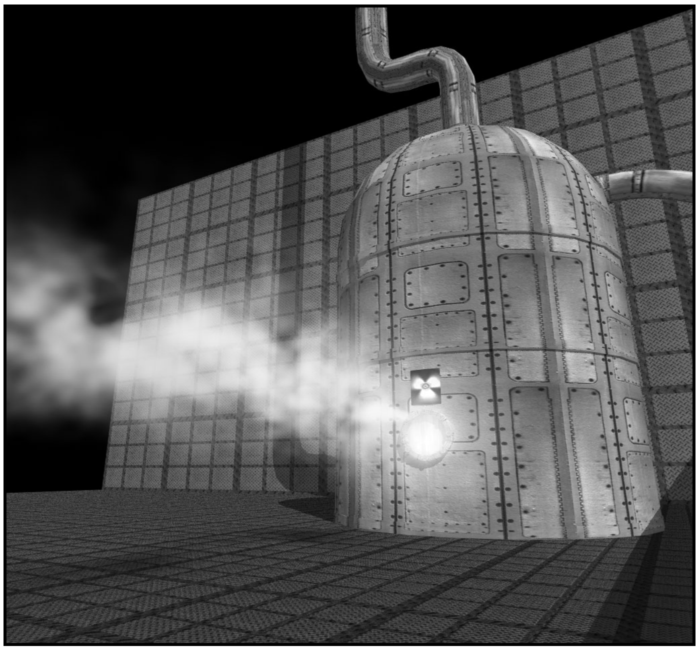


Dynamics

Version 6

Alias®

ALIAS ■ 210 KING STREET EAST ■ TORONTO, CANADA M5A 1J7

Table of Contents

Table of Contents

1 About Dynamics

About Dynamics

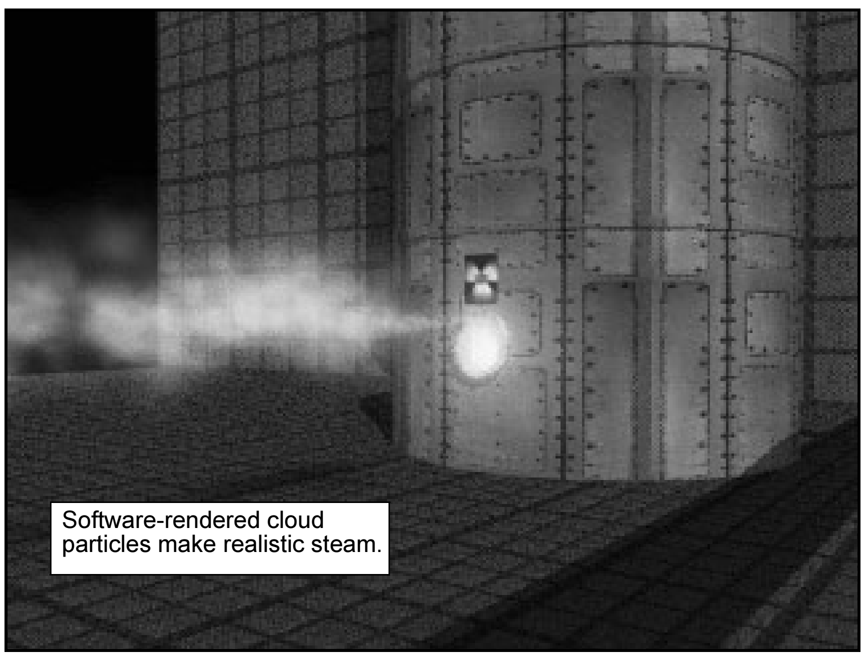
Introduction to dynamics

Introduction to dynamics

Dynamics is a branch of physics that describes how objects move. Dynamic animation uses rules of physics to simulate natural forces. You specify the actions you want the object to take, then let the software figure out how to animate the object.

Dynamic animation lets you create realistic motion that's hard to achieve with traditional keyframe animation. For instance, you can make effects such as tumbling dice, waving flags, and exploding fireworks.

To get started quickly using Maya Dynamics, see the Dynamics lessons in the *Getting Started with Maya* tutorials.



What you can do with Maya Dynamics

- Create, color, and animate *particles* (see "Particles" in Chapter 2).

- Use *emitters* to launch particles for effects such as steam, fire, rain, fireworks, and explosions (see “Emitters” in Chapter 2).
- Use *soft bodies* to create geometry that bends and deforms when influenced by a field or struck by a collision object (see “Soft bodies” in Chapter 4).
- Use gravity and other force *fields* to move particles, soft bodies, and rigid bodies (see “Fields” in Chapter 3).
- Create *collisions* between particles or soft bodies and geometry. You can make the particles split, emit new particles, or disappear when they collide with geometry (see “Particle collisions” in Chapter 2).
- Use *goals* to make particles or soft bodies follow other objects or object components (see “Goals” in Chapter 2).
- Use *springs* to give soft bodies and groups of particles internal structure (see “Springs” in Chapter 4).
- Use *rigid bodies* to create collisions between polygons or NURBS (see “Rigid bodies” in Chapter 4).
- Use *constraints* to restrict the motion of rigid bodies (see “Rigid body constraints” in Chapter 4).
- Use built-in dynamic *effects* to quickly create complex, popular animations such as smoke and fire (see “Effects” in Chapter 5).
- Tune playback efficiency and fix common problems with dynamics (see “Animating with dynamics” in Chapter 7).
- Store dynamic simulations either to disk or to memory (see “Particle caching” in Chapter 6).
- Use the Dynamic Relationship editor to connect and disconnect dynamic relationships between objects and fields, emitters, or collisions (see “Dynamic relationship editor” in Chapter 8).
- Render particles in software or hardware (see “Rendering particles” in Chapter 2).
- Work with advanced features of particles such as substituting animated geometry for moving particles (see “Advanced particle topics” in Chapter 2).

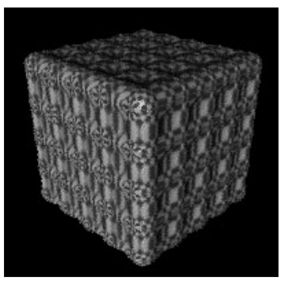
2 Particles

About Dynamics

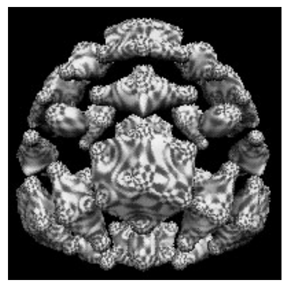
Particles

Particles

Particles are points that display as dots, streaks, spheres, blobby surfaces, or other items. You can animate the display and movement of particles with various techniques; for example, keys, expressions, and fields such as gravity.



Particles displayed as Blobby Surfaces

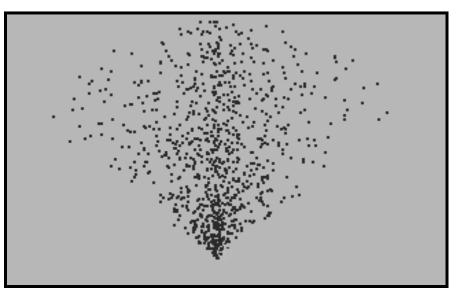


❖ "Create particles" on page 28

Particle objects

A *particle object* is a collection of particles that share the same attributes. You can create particle objects containing a single particle or millions of particles. Each particle in a scene belongs to some particle object. An overview of how you create particle effects follows:

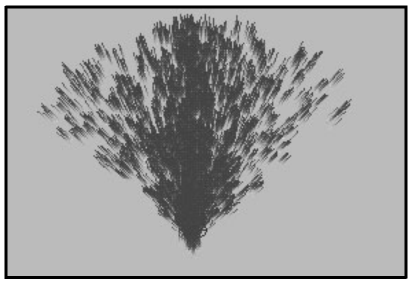
Creating particles



You can create particles several ways:

- Place particles in the workspace with the Particle Tool (see ”Create particles” on page 28).
- Create a particle emitter, which generates and animates the motion of particles automatically (see ”Emitters” on page 19).
- Cause particles that collide with geometry to create new particles upon contact (see ”Particle collisions” on page 23).

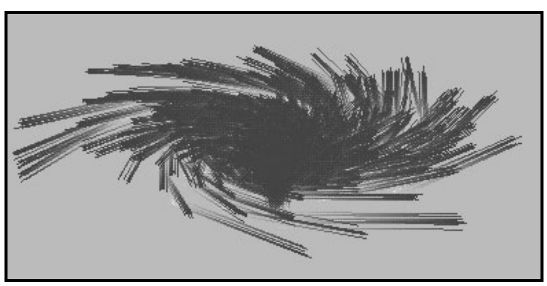
Set display attributes



You change the way particles look with several techniques:

- Set the render type of the particle object to select its form. For example, you can display particles as small spheres or streaking tails (see ”Choose how particles render” on page 37).
- Illuminate particles with lights you add to the scene (see ”Use lights, reflections, refractions, and shadows” on page 46).
- Color particles (see ”Set particle color” on page 48).
- Give particles transparency (see ”Set particle opacity” on page 50).
- Make particles disappear as they age (see ”Set particle lifespan” on page 51).

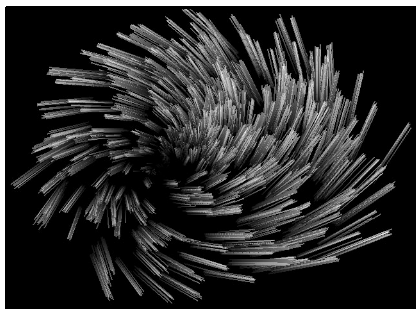
Animate the particles



You can animate particle motion several ways:

- Set the position, velocity, or acceleration attributes of particles. You can also set keys to animate an entire particle object's translate, scale, and rotate attributes. See "Choose how particles render" on page 37).
- Apply fields, such as gravity, to particles (see Chapter 3, "Fields").
- Turn geometry into a collision object and bounce particles off it (see "Particle collisions" on page 23).
- Make the particles follow a moving goal object (see "Goals" on page 21).

Render the particles



Rendering is the final step to creating particle effects. Depending on the render type you select, you render the particles with hardware rendering or software rendering. See "Choose how particles render" on page 37.

Advanced particle topics

Advanced particle topics

This chapter includes topics that enhance your work with particles. The topics require a basic understanding of particles, and in some cases, an understanding of related topics such as fields and expressions.

Control complex motion and forces

The following sections describe techniques for tuning dynamic animation of particles.

> Note The dynamic state of a particle object is the value of its position, velocity, acceleration, and mass attributes at any frame. Maya uses these attributes to compute any dynamics that influence the object's positioning in any frame.

Scale the effect of dynamics

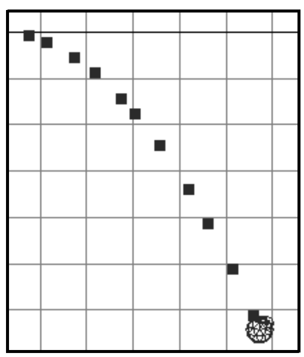
You can scale the effect of fields, collisions, springs, and goals on particles. To scale the effects of dynamics, select the particle object, display the Attribute Editor, and set the Dynamics Weight to a value between 0 and 1.

A value of 0 causes fields, collisions, springs, and goals connected to the particle object to have no effect. A value of 1 provides the full effect. A value less than 1 sets a proportional effect. For example, 0.6 scales the effect to 60% of full strength.

Expressions are unaffected by Dynamics Weight.

Combine keyed transform attributes and dynamics

You can combine dynamics with keys on particle objects or soft bodies. For instance, if you make a particle object fall with gravity, you can add sideways motion by keying its Translate X attribute.



Gravity moves the particles down while keys move it horizontally.

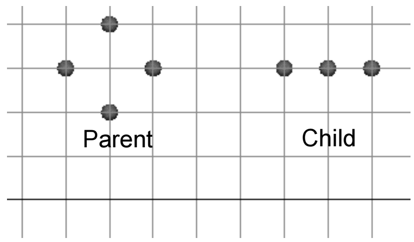
Keys on transform attributes affect the worldVelocity, worldPosition, and worldVelocityInObjectSpace attributes. See "Obtain world position, velocity, and centroid" on page 74 for details.

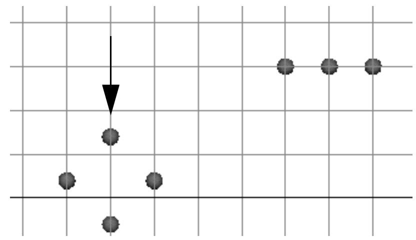
You can also combine dynamics with motion path animation of particles.

Parent an object to dynamic motion

When you parent an object to a dynamically animated parent, the object's transform attributes control its positioning and orientation relative to the positioning and orientation of its dynamic parent's transform attributes.

Dynamics do not affect the child object's transform values. For example, a child particle object won't fall with a parent particle object if you make the parent fall with gravity. The gravity changes the dynamic parent's particle positions but not its transform attributes. For this reason, the gravity doesn't change the child's position.





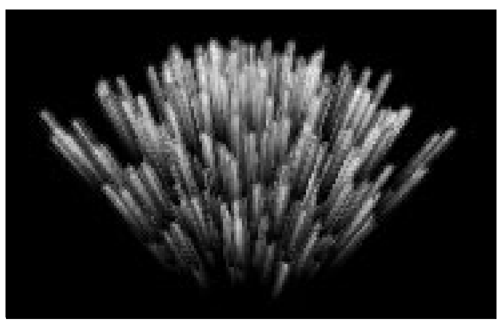
The child doesn't move as the parent falls with gravity.

There is a technique, however, to make an object move with a dynamic object's motion.

Emitters

Emitters

Emitters generate moving or stationary particles as an animation plays. You can use emitters to create smoke, fire, fireworks, rain, and similar objects.



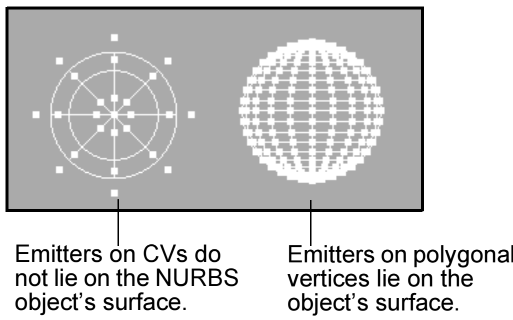
Emitted particles

Maya includes the following types of emitters:

- *Point emitters (directional and omni)* emit particles from a position in the workspace or from particles, vertices, CVs, edit points, or lattice points.
- *Surface emitters* emit particles from random, evenly distributed positions on the outer faces of NURBS or polygonal surfaces.
- *Curve emitters* emit particles from random, evenly distributed positions of a NURBS curve.

- *Volume emitters* emit particles from a closed volume. You can choose from cube, sphere, cylinder, cone, and torus.

When you select a NURBS surface or curve and add a default emitter, you create a point emitter that emits from all CVs. Most CVs do *not* lie exactly on the surface of the object. When you select a polygonal surface and add a default emitter, it emits from all vertices, which do lie on the surface of the object.



When you create any type of emitter, a particle object is automatically created and connected to it. The connected particle object originally has no particles. As the animation plays and the emitter generates particles, the particle count of the connected particle object increases. We refer to the connected particle object as the emitted particle object.

The connection between the emitter and emitted particle object is not a spatial relationship. The emitted particle object's attributes define the appearance and other characteristics of the emitted particles. The emitter's attributes control the initial position, direction, quantity, and velocity of the emitted particles.

❖ "Create emitters" on page 86
❖ "Edit attributes of an emitter" on page 89
❖ "Edit attributes of emitted particles" on page 93
❖ "Duplicate emitters" on page 95
❖ "Connect emitters and particles" on page 96
❖ "Vary emission from different points of point emitters" on page 97
❖ "Use a texture to color emission or scale the rate" on page 100

Goals

Goals

A goal is an object that particles follow or move towards. You can use goals to give trailing particles a flowing motion that's hard to generate with other animation techniques. The trailing particles move as if connected to the goal by invisible springs. In the context of goals, soft bodies are considered particles.

Tentacles created by Jason Schleifer

A goal can be any object except a curve on surface. The movement or motion of the trailing object depends on the type of goal object and the number of goal objects connected to the trailing object.

- "Create goals" on page 107
- "Edit goal attributes" on page 108
- "Animate goal behavior" on page 112

Particle goals

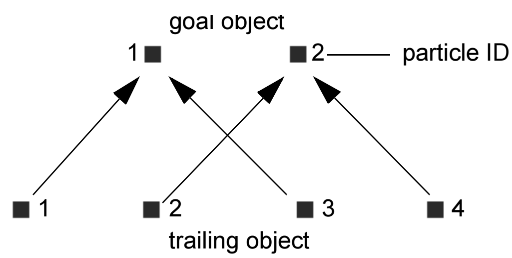
Particle objects are useful as goal objects because of the many techniques available for animating particle motion. You can't add a goal to individual particles of the particle object, but you can control how influential each particle is on the trailing object (see "Set goal weight on a per particle basis" on page 109).

If the goal is a particle object, its particles attract the particles of the trailing object one for one as the animation plays. If particles in the objects do not die, the trailing particles follow goal particles based on the creation order. For example, the particle created first in the trailing object follows the particle created first in the goal object. The particle created last in the trailing object follows the particle created last in the goal object.

If particles in either object die, the preceding scheme no longer applies. You can no longer visually predict which trailing particle will follow a particular goal particle.

If the trailing particle object has more particles than the goal object and particles don't die in either object, the extra particles follow the first-created particles of the goal.

For instance, suppose you create a goal object with two particles and a trailing object with four particles. The four particles would move toward the two particles like this:

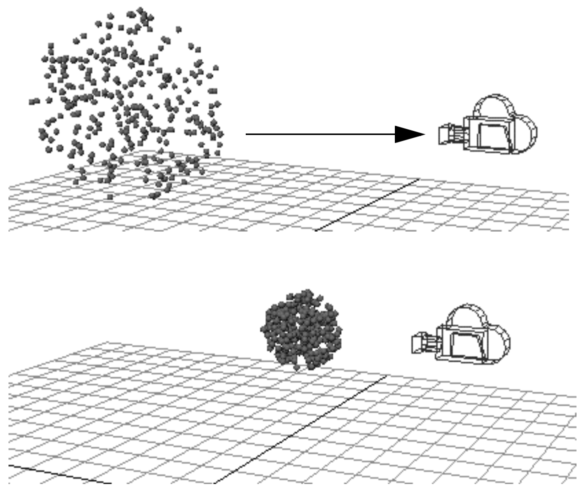


❖ "Create goals" on page 107
❖ "Edit goal attributes" on page 108
❖ "Animate goal behavior" on page 112

Nonparticle goals

If the goal is a NURBS object, polygonal object, or lattice, its CVs, vertices, or lattice points attract the particles one for one. Extra particles follow the points in the goal object by starting over at the beginning points. The method is the same as described for particle objects.

If the goal is an object other than a lattice, NURBS, or polygonal object, for example, a light or camera, the object's transform becomes the goal.



When the goal is a camera, the trailing particles move toward its transform position.

❖ ”Create goals” on page 107
❖ ”Edit goal attributes” on page 108
❖ ”Animate goal behavior” on page 112

Multiple goals

You can use more than one goal object to affect a particle object. For each goal object, the trailing particle object has a goal weight that sets the relative weighting of the attraction. If the goal weights are the same, each goal object attracts the trailing object with equal strength. The trailing object moves to a position between the two goal objects, typically oscillating back and forth before coming to equilibrium.

