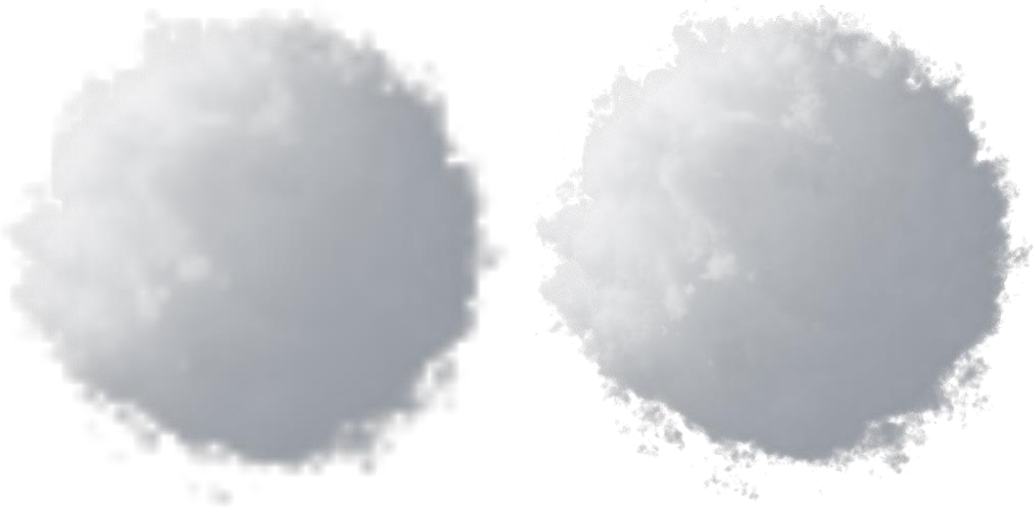
Manual

In this manual we will provide a thorough description of all the attributes of the Elementacular Cloud Shader. We will also discuss how to light your clouds, and finally we will provide some tips on how to improve the performance of the plug-in.

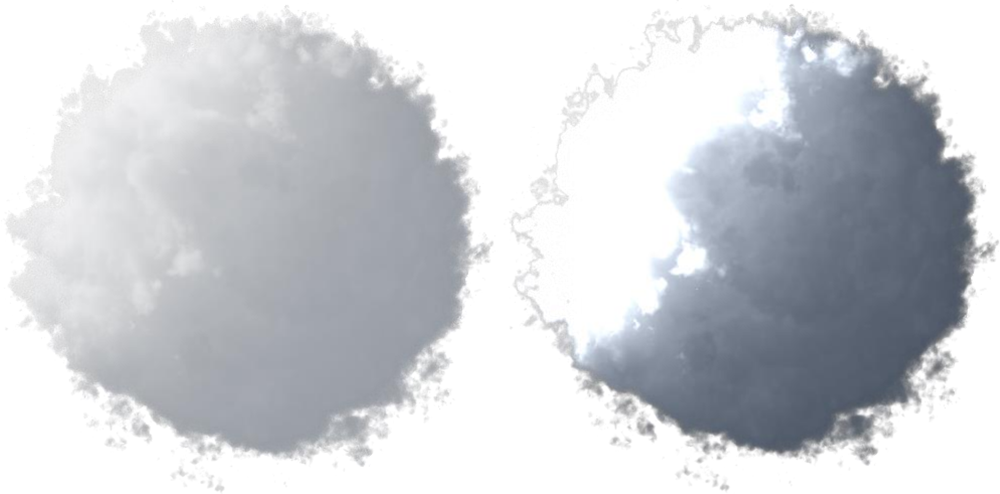
Attributes

In this section, we will walk through all the attributes and describe their purpose. We will provide image examples help illustrate the range of values the given attribute can assume. We will also explain how a given attribute can be coupled to others and how changing that attribute may influence the result in unexpected ways. Finally, we will discuss performance considerations whenever the value of an attribute affects performance.

Render Quality: This attribute controls the resolution of the output. Decreasing its value will give a coarser look, but will also increase performance when moving the camera around and working with other parts of the scene (not including lighting). Set the value back to 1.0 before doing the final rendering.



