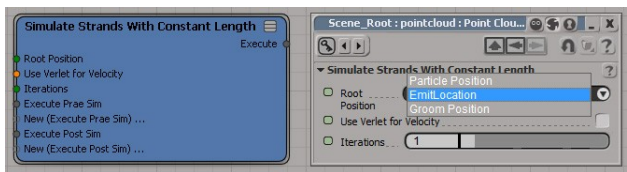


Iterations

The number of sub frames to process the simulation

Simulate Strands with Constant Length

This compound simulate the strands with a simple length-maintenance solver, which is a lot faster than the Simulate Hair solver.



Root Pos.

A dropdown to define what to use for the root position of the strand.

Verlet

If checked velocity is corrected

Iterations

The number of sub frames.