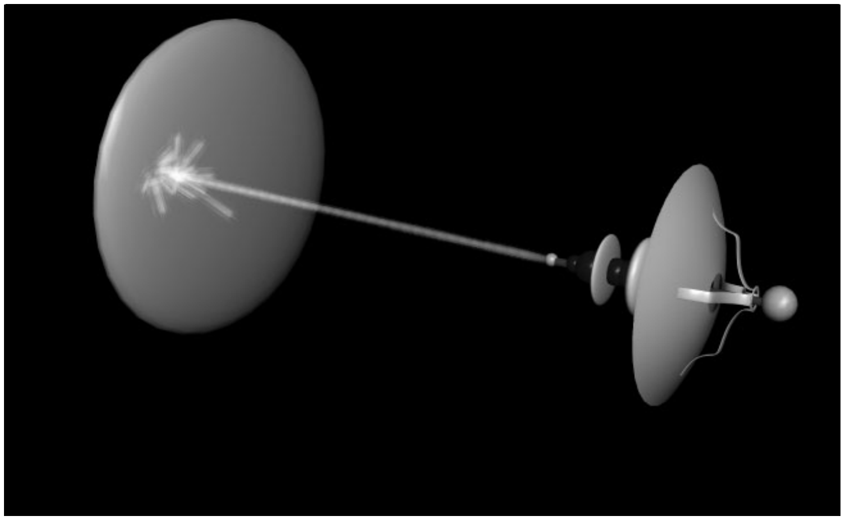
If the goal weights differ, each goal object attracts the trailing object with different strength. The trailing object comes to rest at a position closer to the goal with the higher goal weight.

❖ ”Create goals” on page 107
❖ ”Edit goal attributes” on page 108
❖ ”Animate goal behavior” on page 112

Particle collisions

Particle collisions

You can make particle objects collide with geometry. Either or both objects can be moving at the moment of impact. You can also make particles split, emit new particles, die, or run a MEL script when they collide with geometry.



- "Make particles collide with a surface" on page 113
- "Edit particle collision attributes" on page 114
- "Duplicate collision effects" on page 116
- "Emit, kill, or split particles upon contact" on page 117

Rendering particles

Rendering particles

If your scene contains particles, the way you render your scene depends on the type of particles it contains. There are two main types of particles: hardware particles and software particles.

- Hardware rendered particles have a render type of render type of MultiPoint, MultiStreak, Numeric, Points, Spheres, Sprites, or Streak.
- Software rendered particles have a render type of Blobby Surface, Cloud, or Tube.

For more information on setting particle render types, see "Choose how particles render" on page 37.

Hardware rendered particles

Hardware rendering makes use of your computer's graphics hardware, and is much faster than software rendering. However, the quality of hardware rendered images may not always be as high as software rendered images.

If your scene contains hardware particles and you want to software render the scene, you must do the following:

- Software render your scene. The hardware particles won't appear in the software-rendered scene. See *Rendering* "Rendering a frame"and "Rendering an animation."
- Hardware render the hardware particles. See "To prepare to preview or render hardware particles" on page 120.
- Composite the two sets of rendered images using a compositing software.

For more information, see the following topics:

- "To prepare to preview or render hardware particles" on page 120
- "Preview hardware particles" on page 121
- "Render hardware particles at final production quality" on page 123
- "View rendered hardware particles" on page 124.

Use the Particle Cloud shader

The particle cloud shader is a volume material that you can assign to particles with a Cloud render type to achieve effects such as gas or clouds.

This section describes the attributes you can modify in a particle cloud shader. For information on creating and assigning materials, see "Creating Materials" in *Rendering.*

Particle Sampler Info node

The Particle Sampler Info node is a particle utility that lets you:

- Use particle object attributes (including ramps and particle expressions) to drive particle shader attributes on a per-particle basis.
- Use rgbPP and opacityPP attributes in software rendering. Previously, you could use these attributes only in hardware rendering.
- Have more control over texture placement nodes for particles, including the use of per-particle attributes for texture coordinates.

The Particle Sampler Info node provide all the functionality of the particle color mapper, transparency mapper, incandescence mapper, and age mapper, which were available in versions of Maya prior to 3.0.

Using the Particle Sampler Info node

The Particle Sampler Info node provides particle shape attributes to either a particle shader, texture, or texture placement node for use in software rendering.

When you use a Particle Sampler Info node, you connect one or more of its outputs to attributes of a particle shader or texture placement node. This tells the shader to get the per-particle information from the Particle Sampler Info node. It also tells the Particle Sampler Info node which attributes to get from the particle shape.

Types of outputs

The Particle Sampler Info node has the following types of outputs:

- A uv coordinate output designed to drive the uv coordinate input for texture placement nodes. This is somewhat like the output from the old particle utilities, but more powerful.
- Outputs that correspond to predefined attributes of the particle shape, such as Rgb PP. These are used to drive attributes of shaders or textures on a per-particle basis.
- Ten predefined outputs for user-defined attributes that you can add to the particle shape. Five are scalar attributes and five are vector (see ”” on page 137).
- Two outputs, birthPosition and worldBirthPosition, that correspond to two optional attributes (with the same name), which you can add to your particles.

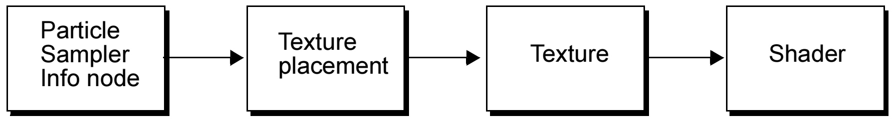
Note	You still have to add attributes to the particle shape, where applicable. The Particle Sampler Info node just gets attribute values, if they are present. For example, making a connection from the Particle Sampler Info node parentU to the shader tells the sampler info node to get parentU if it’s there, and use it in the shading. You still have to add parentU to the particle shape.

The Particle Sampler Info node has no association with any particular camera, unlike the samplerInfo node. The two nodes are similar in that both provide data to shaders to use in shading samples. They are different in that much of the data the samplerInfo node provides is camera-dependent.

Strategies for using the Particle Sampler Info node

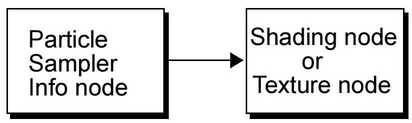
There are two basic strategies for using the Particle Sampler Info node to control your particles:

- The first strategy uses the Particle Sampler Info node to drive a texture placement node. The setup is:



The advantage of this strategy is two-fold: you can use any type of texture, and you can use the controls on the texture placement node.

- The second strategy uses the Particle Sampler Info node to feed per-particle attributes, such as rgbPP, directly to the shading or texture node. The setup is:



The main advantage of this strategy is that it lets you control the per-particle attribute using expressions, ramps, or the Component Editor and see the result in the shading or texture node.

Use user-defined attributes

In addition to its many outputs for predefined attributes of the particle shape (such as position, radiusPP, and so on), the Particle Sampler Info node includes ten predefined outputs for user-defined attributes, which you can add to the particle shape.

Five of these are scalar and five are type; they are called userScalar1PP, userVector1PP, and so on.

If you connect one of these to a shading attribute, the Particle Sampler Info node looks for an attribute of the identical name on the particle shape, and fetches those values. For example, if you connect userScalar1PP to shader noise, then the Particle Sampler Info node looks for a userScalar1PP attribute on the particle shape.

How do I? Create dynamic effects

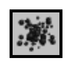
Create particles

Create particles

The Particle Tool lets you create and position particles individually or in grids or spherical regions. By default, the Particle Tool creates particles individually, one particle per mouse click. To change the tool options, see ”Set the Particle Tool options” on page 29.

> **Note** Unless otherwise noted, the directions in this book for making menu selections assume you’ve already selected the Dynamics menu set. Also, this book describes how to edit attributes of objects only with the Attribute Editor. You can also set many of the attributes with the Channel Box or Attribute Spreadsheet. See *Basics* for details.

To create particles

1. Select Particles > Particle Tool or click  .
2. Click the positions where you want to place the particles.
3. Press Enter (Windows, Linux, and IRIX) or Return (Mac OS X).

   This creates a new particle object consisting of the particles you positioned.

Tips

A particle object is not complete until you press Enter (Windows, Linux, and IRIX) or Return (Mac OS X). You cannot use Edit > Undo to undo the entire object until you press Enter or Return.

If you want to undo individual particles before you press Enter or Return, use the Backspace (Windows, Linux, and IRIX) or Delete (Mac OS X) key to remove the particles in the reverse order of creation. Alternatively, you can press the Insert (Windows, Linux, and IRIX) or Home (Mac OS X) key to go into editing mode, then click any particle and press Backspace or Delete. Press Insert or Home to leave editing mode.

You can also change the position of one or more particles before you press Enter or Return. Press the Insert or Home key then drag the particle. Press Enter or Return to complete the particle object.

Set the Particle Tool options

When you create a particle object, you can set several tool options in the Particle Tool Settings window. When you change these settings, it affects only particle objects you create after the change.

After you create the object, you can change some tool options and set additional attributes (see ”Edit particle attributes” on page 36).

To set the Particle Tool options

1. Select Particles > Particle Tool > ❐ to display the options window.
2. Set the options as described in the following procedures.

To name the particle object

- Enter a name in the Particle Name box.

  The name helps you identify the object in the Outliner. If you don’t enter a name, the particle object receives a default name such as *particle1*.

To adjust frame-to-frame dynamic velocity of moving particles

- Set the Conserve attribute to influence the motion of particles whose velocity and acceleration attributes are controlled by dynamic effects.

See ”Adjust frame-to-frame velocity conservation” on page 74.

> Note The terms *dynamics* and *dynamic animation* refer to animation motion created by fields, emission, collisions, springs, goals, and particle expressions. Non-dynamic animation is motion created by keys, motion paths, non-particle shape expressions, deformers, and constraints unrelated to rigid bodies.

To set the number of particles per click

1. In the Number of Particles box, enter the number of particles you want to create per mouse click and press Enter (Windows, Linux, and IRIX) or Return (Mac OS X).
2. If you choose a number greater than 1, you can distribute particles randomly in a spherical region where you click. To choose the spherical region, set Maximum Radius to a value greater than 0.

To sketch a continuous curve of particles

You can drag the mouse to sketch a continuous stream of particles.

1. Set the Number of Particles to 1.
2. Turn on Sketch Particles.
3. Set the Sketch Interval value.

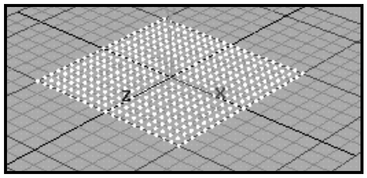
   This sets the pixel spacing between particles. A value of 0 gives you nearly a solid line of pixels. The higher the value, the more space between the pixels.
4. Drag the mouse in the workspace.
5. Release the mouse button and drag in another location, if desired.
6. Press Enter (Windows, Linux, and IRIX) or Return (Mac OS X).

   The sketched particles are a single particle object.

To create a 2D grid of particles by clicking in the workspace

1. Turn on Create Particle Grid.
2. Set the Particle Spacing value.

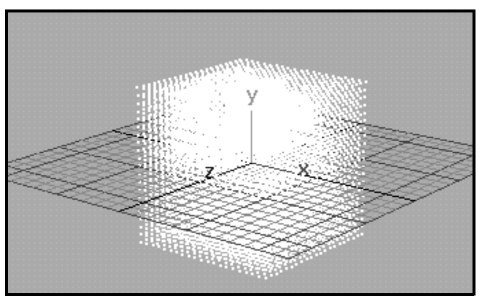
   This sets the spacing (in units) between particles in the grid.
3. Turn on Placement with cursor.
4. Click to place the left corner of the grid; click again to place the upper right corner of the grid.
5. Press Enter (Windows, Linux, and IRIX) or Return (Mac OS X) to create the grid.



To create a 3D grid of particles by clicking in the workspace

1 Turn on Create Particle Grid.
2 Select Placement with cursor, if it is not already selected.
3 Set the Particle Spacing value.

   This sets the spacing (in units) between particles in the grid.
4 In the perspective view, click the left mouse button at the lower left and upper right corners to specify the X and Z grid dimensions of the base or top of the 3D grid—*don't* press Enter (Windows, Linux, and IRIX) or Return (Mac OS X) yet.
5 Move the cursor into the front view or side view. Press the Insert (Windows, Linux, and IRIX) or Home (Mac OS X) key to enter edit mode.
6 Drag either the left or right point up or down to create the height of the grid. Do not drag both. To constrain placement, hold Shift down as you drag.
7 Press Enter or Return.



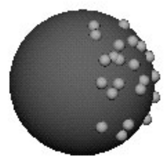
To create a 2D or 3D grid by entering values

1 Turn on Create Particle Grid.
2 Set the Particle Spacing value.
3 Turn on Placement with textfields.
4 For the Minimum Corner, enter the coordinates of the lower left corner. For the Maximum, enter the coordinates of the upper right corner.

5 Move the pointer into the workspace and press Enter (Windows, Linux, and IRIX) or Return (Mac OS X) to create the grid.

Place particles on a surface

You can place particles directly on the surface of a NURBS or polygonal object, or on a construction plane.



To place particles on a surface

1 Select the surface.

2 Select Modify > Make Live or click .

3 Select Particles > Particle Tool > ❐.

4 Set the desired tool options.

See ”Particles > Particle Tool” on page 142.

5 Place the particles on the surface.

6 Deselect the particle object, then select Modify > Make Not Live.

This deselects the surface as the live object.

Note To make the particles move with the surface, parent them to the surface. For more information, see the *Basics* guide.

Animate particles

You can animate particles in several ways:

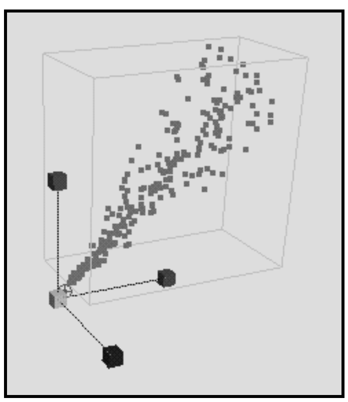
- Animate transform attributes of a particle object (described in this section).
- Edit the position, velocity, or acceleration attributes on an individual particle basis. See ”Set attributes on a per particle basis” on page 54.
- Use or modify the motion an emitter imparts to the particles it generates (see ”Emitters” on page 19).
- Apply fields, such as gravity, to particles (see ”Fields” on page 207).

- Turn geometry into a collision object and bounce particles off it (see ”Particle collisions” on page 23).
- Create goals for particles and make the particles follow the moving goal objects (see ”Goals” on page 21).

To animate transform attributes of an entire particle object

1. Select the particle object.
2. Do one of the following:
   - In the particle tab of the Attribute Editor, enter values for the translate, scale, and rotate attributes.
   - Use the Move Tool, Rotate Tool, and Scale Tool to manipulate the attributes.
3. Set keys at the desired frames for the attributes. See *Animation* for details on setting keys.

Note	You can key transform attributes of an entire particle object, not for individual particles in a particle object. If you use a field or other forces on particles for which you key transform attributes, also see ”Advanced particle topics” on page 17



The Move, Scale, and Rotate manipulators work on particle object transform attributes.

Work with particle attributes

You change the way particles look and behave by setting attributes. The type of attribute affects how you work with it.

Static attributes

Static attributes are attributes the particle object has by default. For example, a particle object's transform node has the static attributes Translate X, Rotate Y, Scale Z, and so on. A particle object's shape node has many static attributes that are listed in the Particle Attributes section of the Attribute Editor.

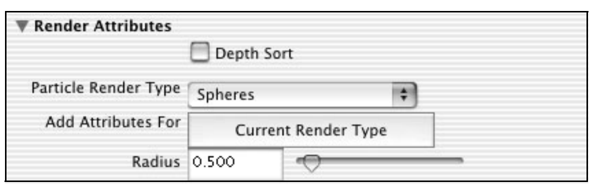
You can set the values of these attributes with the Attribute Editor, expressions, and other techniques after you create the object. The values you specify for static attributes apply to all particles in the particle object.

Dynamic attributes

Dynamic attributes have predefined names and purposes. Maya adds dynamic attributes to the particle object in response to your user interface selections. An object has no dynamic attributes unless your actions cause Maya to add them to the object.

You can set per object and per particle opacity and color of a particle object. The procedures require you to add dynamic per particle or per object attributes for the opacity and color. See "Per particle and per object attributes" on page 36 for details on per particle and per object attributes.

You can also add default dynamic attributes for tuning particle render types (see "Add dynamic attributes" on page 34). For example, suppose you select a particle object and display the Render Attributes section of the Attribute Editor. If you select Spheres as the Particle Render Type then click the Add Attributes For Current Render Type button, Maya adds a Radius attribute. The Radius attribute appears at the bottom of the Render Attributes section:



By adding dynamic attributes when they're needed, Maya runs faster. When a dynamic attribute is added to an object, the attribute appears in the Attribute Editor for the selected object.

Add dynamic attributes

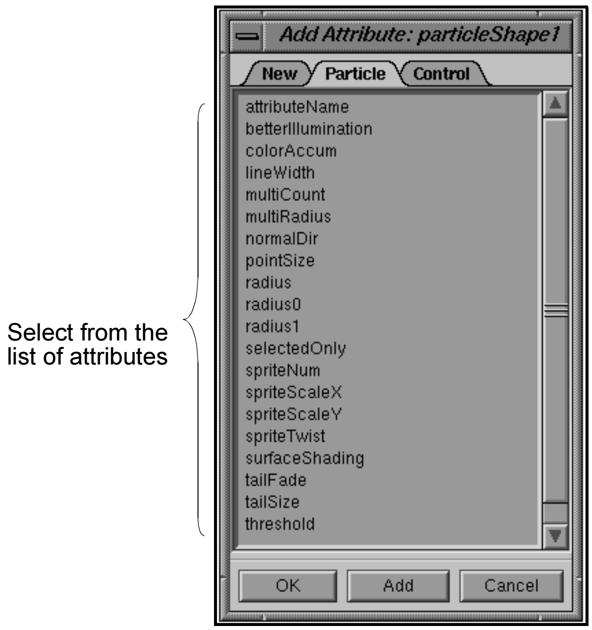
You can add dynamic attributes on an individual basis. For instance, after you become familiar with the attributes of a particle render type, you can selectively add only those you need without adding all the default attributes. By adding the fewest attributes necessary, you'll prevent unnecessary processing and avoid cluttering the Attribute Editor.

For example, suppose you select the Points render type and want to adjust only the point size of the particles. You can add only the attribute that sets the point size, while not adding any others.

To selectively add a dynamic attribute to a particle object

1 Select the particle object.
2 In the Add Dynamics Attributes section of the Attribute Editor, click the General button.

   The Add Attribute window is displayed.
3 In the Add Attribute window, select the Particle tab.
4 Select an attribute.



5 Click Add to add an attribute.
6 Click OK when you are done adding attributes.

Add custom attributes

Custom attributes are attributes you optionally add to an object, for example, by selecting Modify > Add Attribute. Although custom attributes are dynamically added to an object, we refer to them as *custom* to distinguish them from the static dynamic attributes.

Custom attributes have no direct effect on any characteristic of an object in Maya. A custom attribute is typically used in an expression to control a combination of other attributes. See *Expressions* for details.

