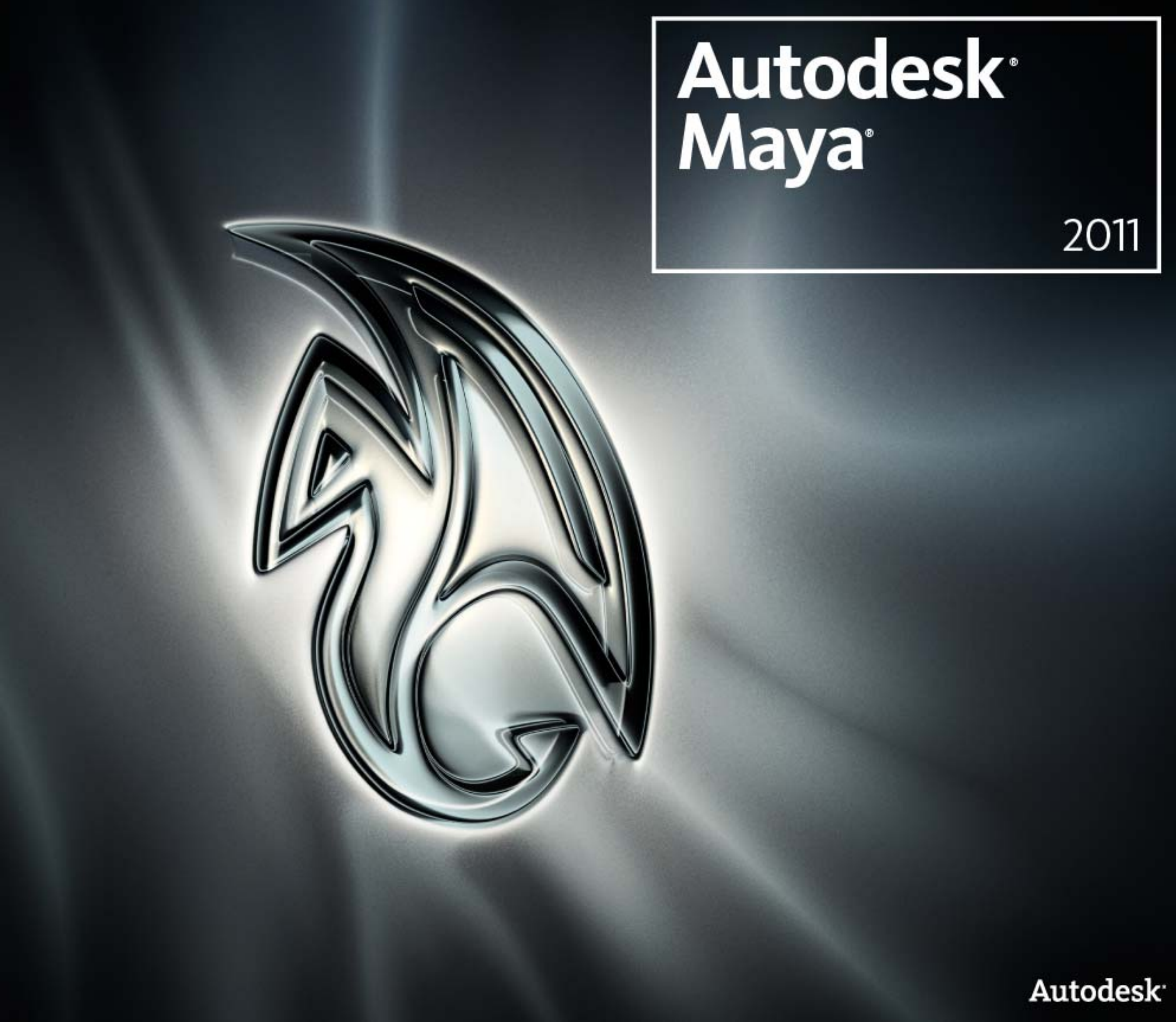


Autodesk® Maya® 2011

Autodesk®

Toon

Copyright Notice

Autodesk® Maya® 2011 Software

Disclaimer

THIS PUBLICATION AND THE INFORMATION CONTAINED HEREIN IS MADE AVAILABLE BY AUTODESK, INC. "AS IS." AUTODESK, INC. DISCLAIMS ALL WARRANTIES, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE REGARDING THESE MATERIALS.

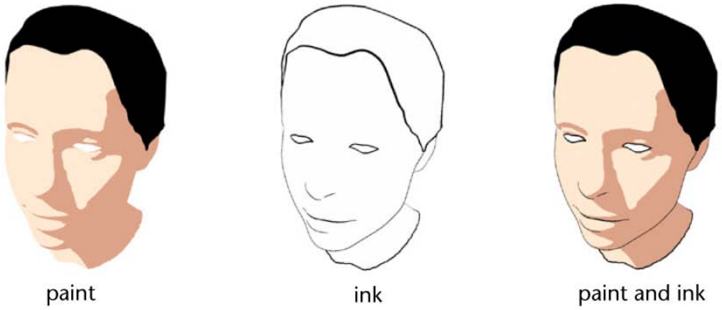
Contents

Toon shading

1

Toon shading overview

Toon shading creates the look of 2D cel or cartoon animation using 3D modeling and animation software. Elements of the “toon” look include profile lines (outlines), border lines, crease lines, intersection lines, and solid color shading. Combined, these elements recreate the look of traditional animation’s “ink and paint” technique, where *ink* refers to lines and *paint* refers to shading.



Model by Inspeck

The Toon menu in the Rendering menu set contains various items and tools specifically for creating toon shading, which lets you:

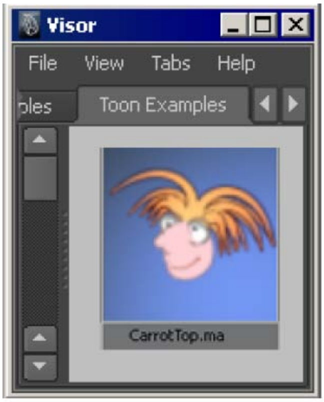
- add fill shaders to surfaces or use the Ramp shader to create a custom fill shader
- add profile lines, with the option of using offset surfaces or Maya® Paint Effects™, to create a toon line look to outline everything
- add border lines

- add crease lines
- add intersection lines
- set a background color for rendering
- assign a Paint Effects brush to toon lines
- convert toon lines to polygons
- modulate line character by painting, applying a toon line modifier, or editing other toon line attributes

NOTE Maya displays a warning message if your video card is not qualified for hardware rendering, which can affect the display in the scene view (for example, Toon shading). In this case, shading in the scene view is what you get with non-High Quality rendering, or when not using the hardware renderer.

Toon examples

You can import example files of toon line and shading effects. Modify these files to customize the toon styles, use them as learning tools to understand how to create your own toon styles, or use them simply for ideas.



Character by Duncan Brinsmead and Rick Kogucki

To import a toon example into a scene

1. In the Rendering menu set, select Toon > Get Toon Example. The Visor window opens to the Toon Examples tab.

2. Right-click the desired Toon example and select Import. Autodesk® Maya® imports the pfxToon line nodes, shaders and surfaces into your scene.

   The Notes section at the bottom of the Attribute Editor provides information about the Toon example, pointing out the key aspects of its construction.

Adding toon lines and shading to a scene

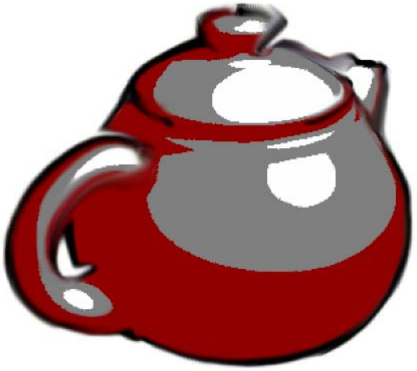
You can use toon lines on all three types of geometry—NURBS, polygons and subdivision surfaces—as well as on strokes, Paint Effects brushes and Paint Effects hair. The lines are generated directly from a combination of the input geometry and the current view. You can control the appearance of these lines with the attributes in the **pfxToon** node; for descriptions of these attributes, see pfxToonShape node on page 29. You can also paint additional lines using Paint Effects.

To add toon shading to a scene

1. Create the 3D geometry you want to render in 2D.
2. Assign Toon shaders to the geometry (Toon > Assign Fill Shader).
3. Assign Toon outlines to the geometry (Toon > Assign Outline).
4. Optionally, set a background color for the rendered scene (Toon > Set Camera Background Color).
5. Render the scene.

   Notes on rendering:

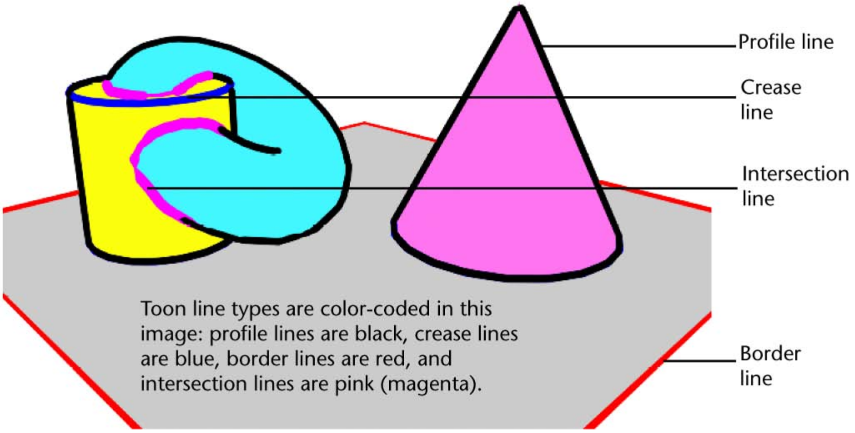
   - You can render all toon lines using the Maya Software renderer.
   - If you used the Paint Effects method for profile lines, then before you render in the Maya Software renderer, turn on the Oversample and Oversample Post Filter checkboxes in the Paint Effects Rendering Options section of the Maya Software tab in the Render Settings window.
   - To render Paint Effects toon lines using the mental ray® for Maya or the Maya Hardware renderers, you must first convert the toon lines to polygons. See To convert toon lines to polygons on page 14. Alternatively, use the Offset Mesh method for profile lines, which will render for these renderers. (You can set this method in the Profile Lines section of the pfxToonShape node.)



The blurry toon line effect was created by using a second toon line with a Paint Effects "smear" brush attached to it.