Tone Mapping: When rendering clouds it is very important to have a high dynamic range since the difference between lighting intensities on the lit and unlit sides is immense. This means that colors are easily saturated in the viewport. The tone mapping attribute enables you to work with the tone mapped, final rendering result directly in the viewport. However, the tone mapped result will not blend correctly with the remaining scene since linear blending does not work after tone mapping. Also, the tone mapping operator tends to wash out the colors of the lighting, so if you're not trying to recreate photorealistic clouds, it might be better to turn it off to get more clear and bright colors.
Before rendering the final result you can turn off tone mapping and render the linear result to a HDR format, and then do the tone mapping during composition.



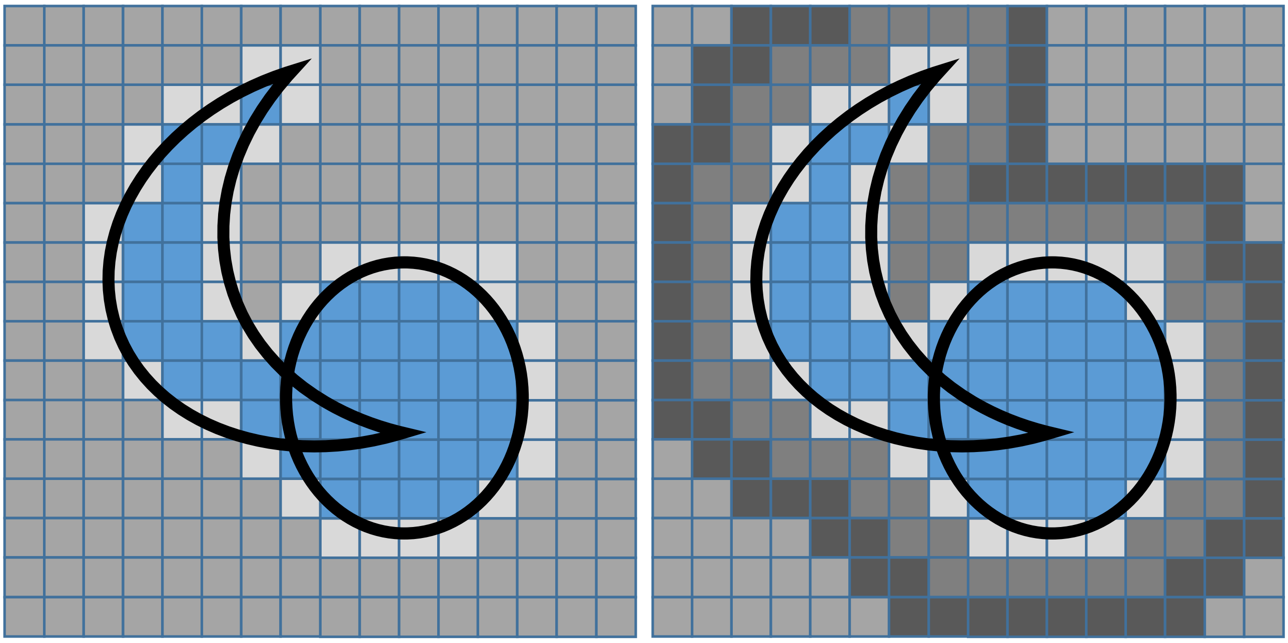
Tone Mapping Exposure: This attribute adjusts the exposure of the tone mapping operator.

Tone Mapping Gamma: The tone mapping operator includes gamma correction. This attribute adjusts gamma.

Distance Resolution: This attribute controls the voxel resolution of the grid, which the input mesh is voxelized unto when it is converted to a cloud. It is important to understand that this resolution does not alone dictate the appearance of the final, rendered cloud. Specifically, the procedural details that are added on top of the basic cloud shape may be sampled at a higher resolution (see the Noise Resolution and Wispy Resolution attributes) and thus provide the appearance of a more detailed cloud. Thus, the Distance Resolution attribute does not need to have a particularly high value. It just needs to match the complexity of the input mesh such that all the desired details of the mesh are correctly voxelized.

Higher resolutions are more expensive to compute, so it might be helpful to decrease the resolution while modifying or adjusting the input mesh in order to improve the responsiveness.

Dilation Layers: This attribute controls the thickness of the band of voxels away from the mesh in which the cloud details can be generated. If this value is set too low, the cloud will appear to be “cut off” by an invisible wall thus creating a very sharp edge. The required value depends on the cloud detail parameters in the sense that more large-scale details require a thicker band of voxels.