Per particle and per object attributes

A *per object attribute* lets you set the attribute value for all particles of the object collectively with a single value. For instance, the per object opacity attribute lets you set a single opacity value for all the particles in the object.

A *per particle attribute* lets you set the value of the attribute individually for each particle of the object. For example, the per particle opacityPP attribute lets you set a unique opacity value for each particle. Though there is only one opacityPP attribute in a particle object, the attribute holds the value for each particle’s opacity value. The attribute holds the values in an array. In simple terms, an array is a list.

Though per particle attributes are best for creating complex effects, you can’t keyframe them. You *can* keyframe per object attributes.

”Dynamic attributes” on page 34 describes how to add attributes to a particle object to enhance your control of the behavior and appearance. After you add a per object attribute to a particle shape node, the attribute appears in the particleShape tab of the Attribute Editor, for example, in the Render Attributes section. You typically set per object attribute values in the Attribute Editor or Channel Box.

After you add a per particle attribute to a particle shape node, the attribute name appears in the Per Particle (Array) Attributes section of the particleShape tab of the Attribute Editor. For details on setting values, see ”Set attributes on a per particle basis” on page 54.

Note that a static, dynamic or custom attribute can be a per particle or per object attribute, not both. Also be aware that dynamically added per particle attributes often have a name that ends in PP. PP stands for per particle.

Edit particle attributes

You use the Attribute Editor to edit various display and motion attributes of a particle object after you create it.

To display the particle attributes in the Attribute Editor

1. Select the particle object.
2. Select Window > Attribute Editor to display the Attribute Editor.

   The table below lists the contents. For many sections, the attributes are displayed only after you create the corresponding dynamic effect.

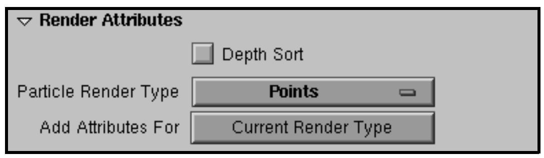
   To instead edit the transform node attributes of a particle object, click the particle tab in the Attribute Editor.

Tip	You can also use the Attribute Spreadsheet or Channel Box to edit particle attributes. Be aware, however, that only the Attribute Editor displays *all* particle attributes that you can edit interactively.

Choose how particles render

The particle render type of a particle object specifies the form of its particles. For example, you can display particles as small spheres, streaking tails, or 2D images of your favorite snapshot. Once you select the particle render type, you can add attributes specific to the render type to tune the appearance.

For information on hardware and software rendering of particles, see "Rendering particles" on page 24.



To set the Particle Render Type

1 Select the particle object.

2 In the Render Attributes section of the Attribute Editor, select the type from the Particle Render Type pop-up menu.

Workspace preview rendering doesn't display render types with full detail. You must hardware render or software render. Turn on Shading > Smooth Shade All if you want workspace rendering to display color that you add to particle objects.

When you test render or render to disk, use hardware rendering for the particle render types that don't have *s/w* after their names. Blobby Surface, Cloud, and Tube must be software rendered. If you use the wrong rendering technique, the particles won't be displayed. If you use hardware rendering, you must composite the particle images with software rendered images of the other objects in the scene.

The following rules apply to the software render types:

- You must add a light to the scene. If you render without adding a light, the particles won't be visible.
- You must apply a material with shading group to the particle object. Tube and Cloud render types use specific particle cloud materials. You can use any surface material with Blobby surface render type.

See *Rendering* for details on using materials and lights.

Points

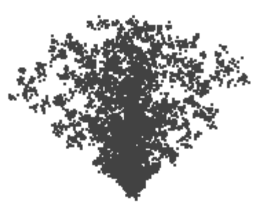
The Points render type displays particles as points. This is the default render type.

To set Points render type

1 Select the particle object.
2 In the Attribute Editor, set Particle Render Type to Points.
3 To add default Particle Render Type attributes that let you tune the appearance, click the Add Attributes For Current Render Type button.

MultiPoint

The MultiPoint render type displays each particle as multiple points. The extra points make the particle object appear denser. You can use this render type to create dust, clouds, mist, or other gaseous emissions.

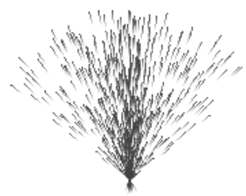


To set MultiPoint render type

1 Select the particle object.
2 In the Attribute Editor, set Particle Render Type to MultiPoint.
3 To add default Particle Render Type attributes that let you tune the appearance, click the Add Attributes For Current Render Type button.

Streak

The Streak render type displays moving particles with an elongated tail. This render type enhances the display of objects such as meteors or rain. The streak length is based on the velocity of the particle, so if the particles are stationary or moving slowly, you won't see the particles.



To set Streak render type

1. Select the particle object.
2. In the Attribute Editor, set Particle Render Type to Streak.
3. To add default Particle Render Type attributes that let you tune the appearance, click the Add Attributes For Current Render Type button.

MultiStreak

The MultiStreak render type is a combination of Streak and MultiPoint render types; it displays multiple points with tails for each moving particle. The faster the particles move, the longer the tails.

To set MultiStreak render type

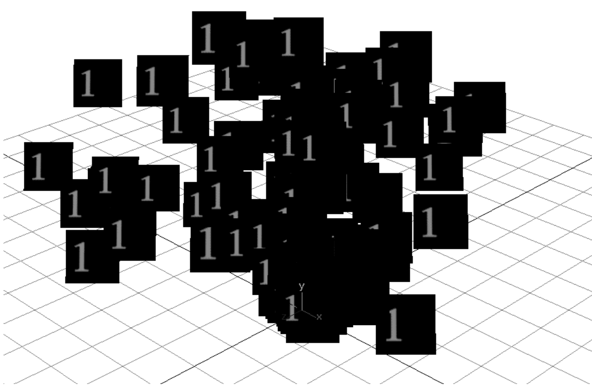
1. Select the particle object.
2. In the Attribute Editor, set Particle Render Type to MultiStreak.
3. To add default Particle Render Type attributes that let you tune the appearance, click the Add Attributes For Current Render Type button.

Sprites

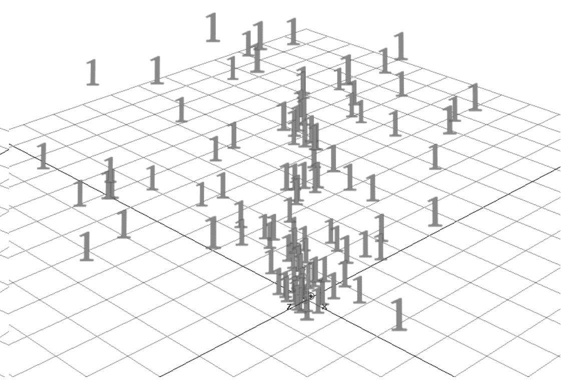
The Sprites render type lets you display a texture image or image sequence at each particle. Each particle can display an identical or different image or image sequence. Depending on the type of texture image you use, you can use sprites to create effects such as smoke, clouds, fog, and stars.

A sprite appears as a small rectangle until you map a texture image to it. The image faces the camera directly regardless of the camera's position or orientation.

If the texture image lacks an alpha channel, the image is opaque and occupies the sprite rectangle. If the texture image has an alpha channel, the sprite uses the image's transparency. To avoid displaying the rectangle, you must make the peripheral parts of the original texture image transparent.



These sprites use a texture image lacking an alpha channel, so the rectangles are displayed.



These sprites use a texture image that has an alpha channel. The "1" in the image is opaque. The rest of the image is transparent. The rectangles are invisible.

Use the Sprite Wizard

The particle Sprite Wizard simplifies the process for displaying a texture image or image sequences on particles. The particle Sprite Wizard leads you through the steps necessary to associate image files with sprites. You can assign a single image or a sequence of images to each particle. The images can be assigned randomly, or using various criteria such as the particleID or a ramp. You can also edit the sprites once you've created them with the Sprite Wizard.

The Sprite Wizard automatically:

- assigns a Lambert material to the particle object

- adds a creation expression and runtime expression that enables all the wizard options
- connects a ramp to the spriteNumRamp attribute
- sets the particle render type to Sprites and adds the attributes for sprites
- assigns the images you specify to the sprites

Note	The image files you use as sprites must have filename extensions in the format *file.n*, not *file*.000*n* (zero-padded extensions).

To use the Sprite Wizard

1. Select the particle object to which you want to assign sprites.
2. Select Particles > Sprite Wizard.

   The Sprite Wizard leads you through a series of choices. For help, read the pop-up help that appears with each text box.

   For additional information on the Custom Start and Custom Cycling options, see "Customizing sprites with the Sprite Wizard" on page 42.
3. To see the sprite images, turn on Shading > Smooth Shade All and Shading > Hardware Texturing.

To edit the sprites

- Do one of the following:
  - Select the particle object and display the Attribute Editor. Open the Sprite Attributes section of the Particle Attribute Editor to edit the attributes. You can also edit the sprite attributes using the Channel Box.
  - Select the particle and select Particles > Sprite Wizard to re-open the Sprite Wizard window. You can change the file names, the frame ranges, or other settings in the Sprite Wizard window. When you accept the changes, the wizard updates the appropriate attributes for you.

To change the images of an existing sprite animation:

1. Select the particle object.
2. Select Particles > Sprite Wizard.
3. Select a new image sequence for the Sprite File.
4. Turn on Use existing setup with new images in the second Wizard screen.

All the existing settings in the Sprite Wizard are used with the new image sequence.

Editing sprite attributes

Once you have created the sprites, you can optionally edit these Render and Sprite attributes in the Attribute Editor or in the Channel Box:

Customizing sprites with the Sprite Wizard

You can customize how the initial sprite images are selected and how the images are cycled by modifying the expressions that are automatically added to the particle object when you use the Sprite Wizard.

To customize the selection of the initial sprite image

You can customize the selection of the initial sprite image whether or not you choose to animate the selection of the initial image.

1. Create the particles and select Particles > Sprite Wizard.
2. Select the Sprite File you wish to use and click Continue.
3. Select how you want to assign images to the particles: either No animation or Cycle through the images for each particle, and click Continue.
4. Select Custom Start and click Continue.
5. Select how you want the images to be cycled for each particle If you select Custom Cycling, see "To customize sprite cycling" on page 43. Click Continue.
6. Click Apply to apply the wizard.
7. Select the particle and select Window > Animation Editors > Expression Editor.
8. In the Expression Editor, select Select Filter > By Expression Name.
9. Select the expression with the name of the particle object to which you have applied the sprites.
10. Click the Creation checkbox.
11. If you selected No Animation for the image assignment, look for the first occurrence of the line "Custom Start: If your sprites are not animated" in the creation expression.

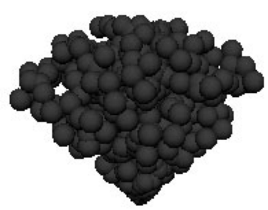
    If you selected Cycle through the images for each particle for the image assignment, look for the second occurrence of the line "Custom Start: If your sprites are animated" in the creation expression.
12. Insert your custom expression. For information on writing expressions, see *Expressions*.

To customize sprite cycling

1. Create the particles and select Particles > Sprite Wizard.
2. Select the Sprite File you wish to use and click Continue.
3. Select Cycle through the images for each particle and click Continue.
4. Select the technique used to assign the initial sprite to each particle and click Continue.
5. Select Custom Cycling for how you want the images to be cycled for each particle and click Continue.
6. Click Apply to apply the wizard.
7. Select the particle and select Window > Animation Editors > Expression Editor.
8. In the Expression Editor, select Select Filter > By Expression Name.
9. Select the expression with the name of the particle object to which you have applied the sprites.
10. Click the Runtime checkbox (before or after dynamics calculation).
11. In the runtime expression, look for the line “Custom Cycle.”
12. Insert your custom expression. For information on writing expressions, see *Expressions*.

Spheres

The Sphere render type displays particles as opaque spheres. You cannot display the spheres with transparency.

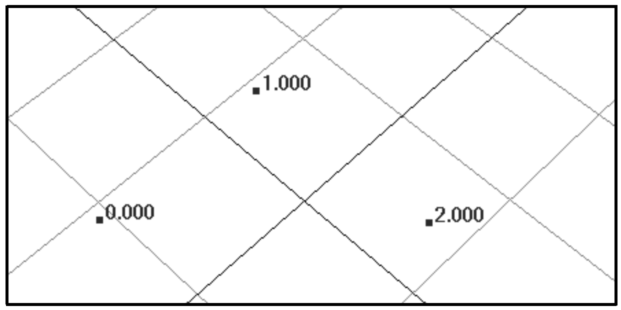


To set Spheres render type

1. Select the particle object.
2. In the Attribute Editor, set Particle Render Type to Spheres.
3. To add default Particle Render Type attributes that let you tune the appearance, click the Add Attributes For Current Render Type button.

Numeric

The Numeric render type displays the current values of any attribute of the particle object. This is useful when you want to know the value of an attribute such as velocity at a certain frame. By default, Maya displays particle ids for the Numeric render type.



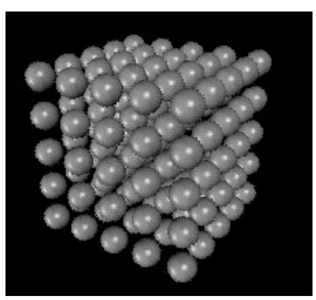
Particles displaying particle id

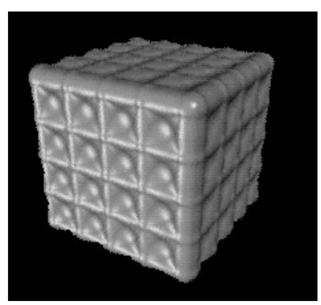
To set Numeric render type

1 Select the particle object.
2 In the Attribute Editor, set Particle Render Type to Numeric.
3 To add default Particle Render Type attributes, which let you tune the appearance, click the Add Attributes For Current Render Type button.

Blobby surface

The Blobby Surface render type displays particles as metaballs. Metaballs are spheres that blend together to form surfaces. Blobby Surfaces appear only in software rendered images.





Note Inside edges of blobby surfaces seen through other blobby surfaces are not anti-aliased

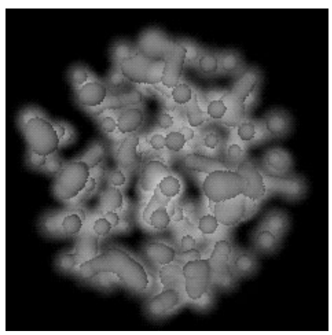
To avoid this, Increase Render Quality > particleSamples to 4 or 8 when very high quality is desired.

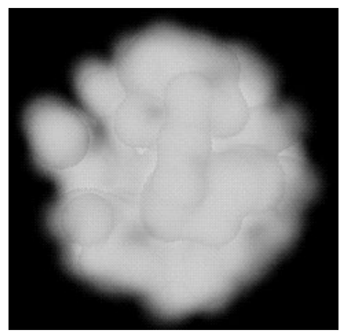
To set Blobby Surface (s/w) render type

1. Select the particle object.
2. In the Attribute Editor, set Particle Render Type to Blobby Surface.
3. To add default Particle Render Type attributes that let you tune the appearance, click the Add Attributes For Current Render Type button.
4. Apply a shading group to the particle object.
5. Add a light to the scene.
6. Software render the scene.

Cloud

The Cloud render type displays particles as blurred or cloudy metaballs. Metaballs are spheres that blend together to form surfaces. Clouds appear only in software rendered images. See "Create raytraced shadows with particles" on page 126.



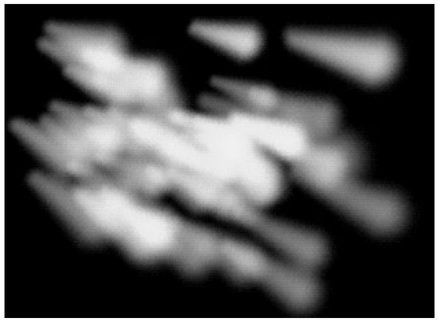


To set Cloud (s/w) render type

1. Select the particle object.
2. In the Attribute Editor, set Particle Render Type to Cloud.
3. To add default Particle Render Type attributes that let you tune the appearance, click the Add Attributes For Current Render Type button.
4. Add a light to the scene.
5. Software render the scene.

Tube

The Tube render type displays particles as tubes. Tubes appear only in software rendered images.



To set Tube (s/w) render type

1. Select the particle object.
2. In the Attribute Editor, set Particle Render Type to Tube.
3. To add default Particle Render Type attributes that let you tune the appearance, click the Add Attributes For Current Render Type button.
4. Add a light to the scene.
5. Software render the scene.

Use lights, reflections, refractions, and shadows

Lighting effects are an essential element of animation. You can render particles with several lighting techniques that are commonly used for conventional surfaces.

Add reflections, refractions, and shadows

You can turn on reflections, refractions, and shadows when you software render Clouds, Tubes, and Blobby Surfaces.

To turn on reflections, refractions, and shadows

1. Select the particle object.
2. In the Render Stats section of the Attribute Editor, turn on:
   - Visible In Reflections
   - Visible In Refractions
   - Casts Shadows

   You must use ray tracing to create reflections and refractions. You can use ray tracing or depth map shadows to create shadows. See *Rendering* for details.

> Tip — If you've created clouds, tubes, or blobby surfaces in your scene and are not currently working with them, you can speed up software test rendering by selecting the particle object and turning off Primary Visibility.

Use lights with moving particles

By default, all particles are evenly lit, regardless of where you place lights in the scene. For some effects, you might want moving particles to drift in or out of the lighting, disappearing when not illuminated.

For example, you might want cigar smoke rising under a lamp to show only when it passes through the lighting. Or, you might want rain to show only when it passes beneath a street light or in front of the headlights of a car.

If you use Streak, Point, MultiStreak, and Multipoint render types, you can use lights to create these effects.

To use scene lighting with particles

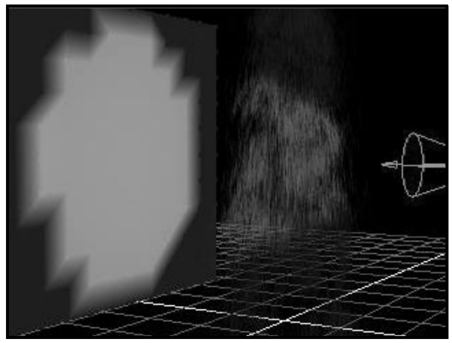
1. Add a spot light, point light, or directional light to your scene. A spotlight gives the most obvious effect. See *Rendering* for details on adding lights.
2. Aim the light at the particles.
3. Select Lighting > Use All Lights.
4. Select Shading > Smooth Shade All.
5. Select the particle object you want to light.
6. In the Attribute Editor, set the Particle Render Type to Streak, Point, MultiStreak, or MultiPoint.
7. Click the Add Attributes For Current Render Type button.

   The default render attributes are displayed in the Attribute Editor.
8. In the Render Attributes section, turn on Use Lighting.
9. To maximize particle illumination, set Normal Dir as follows:
   - Set to 1 if most or all particles are moving towards the light. Example: smoke rising toward a light.
   - Set to 2 if most or all particles are stationary or passing in front of the light. Examples: rain passing in front of headlights, or stationary particles creating a glow around a point light.
   - Set to 3 if most or all particles are moving away from the light. Example: rain falling down past a street light.