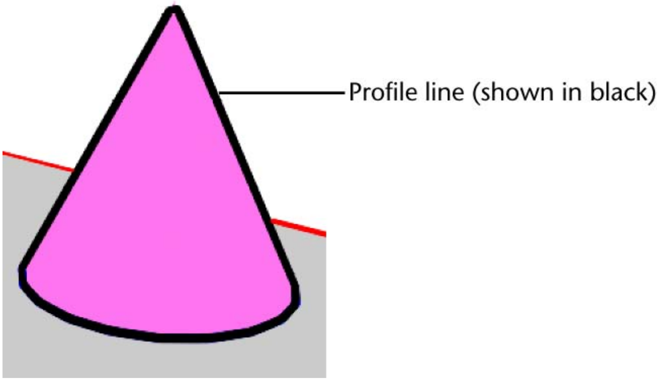
Types of toon lines

There are four types of toon lines you can edit in Maya: *profile*, *border*, *crease*, and *intersection* lines. These lines are defined in terms of mesh properties; therefore all geometry types other than meshes are converted to polygon meshes before being passed to the pfxToon node.

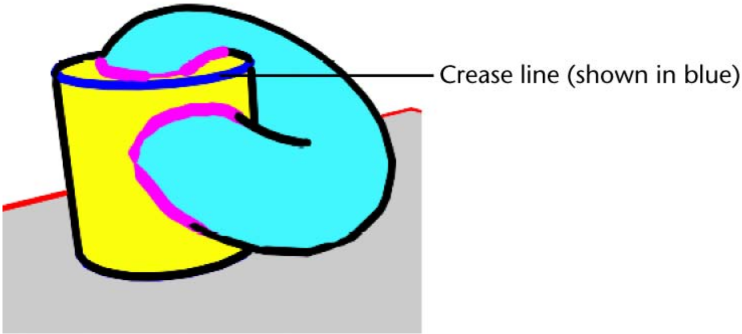


Toon line types are color-coded in this image: profile lines are black, crease lines are blue, border lines are red, and intersection lines are pink (magenta).

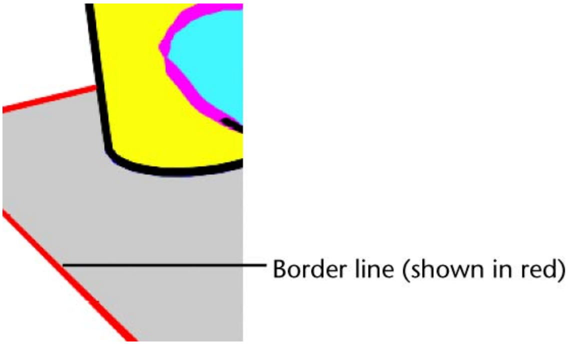
Profile lines Profile lines are silhouette lines occurring where an object turns to be facing away from the view. If using Paint Effects for profile lines, Maya regenerates these lines when the view changes. If using Offset Mesh for profile lines, updating is inherent, so the refresh is not slowed down the way it is with the Paint Effects method. Depending on the view, profile lines may coincide with crease lines (when the crease edges are also in silhouette). For attributes you can use to modify the appearance of profile lines, see Profile Lines.



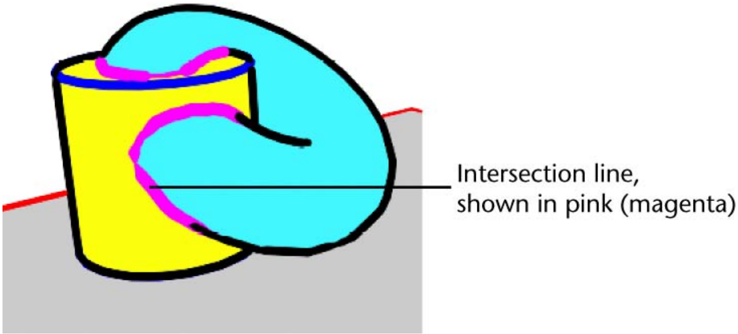
Crease lines Crease lines are formed on polygon edges where the bend angle is greater than the defined Crease Angle Min value. Unlike profile lines, crease lines are not view-dependent. For example, you would use them to draw the edges of a cube, even when the edge is facing you. For attributes you can use to modify the appearance of crease lines, see Crease Lines on page 30.



Border lines Border lines are on edges that are only part of a single polygon face. For example, the edges of a simple plane are border lines. For attributes you can use to modify the appearance of border lines, see Border Lines on page 30.



Intersection lines Intersection lines occur at the intersection where objects overlap other objects; for example, toon eyeballs can be created by sticking spheres into a head and using the intersection lines to show the cartoon eyes.

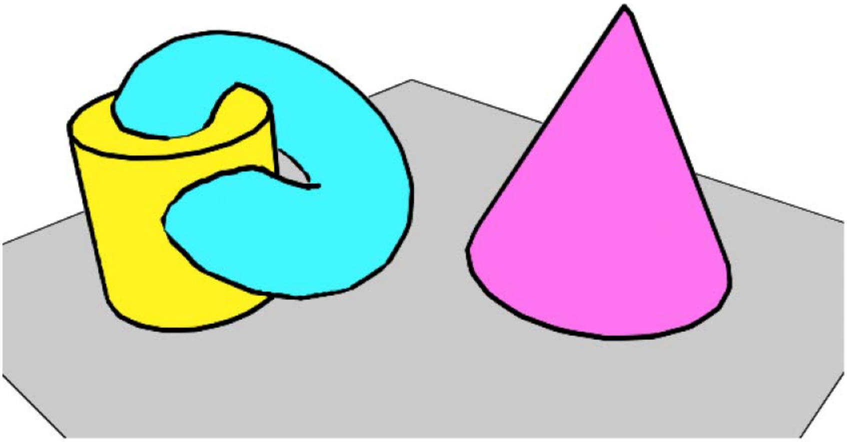


For two objects to have intersection lines between them they must be assigned to the same toon line node. However, you can layer multiple toon line nodes on objects, and if desired use several different toon line nodes just for intersections and others for profile and crease lines.

You can control the appearance of intersection lines using the attributes in the Intersection Lines section of the pfxToon node. See Intersection Lines on page 30. These attributes are similar to those for profile and crease lines and they work in much the same way as they do for the other line types. There is a unique Self Intersect attribute. The Intersection Angle Min/Max attributes control the thick/thin variation of the lines based on how steep the angle is where the surfaces intersect. By making the Min value greater than zero you can avoid creating intersection lines where the two surfaces are relatively flush. You can also generate NURBS curves with history along the intersections using Modify > Convert > Paint Effects to Curves.

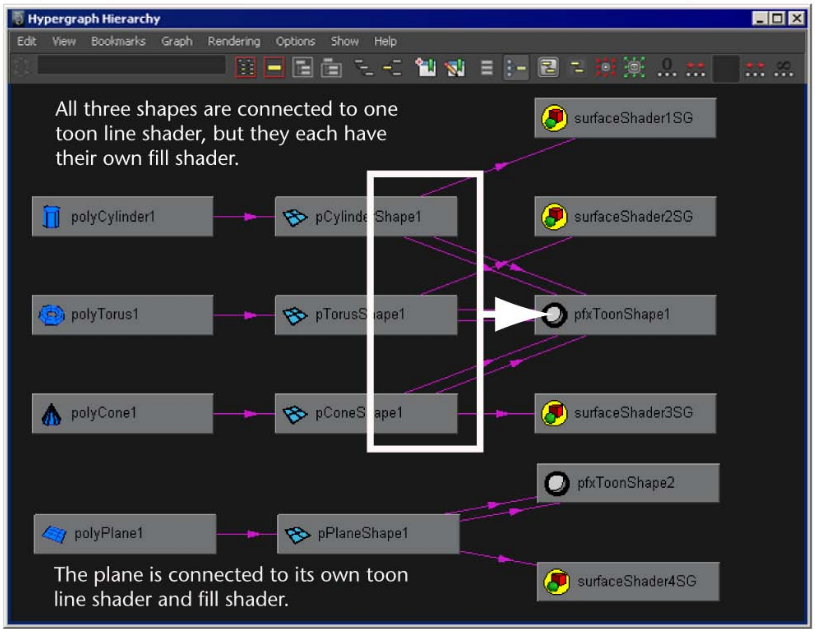
Toon line connections

The various toon shading nodes are connected to the object and the camera so that changes in curvature are detected. Each fill shader and toon line shader has its own node, but you can connect multiple objects to the same fill shader node or toon lines node. For example, in the following image all three objects—the torus, cone and cylinder—are all connected to the same toon line node (pfxToonShape1). However each of these objects has its own unique fill shader. The plane that represents the floor has its own toon line node (pfxToonShape2) and fill shader (surfaceShader4SG). You can see all the connections in the second image.



In this image, the three objects—a torus, cone and cylinder— share the same toon line node. These toon line settings include using the Paint Effects mode for Profile Lines, turning on Intersection Lines, and using the profile line Resample option. The torus is intersecting the cylinder, thereby illustrating intersection toon lines. The toon line node connected to the plane uses the Open Edge mode for Border Lines.

The geometry, toon line and fill shader connections below are related to the previous image.



All three shapes are connected to one toon line shader, but they each have their own fill shader.

The plane is connected to its own toon line shader and fill shader.

Work with toon lines

Hide and show toon lines

Toon lines are Paint Effects strokes, so you can hide them using any of those related methods, or you can turn off the Display In Viewport option in the pfxToon node. For more information, see Troubleshoot toon lines on page 21.

To hide toon lines

➤ Do one of the following:

- Select the toon lines you want to hide and, in the pfxToon node, turn off the Display In Viewport checkbox.
- In the panel menu, select Show > Strokes to unchecked.

To show toon lines

➤ Do one of the following:

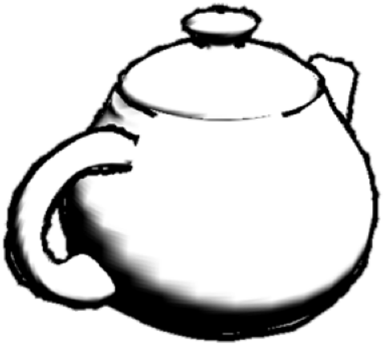
- Select the geometry containing toon lines you want to show and, in the pfxToon node, turn on the Display In Viewport checkbox.
- In the panel menu, select Show > Strokes.

Modify toon line attributes

You can modify various toon line attributes in the pfxToon node. There are also attributes unique to line type, including profile, crease, border and intersection lines. You can create multiple toon line nodes for different effects; for example, you may want to have a unique toon line node for crease lines, including curvature modulation.

To modify toon line attributes

1. Select the toon lines you want to edit.
2. Edit the desired toon line attributes in the pfxToon node of the Attribute Editor. For descriptions of these attributes, see pfxToonShape node on page 29.



In this image, a fractal texture was mapped to the Line Offset Map attribute in the pfxToon node.

Paint toon line attributes

You can paint toon lines with certain attributes—Line Width, Line Offset, Line Opacity, Profile Color, Crease Color, and Border Color. By painting your own toon lines, you can control the look of toon lines in specific places.

To paint toon line attributes

1. Select the toon line you want to paint.

   Click the line itself, or click a surface if that surface has only one toon line on it.