Num Steps: This attribute determines the number of steps performed by the ray marcher to march through a single cloud during rendering. Increasing this value improves the quality of the rendered result. When the value is lowered, the rendered result will become noisier. Taking more steps requires more computation time, so it can be helpful to decrease this value while working with the input mesh or the other attributes.

Num Light Steps: This attribute determines the number of steps taken towards the light sources when approximating the lighting due to single scattering. Note that this lighting contribution is only recomputed when necessary, such as when a light source is moved. Decreasing this value will lead to larger errors in the approximated lighting due to single scattering, but the rendered result will not suffer from more noise. If you need to move the lighting in a scene with many light sources, performance should improve when decreasing this value.

Density Scale: This attribute controls the overall density of the cloud. Increase this value to make the cloud appear denser and give the impression of a larger scale. Decrease this value to make a thinner, smaller-looking cloud. Note that the density scale affects the amount of emitted light (see the Density Emission attributes).

Density Falloff: To give the cloud a soft appearance, its density decreases as the distance to the input mesh increases. This attribute determines how quickly the density falls off as the increases. Increasing the falloff makes the cloud appear sharper and wrap the input geometry more tightly.

Density Bias: This attribute adjusts the bias of the mapping from mesh distance to cloud density. A positive value will increase the overall density of the cloud and make it grow while a negative value will subtract density from the cloud.

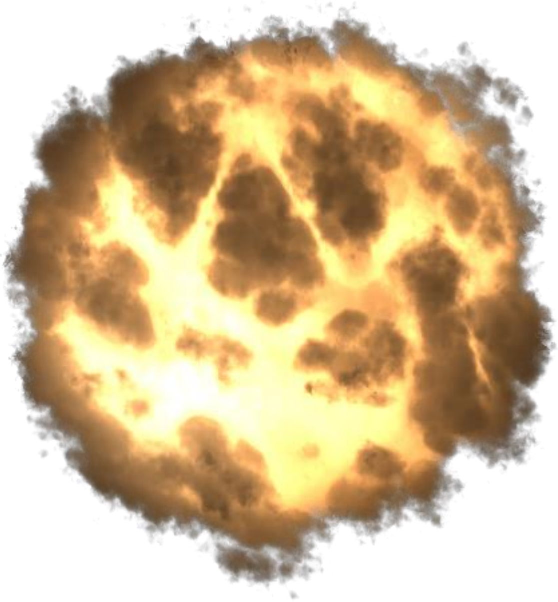
For the technically inclined, the cloud density *d* is computed from the distance *h* in the following manner using the above attributes:

$$d(\boldsymbol{x}) = \left(-\mathbf{Falloff} * h\big(\mathrm{Wispy}(\boldsymbol{x})\big) + \mathbf{Bias} + \mathrm{Noise}(\boldsymbol{x})\right) * \mathbf{Scale}$$

See the Noise and Wispy related attributes for an explanation of the Wispy and Noise functions.

Density Emission Min and **Max:** These attributes control how the distance to the input mesh is mapped to the domain of the Density Emission Ramp and Density Albedo Ramp, which goes from zero to one. Points with a distance equal to or less than the *min* attribute are mapped to zero while points with a distance greater than the *max* attribute are mapped to one.

Density Emission Ramp: This ramp allows for emission of light from the cloud volume. This can be used to create effects such as fireballs. Note that the color values must be quite large since Elementacular attempts to use physical units when computing lighting.



Density Albedo Ramp: This ramp allows for a custom definition of albedo, which changes with the distance to the input mesh. Albedo indicates the amount of scattering events compared to the amount of absorption events in a material. A value of 1 means that all events are scattering events and thus no energy/light is lost. A value of 0 means that all events are absorption events and thus all energy/light is lost at the first event. Clouds are typically high-albedo while smoke could have a lower albedo since the soot particles tend to absorb more light.

Noise Gradient Ramp: This ramp allows you to modify the noise signal used to create cloud details (see Noise attributes) as a function of the y-position within the cloud. This can for instance be used to create cumulus clouds, which tend to be smoother at the bottom.

Ambient Color: This color is added to the rendered result. Think of it as a constant emission of light to adjust the hue and lightness of the cloud.

Ambient Scale: The value of this attribute is just a convenient way of scaling the ambient color attribute.