10 From the Hardware Render Buffer window, select Render > Attributes.

11 In the Render Modes section of the Attribute Editor, select All Lights from the Lighting Mode pull-down menu.

12 Hardware render the scene to see the effect.



Normal Dir set to 2 (particles pass in front of the light)

Note Sometimes, unlit particles may appear brighter than lit ones when viewed in the hardware renderer. To fix this problem, subsitute the particle type with small radius spheres.

Tips If the particles aren’t illuminated as expected, make sure the light points at the particles. Also, position and rotate the camera so your view of the particles is from behind the light. Because changing the Normal Dir setting is convenient, it’s often fastest to simply try each setting and see which looks best. If you use stationary particles with Normal Dir set to 1 or 3, the particles won’t be displayed.

Set particle color

You can color particles with one of these methods:

- You can add and set *per object* attributes for the three RGB (red, green, and blue) components of the color. All particles in the object use the same color.
- You can add a *per particle* rgbPP attribute, which means you can set the color of each particle of the object independently.
- You can assign a shading group to the particle object.

If you use two or more of the methods, the one highest in the above list takes precedence. For example, if you add both per object and per particle attributes, the per particle attribute (rgbPP) controls the color.

Note	See "Per particle and per object attributes" on page 36 for more details on per particle and per object attributes.

To add and set a per object color attribute

1. Select the particle object and display the Attribute Editor.
2. In the Add Dynamic Attributes section, click Color.

   The Particle Color window is displayed.
3. Turn on Add Per Object Attribute and click Add Attribute.

   The Color Red, Color Green, and Color Blue attributes are displayed in the Render Attributes section of the Attribute Editor.
4. Set the color attributes to the values you want for the particle object color. These attributes are keyable.
5. Turn on Shading > Smooth Shade All to see the particle color in the workspace.

To add and set a per particle color attribute

1. Select the particle object and display the Attribute Editor.
2. In the Add Dynamic Attributes section, click Color.

   The Particle Color window is displayed.
3. Turn on Add Per Particle Attribute and click Add Attribute.

   The rgbPP attribute is displayed in the Per Particle (Array) Attributes section of the Attribute Editor.
4. Right-click the rgbPP data box and choose an attribute editing technique from the pop-up menu. See "Techniques for setting per particle attributes" on page 55 for details. Note that you cannot key this attribute or any other per particle attributes.
5. Turn on Shading > Smooth Shade All to see the particle color in the workspace. If necessary, deselect the particle object to see the color.

To assign a shading group

1. Select the particle object and display the Attribute Editor.
2. In the Add Dynamic Attributes section, click Color.

   The Particle Color window is displayed.
3. Turn on Shader.

The Multilister is displayed.

4 Create and assign the desired material with shading group to the particle object (see *Rendering*).

5 Turn on Shading > Smooth Shade All to see the particle color in the workspace.

Set particle opacity

You can give particles any amount of transparency. For software render particle types (Cloud, Blobby Surface, Tube), you must set a transparency attribute in the material that controls the particle object's color. See *Rendering* for details. Do not add a per object or per particle opacity attribute to a software render particle type. The attribute value has no effect when you software render the scene.

For hardware render particle types, you can add a per object or per particle opacity attribute. See the procedures that follow. Note that if you add per object and per particle opacity attributes to a hardware render type, the per particle attribute controls the opacity.

To add and set a per object opacity attribute

1. Select the particle object and display the Attribute Editor.
2. In the Add Dynamic Attributes section, click Opacity.

   The Particle Opacity window is displayed.
3. Turn on Add Per Object Attribute and click Add Attribute.

   The Opacity attribute is displayed in the Render Attributes section of the Attribute Editor.
4. Set Opacity value.

   A value of 0 makes all particles in the object completely transparent. A value of 1 makes the particles completely opaque. Values between 0 and 1 create partial transparency. This attribute is keyable.
5. Turn on Shading > Smooth Shade All to see the effects of the opacity.

To add and set a per particle opacity attribute

1. Select the particle object and display the Attribute Editor.
2. In the Add Dynamic Attributes section, click Opacity.

   The Particle Opacity window is displayed.
3. Turn on Add Per Particle Attribute and click Add Attribute.

   The opacityPP attribute is displayed in the Per Particle (Array) Attributes section of the Attribute Editor.

4 Right-click the opacityPP box and choose an attribute editing technique from the pop-up menu. See "Techniques for setting per particle attributes" on page 55 for details. Note that you cannot key this attribute or other per particle attributes.

5 Turn on Shading > Smooth Shade All to see the effects of the opacity.

Tips	If you see unexpected colors when you hardware render overlapping, colored, transparent particles, select the particle object and turn on Depth Sort in the Render Attributes section of the Attribute Editor. Maya will draw distant particles first and closer particles last. This creates accurate coloring at the expense of slower scene play. To eliminate the display of individual particles, add a per particle opacity attribute to the particle object, then use the Component Editor to assign the particle an opacityPP value of 0.

Set particle lifespan

You can give particles a lifespan to make them disappear from the scene after they reach a specified age. Though you can give a lifespan to particles you've created with the Particle Tool, lifespan is typically used with emitted particles. For example, you can use lifespan to fade out emitted smoke or fire. You can set the lifespan for all the particles or on a per-particle basis.

All particle shapes have the following attributes:

Per-object lifespan attributes

- lifespanMode
- lifespan
- lifespanRandom

Per-particle lifespan attributes

- lifespanPP
- finalLifespanPP

These attributes are used together to control lifespan.

Set per-object lifespan

Setting per-object lifespan assigns the same lifespan to all particles in the particle object. You can make particles live forever or make all particles die at the same age.

To make particles live forever

1. Select the particle object and display the Attribute Editor.
2. In the Lifespan Attributes section, set Lifespan Mode to Live forever.

To make particles die at the same age

1. Select the particle object and display the Attribute Editor.
2. In the Lifespan Attributes section, set Lifespan Mode to Constant.
3. Set Lifespan to the desired value.

   The Lifespan value is the number of seconds that the particle object exists after it is created. A particle object you create with the Particle Tool is considered to be created immediately at the beginning of the first frame of the scene. An emitter particle is created when it is emitted. The default value of 1 makes the particle object disappear after one second.

> Tip — If your particles aren't dying, make sure you haven't set Lifespan to a large number—one that exceeds the time duration of the scene's Time Slider.

Set per-particle lifespan

Setting per-particle lifespan assigns different lifespans to each particle in the particle object. You can make particles die at random ages, assign a lifespan to individual particles, or use an expression to control the lifespan.

To make particles die at random ages

1. Select the particle object and display the Attribute Editor.
2. In the Lifespan Attributes section, set Lifespan Mode to Random range.
3. Set Lifespan to the desired value.
4. Set Lifespan Random to the desired value.

   The lifespan is uniformly distributed with Lifespan as the mean and Lifespan Random as the width of the distribution.

   For example, if Lifespan is 3 and Lifespan Random is 2, then each particle will have a lifespan between 2 and 4.

   The random number stream for Random Range lifespan mode is maintained inside the particle object and is automatically reseeded when you rewind. You do not have to reseed it yourself.

5 If you want to change the random number stream for random range lifespan, specify a value for General Seed.

   Normally, you don't need to change this value. If you want two identical looking particle objects, you can set this value to match the other particle object's General Seed value.

> Note While lifespan = 4, lifespanRandom = 2 is mathematically equivalent to the expression lifespanPP = 3 + rand(2), these two methods of setting lifespan will not give identical results because the random number streams are different.

To control lifespan of individual particles

1 Select the particle object and display the Attribute Editor.

2 In the Lifespan Attributes section, set Lifespan Mode to lifespanPP only.

3 In the Per Particle (Array) Attributes section, right-click the lifespanPP data box and select Component Editor from the pop-up menu.

4 Click (the Select by Component Type icon).

5 In the workspace, select the specific particles you want to edit, and click Load Components.

   See "Set particle attributes with the Component Editor" on page 55 for more information on using the Component Editor.

6 Click the entry box for lifespanPP and enter a value.

To use expressions to control lifespan

1 Select the particle object and display the Attribute Editor.

2 In the Lifespan Attributes section, set Lifespan Mode to lifespanPP only.

3 In the Per Particle (Array) Attributes section, Right-click the lifespanPP data box and select Creation Expression or Runtime Expression (before or after dynamics calculation) from the pop-up menu.

4 Create a creation or runtime expression to assign a unique value to each particle (see *Expressions*).

   Note that you cannot key this attribute or other per particle attributes.

How lifespan is determined

A read-only per-particle attribute called finalLifespanPP stores the final result of lifespan computations no matter what mode you are using. The particle shape uses finalLifespanPP to kill particles and to drive ramps.

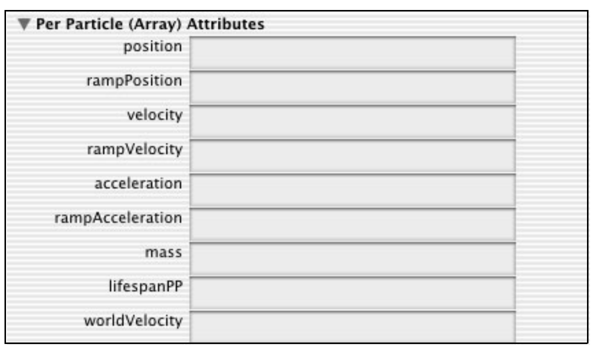
- In Constant or Random Range mode, the value computed from lifespan/lifespanRandom is stored for each particle.
- In lifespanPP only mode, the value of lifespanPP is stored.
- In Live Forever mode, a maximum-value float is stored.

You cannot set the value of finalLifespanPP yourself in any way (for example, with an expression, ramp, or component editor). Like age, it is an attribute the particle shape computes for you. But you can read its value (but not assign its value) in an expression.

In Random Range mode, lifespans are determined for each particle when it is born, and are stored in finalLifespanPP. If you change lifespan or lifespanRandom, it affects new particles but does not affect particles that are already born. If you rewind and re-play, all particles are affected.

Set attributes on a per particle basis

When you select a particle object, the Per Particle (Array) Attributes section of the Attribute Editor displays attributes you can set on an individual particle basis. By default, the following attributes are displayed:



The attributes provide different ways to control position or motion.

The position, velocity, and acceleration attributes offer a direct approach to controlling particle motion. The attributes rampPosition, rampVelocity, or rampAcceleration let you use a ramp texture to control the position, velocity, or acceleration. The mass attribute affects motion calculations resulting from another object’s dynamic influences on the particles, for instance, fields or collisions. The following pages introduce techniques for setting these and other per particle attributes.

If you add other per particle attributes to a particle object, for example, rgbPP or opacityPP, the attributes appear at the bottom of the Per Particle (Array) Attributes section.

Note	Per particle attributes are unavailable in the Channel Box and Attribute Spreadsheet.

Techniques for setting per particle attributes

There are various techniques to set per particle attributes:

- Use the Component Editor to select individual particles in the particle object and assign values (see "Set particle attributes with the Component Editor" on page 55).
- Use a ramp texture to vary values (see "Set particle attributes with a ramp texture" on page 57).
- Create a creation or runtime expression to assign a unique value to each particle (see *Expressions*).

Note the following:

- You cannot key per particle attributes.
- The techniques above are listed in order of increasing complexity and versatility.
- Not all techniques are available for each attribute. To see which techniques are available, right click the box to the right of the attribute name in the Per Particle (Array) Attributes section of the Attribute Editor.
- If you set the position, velocity, or acceleration *and* use a ramp to set its counterpart ramp attribute, only the ramp value is used.
- To avoid unexpected results, do not set a combination of values for position, velocity, and acceleration.

Set particle attributes with the Component Editor

The Component Editor provides an intuitive way to set per particle attributes. You simply enter the desired attribute values for one or more selected particles in the particle object.

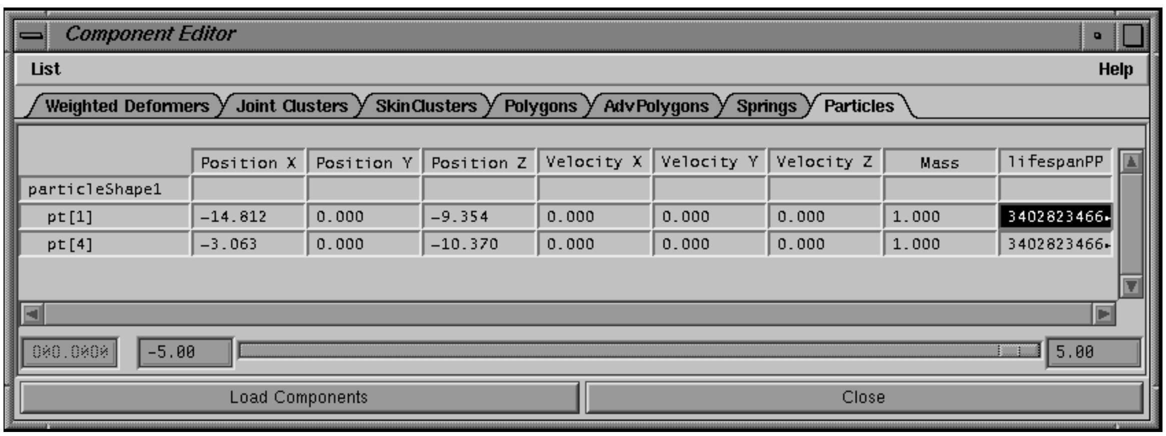
If necessary, add the per particle attribute you want to edit (see "Add dynamic attributes" on page 34 for information on adding attributes).

To set per particle attributes with the Component Editor

1. Select the particle object.
2. In the Per Particle (Array) Attributes section of the Attribute Editor, right-click the attribute data box and choose Component Editor.
3. Click  (the Select by Component Type icon).
4. In the workspace, select the specific particles you want to edit, and click Load Components.

   You can drag a selection box to select adjacent particles, or you can Shift-select non-adjacent particles. The particles turn yellow when selected. If the particles don't become selected, make sure the selection mask is set correctly for particle selection. See *Basics* for details.

   After you select the particles, the Component Editor displays the selected particles and per particle attributes you can set. The current attribute values are shown for each particle.



   The particles are abbreviated pt[0], pt[1], pt[2], and so on. The number in brackets is not the particleId. It's an array index used internally by Maya. If you want to be certain you're setting the value of the correct particle, select only that particle in the workspace. The Component Editor displays only the selected particle.

5. Click the entry box you want to edit and enter a value for the desired attribute.

   You can drag through multiple boxes in rows or columns to enter the same value for all. You can also click an attribute name to select and enter a value for all particles listed in the table.

The value you enter becomes the initial state value—the value of the attribute at the first frame. Unless dynamic effects alter the value of the attribute, this value is used for the entire animation. See ”Set the initial state of dynamic objects” on page 392 for more details.

Many per particle attributes have vector values. A vector is a related group of three components. For example, velocity is made of velocity X, velocity Y, and velocity Z components. As another example, rgbPP has a vector value made of R, G, and B components. These are labeled rgbPP[0], rgbPP[1], and rgpPP[2] in the Component Editor.

If you're unfamiliar with how to use vector components to set position, velocity, or acceleration, experiment with the various component values until you develop an intuitive sense of their effect on motion.

Set particle attributes with a ramp texture

A ramp texture is a 2D texture in which the color or grayscale value changes from one value to another across the extent of the image. You can use a ramp to control per particle attributes as particles age. You can also connect a ramp texture to any other per particle attribute of the particle. You commonly use ramp textures with the rgbPP, opacityPP, and radiusPP attributes. For most per particle attributes, you must add the attribute to the particle object before you use the ramp to control its value.

Note	To set the color, transparency, or incandescence of a Blobby Surface particle object over its lifespan see ”Example: Using the particles' age to color particles” on page 129.

Use color ramps

You can use a color ramp to change vector array (per particle) attributes as particles age. When you create a ramp for a vector array (per particle) attribute, the attribute is connected to a color ramp.

To use a color ramp

1 Select the particle object.
2 Open the Per Particle (Array) Attributes section of the Attribute Editor, then right-click the desired attribute and select Create Ramp.

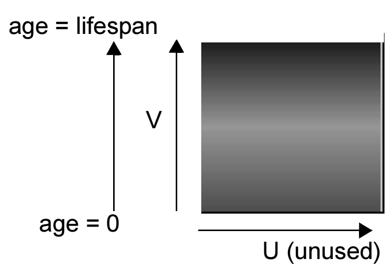
Note	
	To control the position, velocity, or acceleration, right-click the rampPosition, rampVelocity, or rampAcceleration attribute box and select Create Ramp. You cannot use a ramp directly on position, velocity, or acceleration. If you use a ramp *and* an expression to set the position, velocity, or acceleration, the ramp sets the value before the expression controls its value in a frame. The rampPosition, rampVelocity, or rampAcceleration can receive input only from ramps. Each frame, Maya calculates the ramp attributes first, then calculates dynamics based on any value in the ramp attributes.

This creates a default ramp that controls the attribute as the particles age.

3 In the Lifespan Attributes section of the Attribute Editor, set the Lifespan Mode to Constant and set Lifespan to the number of seconds you want the particles to exist in your animation.

4 In the Per Particle Attributes section of the Attribute Editor, right-click the attribute box for which you just created a ramp, then select arrayMapper*n*.outColorPP > Edit Ramp.

The Attribute Editor displays the default color ramp that controls the attribute. The three component values of the attribute are set to the RGB values of the vertical component (V) of the ramp over the particle's lifetime. Values at the bottom of the ramp are used at the beginning of the lifespan, values at the top are used at the end of the lifespan.



Default ramp for vector array attributes

By default, at the beginning of each particle's lifespan, the left, middle, and right vector components of the attribute equal 1, 0, 0 because the RGB value at the bottom of the ramp is red (1, 0, 0).

At the middle of the lifespan, the vector components equal 0, 1, 0 because RGB is green (0, 1, 0). At the end of the lifespan, the vector components equal 0, 0, 1 because RGB is blue (0, 0, 1). Maya interpolates vector components between these three points.

5 Edit the ramp's RGB color values as desired (see *Rendering* for details).

You can use any numerical values for R, G, and B in the Color Editor when you edit ramp values. Values outside the range 0 to 1 are valid, but they have no meaningful color equivalent in the display of the Texture Sample swatch. Do not edit HSV values or you'll likely get incorrect results.

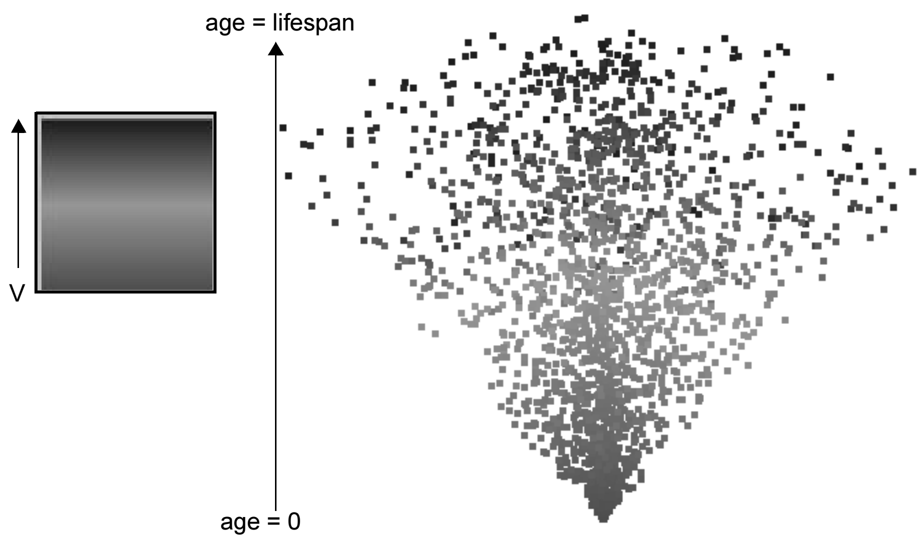
The only attributes for which the color in the swatch necessarily represent the component values of the attribute are rgbPP and incandescencePP. For other attributes that don't have color equivalents, you can edit color values in the Color Editor by entering values in the R, G, and B boxes rather than by clicking colors directly.