2. Select Toon > Paint Line Attributes and from the menu select the line attribute you want to change.

   The 3D Paint Tool Settings editor appears. If you don't see it, click on the Show or hide the Tool Settings icon in the upper right of the Maya window.

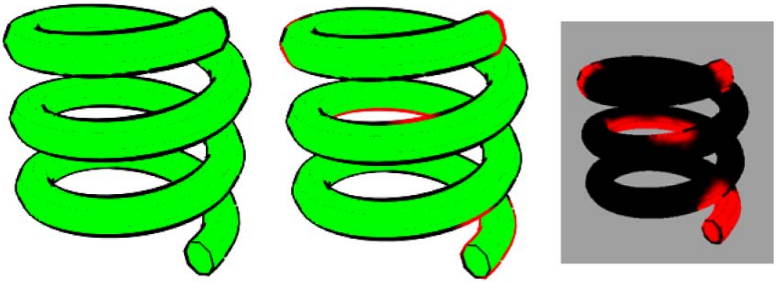
3. The Attribute to Paint option in the File Textures section tells you the line attribute you are painting. To change this attribute, you must use the menu items from Toon > Paint Line Attributes.

4. Make sure Update on Stroke is turned on in the File Textures section.

5. Drag the Color slider in the Color section. For example, if painting Line Width, painting with the color black will reduce the line width value and painting with color white will return the line width to its original value before painting.

6. Paint the toon lines.

NOTE While painting, a temporary shader for painting feedback is applied to the surface. The original shader is restored on exiting the tool. Simply re-entering the 3D Paint Tool will not resume painting. You must always invoke the tool from the menu, Toon > Paint Line Attributes. Do not select the last used tool in the Toolbox as this will load the 3D Paint Tool, but will not load it properly for painting toon line attributes.



After painting the Profile Color of the helix toon lines red, you can see in the second image which toon lines were affected. The third image is the texture map illustrating the areas painted red, resulting in red profile lines.

Use Paint Effects with toon lines

You can assign a Paint Effects brush to toon lines to create interesting effects, for example, a smeared toon line, a metallic toon line, and so on.

You can also paint additional toon lines using a Paint Effects brush. See Paint additional lines in toon scenes on page 13.

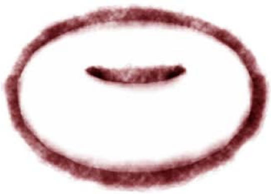
To attach a Paint Effects brush to toon lines

1. Select the toon lines to which you want to assign a Paint Effects brush.
2. Select a Paint Effects brush by doing one of the following:
   - Select Paint Effects > Get Brush and select a brush from the Visor window. Any Paint Effects brush can be used, but those in the Toon folder are particularly suitable.
   - Select a brush from the Paint Effects shelf.
3. Select Toon > Assign Paint Effects Brush to Toon Lines. The selected brush is attached to the selected toon lines.
4. Edit the brush attributes. Consider changing the Global Scale value of the brush to 1 so that you can scale the line using toon attributes.

TIP After assigning a Paint Effects brush with tubes turned on (for example, a vine brush), in order to eliminate the “jumping around” that occurs when the toon lines redraw, do the following:

- turn on Smooth Profile
- turn on Resample Profile
- increase the Min Segment Length for resampling
- set the Max Segment Length equal to the Min for resampling

Before assigning a Paint Effects brush to toon lines



After assigning a Paint Effects pastel brush to the toon lines

After decreasing the brush’s Global Scale value

Use a modifier to alter toon line and brush attributes

You can use a modifier to manipulate certain toon line and brush tube attributes. This can be used to locally edit line width, color and other characteristics. For example, you can edit the shape of Paint Effects trees, prune their leaves, locally modify color, get rid of branches, and so on. And with toon lines, you can locally edit line width, surface offset, line color, and more.

The same modifier can be connected to multiple nodes; for example, you could have many line modifiers assigned to the same object. The influences of the different modifiers blend together. The line modifier node is a sphere or cube shape that affects lines based on worldspace position. You may want to parent or constrain the modifier to objects so that it moves with them.

To apply a line modifier to toon lines

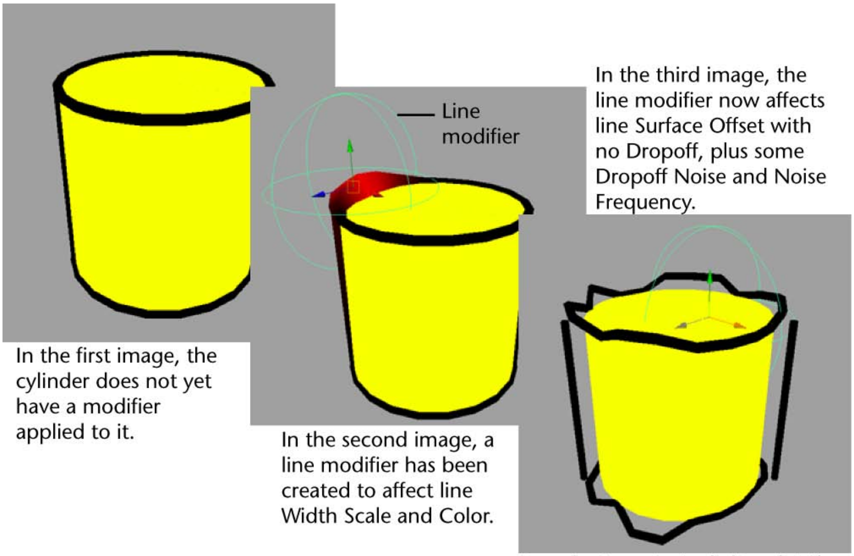
1. Select the toon lines to be influenced by the modifier.
2. Select Toon > Create Modifier. A lineModifier node is created.

3 Select the shape of the modifier, a Cube or Sphere.

4 Modify the attributes in the lineModifier node that you want to have affect the toon lines. See lineModifier node on page 46.

5 Transform the modifier to see how the toon lines are influenced.

6 Optionally, parent the modifier to the object(s) so they move together.



In the first image, the cylinder does not yet have a modifier applied to it.

In the second image, a line modifier has been created to affect line Width Scale and Color.

In the third image, the line modifier now affects line Surface Offset with no Dropoff, plus some Dropoff Noise and Noise Frequency.

In order to get wiggly line detail along the sides of the cylinder you can turn on Resample Profile and Resample Crease, plus lower the Max Segment Length value in the Lines Resampling section of the pfxToon node.

Paint additional lines in toon scenes

If you want to paint additional toon lines (for example, a mouth on a character or a single line down the very center of a nose), you can use a Paint Effects brush. You can even animate the painted lines by keyframing the Pressure Scale attribute on the brush.

To paint additional lines

1. Select the surface on which you want to paint and select Paint Effects > Make Paintable.
2. Select a Paint Effects brush (Paint Effects > Get Brush).
3. Draw on the surface of the object where you want the toon line to appear.
4. Optionally, edit the stroke Pressure Scale ramp to modulate the line width.

The character's mouth is a Paint Effects stroke.

Convert toon lines to polygons

You may want to convert toon lines to polygons to manipulate them as geometry, or to render the toon lines in mental ray or in the Maya Hardware renderer.

To convert toon lines to polygons

1. Select the geometry containing the toon lines you want to convert.
2. Select Toon > Convert Toon to Polygons > ❐ . The Convert Paint Effects to Polygons Options window appears (converting toon lines is really converting Paint Effects).
3. Set the desired options and then click Convert. See below for descriptions of the options in the Convert Paint Effects to Polygons Options window. **Vertex Color Mode** The default is None. Other options include Color and Illuminated.

Quad Output Turn this on to output to quads. The default is off, defaulting output to triangles.

Hide Strokes Turn this on to hide the toon lines once they're converted to polygons. Even if a toon line is hidden, its attributes can still be updated (in the pfxToonShape node) and will affect the polygon mesh. The default is on.

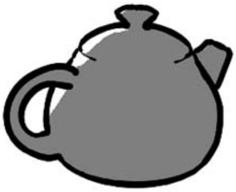
Poly Limit This value indicates that when converting the toon lines to polygons, it should stop after it has reached approximately this number of faces. This can be used to avoid creating excessively large meshes that could cause you to run out of memory. The default is 100 000. A value of 0 indicates no limit.

Work with toon shaders

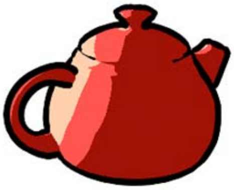
There are a series of default toon shaders available in the Toon menu, all designed to work with simple lighting. When you assign a toon shader to selected geometry, a shader node is created. You can edit the shader in the Attribute Editor. All of the toon shaders, except the Solid Color shader, are based on the Ramp shader. On a Ramp shader you can create multiple indices (tones); for more information, see Ramp Shader in the *Shading* guide. By turning on High Quality Rendering, you can get an accurate preview in the scene view of what the toon shaders will look like when rendered.

Solid Color - fills the surface with a constant non-shaded color

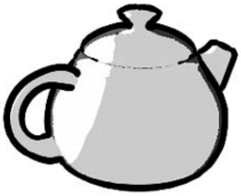
Light Angle Two Tone - indexes a ramp color based on the light angle, brightness based on light brightness



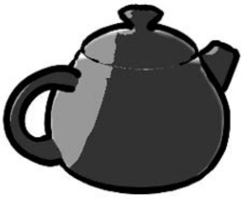
Shaded Brightness Two Tone - indexes a ramp color based on the diffuse shaded brightness



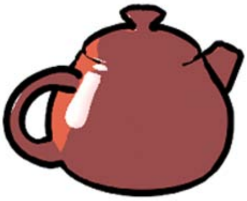
Shaded Brightness Three Tone - indexes three colors based on the diffuse shaded brightness



Dark Profile - threshold the color based on the view angle with the edges



Rim Light - edges have a white highlight



Circle Highlight - a light angle shader with a round hard-edged specular highlight

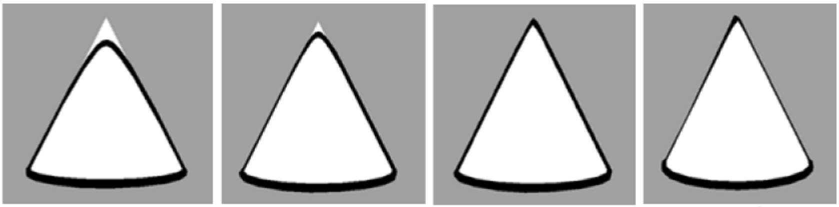
Tips for working with toon shading

- While you can create one toon outline for all toon lines, you may want to use multiple toon line nodes so you can customize toon lines. Otherwise changes to the toon line node affect all objects connected to that toon line (pfxToon) node.

- Consider using multiple toon shaders in a scene; otherwise the scene view update may take a long time and the dependency graph may be cluttered.