Multiscatter Method: One of the most important phenomena to reproduce when rendering realistic clouds is the way much of the light scatters around millions of times inside the cloud before leaving it. This is what creates the very soft and flat lighting on the underside of clouds. Since the raw computation of these light bounces is not feasible, Elementacular provides two different approximations, which differ in performance and quality. *Flux-Limited Diffusion* (FLD) gives the most accurate results by simulating the scattering as a diffusion process. However, it can also be computationally expensive. *Multiple-single-scatter* simply simulates the whole scattering process as (up to) three layers of single scattering.

FLD:

Multiscatter Resolution: This attribute controls the resolution of the FLD computation. It can be set to a low value since the effect of multiple scattering is usually smooth and of a low frequency. However, the resolution should match the level of detail and complexity in the input mesh to capture all the details. Also, note that voxels on the boundary of the computation grid will have an incorrect value and choosing a low value might make them visible as dark spots near the edge of the cloud.

Singlescatter Scale: This attribute allows you to scale the contribution of light coming from single scattering.

Multiscatter Scale: This attribute allows you to scale the contribution of light coming from the multiple scattering computation.

Multiple-single-scatter:

Multiscatter First Band Extinction Scale: This attribute determines how much of the incoming light is lost when traveling through the cloud. Decreasing the value of this attribute will let the light travel further into the cloud before being absorbed or scattered away from the camera.

Multiscatter First Band Inscatter Scale: This attribute determines the fraction of light that is scattered towards the camera at each point in the cloud. Increasing this value will give a larger light contribution.

Multiscatter Second Band Extinction Scale, Multiscatter Second Band Inscatter Scale, Multiscatter Third Band Extinction Scale, Multiscatter Third Band Inscatter Scale: These attributes allow you to control the second and third layer of single scattering.

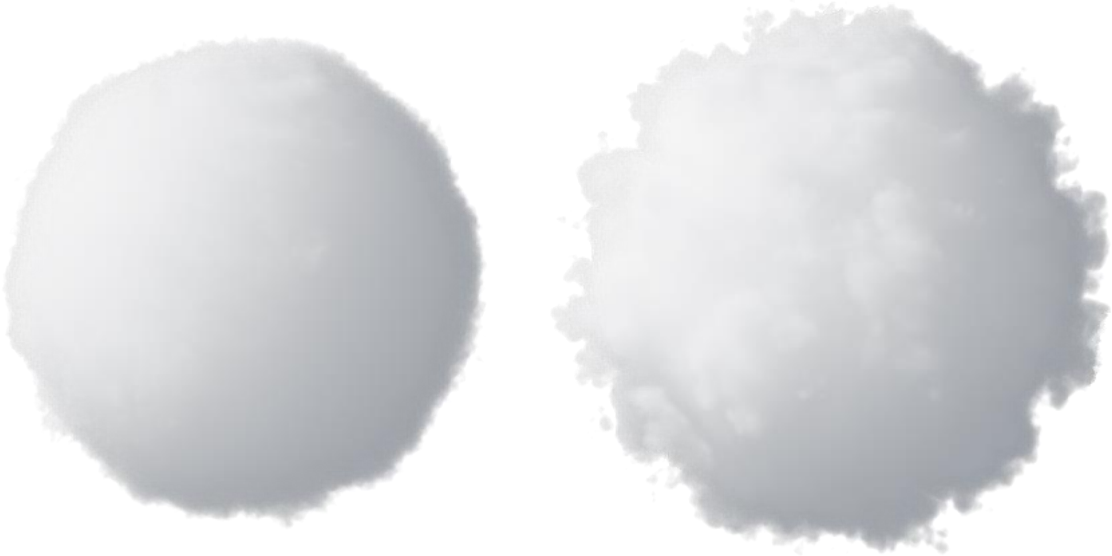
Noise Time: This attribute controls the temporal dimension of the procedural noise layer, which generates the fine-scale details of the rendered cloud. Changing or animating this value will update the noise in a directionless manner.

Noise Offset: This attribute controls the spatial offset of the procedural noise layer. Changing or animating this value will make the noise move in a certain direction.

Noise Resolution: This attribute controls the resolution that the procedural noise layer is sampled at. The resolution of the noise field largely dictates the overall appearance of the cloud and therefore it should be set to a high value if a detailed cloud is desired. The noise field is fast to compute and is only updated when necessary.