For example, you can set rampAcceleration to 100, 0, 0 at some point in the lifespan by setting R, G, and B values of the corresponding point in the ramp to 100, 0, 0. These values create a red color in the ramp that's identical to 1, 0, 0. However, adjacent points in the swatch will be interpolated with 100, 0, 0 differently than with 1, 0, 0.

You can also change the color sequence by changing other ramp attributes in the Attribute Editor such as Noise and Noise Frequency. See *Rendering* for details on the ramp attributes.

Remember, the horizontal component (U) of the ramp has no effect on the attribute by default. More precisely, the controlling component of the ramp is the leftmost vertical edge of the Texture Sample swatch in the Attribute Editor. To use the horizontal component as well, see "Customize per particle attribute control with ramps" on page 61.

An example of a default ramp applied to emitted particles follows:



Color ramp applied to emitted particles

Use grayscale ramps

You can use a grayscale ramp to change float array attributes as particles age. When you create a ramp for a float array (per particle) attribute, the attribute is connected to a grayscale ramp. In grayscale ramps, black represents a value of 0 and white represents a value of 1.

Notes	If you create a ramp for emitterRatePP, you control the emission rate over time for a directional or omni emitting particle object. The attribute exists in the particle object that emits, not in the emitted particles. You cannot control lifespanPP with a ramp.

To use a grayscale ramp

1 Select the particle object.

2 Open the Per Particle (Array) Attributes section of the Attribute Editor, then right-click the desired attribute and select Create Ramp.

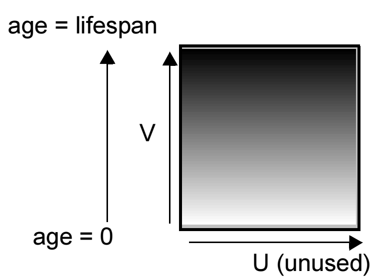
This creates a default ramp that controls the attribute as the particles age.

3 In the Lifespan Attributes section of the Attribute Editor, set the Lifespan Mode to Constant and the LIfespan to the number of seconds you want the particles to exist in your animation.

If you are working with emitterNameRatePP, set the Lifespan to the number of seconds you want the *emitter* to exist. The emitting particle object will disappear after the Lifespan value specified.

4 In the Per Particle Attributes section of the Attribute Editor, right-click the attribute box for which you just created a ramp, then select arrayMapper*n*.outColorPP > Edit Ramp.

The Attribute Editor displays the default grayscale ramp that controls the attribute. The values of the attribute are set to the R color values in the vertical component (V) of the ramp over the particle's lifetime. Values at the bottom of the ramp are used at the beginning of the lifespan, values at the top are used at the end of the lifespan.



Default ramp for float array attributes

By default, at the beginning of each particle's lifespan, the value of the attribute equals 1 because the R component at the bottom of the ramp is 1 (The color is white, with an RGB value of 1, 1, 1.)

At the middle of the lifespan, the value of the attribute equals 0.5, because the R component at the bottom of the ramp is 0.5. (The color is gray, with an RGB value of 0.5, 0.5, 0.5.)

At the end of the lifespan, the value of the attribute equals 0, because the R component at the bottom of the ramp is 0. (The color is black, with an RGB value of 0, 0, 0.)

Maya interpolates the values between the three points in the ramp.

Tip	Age is the attribute used with ramps by default. You can also use other attributes.

5 Edit the ramp's R color values as desired (see *Rendering* for details).

You can use any numerical values for R in the Color Editor when you edit ramp values. Values outside the range 0 to 1 are valid, but they have no meaningful color equivalent in the display of the Texture Sample swatch. Do not edit HSV values or you'll likely have incorrect values applied to the attribute.

Typically, opacityPP is the only attribute for which the color in the swatch represents the values in the attribute. For other attributes that don't have color equivalents, you can edit color values in the Color Editor by entering values in the R box rather than by clicking colors directly.

You can also change the color sequence by changing other ramp attributes in the Attribute Editor such as Noise and Noise Frequency. See *Rendering* for details on the ramp attributes.

Remember that the horizontal component of the ramp has no effect on the attribute. To use the horizontal component as well, see "Customize per particle attribute control with ramps" below.

Customize per particle attribute control with ramps

A ramp has two inputs—U (horizontal) and V (vertical). By default, the V input of the array mapper is connected to the particle's age and the U input is not connected. You can change these input connections when you are creating the ramp.

Other ramp options such as circular, diagonal, or box change on both axes. You select these in the ramp Attribute Editor.

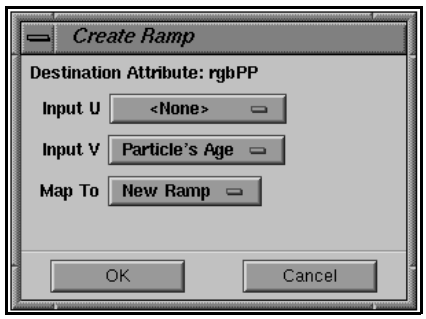
To tune ramp control of per particle attributes, you can:

- Use an existing ramp or customize U and V control of the ramp.
- Ensure the ramp colors portray attribute values meaningfully.
- Disconnect a ramp's control of an attribute.

To specify how particle's attributes get values from the ramp

1. If necessary, add the attribute to the particle.
2. Open the Per Particle (Array) Attributes section of the Attribute Editor, then right-click the desired attribute and select Create Ramp > ❒.

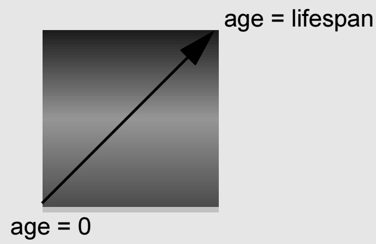
   This displays the Create Ramp options window.



3. To use an existing ramp, select it from the Map To menu.
4. To customize which attributes to control the U (horizontal) and V (vertical) ramp components, pick one of the following for **Input U** and **Input V**.
5. Click the attribute from the Input U or Input V menu.

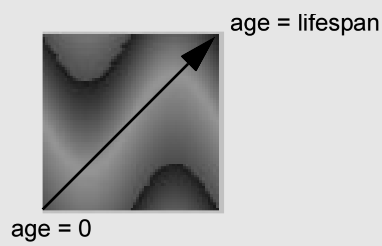
Tip

If you use Particle's Age for both Input U and Input V, the part of the ramp that controls the attribute over the particle lifespan is indicated by the diagonal line in the following figure:

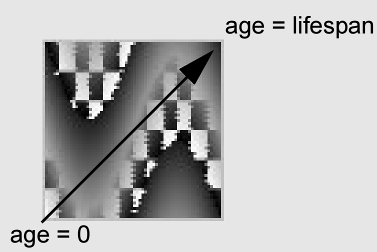


For the default ramp above, using Particle's Age for both Input U and Input V has no advantage, because the ramp color doesn't change in a U (horizontal) direction.

If you edit ramp attributes to create a ramp such as the following example, selecting Particle's Age for both Input U and Input V adds versatility to your control of the attribute:



You can even use a texture as input to the ramp's color. For example, you can map a 2D checker texture onto the color ramp as follows:



Editing the ramp

You can also edit the ramp to change the colors.

To edit the ramp

1. Display the Attribute Editor for the particle shape. To the right of rgbPP, right-mouse-click <-arrayMapper1.outColorPP and select Edit Ramp from the popup menu.

   The ramp for rgbPP is displayed in the Attribute Editor.
2. Edit the colors in the ramp. See “Ramp” in *Rendering* for information on editing the colors in the ramp.

Map to an existing ramp

In addition to mapping to a default ramp, you can map the destination attribute to a ramp that you have already used in the scene.

To map to an existing ramp

1. Place the pointer in the data box of the per particle attribute for which you want to create a ramp. Click the right mouse button to display the pop-up menu and select Create Ramp > ❐ to display the Create Ramp options window.
2. Click the Map to button to display the options menu.

   The ramps you have used in the scene are listed in the pop-up menu.
3. Select one of the ramps from the list.

Ensure ramp colors portray attribute values meaningfully

When you edit ramp values for RGB (or R) in the Color Editor, values outside the range 0 to 1 are valid but have no meaningful color equivalent in the display of the Texture Sample swatch. You can create meaningful color equivalents as follows:

To ensure ramp colors portray attribute values meaningfully

1. Set the lowest RGB (or R) values in the ramp to 0, and set the highest RGB (or R) values in the ramp to 1.
2. Right-click the attribute box of the per particle attribute that’s connected to a ramp, slide the pointer to the arrow to the right, and select Edit Array Mapper.

   The array mapper lets you scale the range of attribute values applied to the particles.
3. Click the array mapper tab of the Attribute Editor.
4. For the Min Value and Max Value, enter the lowest and highest values you want to use for the attribute.

By default, the Min Value sets the lowest value the attribute has at the beginning of its lifespan—the value at the bottom of the ramp. By default, the Max Value sets the highest value the attribute has at the end of its lifespan—the value at the top of the ramp.

Disconnect a ramp’s control of an attribute

To discontinue a ramp’s control of an attribute, right-click the attribute box of the per particle attribute that’s connected to a ramp, slide the pointer to the arrow to the right, and select one of these:

Break Connection

Breaks the array mapper’s input to the particle shape. This menu item doesn’t delete the ramp node or array mapper. Use this menu item if you don’t want to remove the ramp when you break the connection.

Delete Array Mapper

Breaks the array mapper’s input to the particle shape by deleting the array mapper. The ramp will also be deleted if it has no other connections. Use this menu item if you no longer need the ramp in the scene.

Ramp example

Coloring particles according to distance from origin

In this example, we’ll use the V coordinate of a ramp to color particles according to their distance from the origin. This example uses emitters and fields, which are covered in the following chapters. You may want to read about emitters and fields before doing this example. Or you can cut and paste the MEL code into the Script Editor to create the emitters and fields.

To use a ramp to color particles according to their distance from origin

1 Create a sphere volume emitter. Set the Volume Speed attributes as follows: Away From Center to 1; all the other speeds to 0. Set Scale (10, 10, 10).

   To do this, you can copy the following Mel commands from the online help and paste them into the Script Editor:

```
emitter -pos 0 0 0 -type volume -r 100 -sro 0 -nuv
0 -cye none -cyi 1 -spd 1
-srn 0 -nsp 1 -tsp 0 -mxd 0 -mnd 0 -dx 1 -dy 0 -dz
0 -sp 0 -vsh sphere -vof 0 0 0 -vsw 360 -tsr 0.5 -
afc 1 -afx 1 -arx 0 -alx 0 -rnd 0 -drs 0 -ssz 0 ;
```

```
scale 10 10 10;
```

```
particle;
```

```
connectDynamic -em emitter1 particle1;
```

2 Apply a radial field with Volume Shape set to Sphere to the particles. Set the Magnitude to -2, set Attenuation to 0, and disable Use Max Distance. Turn on Volume Exclusion and set Scale to 10, 10, 10. This will keep the particles inside the volume.

   To do this, you can copy the following Mel commands from the online help and paste them into the Script Editor:

```
radial -pos 0 0 0 -name pushIn -m -2 -att 0 -typ 0
-mxd -1 -vsh sphere -vex 1-vof 0 0 0 -vsw 360 -tsr
0.5 ;
```

```
scale 10 10 10;
```

```
connectDynamic -f pushIn particle1;
```

3 Add per-particle color to the particles.

   To do this, you can copy the following Mel command from the online help and paste it into the Script Editor:

```
addAttr -ln "rgbPP" -dt vectorArray particleShape1;
```

4 Select the particle and display the Particle Attribute Editor. In the space to the right of rgbPP, click the right mouse button and select Create Ramp > ❐ to display the Create Ramp Options.

5 Set InputU to None (the default), Input V to rgbVPP, and Map to New Ramp (the default). Click OK.

   Notice that a new attribute called rgbVPP appears in the Attribute Editor.

6 Right-mouse-click next to rgbVPP and select Runtime Expression (before or after dynamics calculation). In the Expression Editor, copy and paste the following line, and click Create:

```
 rgbVPP = mag( position ) / 10;
```

   The quantity mag(position) is always equal to the particle’s distance from the origin. We’ve divided by 10, which is the maximum distance any particle can be from the origin in our setup.

   In other words, this expression says that the V input to the ramp will have a value 0 if the particle is at the origin, and 10 if it’s all the way out at the boundary of the sphere.

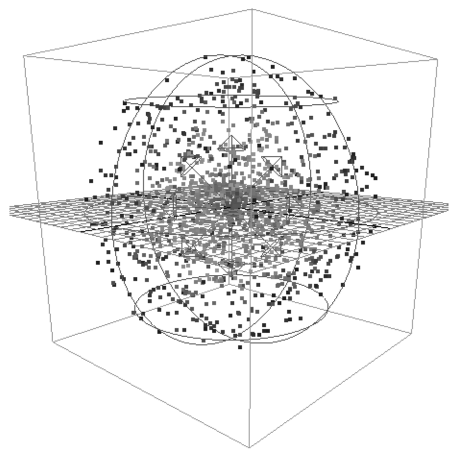
7 You may also want to make the particles a little bigger.

   To do this, you can cut the following Mel commands from the online help and paste them into the Script Editor:

```
addAttr -is true -ln "pointSize" -at long -min 1 -
max 60 -dv 2 particleShape1;
```

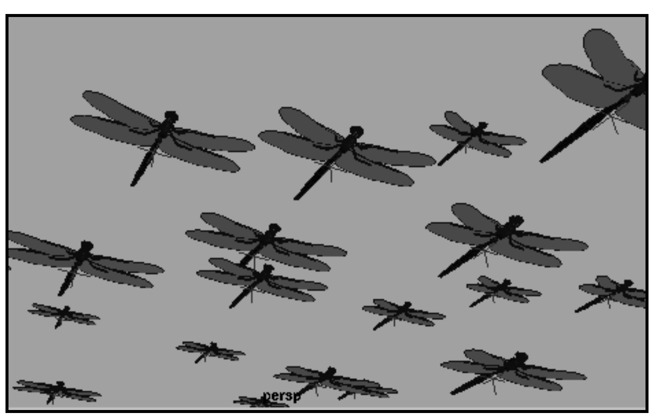
```
setAttr "particleShape1.pointSize" 4;
```

8 Now turn on smooth shading, rewind. and play the scene.



Instance geometry to particles (single and animated)

Maya's particle instancing feature saves you time when you need to animate many identical objects in a scene. For example, suppose you want to create a group of flying bugs where only the placement and orientation of the bugs differ. With Maya, you can animate a single bug, then create instances of the bugs that move with the position and orientation of animated particles.



The instances are not copies. They are references to the original object. Any changes you make to the original object changes the instanced objects. You can control the motion of the individual instanced objects by animating the per particle attributes that control them.

The instanced geometry object, called the *source geometry*, can be:

- a single object, animated or not animated.
- a sequence of objects in different shapes or positions. An example of an object sequence is a series of nearly identical bird objects in different wing-flapping positions.
- different objects to be displayed at different particles.

You can use object hierarchies instead of individual objects as the source geometry. Do not instance lights; they'll have no effect in rendering.

Create animated instances

You can instance a single object to a particle shape or a sequence of objects to a particle shape.

Before creating the instance, you create the source geometry and apply shading groups to it. You can animate the source geometry and the particles either before or after instancing, and you can place the source geometry anywhere in the workspace. You can optionally hide it as described in the procedure. You cannot control the color of the instanced geometry on a per particle basis.

To instance a single object to particles

1. Select source geometry.
2. Shift-select the particle shape.
3. Select Particles > Instancer (Replacement).
4. To hide the source geometry, select the geometry and select Display > Hide > Hide Selection.

To instance a sequence of objects to particles

1. Select the source geometry.
2. Select Particles > Instancer (Replacement) > ❐.

   The Particle Instancer Options window appears. The selected source geometry is listed in the Instanced Objects list.
3. To instance different objects at different particles, include them all on the list.

   To instance a sequence of objects, list the objects in the order you want them to appear in the sequence.

Use the Move Up and Move Down buttons to change the order of the objects. Use Add Selection if you want to add selected objects to the list. Use Remove Items to remove objects from the list.

For example, suppose you create four birds in different flapping positions:

- birdWingsHigh shows the wings over the bird's head.
- birdWingsMid shows the wings at mid position.
- birdWingsLow shows the wings below its body.
- birdWingsMidCopy, is a copy of birdWingsMid. This object is necessary so the wings cycle from low to high and high to low positions.

An appropriate ordering for the objects in the Instanced Objects list is:

0: birdWingsHigh
1: birdWingsMid
2: birdWingsLow
3: birdWingsMidCopy

The number to the left of the object indicates the sequence position. The 0 indicates the first object in the sequence.

4 From the Particle Object To Instance menu in the Particle Instancer Options window, select the animated particle object that the instances will follow.

Note that you can use emitted particles as the animated particles. When the particles emit, the instanced geometry appears with the particles.

5 Set the options as described in "Particles > Instancer (Replacement)" on page 153, then click Create.

Maya creates an instancer node that controls the instanced objects. This node is where you can alter the attribute settings you made in the Particle Instancer Options window.

6 To hide the source geometry, select the geometry and select Display > Hide > Hide Selection.

You might also choose to hide the particle object being instanced. The instancer doesn't make the particles invisible. Note, however, that you don't need to hide the particles if you will software render the animation. Particles don't appear in software rendered images.

> Tip If you want to move the rotation pivot point of the source object, select the object and enter *createInstancerPivot* in the Command Line. Thereafter, when you manipulate the rotation pivot in the source geometry, the instanced geometry rotates the same. Maya reorients any animation you previously applied to the translate, rotate, scale, and shear attributes of the source object.

Aim instanced geometry

The following general procedure is a way to aim instanced geometry.

To aim instanced geometry

1 Orient the geometry so the front points along the x-axis and the up-direction points along the y-axis.

2 Group the geometry and instance the group.

3 Set the "AimDirection" or "AimPosition" options to the direction you want the instanced objects to aim.

   For example, set AimDirection to Velocity to aim the instanced objects in the direction of movement. See "particleShape" on page 156 for information.

   The choices in the General Options popup menus include all the per particle attributes. You can control the motion of the individual instanced objects by animating the per particle attributes that control them.