- You can turn off the display of toon lines in the scene view by turning off the Display In Viewport attribute in the pfxToon node. Turning this off does not affect the display of Offset Mesh profiles, only the Paint Effects toon lines. To turn off display of Offset Mesh profiles, turn off the Visibility option in the pfxToonShape node. (Offset meshes are grouped and you can set visibility on the group.)

- For some types of geometry, toon lines may not appear correctly. For cones, the toon lines may appear rounded at the tip because the cone has long thin triangles meeting at a sharp point. (See the image below.) To correct this, turn on Tighter Profile (in the Profile Lines section of the pfxToonShape node). Alternatively use the Offset Mesh option for profile lines, or increase the number of subdivisions on the mesh.



Smooth Profile turned on

Smooth Profile and Tighter Profile turned on

Smooth Profile turned on and the number of Subdivisions Height increased

No Smooth Profile

- If the profiles lines are jagged when using the Paint Effects option, you can resample the segment length. To adjust line segments, use the attributes in the Line Resampling section of the pfxToonShape node.

- Set the Line Width to zero where you want no lines to appear.

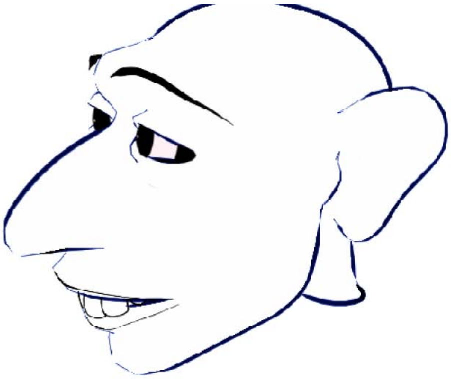
- Some lines are best created with texture mapping, for example, eyebrows.

- Toon lines do not respect switch nodes.

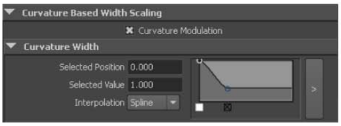
- Use the Line Extend attribute in the pfxToonShape node to extend toon lines at intersections.

- Use the Edge Clip and Edge Clip Width attributes in the Paint Effect’s brush Attribute Editor for drop shadows and to fake shadows. For more information, see Edge Clip, Edge Clip Depth in the *Paint Effects, Artisan, and 3D Paint* guide.

- Use Curvature Modulation to scale line width based on curvature (curvature is defined relative to the base line width). In the Curvature Width ramp, the left of the graph represents a straight line, while the right side represents a 180 degree kink. The top of the graph leaves the width unchanged, whereas the lower values reduce line width.



In this image Curvature Based Width Scaling is used (see graph below) to lower the line thickness along the curves of the face, especially along the nose.



Curvature Based Width Scaling

☒ Curvature Modulation

Curvature Width

Selected Position 0.000

Selected Value 1.000

Interpolation Spline

Tips for creating wiggly toon lines

While several methods exist for creating wiggly toon lines, we’ve provided information about four possible techniques in this section. Depending on the level of detail needed and the density of triangles on the model you might want to turn on resampling for the lines and lower the Max Segment Length to produce enough CVs to see the wiggle effects.

For this "shiny teapot" by Duncan Brinsmead, he used the teapot Paint Effects brush with a fractal displacement and converted it to polygons. He then assigned a ramp shader and used separate brushes for crease and border toon lines. For the profile lines he assigned the spike brush and used depth offset on the toon lines. To push the spikes "behind" the image to create the sparkle effect, he used a line modifier with force.

To create wiggly toon lines

1. Map the line offset and/or width attributes with a noise texture.
   - Make the noise type Perlin Noise because it is faster.
   - Adjust the alpha gain on the texture to increase/decrease the amount of offset.
   - The line offset is relative to the line width, so increasing the width scales the offset as well.
   - To make it wiggle you can animate the noise texture by typing the following in the Time attribute:
   - `“= time * 5"`

- You could also step the keys on time or oscillate for various looks. You can then play with the various noise parameters to get the desired wiggle.
- You can also map opacity or line width to create gaps.

2 Create a line modifier for the toon line.

- In the lineModifier node, make Width Scale 1, Surface Offset something like 0.01, and Dropoff Noise around 0.8.
- Adjust the Noise Frequency for the desired level of detail. By default the effect is only inside the line modifier, however by deleting the second dropoff index it will also affect areas outside the modifier. This noise is a volume noise that moves with the modifier, so rotating or translating the modifier will create a wiggle.
- You could also oscillate the Noise Frequency up and down a little.

3 Use a Paint Effects brush. If you assign a Paint Effects brush to the toon line, you can then use the full set of attributes on the brush node. The pfxToon node overrides a few of these attributes, although for simplicity it has an internal default brush and does not show the full set of Paint Effects attributes.

- You can use watercolor brush strokes and so on; when doing this, it is a good idea to use Resample Profile and set the Min/Max Segment Length to the same value. This will result in better coherence in the lines as they change.
- One technique to use some of the brush displacement attributes is to create a single tube, but force it to follow the stroke path:

Tubes = on

Tube Completion = off

Tubes Per Step = 0

Start Tubes = 1.0

Path Follow = 1.0

Segments = the largest number of segments in any of your lines or more

You can then play with displacement values such as Noise, and world Displacement turbulence. Then increase the turbulence speed for an animated wiggle.

If desired you can layer multiple toon lines on the same object, each with a different brush. The Draw Order (under Stroke attributes) can be used to set the order of these different strokes; for example, creating a pencil line on top of a watercolor smear or blur.

4. Create a toon line using the Offset Mesh option for Profile Lines and turn off Border and Crease lines.
   - Assign your shader to the resulting offset mesh and turn on Double Sided (Render Stats section of the mesh's Shape Attribute Editor). You will have to select the offset mesh in Outliner because it is not pickable in the scene and then turn off Enable Overrides (Object Display section of the Attribute Editor).
   - Hide your original mesh. By texturing the line offset or width on the toon shader you now have a displacement to polygon with construction history.
   - You can animate the displacement in the same way as described in the first technique. A toon line can then be assigned to this wiggling displaced mesh.

Troubleshoot toon lines

Here are solutions to problems you may encounter when using toon lines.

Staircasing or aliasing on toon lines

- Turn on Oversample and Oversample post filter in the Paint Effects Rendering Options.
- Normal scene Anti-aliasing Quality in the Render Settings should at least be set to intermediate.

NOTE If there is aliasing only on the inside edge of toon lines, then it is likely that Occlusion Width Scale was turned off (pfxToon node). Turn it back on. Also make sure Paint Effects Oversample is turned on.

Many of the problems listed below may also appear as anti-aliasing issues. If none of these solutions work, setting profile lines to the Offset Mesh method may work well. Alternatively convert the toon lines to polygons, although for thick lines you may see little cracks at line breaks. As another alternative, render at a higher image file resolution, then resize down to the target resolution with filtering.

Kinks and pops in line width along profile lines

If lines are not smooth enough (that is, there are kinks in them or they pop out in places), try adjusting the following settings in the pfxToonShape node:

- To use the Profile Width Modulation or Curvature Based Width Scaling options, Smooth Profile should be turned on.
  The Profile Width Modulation and Curvature Based Width Scaling options both require smoothly changing lines to work correctly. If they're on but Smooth Profile is turned off, there will be places where the profile line shows kinks. (These kinks may not always be noticeable because they appear along the view line, which may be fine for the current view.)
- Avoid the Profile Width Modulation and Curvature Based Width Scaling options for hard edges, which behave as if Smooth Profile is turned off.
- Turn on Resample Profile in the Line Resampling section. This will avoid very small segments that can cause sudden direction changes in the stroke.
- In some cases, having a more uniform triangle density can help avoid kinks. Very small or skinny triangles may cause problems.

Profile line does not fit to the exact edge

This typically happens with Smooth Profile when the geometry does not accurately match the surface (or the sense of the surface defined by the vertex normals). A simple example is a cone, which has a pinned normal pointing up at its peak. Therefore when viewed from its vertex normals, a default polygon cone is more like a poorly tessellated smooth hump. As a solution turn on Tighter Profile. If it is still not good enough, then either turn off Smooth Profile (and make Profile Width Modulation zero and turn off Curvature Modulation) or increase the mesh resolution of your geometry in the problem regions.

Wobbly or uneven profile line widths

This could be either of the previous two problems (Kinks and pops in line width along profile lines and Profile line does not fit to the exact edge). If those solutions don't fix the problem, then consider using Offset Mesh profile lines.

Line ends continue too far, slightly interpenetrating

Turning on Oversample in Paint Effects Rendering Options may help. Try turning off Occlusion Width Scale as well, although this may cause aliasing on the inside edges of lines. Increasing Line Offset to around 0.5 may help the anti-aliasing; however Line Offset can also cause line edges to extend too far. In some cases Offset Mesh profile lines may be a good solution.

Lines appear along joins between surfaces

Border lines work well along open edges, such as the edges of a sheet of paper, but less so in places like fillets and smooth seams. You can usually turn off Border Lines. However if one edge is open and needs a border line while another lies along a seam, you can get rid of just the border on the seam using Remove Flush Borders.

Can't create good crease lines on organic surface, like a human face

The Crease Angle Min/Max are generally insufficient to generate creases along a highly curved organic surface like around nostrils or along the boundaries of lips. You can explicitly define the edges you want the creases to be on as hard, and use Hard Creases Only. However the triangulation of the model will generally not be fine enough to allow good feature lines along triangle boundaries. In these cases it is best to model the feature lines, or paint them on to the geometry with Paint Effects brushes. You can also edit the Pressure Scale ramp on a stroke after painting to get good thick/thin line effects.

Complex Paint Effects brushes assigned to profile toon lines jump around too much when the object moves or the view changes

It is difficult to keep the same profile line start points when rotating an object with respect to the view. However, the effect will be more consistent if Resample Profile is turned on. Additionally setting the Min/Max Segment Length to a fixed value can also help. Profile Width Modulation should be set to zero and Curvature Modulation turned off. Note that if the only effect desired is a wiggly line, then you might be better off mapping the Line Offset attribute instead of using a complex Paint Effects brush. See Tips for creating wiggly toon lines on page 18.

Lines don't show up in mental ray or the Maya Hardware renderers

Convert toon to polygons. For profile lines you can also use Offset Mesh profiles, instead of converting toon lines to polygons. Paint effects to NURBS or curves may also be useful.