Noise Octaves: The procedural noise layer consists of several octaves of noise. This attribute controls the number of octaves included when computing the noise. Increasing this value enables a larger span of frequencies in the noise signal but also increases the computation time.

Noise Base Roughness: This attribute controls the amplitude of the first, base octave of noise. A higher value will introduce noise that is more pronounced.



Noise Base Frequency: This attribute controls the frequency of the first octave of noise. Increasing this value results in more detailed, high frequency noise.

Noise Base Frequency Scale: This attribute allows you to scale the frequency of the first octave of noise individually along the x-, y- and z-axis.

Noise Gain: This attribute determines the amplitude gain of each octave in the noise signal. A value of 0.5 means that each octave will have half the amplitude of the previous octave.

Noise Lacunarity: This attribute determines the frequency gain of each octave in the noise signal. A value of 2.0 means that each octave will have twice the frequency of the previous octave.

Noise Billow: Clouds tend to have less detail in the “canyons” or “valleys” between the highly detailed “peaks” – that is, the high frequency parts of the noise signal should be suppressed in areas where the contribution of the previous octave was small. This attribute determines how much the high requency parts of the noise signal is dampened in areas with a small low frequency noise signal. Increasing this value will increase the dampening.

Noise Q: This attribute controls how large the contribution of an octave should be before avoiding any dampening. Increasing the value will apply the dampening in more areas as well as increase its effect. The desired value of this attribute depends on the chosen noise signal and you should expect to adjust it whenever you change the Noise Base Roughness or Noise Gain.

For the technically inclined, the dampened value of the *i*th octave N_i^C is computed from the original value N_i and the dampened value of the previous octave N_{i-1}^C as follows:

$$N_i^C(\boldsymbol{x}) = N_i(\boldsymbol{x}) * \text{clamp}(\frac{N_{i-1}^C(\boldsymbol{x})}{\mathbf{Q}}, 0, 1)^{\mathbf{Billow}}$$

The final value of the noise signal is simply the sum of the dampened octaves:

$$\text{Noise}(\boldsymbol{x}) = \sum_{i=0}^{\mathbf{Octaves}} N_i^C(\boldsymbol{x})$$

Wispy Method: On top of the procedural noise layer described in the previous attributes, is a layer dedicated to creating the wispy look of clouds. This look is created by advecting the cloud densities through a procedural noisy velocity field in a timeless manner. Elementacular provides two methods for generating this velocity field. *Perlin* simply creates a velocity field by sampling a simplex noise function at slightly offset positions for the x, y and z components. This is very fast to compute, but the generated velocity field is not incompressible, which is physically incorrect. The other method *Curl* computes the velocity field as the curl of three noise-based scalar fields. This is more expensive to compute but the result is incompressible, which can give better results for long wisps.

Wispy Time: This attribute controls the temporal dimension of the procedural wispy layer. Changing or animating this value will update the wisps in a directionless manner.

Wispy Offset: This attribute controls the spatial offset of the procedural wispy layer. Changing or animating this value will make the wisps move in a certain direction.

Wispy Resolution: This attribute controls the resolution that the procedural wispy layer is sampled at. If highly detailed wisps are required, you should set the resolution to a high enough value to capture the high frequencies of the wisps.