Command-line instancing limitations

The `runup` command may fail to create instanced geometry when rendering arbitrary frames in an animation. Adding a `currentTime` setting before the runup creates the instanced geometry properly; that is, if something like `runup -mxf 60;` does not create the proper results when rendering with instanced geometry, use:

```
currentTime -e 1;
runup -mxf 60;
```

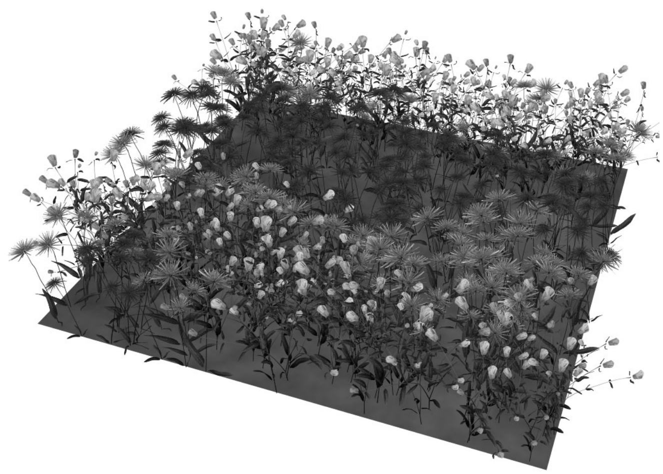
Particle instancing options

❖ "Particles > Instancer (Replacement)" on page 153.

Instance strokes from Paint Effects

You can use strokes from Paint Effects as instanced objects for particles. The image below was created using Paint Effects strokes instanced to particles.

See *Paint Effects* for information on painting in your scene, and see "Instance geometry to particles (single and animated)" on page 67 of this book for information on instancing.



Center strokes on particles

When you instance an object to a particle, the local origin of the object is instanced to the particle. With spheres and cubes, the local origin is the center of the object.

With the NURBS curves that comprise the strokes in Paint Effects, however, the local origin is not directly related to the geometry of the curve. No matter where you actually draw the curve, its origin is the world origin. For example, if you draw a short stroke in the workspace at position 5,0,5 to get a single flower and instance it to a particle object, the flowers will be offset from the particles by 5,0,5.

To center the Paint Effects geometry on your particles, there's a few things you can do. The obvious one is to start your curve at the origin. Unfortunately, this is hard to do in regular Paint Effects mode because you draw the curve, instead of placing down the CVs with grid snapping. We suggest the following procedures:

To center instanced strokes on particles (method 1)

1. Draw your curve near the origin.
2. Unhide the curve.
3. In component mode, select all of the CVs in the curve and use the Translate tool to move the CVs so the base of Paint Effects geometry (like the base of the flower) is at the origin.

   You move the CVs instead of the transform because the top-most translation and rotation of the instanced hierarchy is ignored.

To center instanced strokes on particles (method 2)

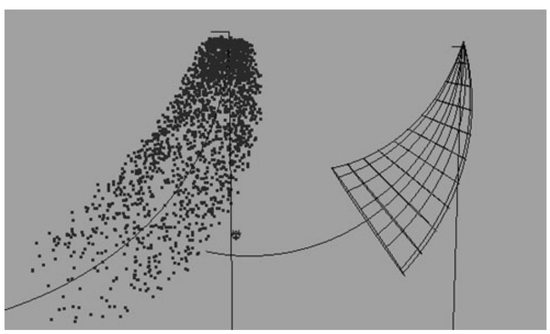
1. Group the stroke.
2. Move the transform of the stroke so that the base of the Paint Effects geometry is at the origin of the new group node.
3. Instance the group instead of the stroke.

   This has the same effect as moving the CVs.

Tip	Turn on Tube Completion. This lets you make a very short stroke while the tube continues to grow until it reaches the end of its number of segments.

Deform particles

Deformers let you influence particle positions in a variety of ways.



A bend deformer applied to particles (left) and geometry (right)

To deform particles

1. Create a scene with particles.
2. Select the particle object.
3. In the Animation menu set, click Deform.

4 Select the deformer you want to use on the selected particle object. You can deform particles with all deformers except for the skinning and blend shapes deformers.

   The deformers operate in the same way as they do for geometry. You can also layer deformers.

You can also combine particle deformation with forces: deform the particles, then apply the desired forces to the intermediate object. Alternatively, you could apply forces to particles before the application of a deformer.

Note	Particles affected by deformers do not exhibit inertia. Attributes such as velocity are not affected by deformations of particle shapes, and should be accounted for when animating particles using deformers. For example, particles within a flow deformed with a bend deformer will not slow down or speed up in relation to their position within the bend.

Work with advanced dynamics

Make an object move with a dynamic parent

1 Select the object to be parented.

2 Select Edit > Group.

   This parents the object to an empty group transform node.

3 Select Windows > General Editors > Connection Editor.

4 Load the particle shape of the dynamic parent into the Connection Editor's left side. Load the group object's transform node into the right side.

5 Connect World Centroid in the left side of the Connection Editor to Translate in the right side.

   This provides the dynamic parent's World Centroid (worldCentroid) values to the Translate attributes of the group node. The World Centroid of an object is the average position of its CVs, vertices, particles, or lattice points.

   In some instances, you'll need to connect the dynamic parent's Centroid attribute (local space) rather than its World Centroid to the group node's Translate attribute. If in doubt, try both to see which works as desired.

Note that you can connect to any vector attributes of the group node. For instance, you can connect a World Centroid attribute to the group node's Rotation or Scale attributes.

6. Select the child object (below the group node) and adjust its Translate X, Y, and Z attribute values as necessary to relocate it to its original position.
7. Play the animation.

Adjust frame-to-frame velocity conservation

A particle object has a Conserve attribute that influences the motion of particles whose velocity or acceleration attributes are controlled by dynamics (including particle expressions).

Set Conserve before you create the particles with the Particle Tool

- Select Particles > Particle Tool > ❐ to and set Conserve in the Tool Settings window.

To set Conserve after you create the particles

- Select the particle object and set its value in the Attribute Editor.

Apply forces in an object's local space

Dynamic forces influence NURBS and polygonal objects in the world space coordinate system. Regardless of how you rotate a cone, for instance, a gravity field with default option settings moves the cone down along the world space Y-axis.

By default, fields also influence particle objects in world space. However, you can cause a field to affect a particle object in the particle object's local space. For example, you can make a gravity field pull a particle object down the particle object's local Y-axis rather than down the world space Y-axis.

See "Forces In World" on page 158.

Obtain world position, velocity, and centroid

A particle object's position, velocity, and centroid attributes contain local space values. (The centroid of a particle object is the average position of its particles.) There are times when you need to know world space values for a particle object.

For example, suppose you apply gravity to a particle object to make it fall through a glass floor. You also key the particle object's Rotate Y attribute to make it rotate around the origin's Y axis as it falls. Suppose you need to know the exact world space position of the particles so you can change their color when they pass through the floor. You cannot get this information with the position attribute, because it contains the local space position.

You can get this information from the three world space counterpart attributes for position, velocity, and centroid:

- worldPosition
- worldVelocity
- worldCentroid

The centroid attribute is a compound attribute consisting of centroidX, centroidY, and centroidZ attributes. The worldCentroid attribute is a compound attribute consisting of worldCentroidX, worldCentroidY, and worldCentroidZ attributes.

You can read the value of worldPosition, worldVelocity, and worldCentroid as inputs in the Connection Editor. You can read the value of worldCentroidX, worldCentroidY, and worldCentroidZ in expressions. You cannot set the values of any of these attributes.

The worldVelocity contains the difference between the worldPosition in the current and prior frame. The worldVelocity is influenced by the object's (and object parent's) Translate, Rotate, and Scale values, so it includes the effect of keys you apply to them.

If you need to convert a particle object's world space velocity to local space velocity, be aware that you can use the worldVelocityInObjectSpace attribute of a particle object.

There is no worldAcceleration attribute.

Set input force with an attribute connection

Each field or spring connected to a particle object writes its vector force value to the Input Force[ ] attribute in the object. This attribute is an array of vectors. There is one Input Force array for each field connected to an object. For example, an object connected to three fields has Input Force[0], Input Force[1], and Input Force[2]. Each vector in the array represents the force on a single particle.

You can use the Connection Editor or API to feed one particle shape attribute into another's Input Force[ ] attribute. The attribute must be a vector array, for example, velocity.

When you connect an attribute to Input Force[0], if the two particle objects have the same number of particles, the values input to Input Force[0] match one for one. If the object containing Input Force[0] has less particles than the object providing input, the remaining values are ignored. If it has more, the last one is repeated.

Input Force[ ] also exists in the shape nodes of rigid bodies. It works on CVs or vertices the same as for particles.

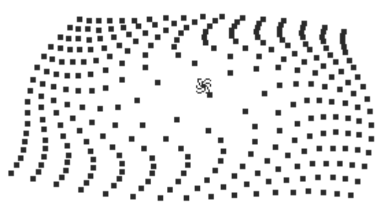
Control execution time of particle dynamics

There are several ways to control when particle dynamics operate:

- ”Change the start frame for a particle object” on page 76
- ”Control the timing of particle dynamics” on page 77
- ”Execute expressions after particle dynamics” on page 80

Change the start frame for a particle object

Each particle object has a Start Frame attribute that sets the time at which effects begin. For example, you can delay the effect of a vortex on a particle object until the frame of your choice.



This particle object’s Start Frame is set to 50 to delay the vortex field’s effect until frame 50.

As another example, suppose you create a character that walks through dust particles lying on a floor. The dust is connected to gravity, a floor collision object, and an air (wake) field on the character’s feet. You want the air field on the feet to stir up the dust starting at frame 100.

You can set the dust’s Start Frame to 100 to delay dynamic calculations for the dust until there’s a need for the calculations. This speeds up scene play and prevents the particles from bouncing slightly from the gravity and collision effect.

You could also prevent the dynamic calculations for dust by keying Is Dynamic off for the first 100 frames. However, by setting the Start Frame to 100, you also avoid having to wait for run-up calculations for the first 100 frames when you click a frame or scrub the Time Slider.

Note that when you create a particle object or an emitter emits a particle object, Start Frame is set to 1.0, by default. The value is the duration of a single frame in the current setting for time units.

To set the Start Frame of a particle object, select the particle object and enter a value for Start Frame in the Time Attributes section of the Attribute Editor.

Important	If you change the time units setting, you must set the Start Frame to the correct initial value so that Maya computes the start time again. An emitted particle object accepts emitted particles only at frames greater than or equal to its Start Frame. For example, to emit particles at frame number -5, set the Start Frame of the emitted particle object to -5 or less.

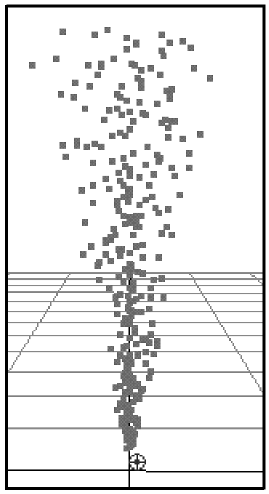
Control the timing of particle dynamics

With dynamic animation, you specify the actions you want an object to take, then let the software figure out how to animate the object. Historically, it has been challenging to synchronize this animation method with other actions. For example, it has been hard to increase the velocity of emitted particles in synch with a musical score coming to a crescendo.

Maya has a feature that overcomes such difficulties. Each particle object has a Current Time attribute that you can animate to slow or hasten a particle effect. This attribute contains the value of an independent clock time. In essence, you slow or speed the particle object's clock to slow or speed the dynamics that affect it. See "Understand the Current Time attribute" on page 79 for more details.

Example

An emitter shoots particles up in a 150-frame animation. The following steps show how to control the velocity of the particles by keying Current Time.



1. Select the emitted particle object (not the emitter).
2. In the Channel Box, click the Current Time attribute, then right-click and select Break Connections.
3. Set keys for Current Time at the following frames:

Frame	Current Time
0	0
50	100
100	110
150	200

At the beginning of the animation (frame 0), the Current Time of the emitted particles is synchronized with the scene time. Both have the value 0.

At frame 50, the Current Time is keyed to 100, twice the value of the scene time. For frames 0 through 50, this adds twice as much time to the action of the emitted particles as the scene time. The emitted particles therefore move at twice the speed that would occur without the manipulation of the Current Time.

At frame 100, Current Time is keyed to 110. From frames 50 to 100, this slows the progression of time for the emitted particles relative to the scene time. The emitted particles move at a slower speed than would exist without the manipulation of the Current Time.

At frame 150, Current Time is keyed to 200, once again speeding the progression of time for the emitted particles. The particles move faster again.

Tip	To conveniently slow or speed up dynamic effects, use the Graph Editor to tune the animation curve of Current Time after you key its values. See *Animation*.

Understand the Current Time attribute

Each animated particle object runs according to its own clock. You can make this clock independent of the scene clock that advances with the ticks in the Time Slider.

Each particle object has a Current Time attribute that contains its clock value. Unlike the scene clock's time variable, you can set the value of the Current Time attribute with keys, expressions, or other animation techniques. While the scene clock is a timekeeper with fixed length increments, a particle object's clock has variable increments that you set through the Current Time attribute. You can vary the increments to slow or speed dynamic effects on the particles.

For animation to occur in the particle object, its Current Time requires an input connection that changes as the scene plays. By default, the scene's predefined time variable provides input to Current Time.

If you use a nonparticle shape expression to assign a value to Current Time, the value is interpreted as the type of Time units specified in the Window > Settings/Preferences > Preferences > Settings window. If an fps rate is selected, for instance, Film (24 fps), the Current Time value is interpreted as frames.

For example, suppose you've set Working Units Time value to Film (24 fps). This means your animation uses the frame number as the time value. Suppose further you use the following expression statement:

```
currentTime = 50;
```

The 50 is considered *frame* 50, rather than, say, 50 milliseconds.

Note that you can animate a decreasing Current Time to achieve effects such as an emitter sucking previously emitted particles back into itself. To see the results of this effect, you must turn on Cache Data for the emitted particles and play the animation.

> **Tip** If you've broken the connection to Current Time and want to reconnect it to the scene time, select the particle shape node (not the transform), then select Particles > Connect to Time.

Execute expressions after particle dynamics

Runtime expressions execute before or after dynamics in each frame, depending on what you selected when you created the runtime expression.

In pervious version of Maya, you had to choose whether a particle shape runtime expression executed before or after its dynamics were computed. In Maya 6, you can specify when you create the runtime expression whether it is to be executed before or after dynamics calculation.

For example, suppose you use a gravity field to drop particle blobs on a character, then you write a runtime expression for the blob object's position to prevent its particles from penetrating the character's surface. If you created that expression as a runtime before dynamics, you might see some surface penetration because the gravity field executes after your position-constraining expression. You can create or copy this expression to Runtime after dynamics so the residual effect of the dynamics doesn't interfere with your expression.

Duplicate particle objects

You can duplicate a particle object with or without the original's input connections from dynamics such as fields.

To duplicate a particle object with input connections

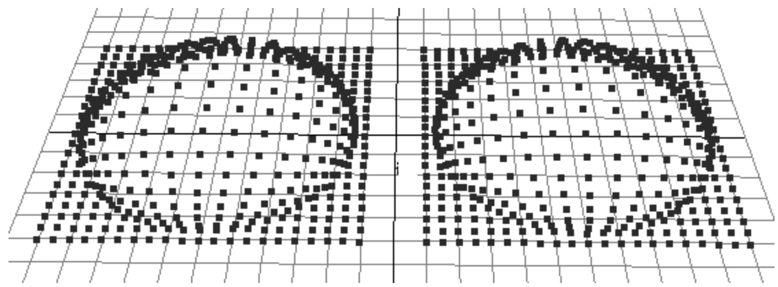
1. Select the particle object.
2. Select Edit > Duplicate > ❐.
3. In the Duplicate Options window, turn on Duplicate Input Connections. If this option is dim, turn off Duplicate Input Graph first.
4. Click Duplicate.
5. Reposition the duplicate as desired.

   The Duplicate menu item copies incoming connections from dynamic effects, so the same effects that influence the original object influence the duplicate. This doesn't copy outgoing connections from the particle object. This means that if you apply a field to the original particle object from some position, the duplicate particle object will be influenced by the field from a corresponding local space position.

You can cause the field to influence the duplicate from the world space position. In the dependency graph for the duplicate particle shape node, break the field's input connection to the particle shape node, then use the Dynamic Relationships Editor to connect it to the field.

Example

Suppose you create a particle grid and connect it to a radial field positioned below it. You create a duplicate of the particle grid with Duplicate Input Connections, then move the duplicate grid to a new position. The right object mirrors the radial field's effect even though the radial field is positioned under the left object.



The right object mirrors the radial field's effect even though the radial field is positioned under the left object.

To duplicate a particle object without input connections

1. Select the particle object.
2. Select Edit > Duplicate > ❐.
3. In the Duplicate Options window, turn on Copy but no other options, then click Duplicate.
4. Reposition the duplicate as desired.
5. Select Particles > Connect to Time.

   For a particle object's dynamic effects to work subsequently, the object needs an incoming connection to its Current Time attribute. By default, the scene's predefined time variable provides this input. If you select Edit > Duplicate with default options, it doesn't copy incoming connections to the new object. The dynamics therefore won't work for the new object unless you select Particles > Connect to Time.
6. Rewind and play the animation.

Assign image sequences to sprites

We've supplied a MEL script (applySprite.mel) in the Gifts/smoke directory (under the main Maya install directory) to make it easier to assign image sequences to sprites. To use the MEL script, see "Sprites" on page 40.

If you want to assign an image sequence to sprites manually (without using the MEL script), you can use the following procedure.

To assign an image sequence to the sprites

1. In the menus above the workspace, turn on Shading > Smooth Shade All and Shading > Hardware Texturing.
2. Select the particle object.
3. In the Attribute Editor, set Particle Render Type to Sprites.
4. To add default Particle Render Type attributes that let you tune the appearance, click the Add Attributes For Current Render Type button.
5. Create a Lambert material with shading group, and assign it to the selected particle object. See Rendering for details.
6. From the Rendering menu set, select Lighting/Shading > Shading Group Attributes to display the Lambert node in the Attribute Editor.
7. In the Common Material Attributes section of the Attribute Editor, click the map button to the right of the Color slider.
8. In the Create Render Node window, click the Textures tab. In the 2D Textures section, click the File button to create a texture file node.
9. In the File Attributes section of the Attribute Editor, click the file browser button to the right of the Image Name box.
10. Select the first file in the sequence and click Open.

Important	The image files you use as sprites must have filename extensions in the format *file.n*, not *file*.000*n* (zero-padded extensions).

11. In the Hardware Texture Cycling Options section, turn on Use Hardware Texture Cycling.
12. Specify the Start Cycle Extension, End Cycle Extension, and By Cycle Increment.

    These attributes specify the files available for use in cycling. These attributes do not select the actual files cycled. You do this by setting the Sprite Num or spriteNumPP attribute in a later step.

The Start Cycle Extension and End Cycle Extension specify the filename extension number of the first and last files available for cycling. The By Cycle Increment sets whether each frame is available for cycling every other frame, every third frame, and so on.

For example, to use every file of a 30-file sequence starting with Smoke.0, you would set the Start Cycle Extension to 0, the End Cycle Extension to 29, and the By Cycle Increment to 1. To use every other file of the same file sequence, you would set By Cycle Increment to 2. In other words, you could cycle through Smoke.0, Smoke.2, Smoke.4, and so on.

13 Turn on Use Frame Extension.

14 Because file textures load only on demand, you must set keys to force the Frame Extension value to go from Start Cycle Extension to End Cycle Extension consecutively during the frames are required. You can do this with simple keyframes, set driven key, or an expression.