Tumble in the scene view and transforming toon objects is slow after assigning toon outlines

This is due to the recomputation of the profile lines for the new viewpoint. There are different ways to deal with this issue:

- Turn off Display In Viewport on the toon line node.
- Turn off Show > Strokes in the panel menu (this will also affect Paint Effects strokes and hair as well).
- Use Offset Mesh profile lines - these do not require rebuilding when the view changes.
- Set Paint Effects to Interactive in the Performance Settings window. This will disable building of the toon lines when tumbling the view.
- Assign different toon lines to different objects, rather than one pfxToon node for all objects. This will help when transforming individual objects, as only that object will need to update. (This won't help with tumbling.)
- If an object has many non-profile lines, such as creases, then it might be good to assign one pfxToon node with just profile lines on, and another to the same objects with only the non-profile lines. Thus when the profile node rebuilds there is no additional wait for the non-profile lines to also build.
- Turn off just the profile lines, but turn them back on before rendering.
- In the Stroke Attributes section lower the Display Quality. By default toon lines are drawn as six-sided tubes. At lower display quality this goes down to three-sided tubes, then becomes a simple line at zero quality. Alternately you could turn off Draw As Mesh if the width display of the lines is not important. This will speed up display somewhat, but not dramatically.

Toon menu

2

Toon

Toon > Assign Fill Shader

Select one of the options to attach a fill shader to the selected geometry: Solid Color, Light Angle Two Tone, Shaded Brightness Two Tone, Dark Profile, Rim Light, Circle Highlight. For more information, see Adding toon lines and shading to a scene on page 3.

Toon > Assign Outline

From this menu, you can add toon lines to the outline of selected geometry or remove them. To assign them, you can either add a new set of toon outlines or assign the attributes of an existing set. See Adding toon lines and shading to a scene on page 3.

Toon > Create Modifier

When you select Create Modifier, a cube or sphere shape locator appears, which you can transform. Use the attributes in the **lineModifier** node to control the appearance of the toon lines within the realm of the locator. This is useful for locally editing line width, transparency, color, and other attributes. This affects all Paint Effects entities. The same modifier can be assigned to multiple nodes, for example, it could affect both Paint Effects strokes and toon lines at the same time. You can also have multiple modifiers assigned to one object; the influences of the different modifiers blend together. You may want to group the modifier with the object so they move together. For more information, see Use a modifier to alter toon line and brush attributes on page 12.

Toon > Set Camera Background Color

Sets a background color for a camera view: perspective, front, side, top. See Adding toon lines and shading to a scene on page 3.

Toon > Get Toon Example

Imports example files of toon line and shading effects. Modify these files to customize the toon styles, use them as learning tools to understand how to create your own toon styles, or use them simply for ideas. See Toon examples on page 2.

Toon > Reverse Surfaces

Reverses the normals of the selected polygon mesh. The normals define the inside and outside of the surface, which is important when using the Offset Surface method for profile lines. If the surface is offset on the inside instead of the outside you will not get correct profile lines. Also if the normals are not in the correct direction and you use the Line Offset attribute, the lines could be offset into the surface, rather than out from the surface.

In the case of NURBS or subdivision surfaces that are assigned toon lines, a conversion poly mesh would have been generated. Reverse Surfaces knows how to find and reverse this conversion polygon mesh when only the NURBS is directly selected. (The conversion meshes are drawn, but have displays set to reference object, so they can't be directly picked.) Therefore for NURBS or subdivision surfaces you must do the reverse after creating the toon line. Alternatively use Edit NURBS > Reverse Surface Direction (in the Surfaces menu set); this will affect mapped textures. This is more efficient, as otherwise the reverse normals on the conversion mesh are recomputed whenever the NURBS is modified.

Toon > Assign Paint Effects Brush to Toon Lines

Assigns the selected Paint Effects brush to the toon lines, specifically to the pfxToonShape node connected to the selected geometry. See Use Paint Effects with toon lines on page 11.

Toon > Convert Toon to Polygons

Converts the toon lines of the selected geometry to polygons. The benefit of this is that the poly lines can be rendered in other renderers. For more information, see Convert toon lines to polygons on page 14.

Toon > Paint Line Attributes

You can paint the following attributes for the selected toon lines: Line Width, Line Offset, Line Opacity, Profile Color, Crease Color, and Border Color. For more information, see Paint toon line attributes on page 10.

Toon nodes 3

pfxToonShape node

This node is created whenever you assign toon lines to a selection (Toon > Assign Outline). When you edit the attributes in this node, you change the look of the toon lines on the geometry connected to this node. Below are descriptions of the attributes in this node. See also Types of toon lines on page 4.

Common Toon Attributes

These attributes are in the Common Toon Attributes section of the pfxToon node.

Display In Viewport Turns on and off the display of toon lines. On by default.

NOTE Turning off Display In Viewport has no effect on profile lines set to Offset Mesh.

Profile Lines Specifies the mode for profile lines: Off, Paint Effects, or Offset Mesh.

If Paint Effects lines are used, then curves following the outline are constructed; these lines are relative to the view and therefore Maya rebuilds them when the view changes.

If Offset Mesh lines are used, an output mesh is generated that is displaced along the surface normal by the line width and set to single-sided with reversed facing. This creates a shell that results in an outline (or silhouette) when rendered with the original mesh, although no true line is ever generated. This has the advantages of built-in thick/thin line properties that are stable when animated, and it does not need to be updated when the view changes. It will also appear in raytraced reflections and refractions.

The pfxToon node contains various attributes to control the appearance of profile lines. For details, see Profile Lines on page 38.

Border Lines Specifies how to generate lines along boundaries. *Open Edges* are considered to be edges that are shared by only one face, which, for example, can be used to create lines around the border of a simple plane. Profile lines do not handle this sort of edge, which only occurs when the surface is not fully closed. *Shader Boundary* edges occur where the faces attached to the edge do not share the same shader. The pfxToon node contains various attributes to control the appearance of border lines. For details, see Border Lines on page 40. *Shader Boundary* edges occur where the faces attached to the edge do not share the same shader. *Edge and Shader Boundary* are both attributes applied to the border lines.

Crease Lines If turned on, lines are generated along internal hard edges. The pfxToon node contains various attributes to control the appearance of crease lines. For details, see Crease Lines on page 39.

Intersection Lines Off by default. Turn it on to display intersection lines. For more information about intersection lines, see Adding toon lines and shading to a scene on page 3. The pfxToon node contains various attributes to control the appearance of intersection lines. For details, see Intersection Lines on page 41.

Line Width Controls the global width of the profile, crease and boundary lines.

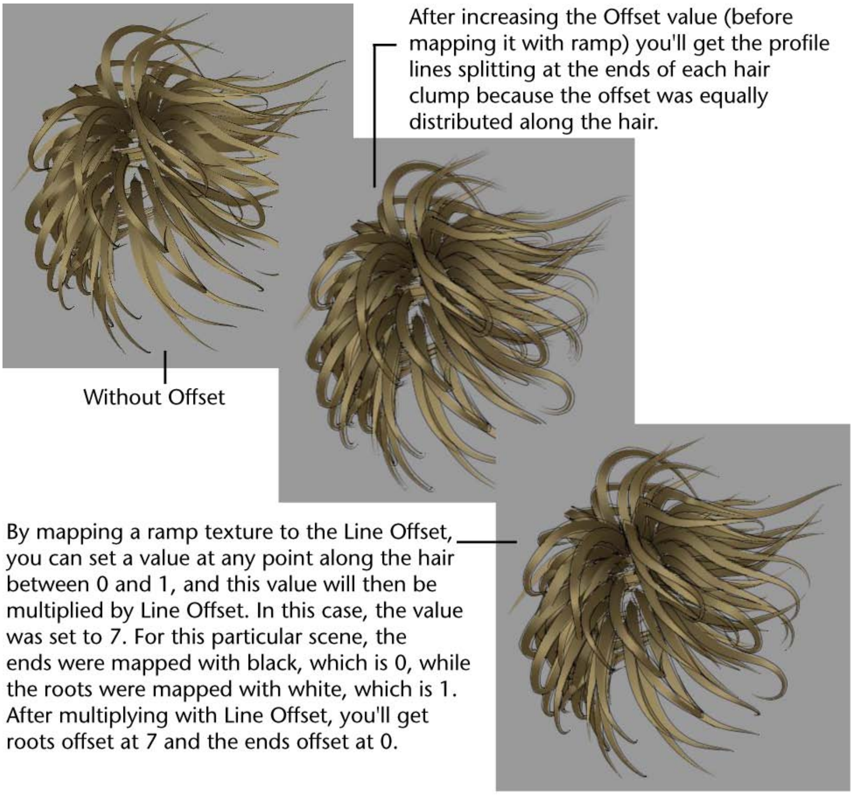
Line Width Map Varies Line Width with a 2D texture map. This attribute scales the currently defined Line Width, but by two times the Line Width Map value, so that a value of 0.5 has no effect. Where the texture value is 0.25 the line width would be half of the Line Width attribute value and where the value is 1.0 the value would be double. Values outside the zero-to-one range can also be used (for file textures this generally requires increasing the alpha gain on the texture). The texture is mapped onto the UV space of the surface the lines are on. In addition to thick and thin effects, you can also block line formation entirely in places by using a zero texture value.

NOTE 3D, projective 2D, and environment textures are not supported.

Line Offset Offsets lines along the surface normal. The amount of offset is relative to the Line Width. With a value of 1.0, the line just touches the surface and with a value of 0 the line is half intersecting the surface.

Line Offset Map Varies Line Offset with a 2D texture map. If a noisy texture is used, this can create a wiggly line. This attribute scales the currently defined line offset. Where the texture value is 0.5 the Line Offset is half the Line Offset attribute value. The texture is mapped into UV space of the surface the lines are on.

NOTE 3D, projective 2D, and environment textures are not supported.



After increasing the Offset value (before mapping it with ramp) you'll get the profile lines splitting at the ends of each hair clump because the offset was equally distributed along the hair.

Without Offset

By mapping a ramp texture to the Line Offset, you can set a value at any point along the hair between 0 and 1, and this value will then be multiplied by Line Offset. In this case, the value was set to 7. For this particular scene, the ends were mapped with black, which is 0, while the roots were mapped with white, which is 1. After multiplying with Line Offset, you'll get roots offset at 7 and the ends offset at 0.

Toon outlines can be assigned to hair; however to render thick hairs you must assign a Paint Effects brush to the hair system. Hair uses the Thin Line brush method when no brush is assigned, but Thin Line has kinks when drawing thick tubes.

Toon Hair images created by Yiqun Chen.

Line End Thinning Controls the distance from line ends beyond which they become fully thick. Depending on the object you might need to lower the Profile Break Angle values in order to see open line ends. For most surfaces the profile lines form closed lines with no open ends. The continuation of the profile is sometimes hidden where it goes behind the object. It may therefore look like it has open ends even though it is closed and will not show the effect of Line End Thinning. Crease lines are more likely to show open ends than profile or border lines. You can use the break angles to create open ends where the line curves strongly.