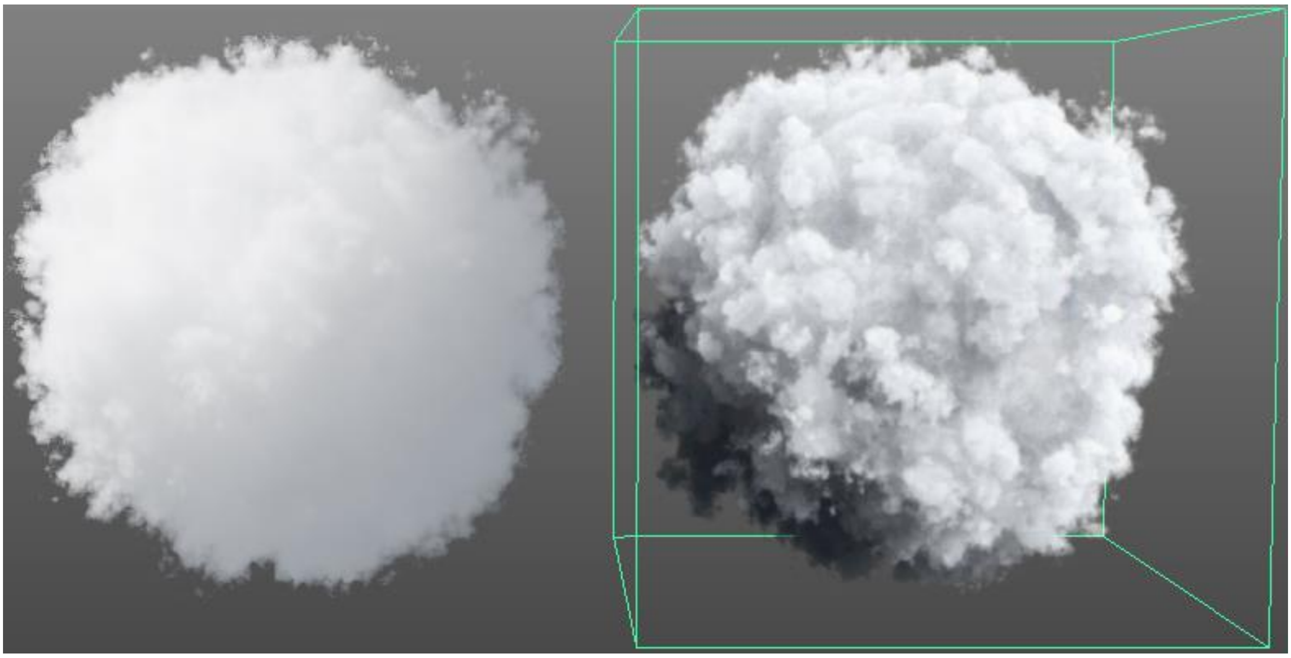
Wispy Roughness: This attribute controls the length of the wisps. A higher value will introduce wisps that are more pronounced. Please note that changing this value will recompute the wispy field. In contrast, changing the Wispy Length will simply modify the value every wispy sample is multiplied with during rendering.

Wispy Frequency: This attribute controls the frequency of the wisps. Increasing this value results in more detailed, high frequency wisps.

Wispy Length: This attribute controls the length of the wisps. A higher value will introduce wisps that are more pronounced. Please note that changing this value simply modifies the length of the wisps during rendering without recomputing the wispy field. Also, note that having very long wisps incurs a performance penalty on the rendering due to more costly memory access patterns on the GPU. Instead, it might be possible to achieve the same effect using the noise layer.

Wispy Steps: Since the wispy look is achieved by advecting the cloud densities, it can be beneficial to break up the advection into several minor steps to achieve a more accurate, detailed look for long wisps. This attribute controls how many steps the advection is split into. Note that a higher number of steps is more expensive to compute.

Fluid resolution: Elementacular has support for converting the densities of your cloud to a Maya fluid container. This attribute controls the resolution of the resulting fluid container. Currently, you cannot convert other values such as emission or color.



Lighting

Elementacular has support for eight simultaneous directional lights as well as eight simultaneous point lights. Currently, no other light types are supported.

Elementacular uses physical quantities when doing lighting. This means that the light source intensities should be set to a fairly high value to achieve a visible lighting. Typically, setting the intensity to around 100-500 gives a good result. You might need to use dedicated light sources for lighting the cloud, as these intensities typically do not work well for the remaining scene.

Elementacular supports light linking for disabling and enabling the contribution of certain light sources. You should disable all light sources that are not explicitly used for lighting the cloud as the performance of the lighting computation scales with the number of involved light sources.

Performance

To improve the rendering performance when moving the camera around, you can try the following steps:

- Decrease the Num Steps attribute
- Decrease the Render Quality attribute
- Decrease the Wispy Steps attribute

To improve the performance when moving light sources around, you can try the following steps (in addition to the previous steps):

- Decrease the Num Light Steps attribute
- Decrease the Multiscatter Resolution attribute

- Change the Multiscatter Method attribute to Multiple-single-scatter
- Decrease the number of lights interacting with the cloud through light linking

To improve the performance when sculpting the cloud, you can try the following steps (in addition to the previous steps):

- Decrease the Distance Resolution attribute
- Decrease the Noise Resolution attribute
- Decrease the Wispy Resolution attribute