For example, if Start Cycle Extension is set to 1, End Cycle Extension is set to 10, and By Cycle is set to 1, you would do the following:

- Go to frame 1, set Frame Extension to 1, and set a key.
- Go to frame 10, set Frame Extension to 10, and set a key.

15 If you want to assign an identical image sequence to all the sprites, do the procedure that follows, “Assign an identical image sequence to all sprites.” If you want to assign a different image sequence to each sprite, do the procedure ”Assign a different image sequence to each sprite” on page 85.

Assign an identical image sequence to all sprites

When you added default Particle Render Type attributes for sprites in a prior step, Maya added a Sprite Num attribute to the particle object. Sprite Num sets which files are cycled from the pool of files specified by Start Cycle Extension, End Cycle Extension, and By Cycle Increment. More specifically, Sprite Num is an index into the pool of files.

For example, suppose you have a file sequence starting with Smoke.0. If you set the Start Cycle Extension to 0, the End Cycle Extension to 29, and the By Cycle Increment to 1, the pool of images and Sprite Num values that represent them are as follows.

Image	Sprite Num
Smoke.0	1
Smoke.1	2

Image	**Sprite Num**
Smoke.2	3
Smoke.3	4
—	—
Smoke.29	30

If you set the Start Cycle Extension to 0, the End Cycle Extension to 29, and the By Cycle Increment to 2, the pool of images and Sprite Num values that represent them are:

Image	**Sprite Num**
Smoke.0	1
Smoke.2	2
Smoke.4	3
Smoke.6	4
—	—
Smoke.28	15

You can set keys or write an expression to animate Sprite Num and thereby cycle through the images. All sprites display the same image each frame.

For instance, if the start frame of your scene is frame 0 and you have 30 images named Smoke.0 through Smoke.29, you can use this creation expression:

```
particleShape1.spriteNum = (frame % 30);
```

and this runtime expression:

```
particleShape1.spriteNum = (frame % 30);
```

As the animation plays, all sprites cycle through the 30 images. At frames 0 through 29, the sprites display Smoke.0, Smoke.1, Smoke.2, and so on through Smoke.29. At frame 30, the cycle repeats for the next 30 frames.

See *Expressions* for details on writing expressions.

Assign a different image sequence to each sprite

1. With the particle shape node selected, open the Add Dynamic Attributes section of the Attribute Editor and click the General button.
2. In the Add Attribute window, click the Particle tab and select spriteNumPP. Click OK.

   This adds the per particle spriteNumPP attribute to the particle shape node. The spriteNumPP attribute works like Sprite Num, but you can set spriteNumPP on a per particle basis. See ”Assign an identical image sequence to all sprites” on page 83 for details on Sprite Num.
3. Write creation and runtime expressions to animate the particle shape node’s spriteNumPP attribute. (If the particle object’s transform node is currently selected rather than the shape node, press the down arrow to correct the selection.)

   For example, if the start frame of your scene is frame 0 and you have 30 images Smoke.0 through Smoke.29, you can use this creation expression:

   ```
   particleShape1.spriteNumPP = rand(0,30);
   ```

   and this runtime expression:

   ```
   particleShape1.spriteNumPP = (particleShape1.spriteNumPP
   + 1) % 30;
   ```

   When you rewind the animation (or the sprite is emitted), the creation expression assigns a random number between 0 and 30 to the particleShape1.spriteNumPP attribute. Each particle therefore starts its sprite display with an image randomly selected from the 30 images.

   When you play the animation, the runtime expression executes each frame (before or after dynamics calculation) and increments each sprite’s number by 1. This makes each sprite cycle through the entire range of images. After Smoke.29 appears for a sprite, the animation cycles through the images again starting at Smoke.0.

   For more information, see the *Expressions* guide.

Export particle data

If you want to use your Maya particle data with another renderer or particle system, you can export it in binary format (.pdb) or ASCII format (.pda). See the *dynExport* command in the online MEL Command Reference.

Work with emitters

Create emitters

The following procedures describe how to create emitters with default settings. For details on other settings, see ”Set emitter attributes by typing entries” on page 90. In the following procedures, you must play the animation as the final step to see the particles emit.

You cannot add an emitter to an object if the emitter already has some other technique controlling its translate attributes. Furthermore, after you add an emitter to an object, you cannot control the emitter’s translate attributes with another technique, for example, an expression.

If you select only some of an object’s CVs, vertices, or particles, the emitter applies only to those selected.

To emit from a position

1. Make sure nothing is selected.
2. Select Particles > Create Emitter.

   This creates a point emitter at the origin.

To emit particles from points on a surface

1. Select the NURBS or polygonal surface.
2. Select Particles > Emit from Object > ❒ to display the options window.
3. In the options window, select Surface from the Emitter Type pop-up menu.
4. In the options window, click Create.

   For details on emitting from entire surfaces evenly, see ”Spread emission more evenly from NURBS surfaces” on page 106.

To emit particles from points on a curve

1. Select the curve.
2. Select Particles > Emit from Object > ❒ to display the options window.
3. Select Curve from the Emitter Type pop-up menu.
4. In the options window, click Create.

To emit from all CVs or vertices of a selected object

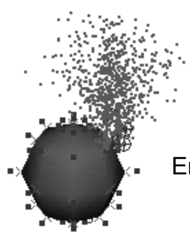
1. Select the object.

2 Select Particles > Emit from Object > ❐ to display the options window.
3 Select Omni or Directional from the Emitter Type pop-up menu.
4 In the options window, click Create.

To emit particles from selected vertices, CVs, or edit points

1 Select the object.
2 Click  (the Select by Component Type icon).
3 Select the components you want to emit particles. See *Basics* for details.
4 Select Particles > Emit from Object > ❐ to display the options window.
5 Select Omni or Directional from the Emitter Type pop-up menu.
6 In the options window, click Create.

Adding an emitter to components creates a set of components named for the emitter. You can change the components that emit by editing the set membership (see *Basics*). Note that you cannot emit from components of an object in a referenced scene.



Emitting from selected CVs

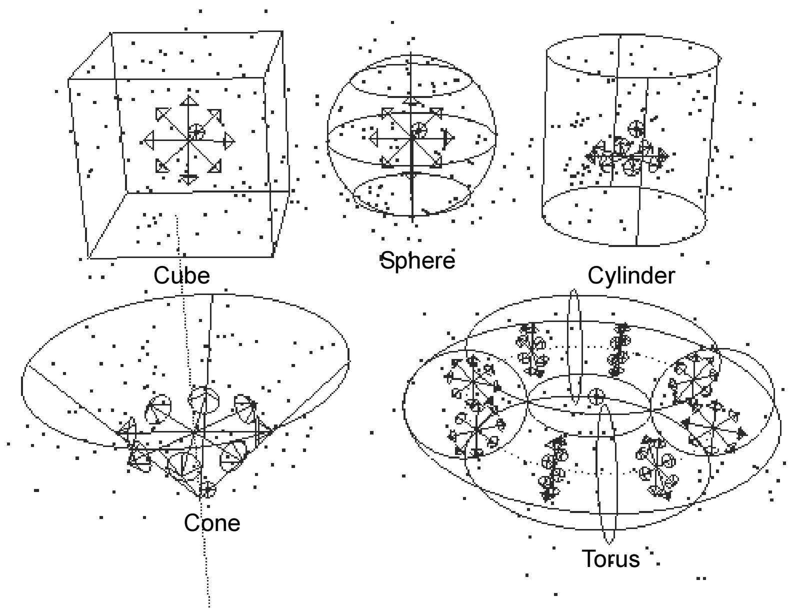
Tips The MEL *emit* command lets you add particles to selected positions of an existing particle object without creating an emitter. You can set the values of any per particle attributes for the created particles. This lets you create effects similar to emission but with greater control.

For example, you can make emitted particles use the attribute values of the emitting particles. You can also use the emit command in an expression to make particles emit at the point where particles die.

For detailed examples of the emit command, see the online MEL Command Reference: Alphabetical.

To emit particles into a volume shape

1. Select Particles > Create Emitter > ❐.
2. Under Basic Emitter Attributes, set the Emitter Type to Volume.
3. Under Volume Emitter Attributes, click Volume Shape and select a shape from the menu.



4. Set the Volume Emitter attributes as desired. See ”Volume Emitter Attributes” on page 148 for information on the volume emitter attributes.
5. Click Create.

   The implicit shape of the volume emitter is displayed on the screen. You can move, rotate, scale, or shear the emitter.

Note	You can’t deform the volume or use an arbitrary volume.

To hide the shape of the volume emitter

- Select the emitter and select Display > Hide > Hide Selection.

To hide the emitter icon that represents the emitter

1. In the workspace, select the emitter icon.

Sometimes it's easier to select the emitter in the Outliner. If you are emitting from an object rather than position, the emitter is indented under the object's name in the Outliner.

Select Display > Hide > Hide Selection.

2 To display the icon again, select Display > Show > Show Last Hidden.

To delete an emitter

1 Select the emitter.

2 Press Backspace (Windows, Linux, and IRIX) or Delete (Mac OS X).

When you delete an emitter, the emitted particle object is not automatically deleted. If this particle object isn't connected to other items in the scene, you can delete it also. Use the Outliner to delete it.

Note	Emission begins only at the frame specified by the emitted particle object's Start Frame attribute (by default, frame 1). For example, to emit particles at frame number -5, you must set the Start Frame of the emitted particle object to -5.

Edit attributes of an emitter

The attributes of the emitter control the initial position, direction, quantity, and speed of the emitted particles. You can set the emitter options before you create the emitter, or you can create a default emitter and set the attributes after the emitter is created.

To set emitter options before creating the emitter

1 Select Particles > Create Emitter > ❐ or Particles > Emit from Object > ❐.

The options window appears.

2 Set the options described in "Set emitter attributes by typing entries" on page 90.

Changes you make to the options window affect emitters you create after the changes.

3 Create the emitter.

To edit emitter attributes after creating the emitter

1 Select the emitter you want to edit.

2 Do one of the following:

- Display the Attribute Editor or use the Channel Box to set the attributes as described in "Set emitter attributes by typing entries" on page 90.
- You can edit the most commonly used emitter attributes with the workspace manipulators. See "Change emitter attributes with workspace manipulators" on page 90.

Tips	
	You can key the emitter position or parent the emitter to a moving object the same as for any other object. See *Animation* for information on setting keys.
	If you see clumps of particles or irregular emission direction when you animate the motion of an emitter, try the following remedies to smooth the emission:
	Make sure the Playback Speed is set to Play every frame in the Playback Speed pulldown in the Window > Settings/ Preferences > Preferences > Timeline window.
	Increase the Rate or Max Distance setting in the emitter.
	Select Solvers > Edit Oversampling or Cache Settings to increase the Over Samples value.

Set emitter attributes by typing entries

You can set the following attributes in the options window, Attribute Editor, or Channel Box:

Change emitter attributes with workspace manipulators

You can use manipulators in the workspace to edit several emitter attributes:

- Rate
- Direction (directional point emitters only)
- Spread (directional point, curve, and surface emitters only)
- Speed
- Normal Speed (curve and surface emitters only)
- Tangent Speed (curve and surface emitters only)
- Max Distance and Min Distance

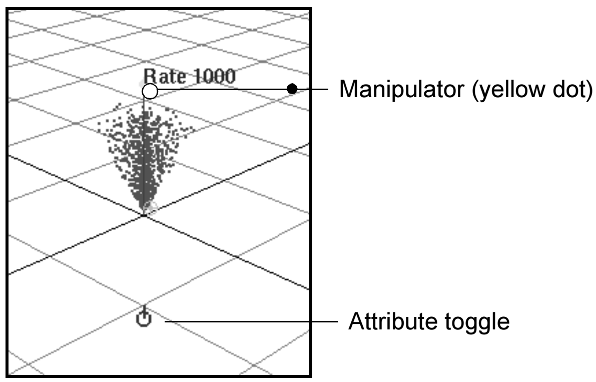
The manipulators offer an interactive alternative to typing entries in the Attribute Editor.

To use a manipulator on an attribute

1 Select the emitter.

2 Select the Show Manipulator Tool from the Tool Box.

A yellow attribute manipulator appears next to the emitter object. Dolly towards the manipulator to get a clearer view. Initially the attribute manipulator controls the Rate setting. An attribute toggle also appears near the emitter. It controls which attribute you can manipulate.



3 Drag the dot next to the Rate to change its value.

4 Click the attribute toggle to display a different attribute.

Each time you click the attribute toggle, you display a different attribute manipulator. The manipulator is an icon (often a dot) you typically drag to change its value. Details specific to attributes are in "Use manipulator icons" on page 92. The color of the manipulator and attribute toggle indicates which is active. Yellow is active, blue is inactive.

After clicking the attribute toggle a number of times, you'll see a display mode where all the manipulator icons are displayed without the attribute names. When you click an icon in this mode, the attribute name appears next to it. You can then manipulate the attribute. Click the attribute one more time and you'll see the first manipulator displayed in the cycle—the Rate attribute.

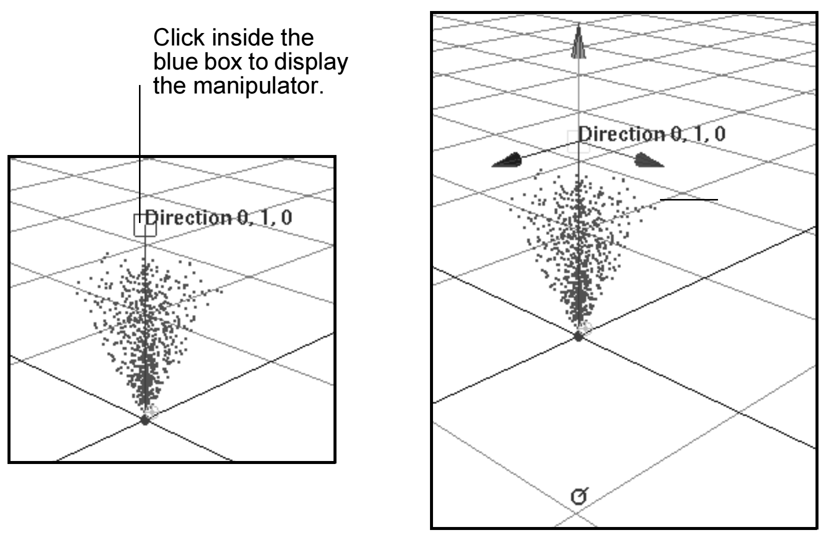
Use manipulator icons

Most attribute manipulators work the same way. Drag the icon away from the emitter icon to increase the value. Drag toward the emitter icon to decrease the value. Attribute manipulators that require different manipulation techniques are described in the following topics.

> **Note** In some instances, the line that represents an attribute is a relative value. For example, if you drag a Rate value to a number over 100, the line snaps back to a fixed position. This occurs so that you can drag the value as high as desired without losing sight of the manipulator in the workspace. Some lines and curves represent accurate measurements, for example, Max Distance.
>
> Note also that the Max Distance and Min Distance dots indicate a range of emission. The two dots originally are superimposed. To improve your view of the dots, drag a dot until the two dots are visible.

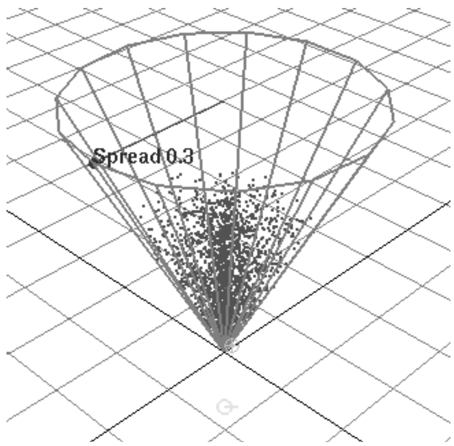
Direction X, Y, Z

Click inside the blue box. A manipulator identical to the Move tool appears. Drag the center to move in all directions, or drag one of the arrows to move with directional constraint. This manipulator appears only for directional point emitters.



Spread

Drag the dot icon roughly perpendicular to the emission direction. This displays a cone that indicates the Spread angle. When you drag the dot, a line appears in the workspace indicating the direction you can drag.



Edit attributes of emitted particles

You can set the usual particle attributes of emitted particles, for example, color and lifespan (see "Edit particle attributes" on page 36). In addition, you can set attributes that pertain only to emission.

To edit the attributes of emitted particles

1. Select the emitter.
2. Open the Attribute Editor.
3. Click the particleShape tab.

Tip	You can enhance the motion of the emitted particles by applying fields such as gravity to the emitter or emitted particles.

Set emitted particle transform attributes

When you manipulate the Translate or Rotate attributes of an emitter, you have no influence on the position of particles after emission. For example, suppose you create a stream of smoke emitting upward. If you key the Rotate X attribute of the emitter, the emitted particles do not rotate after emission. They maintain the direction they had when first emitted.

To change the motion of the emitted particles in mid flight, you can key their Translate, Rotate, or Scale attributes. This alters their position each frame.

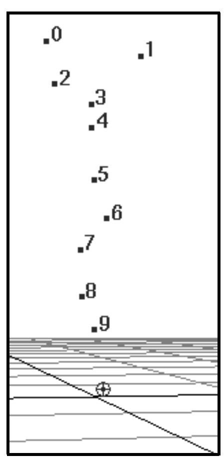
Keying Rotate X of the emitted particles creates different motion than keying Rotate X of the emitter.

Lessen emission quantity

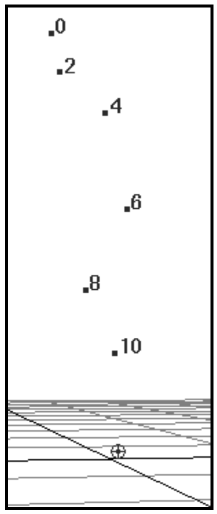
You can limit or lessen emission quantity by setting the Max Count and Level Of Detail attributes of an emitted particle object.

Example

Suppose you emit particles at a rate of 50 per second. If you display the emitted particles as the Numeric render type, you might see the following particles after you play seven frames.



If you set Level Of Detail of the emitted particles to 0.5, the emitted particle object ignores roughly half the emitted particles. The following particles are displayed after you rewind and play seven frames.



Maya removes particles *and* particleId values randomly. This is useful if you create expressions with conditional statements based on a particleId.

Duplicate emitters

You can duplicate an emitter to reproduce the emission elsewhere. For example, you can create an emitter that makes a cloud of particles, then create an identical cloud elsewhere in the scene. You can make the duplicate with or without the original's connections to dynamic effects such as fields. Expressions applied to the emitted particles are not duplicated.

To duplicate an emitter with connections to dynamic effects

1. Select the emitting object (or positional emitter) and the emitted particle object.
2. Select Edit > Duplicate > ❐.
3. In the Duplicate Options window, turn on Duplicate Input Connections.
4. Click Duplicate.
5. Drag the copied emitted particles *and* the copied emitting object away from the originals.

To duplicate an emitter without connections to dynamic effects

1. Select the emitting object (or positional emitter) and the emitted particle object.
2. Select Edit > Duplicate > ❐.

3. In the Duplicate Options window, click Edit > Reset Settings to turn on default options.
4. Click Duplicate.
5. Select Particles > Connect to Time.

   Emitters have a hidden Current Time attribute. For an emitter to work when you play an animation, it needs an incoming connection to its Current Time attribute. By default, the input to Current Time is provided by the scene's predefined time variable. (See "Understand the Current Time attribute" on page 79 for details.)