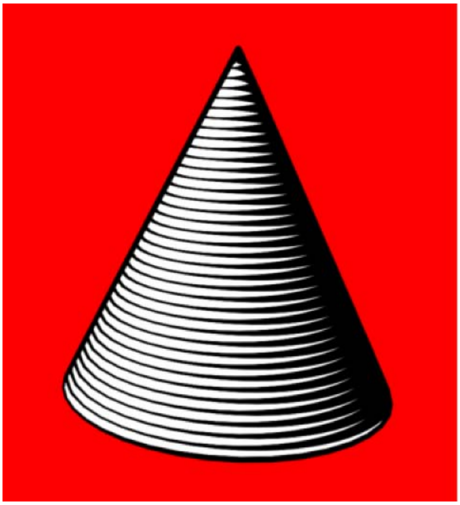
Line Extend Extends the ends of lines so they cross at corners rather than simply intersecting. Line Extend simulates a pencil drafting style of line. Just like Line End Thinning, depending on the object you might need to lower the Profile Break Angle values in order to see open line ends. For most surfaces the profile lines form closed lines with no open ends. The continuation of the profile is sometimes hidden where it goes behind the object. It may therefore look like it has open ends even though it is closed and will not show the effect of Line Extend. Crease lines are more likely to show open ends than profile or border lines. You can use the break angles to create open ends where the line curves strongly.

Line Opacity Controls the opacity of the profile, crease and boundary lines. An opacity of 0 is fully transparent and an opacity of 1 is fully opaque.

Line Opacity Map Varies the line opacity with a 2D texture map. This attribute scales the currently defined line opacity, so where the texture value is 0.5 the line opacity would be half of the Line Opacity attribute value. The texture is mapped into the UV space of the surface the lines are on.

NOTE 3D, projective 2D, and environment textures are not supported.

Lighting Based Width This attribute is useful for creating a thick/thin effect for line modulation. The line width is scaled by the diffuse light intensity. Brighter regions have thin lines. Shadowed and dark regions have full-width lines. Using negative values can have the reverse effect. The lighting used is a diffuse illumination that is relative to the local surface normal.



In this scene we've set the brush Global Scale so that the stroke lines just touch and we set the Lighting Based Width to 1.0. A toon shader was applied to the cone for profile lines and rendered.
The effect of light on width does not show up in the scene view, only in the render. However it will show up if you convert the strokes to polygons.

Occlusion Width Scale *This option is for* Paint Effects *profile lines only, and doesn't apply to* Offset Mesh *profile lines.*

The center of the Paint Effects brush stamp that generates a profile line is located at the surface of an object. When this option is off, the inside part of the brush stroke that is occluded (hidden) by the surface is not rendered. This occlusion results in a smooth inside line edge—that is, the surface's edge creates the inside brush stroke edge. Because it is based on the topology (geometry) of the surface, this edge may exhibit aliasing artifacts when rendered.

When this option is on, the entire brush stamp, including the inside part of it occluded by the surface, is rendered. A wider and more natural looking brush stroke results, with fewer aliasing artifacts.

Some known problems when this option is on include:

- If lines are very thick, they may bite into objects more than desired in places.
- In some unusual situations, lines may have a blobby quality where there are sudden surface changes.

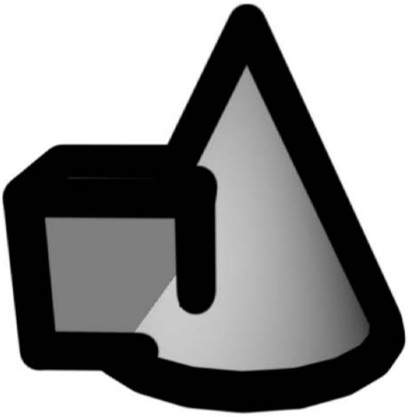
NOTE If a brush is assigned to the toon line, then the setting for Occlusion Width Scale on the brush node will override the setting on the toon node.

Local Occlusion

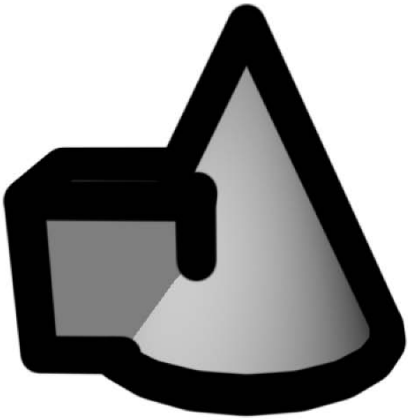
When two faces are closer together than the crease line thickness, toon lines may show up through the faces.

The Local Occlusion option allows visibility occlusion calculations to be performed before the lines are created, rather than at render time. The occlusion can be seen in wireframe display mode and will also reduce the polygon count if you convert toon lines from paint effects to polygons.

All Toon Surfaces
Toon lines are occluded by all the objects assigned to that pfxToon node. If your scene has multiple pfxToon nodes, however, you must still rely on render time occlusion to compute occlusion of one pfxToon group by another.

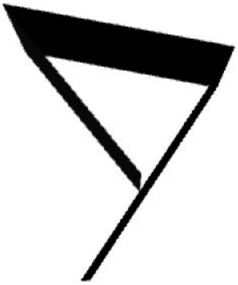


Local Occlusion Off. Crease lines from the cube that should be occluded are showing through the face of the cone.

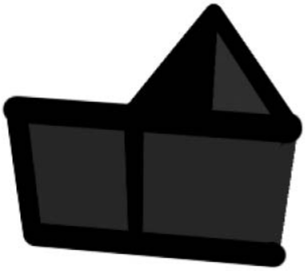


Local Occlusion set to All Toon Surfaces. Crease lines from the cube are properly occluded by the cone. (Both the cube and the cone belong to the same pfxToon node.)

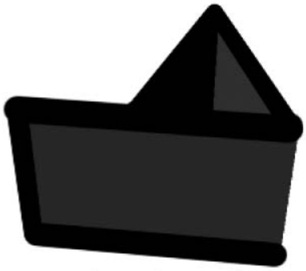
Line Surface
Toon lines are occluded only by the surfaces that generate the lines. The render time occlusion handles occlusions with other objects. The local occlusions are typically the most critical for line quality, however, as this helps avoid problems where the surface occludes its own lines.



Top view of plane. Two faces of the plane are very close together but not actually touching.



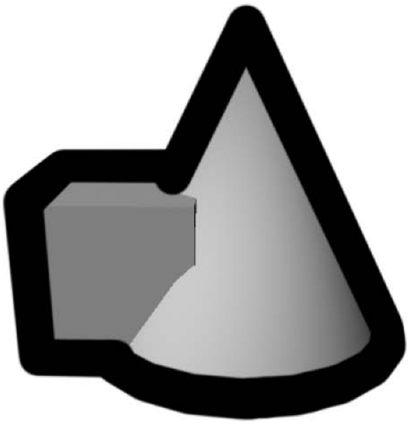
Perspective view of plane. Local Occlusion Off. Line from back plane is showing through the front plane.



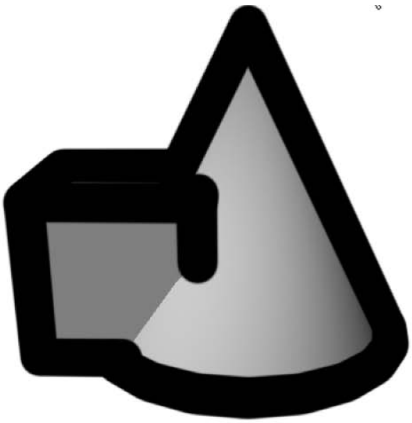
Perspective view of plane. Local Occlusion set to Line Surface. Line from back plane is properly occluded.

Occlusion Tolerance

The amount that the line is moved towards the camera when determining the occluded lines when local occlusion is used. It needs to be a small but non-zero value to avoid precision problems that might push the points on the line behind the triangles they sit on.



Occlusion tolerance set to 0. The cube is missing a few crease lines.



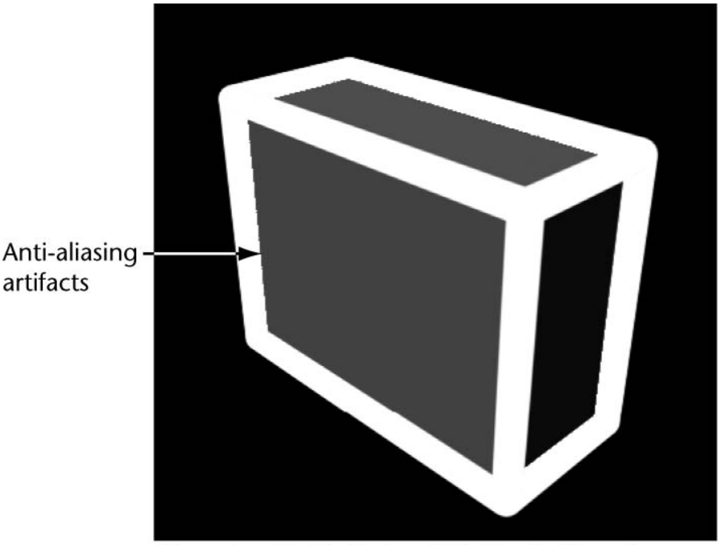
Occlusion tolerance set to 0.02

Depth Bias

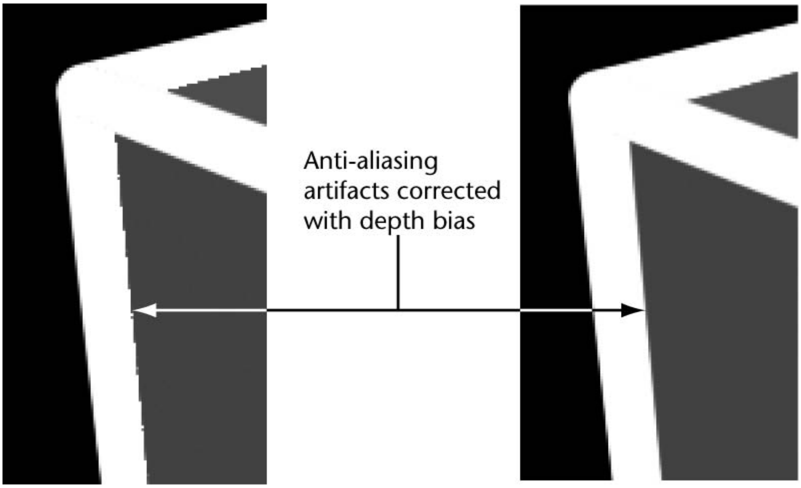
This option moves the line depth values towards the camera such that the lines do not get occluded by the local object they are on. This option is similar to the Occlusion Tolerance, but it is taken into account at render time.

In some cases, Occlusion Width Scale may create blobby artifacts in your scene and thus you may want to turn it off.

When this occurs, the Depth Bias option can be used to eliminate anti-aliasing artifacts when the Occlusion Width Scale option is turned off.



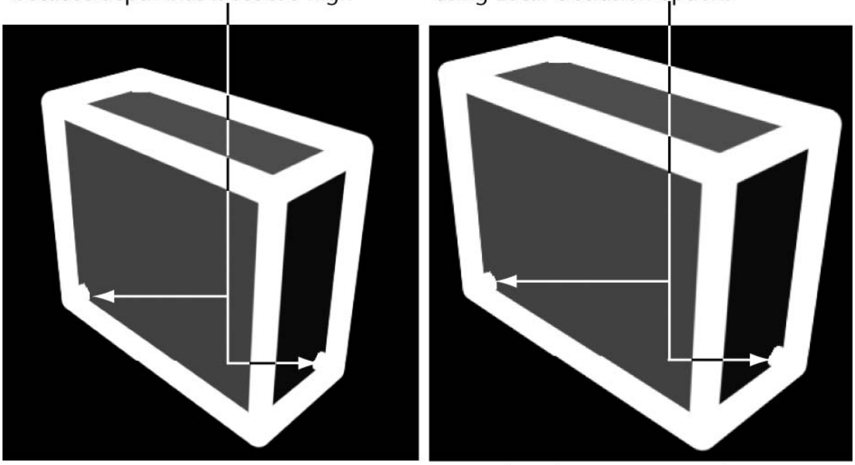
Occlusion Width Scale = Off. Depth Bias =0



Occlusion Width Scale = Off.
Depth Bias =0

Occlusion Width Scale = Off.
Depth Bias =0.2

If the depth bias is set too high, the lines may show through other objects that are in front. In this case, set Local Occlusion to All Toon Surfaces or Line Surface to occlude these lines. Local Occlusion does not rely on the depth buffer occlusion; therefore, depth bias can be increased while crease lines are properly occluded.



Crease lines showing through front because depth bias is set too high

Crease lines are properly occluded using Local Occlusion option.

Occlusion Width Scale = Off.
Depth Bias =0.7

Local Occlusion = Line Surface.
Depth Bias =0.7

Profile Lines

These attributes are in the Profile Lines section of the pfxToon node.

Profile Color The input color for the profile lines. This may be textured with 2D textures; the line color is then derived based on the color of the texture mapped to the surface.

NOTE Mapping 3D, projective 2D, and environment textures is not supported; however you could convert the toon lines to polygons and then apply a shader with 3D textures to the result.

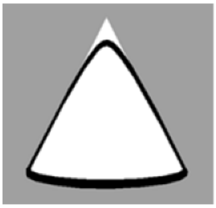
Profile Line Width A relative scale of the width of the profile, or silhouette lines.

Profile Width Modulation Varies the width of profile lines relative to the view angle. If Smooth Profile is turned off, then Profile Width Modulation should not be used.

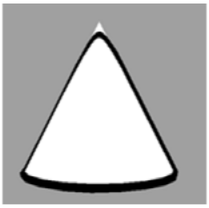
Profile Break Angle The angle in degrees beyond which profile lines are broken. Line segments that are less than this angle, with respect to each other, are joined into lines. At a value of 0 all lines will be only one segment long. At a value of 180 all adjacent segments will be joined together into lines.

Depth Offset Offsets profile lines along the view direction. This can help give lines a more graduated width where the profile is internal to the object.

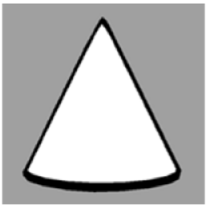
Smooth Profile If this setting is on, then the profile is computed based on the interpolated normals. If it is off, the profile curve exists only on polygon edges. The off state guarantees that the profile is always at the exact visible boundary, although it will jump in discreet steps as the view changes or the object deforms. With Smooth Profile turned on the profile changes in a smooth continuous fashion, but may creep in from edges in places where the surface is crudely triangulated. If all the geometry edges are hard, it will seem as though Smooth Profile is turned off.



Smooth Profile turned on



Smooth Profile and Tighter Profile turned on



Smooth Profile turned on and the number of Subdivisions Height increased

Tighter Profile *This option can only be used if* Smooth Profile *is turned on.* If Tighter Profile is turned on, then Smooth Profile will be a bit closer to the mesh profile edge, particularly in cases where the vertex normals do not match an edge orientation well. A primitive cone is a good example of this situation. Deriving the profile from the vertex normals does not work well in this case because the normals define more of a hump than a pointed cone. For a cone the Tighter Profile method will push the profile about halfway closer to the point of the cone. To fit closer than this, increase the subdivisions on the cone, or turn off Smooth Profile.

Crease Lines

These attributes are in the Crease Lines section of the pfxToon node.

Crease Color The input color for crease lines. It may be textured with 2D textures; the line color is then derived based on the color of the texture mapped to the surface.

NOTE Mapping 3D, projective 2D, and environment textures are not supported; however you could convert the toon lines to polygons and then apply a shader with 3D textures to the result.

Crease Line Width A relative scale of the width of the crease, or hard edge lines.

Crease Width Modulation Varies the width of crease lines relative to the view angle.

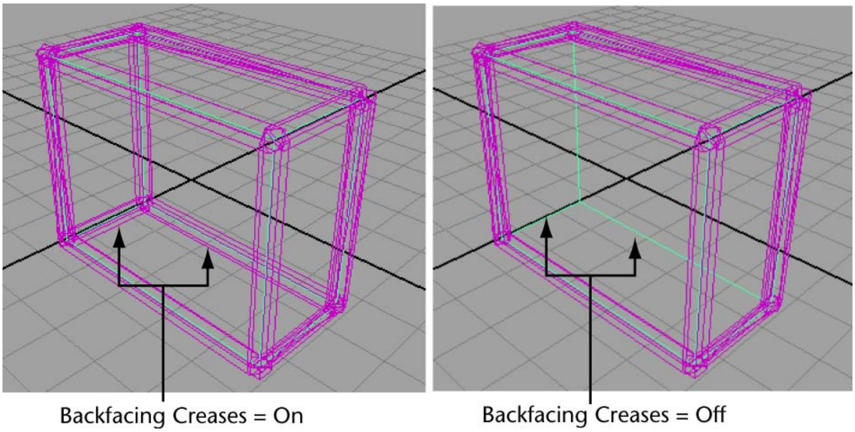
Crease Break Angle The angle in degrees beyond which crease lines are broken. Line segments that are less than this angle with respect to each other are joined into lines. At a value of 0 all lines will be only one segment long. At a value of 180 all adjacent segments will be joined together into lines.

Crease Angle Min The start angle in degrees between faces beyond which creasing forms. Line thickness can be graduated between zero width at the minimum angle and going to full thickness at the maximum. At an angle of zero all edges will be drawn, if Hard Creases Only is turned off.

Crease Angle Max The angle in degrees between faces beyond which crease line thickness is the maximum. At an angle of zero all edges will be drawn, if Hard Creases Only is turned off.

Hard Creases Only If this option is turned on, creases are only generated along hardened edges.

Backfacing Creases Turn this option off to prevent lines from being generated on creases that face away from the view. This option can only be used with crease lines, since profile lines are never on backfacing edges and border lines are along open edges which do not have a sense of side.



Border Lines

These attributes are in the Border Lines section of the pfxToon node.

Border Color The input color for border lines. It may be textured with 2D textures; the line color is then derived based on the color of the texture mapped to the surface.

NOTE Mapping 3D, projective 2D, and environment textures are not supported; however you could convert the toon lines to polygons and then apply a shader with 3D textures to the result.

Border Line Width A relative scale of the width of the border, or boundary lines.

Border Width Modulation Varies the width of border lines relative to the view angle.

Border Break Angle The angle in degrees beyond which border lines are broken. Line segments that are less than this angle with respect to each other are joined into lines. At a value of 0 all lines will be only one segment long. At a value of 180 any adjacent segments will be joined together into one line.

Remove Flush Borders Removes borders where meshes touch and have matching normals.

Flush Tolerance The maximum distance boundary CVs can be from each other when suppressing flush or common border lines.

Flush Angle Max The maximum angle in degrees that boundary vertex normals can be from each other when suppressing flush or common border lines.

Intersection Lines

These are the attributes in the Intersection Lines section of the pfxToon node.

Intersection Color The input color for intersection lines. It may be textured with 2D textures; the line color is then derived based on the color of the texture mapped to the surface.

NOTE Mapping 3D, projective 2D, and environment textures are not supported; however you could convert the toon lines to polygons and then apply a shader with 3D textures to the result.

Intersection Line Width A relative scale of the width of the intersection lines.

Intersection Width Modulation Varies the width of intersection lines relative to the view angle.

Intersection Break Angle The angle in degrees beyond which intersection lines are broken. Line segments that are less than this angle with respect to each other are joined into lines. At a value of 0 all lines will be only one segment long. At a value of 180 all adjacent segments will be joined together into lines.

Intersection Angle Min The start angle in degrees between faces beyond which intersection lines form. Line thickness can be graduated between zero width at the minimum angle, going to full thickness at the maximum. At an angle of zero all intersecting edges will be drawn. By varying the minimum and maximum intersection angles you can avoid lines along flush intersections, yet still have lines where the intersection angle is steeper.

Intersection Angle Max The angle in degrees between faces beyond which intersection line thickness is the maximum. At an angle of zero all intersection lines will be drawn at full thickness.

Self Intersect If this option is turned on, then lines are generated where surfaces intersect themselves.

Curvature Based Width Scaling

These attributes are in the Curvature Based Width Scaling section of the pfxToon node. For an example illustrating the use of Curvature Based Width Scaling, see Tips for working with toon shading on page 17.

NOTE If profile lines are showing, then Smooth Profile should be turned on. Resample Profile may also help with the Min Segment Length value increased.

Curvature Modulation Varies the width of lines relative to the 3D curvature of the line. Use the Curvature Width graph to control how the curvature affects the line width.

Curvature Width When Curvature Modulation is turned on, the Curvature Width graph controls how curvature affects line width. The input position sampled from the Curvature Width graph is the curvature, where the left of the graph is zero curvature (straight) and the right is fully bent at a 180 degree angle. The output value of the graph (vertical position) is then used to determine a scale factor for the current width. Thus if the graph is a horizontal line with a value of 1.0 the line width is unchanged. Where the graph is zero the line width will be zero.

Line Resampling

These attributes are in the Line Resampling section of the **pfxToon** node.

Resample Profile The points along the profile are typically spaced at irregular distances, depending on the surface geometry. This is especially noticeable if you assign to the toon line a Paint Effects brush with the tubes turned on—the tubes may jump around as the scene is animated. Turning on Resample Profile causes the profile lines to be resampled at a uniform CV spacing, resulting in smoother strokes that jump less. The resample spacing needs to be small enough to avoid rounding corner edges.

Resample Crease The points along the crease are normally spaced according to the size of polygon edges. For simple objects this may not provide enough CVs for effects like Line End Thinning. Using Resample Crease provides the ability to sample the crease lines with fewer or more CVs, as well as to fix the worldspace distance between curve CVs. This may make crease lines twitch when animated if Screen Space Resampling is not zero. The resample spacing needs to be small enough to avoid rounding corner edges.

Resample Border The points along the border are normally spaced according to the size of polygon edges. For simple objects this may not provide enough CVs for effects like Line End Thinning. Using Resample Border provides the ability to sample the border lines with fewer or more CVs, as well as to fix the worldspace distance between curve CVs. This may make border lines twitch when animated if Screen Space Resampling is not zero. The resample spacing needs to be small enough to avoid rounding corner edges.

Resample Intersection The points along the intersection are typically spaced at irregular distances. This is due to the way that triangles overlap along the edge.

This is especially noticeable if you assign to the toon line a Paint Effects brush with the tubes turned on; the tubes may jump around as the scene is animated. Turning on Resample Intersection causes the intersection lines to be resampled at a uniform CV spacing, resulting in smoother strokes that jump less. The resample spacing needs to be small enough to avoid rounding corner edges.

Min Segment Length The smallest distance that is allowed between CVs on profiles when Resample Profile is turned on. If this is large the profile curve may be very crudely sampled. It is useful if you want an output, for example to polygon, where the number of segments is kept to a minimum. Also when combined with Screen Space Resampling it can be used to simplify distant profiles.

Max Segment Length The maximum distance that is allowed between CVs on profiles when Resample Profile is turned on. If this is small there may be many CVs on a profile curve. If this value is very large the profile curve will have its original number of CVs, but uniformly distributed across the original curve. You may have rounded corners if the Max Segment Length is not low enough.

Screen Space Resampling Controls the degree to which outline curve resampling is done in screenspace. When this is 1.0 the segments are a fixed distance apart in screenspace. When it is 0.0 the distances are fixed in worldspace. Between 1.0 and 0.0 these two sampling methods are interpolated. This can be used to have profile curves become simpler with distance. Also if you have a brush with tubes turned on and assigned to the toon line, the number of tubes will reduce with distance when this is greater than zero.

Screenspace Width Control

The attributes in the Screenspace Width Control section of the pfxToon node are identical to those in the Screenspace Width Control section (subsection of Brush Profile) in a Paint Effects brush's Attribute Editor. For descriptions of these attributes, see Brush Profile in the *Paint Effects, Artisan, and 3D Paint* guide.

When Screenspace Width is turned off the Paint Effects toon lines appear thinner the further they are from the camera. But when it's turned on, there is no difference in the Paint Effects toon line width no matter how near or far the line is from the camera; with Screenspace Width turned on the Min Pixel Width and Max Pixel Width are respected. Turn this option on to maintain the brush width at all depths in the scene. This option is useful in scenes where you want cartoon-like outlines that retain a fixed width.

The brush stroke width is normally in worldspace and it appears to be constant in the 3D world. *Screenspace* refers to the flat 2D world of the rendered image (like a paint canvas). When the width is defined in screenspace it is relative to the overall size of the image, and the width is constant despite the distance of the object.

Stroke Attributes

These attributes are in the Stroke Attributes section of the pfxToon node. All of these except Pfx Randomize are standard stroke attributes found in the strokeShape node.

Display Quality Sets the display quality of the wireframe for the selected strokes. See strokeShape node in the *Paint Effects, Artisan, and 3D Paint* guide.

Draw As Mesh Previews the rendered look of your strokes interactively. See strokeShape node in the *Paint Effects, Artisan, and 3D Paint* guide.

Seed Sets the sequence of random numbers used for randomized brush attributes. See strokeShape node in the *Paint Effects, Artisan, and 3D Paint* guide.

Draw Order Sets the order that strokes paint in the scene. See strokeShape node in the *Paint Effects, Artisan, and 3D Paint* guide.

Surface Offset Sets the distance (in world units) you want the next stroke offset from the object or plane you paint the stroke on. See Paint Effects > Paint Effects ToolPaint Effects > Paint Effects Tool in the *Paint Effects, Artisan, and 3D Paint* guide.

Pfx Randomize If this option is turned on, then each curve is given a new random seed for Paint Effects. This requires that you have a pfx brush with tubes turned on and assigned to the toon line. Profile line tubes may jump around if this is on when animated.

Control Curves

These attributes are in the Control Curves section of the pfxToon node. For details on these attributes and how to use them, see Modify tube behavior using control curves in the *Paint Effects, Artisan, and 3D Paint* guide.

If a brush with tubes turned on is assigned to toon lines, then Paint Effects control curves can shape tube direction. For example, you can create motion streaks from profile lines.

Mesh Output

These attributes are in the Mesh Output section of the pfxToon node, and are also found in the strokeShape node. See strokeShape node in the *Paint Effects, Artisan, and 3D Paint* guide.

NURBS Output

These attributes are in the NURBS Output section of the pfxToon node, which are also found in the strokeShape node. See strokeShape node in the *Paint Effects, Artisan, and 3D Paint* guide.

Render Stats

These attributes are in the Render Stats section of the pfxToon node. For details on these attributes, see Render Stats in the *Rendering* guide.

pfxToonProfileShader node

This node is created the first time Profile Lines is set to the Offset Mesh option (see Profile Lines in the pfxToonShape node). The pfxToonProfileShader is simply a surface shader and has a solid color. The outColor of the pfxToon node connects to the outColor of this surface shader so that updating and texturing the profile line color will also affect this shader. If you change the Profile Lines type back to Paint Effects or turn it off, the added shader and mesh nodes will still exist, but will have no effect because the output meshes will have no triangles.

NOTE This an internal node used by the pfxToonShape node.

lineModifier node

This node is created when you create a toon line modifier. For more information, as well as illustrations, see Use a modifier to alter toon line and brush attributes on page 12. Below are descriptions of the attributes in the lineModifier node.

NOTE You can have multiple line modifier nodes on a toon line.

Line Modifier Attributes

Shape Determines the shape of the line modifier. There will be no effect outside the bounds of the selected shape. The Dropoff graph attribute is defined relative to the shape, such that the right edge of the dropoff is the outer edge of the shape, and the left is the shape's center.

Width Scale Multiplies the current line width. A value of 1.0 will have no effect and a value of zero will scale the width to zero.

Width Offset Adds on to the current line width. A value of 0.0 will have no effect and a value of 1.0 will increase the width by one.

Opacity Scale Multiplies the current line opacity. A value of 1.0 will have no effect and a value of zero will scale the opacity to zero.

Opacity Offset Adds on to the current line opacity. A value of 0.0 will have no effect and a value of 1.0 will make the lines fully opaque.

Surface Offset Adds on to the current line surface offset. A value of 0.0 will have no effect and a value of 1.0 will increase the offset by 1.0. Negative values will clamp when an offset of 0 is achieved.

Line Extend Adds on to the current line extend. A value of 0.0 will have no effect and a value of 1.0 will increase the extend by 1.0. Negative values will clamp when an extend of 0 is achieved.

Modify Color If this option is turned on, the color value is blended with the line color based on the dropoff value.

Color If Modify Color is turned on, then this color is blended into the local line color based on the dropoff value.

Dropoff Dropoff is the loss of individual leaves, twigs, flowers, tubes, and so on—either a few or many, depending on the Dropoff settings you specify.

This setting determines the range of effect within the bounds of the modifier. If the dropoff is set to a constant value of 1.0 (horizontal line) then lines will be uniformly modified inside the bounds of the dropoff shape, with a hard transition at the shape edge. If the right edge of the dropoff is instead ramped down to zero the transition will be gradual, with no modulation at the shape edge, with increasing modulation towards its center. The dropoff graph attribute is defined relative to the shape, such that the right edge of the dropoff is the outer edge of the shape, and the left is the shape's center.

Dropoff Noise Modulates the dropoff value with a 3D noise. This can provide variation for the parameters being modulated. For example the tube width can be randomized or wiggled in a noisy fashion using this attribute.

Noise Frequency Controls how detailed the Dropoff Noise is. Lower values result in a smoother noise. The 3D noise is defined relative to the local space of the line modifier shape.

Brush Tube Attributes

For these settings to have any effect, they require a Paint Effects brush to be used with tubes turned on.

Force Acts as a force pushing tubes away from the center of the line modifier shape. Positive values will push away from the center while negative values will push towards the center. To have an effect this requires that a brush be used with tubes turned on. For example, use this to push Paint Effects trees as if with the wind.

Directional Force Acts as a force pushing tubes along the local x axis of the line modifier shape. To have an effect this requires that a brush be used with tubes turned on.

Displacement Acts as a displacement pushing tubes away from the center of the line modifier shape. Positive values will push away from the center while negative values will push towards the center. This is similar to the force attribute, but tube segments further down the tube are not affected by the offset. To have an effect this requires that a brush be used with tubes turned on.

Directional Displacement Acts as a displacement pushing tubes along the local axis of the line modifier shape. To have an effect this requires that a brush be used with tubes turned on. This is similar to the directional force attribute, but tube segments further down the tube are not affected by the offset.

Tube Scale Scales the length and width of generated Paint Effects tubes. A value of 1.0 will have no effect and a value of 0.0 will scale the tubes to nothing, resulting in no tubes generated. The behavior is similar to modifying the brush's Global Scale. To have an effect this requires that a brush be used with tubes turned on.

Tube Dropout Reduces the number of Paint Effects tubes generated. A value of 0.0 will have no effect and a value of 1.0 will result in no tubes generated. To have an effect this requires that a brush be used with tubes turned on.

Leaf Scale Multiplies the size of leaves used by the Paint Effects brush. A value of 1.0 will have no effect and a value of 0.5 will reduce leaf size by half. To have an effect this requires that a brush be used with tubes turned on.

Leaf Dropout Adds on to the current leaf dropout value used by the Paint Effects brush. A value of 0.0 will have no effect and a value of 1.0 will increase the dropout by 1.0. To have an effect this requires that a brush be used with tubes turned on.

Flower Scale Multiplies the size of flowers used by the Paint Effects brush. A value of 1.0 will have no effect and a value of 0.5 will reduce flower size by half. To have an effect this requires that a brush be used with tubes turned on.

Flower Dropout Adds on to the current flower dropout value used by the Paint Effects brush. A value of 0.0 will have no effect and a value of 1.0 will increase the dropout by 1.0. To have an effect this requires that a brush be used with tubes turned on.

Branch Dropout Adds on to the current branch dropout value used by the Paint Effects brush. A value of 0.0 will have no effect and a value of 1.0 will increase the dropout by 1.0. To have an effect this requires that a brush be used with tubes turned on.

Twig Dropout Adds on to the current branch dropout value used by the Paint Effects brush. A value of 0.0 will have no effect and a value of 1.0 will increase the dropout by 1.0. To have an effect this requires that a brush be used with tubes turned on.

Index