   If you select Edit > Duplicate with default options, it doesn't copy incoming connections to the new object, so its dynamics don't work. This is why you select Particles > Connect to Time.
6. Drag the copied emitted particles and the copied emitting object away from the originals.

> Note The Seed number of the duplicate emitted particle object is the same as the original. The duplicate's emission randomness is therefore identical to the original, by default. For details on changing the randomness of emission, see "Work with emission randomness" on page 105.

Connect emitters and particles

There are several techniques for connecting emitters and particles:

- An easy way to connect an existing particle object to an emitter is with the Dynamic Relationships Editor (see Chapter 8). You *must* use this method if you want to connect an existing particle object to a selected emitter on an object with multiple emitters.
- After you finish setting the color and other attributes of emitted particles, you can conveniently create a new emitter and connect it to the particles. The new emitter will emit particles with the same attribute settings. The new emitter can emit at a different rate or direction, or even be a different type of emitter.

  For example, suppose you create a fire effect and want to expand its size. You can add emitters to the original emitting object, then connect the existing emitted fire particle object to the new emitters. The particles emitted from the added emitters have the same attributes as the existing particles.
- You can create two or more empty particle objects, define unique attributes for each, and connect them to one emitter. This lets you emit different particle effects, such as fire and smoke, from the same emitter.

If you connect multiple particle objects to an emitter, the emitter emits them all with their respective attributes. For example, if you have a red particle object and a green particle object and you connect both to an emitter, the emitter emits both green and red particles.

- You can connect emitted particles to the emitter of your choice on an object with multiple emitters.
- To replace an existing emitted particle object with another, use the Dynamic Relationships Editor to disconnect the existing particle object so that only the desired particle object is emitted.
- You can add an existing independent emitter to a single selected object. The emitter must not already exist in another object. If you create an emitter with the Maya API, use this technique to add the emitter to the desired object.

To connect a particle object to an emitter

1. Select the particle object you want to connect to the emitter.
2. Shift-click the emitter (or the object that owns it). If you are selecting the emitter in the Outliner, use Ctrl-click (Windows, Linux, and IRIX) or Command-click (Mac OS X) to make a non-contiguous selection.
3. Select Particles > Use Selected Emitter.

To create an empty particle object

1. Select Particles > Particle Tool > ❐ to display the options window.
2. Enter 0 in the Number of Particles box and press Enter (Windows, Linux, and IRIX) or Return (Mac OS X).
3. Move the pointer to the workspace and press Enter or Return.

   This creates an empty particle object in the Outliner.

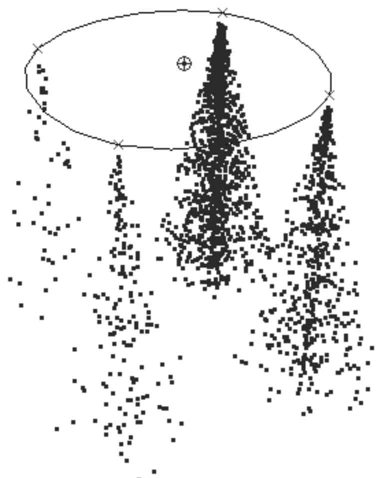
To add an existing emitter to an object

1. Select the object and shift-select the emitter.
2. Select Particles > Emit from Object.

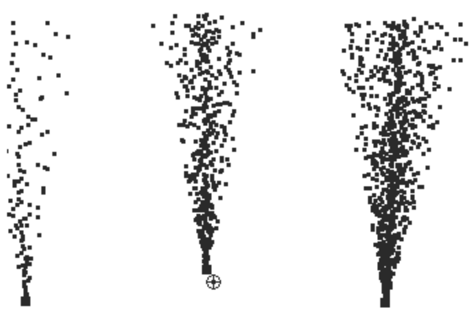
   The emitter becomes indented under the object in the Outliner. Moving the position of the object moves the position of the emitter.

Vary emission from different points of point emitters

You can use a different emission rate for each particle, CV, vertex, edit point, or lattice point of Omni or Directional point emitters. For example, you can emit a ring of fire from a circle’s edit points and vary the emission at each point to enhance the irregularity found in natural fire. *You cannot vary emission on a per-point basis for objects whose Emitter Type is either Surface or Curve..*



This NURBS circle emits from its edit points at rates of 50, 150, 1000, and 500.



This three-particle object emits at a rate of 50, 150, and 300.

Tip	If you emit from a particle object, you can use an expression or MEL command to obtain the id of all particles that emit the particles. You can use the id to query the emitting object's attribute values, for example, acceleration, velocity, and finalLifespanPP. To do this, you must add the parentId attribute to the emitted particle shape node. Note that if you use the MEL emit command to create the particles that emit, the parentId attribute of those emitted particles is always 0.

To vary emission from particles

1 Create and select the emitting particle object.

2 Select Particles > Per-Point Emission Rates.

This creates an attribute named *emitter*RatePP for the particle object that emits. This attribute lets you vary emission rates on a per particle basis.

3 In the Attribute Editor, open the Per Particle (Array) Attributes section.

4 Without breaking the connection to *emittername*RatePP, right click the *emitterName*RatePP box and select the desired method to set its value. See "Set attributes on a per particle basis" on page 54.

Note that you can turn the emitter's Use Per-Point Rate attribute on or off to toggle per-point emission rates. In the Outliner, the emitter is indented under the emitting object.

To vary emission from a NURBS or polygonal point emitter

1 Finish modeling the geometry before adding a point emitter.

2 Select the emitting geometry.

3 Select Particles > Per-Point Emission Rates.

4 Select the emitting object's shape node.

5 In the Extra Attributes section of the Attribute Editor, open the *emitterName* Rate PP section.

For the *emitterName*RatePP attributes, enter values in the Attribute Editor boxes corresponding to the CVs, vertices, edit points, or lattice points. To learn which box corresponds to a point, experiment with a large value in a box. You can also use an expression, ramp, or Artisan to set individual rates.

> Note: If you add CVs, vertices, or edit points after you add and tune per-point emission rates, you can select Per-Point Emission Rates a second time to set the additional entries for *emitterName*RatePP. However, the Attribute Editor *emitterName*RatePP entries no longer correspond to the same identifiers for CVs, vertices, or edit points. You'll need to correct the *emitterName*RatePP entries.
>
> If you delete CVs, vertices, or edit points, delete the emitter, create a new emitter, and choose Per-Point Emission Rates again.

Use a texture to color emission or scale the rate

When you create a NURBS or polygonal surface emitter, you can use a 2D or 3D texture to scale the emission rate at parts of the surface. For instance, you can scale the emitter Rate to 90% at the most luminous parts of a texture, and to 0% at the least luminous parts. Luminance is the measurable brightness of a surface.

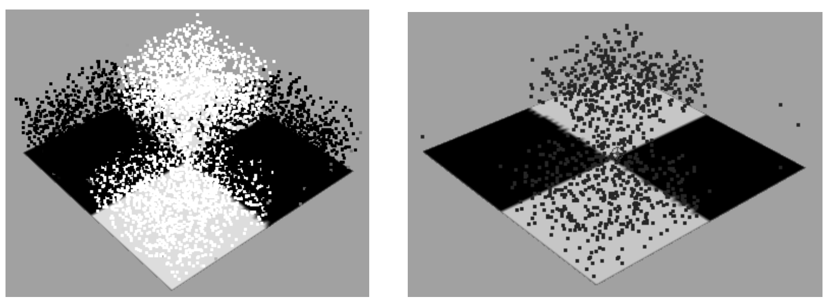
You can also color emitted particles with a 2D or 3D texture. The texture can be the same as the underlying surface or any other texture you create in the scene.

Use a 3D texture with emission

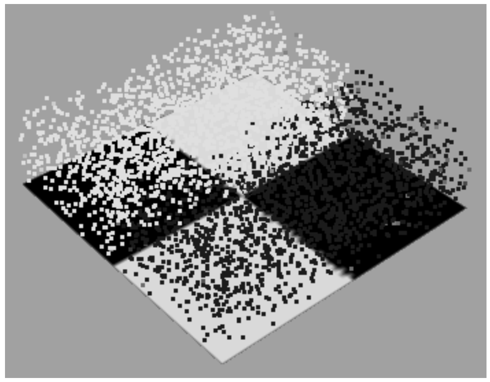
You cannot use a 3D texture to control surface emission, but you can convert it to a 2D texture and then use it to control surface emission For information on converting a 3d texture to a 2D file texture, see the online documentation in the *Rendering Reference* for “Convert to File Texture.”

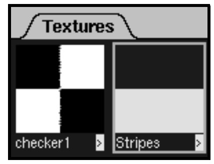
Examples of how to use a texture with emission

Examples of how to use a texture with emission follow:

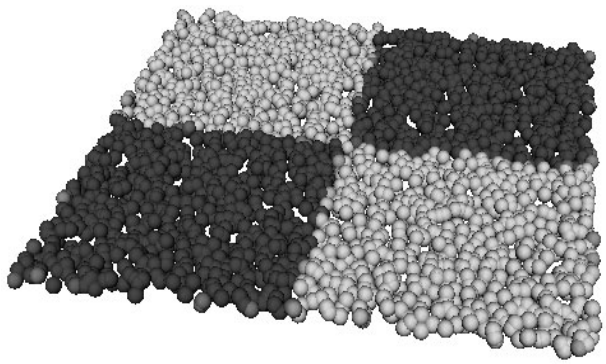


Black particles emit from the black parts, and white particles from the white parts (left). Red particles emit from the white parts (right).

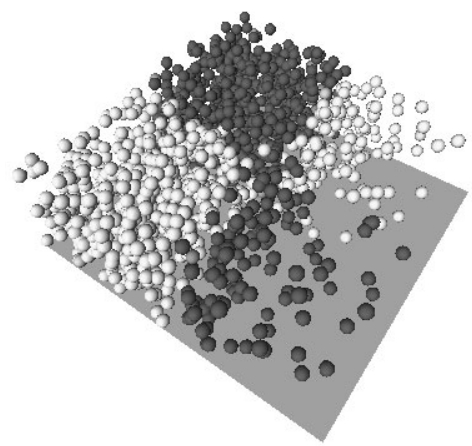


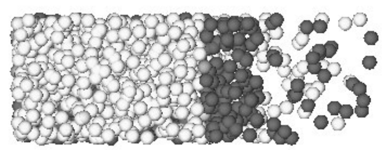


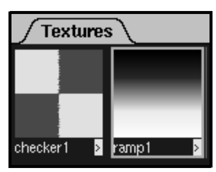
Particles emit from a striped texture not assigned to the plane. The plane's surface has a checker texture that's unused in emission. Note that you can also emit from a hidden plane or from a plane without a texture.



You can display particles as any render type, for example, spheres.







Side view of image

While a checker texture colors the emitted particles, a ramp texture scales the emission rate where the ramp's luminance is low.

To use a texture to scale the emission rate

1 Add a surface emitter to the surface.

2 Select the emitted particles and add a per particle color attribute (rgbPP) from the Add Dynamic Attributes section of the Attribute Editor.

Click the Color button to display a window that lets you add a per particle color attribute.

3 Select Window > Rendering Editor > Hypershade.

If you need to create the texture, do so now. See *Rendering* for details. You don't need to make the texture a part of a shading group. After you create the texture, make sure it's visible in the Hypershade. Leave the Hypershade displayed on your screen.

4 In the Outliner, select the emitter; it's indented under the emitting surface object.

5 Display the Texture Emission Attributes section of the Attribute Editor.

6 Middle-drag the texture's icon from the Hypershade onto the Texture Rate attribute in the Attribute Editor.

7 Turn on Enable Texture Rate.

8 Set the Emit From Dark attribute as desired:

When Emit From Dark is off, fully luminous parts of the texture emit at 100% of the specified emitter rate. Parts of the texture that lack luminance don't emit at all. Other parts emit between 0 and 100% in proportion to the luminance. In simple terms, bright colors emit more than dark colors.

When Emit From Dark is turned on, the opposite luminance scheme applies. Parts of the texture that lack luminance emit at 100% of the specified rate, while fully luminous parts don't emit at all.

Tips	To emit at distinctly different rates from parts of the surface, use a texture with sharp contrast in luminance. For instance, use a striped black and white texture. If you see a few particles emitted from a seemingly wrong position of the texture, check that the coloring in that area has consistent luminance. You can figure the exact luminance value of a color by adding its RGB components and dividing by 3. For instance, the luminance of RGB value 1, 0, 0 equals 1/3, or 0.33. The luminance of RGB value 0, 0, 1 also equals 1/3, or 0.33. In other words, a saturated red has the same luminance as a saturated blue.

9 Rewind and play the scene to see the texture-scaled emission rate.

Note that you must turn on Shading > Smooth Shade All to see particle colors in your scene.

To color emitted particles with a texture

1 Add a surface emitter to the surface.

2 Select the emitted particles and add a per particle color attribute (rgbPP) and a per particle opacity attribute (opacityPP) from the Add Dynamic Attributes section of the Attribute Editor.

Click the Opacity and Color buttons to add per particle opacity and color attributes.

3 Select Window > Rendering Editors > Hypershade.

If you need to create the texture, do so now. See *Rendering* for details. You don't need to make the texture a part of a shading group. After you create the texture, make sure it's visible in the Hypershade. Leave the Hypershade displayed on your screen.

4 In the Outliner, select the emitter; it's indented under the emitting surface object.

5 Display the Texture Emission Attributes section of the Attribute Editor.

6 Middle-drag the texture's icon from the Hypershade onto the Particle Color attribute in the Attribute Editor.

7 Turn on Inherit Color.

When you rewind and play the scene, the particles emit from the texture with the same color and position as the texture. Note that you must turn on Shading > Smooth Shade All to see particle colors in your scene.

8 Turn on the following attributes as desired: Inherit Opacity, Use Luminance, and Invert Opacity.

> Note   If you set the opacityPP attribute of the emitted particle with an expression or other technique, the attribute value overrides the above attributes.

Tips for advanced users

- In the Attribute Editor, you can click the color swatch of the emitter's Particle Color attribute to apply a color rather than a texture to emitted particles. If you've already connected a texture to the Particle Color attribute, you must first break the input connection to it from the texture's outColor attribute. You can break the connection in the Hypergraph's dependency graph.
- If you add multiple emitters to a surface, you create multiple emitted particle objects—one for each emitter. You can use the Dynamic Relationships Editor to disconnect all the emitted particle objects from the emitters and reconnect a single one to all the emitters. Each emitter will emit only that particle object with its attribute settings.

  The Attribute Editor for each of the emitters has its own texture emission attributes, so you can apply a different texture (or color) to each group of particles even though each emitter emits the same particle object.

  With this technique, you can layer a single emitted particle object with various colors to create complex particle effects such as smoke or fire. The particles use the emitted particle object's attributes except for color (and optionally opacity), regardless of which emitter emits them.

  For example, suppose you add two surface emitters to a plane and connect a single particle object to the emitters. The plane can emit particles colored by a black and white checker texture at the same time as particles colored by a purple to yellow ramp texture. The particles are a single emitted particle object. You can display the emitted particles as any render type, for example, spheres, by setting the appropriate attribute in the Attribute Editor for the emitted particle object.

Work with emission randomness

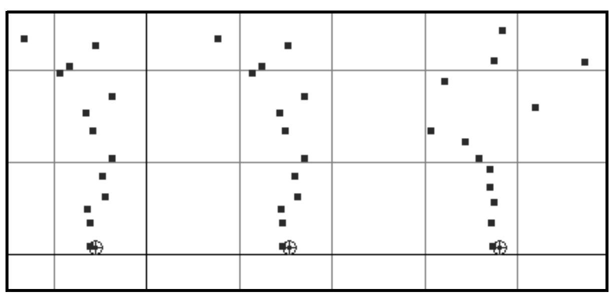
If you create two emitters with identical attributes, the random positioning of the emitted particles is identical, by default. You can cause each emitter to emit particles in different random positions by giving each emitted particle object a different value for its Seed attribute.

If a single emitter emits into several particle objects that have different Seed values, the emission positioning differs for each. If they all have the same Seed value, the positioning is the same.

The default value for each Seed entry is the same (1), so if you want two identical emitters to emit to different positions, you must set one of the Seed values to a different number. A new Seed value takes effect only after you rewind. When you change the Seed used by one emitted object, you won't alter any other emission in the scene.

The size of a Seed value has no significance. For instance, a Seed value of 2 doesn't create twice as much randomness as a Seed value of 1. Each number, for example, 1, 17, or 1000, simply sets a different sequence of random numbers.

To set the value of the Seed, select the emitted particle object and open the Emission Random Stream Seeds section of the Attribute Editor. You'll see a data box and slider named for the associated emitter, for example, *emitter1*. If you use several emitters to emit into a single particle object, there is one data box and slider for each emitter. (Though the attribute name is Seed, the name doesn't appear in the Attribute Editor.)

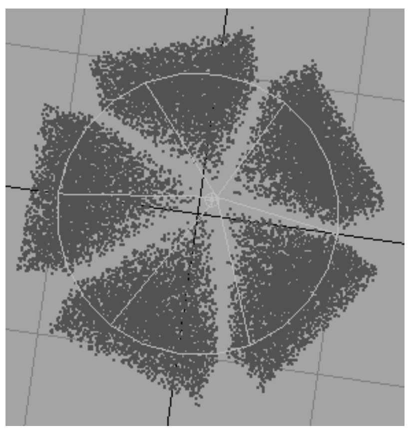


The three emitters are identical. The emitted particle shapes of the left and middle emitters have identical Seed values. The one on the right differs.

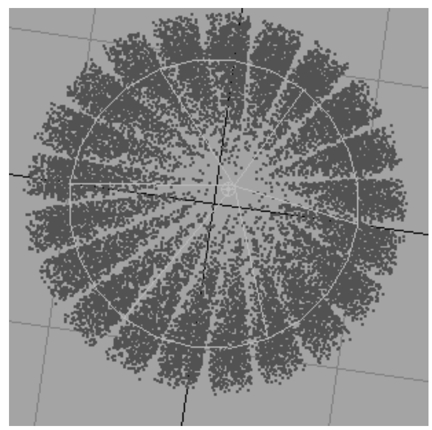
Tips	You can use one emitter to emit two identical streams of emission without having to duplicate an emitter. First, connect two identical emitted particle objects to the emitter. Next, make sure the Seed values are the same, and turn off Emission In World (in the Emission Attributes section of the Attribute Editor) for both emitted particle objects. Finally, move one of the emitted particle objects away from the other. If you connect several particle objects to multiple emitters and want to know which Seed corresponds to a particle object, examine the dependency graph in the Hypergraph.

Spread emission more evenly from NURBS surfaces

You can spread the emission more evenly from NURBS surface emitters. This is often beneficial for surfaces that have sharp curves or bends.



Particles not emitted evenly from surface of cone (viewed from top).



Improved evenness of emission.

To spread emission more evenly

1. Select the emitting NURBS surface.
2. Display the Attribute Editor.
3. Click the geoConnector tab. Expand the width of the Attribute Editor if you don't see the tab.
4. Increase the value of the Tessellation Factor, then rewind and play the animation.

The default value is 200. Try doubling the value as many times as necessary to get the desired evenness. Higher values increase processing time. If you add two or more surface emitters to an object, each uses the same Tessellation Factor.

Tips	You can also tune surface emission direction with the Normal Speed and Tangent Speed attributes. See "Tangent Speed" on page 147. If you add a surface emitter to a NURBS or polygonal surface, you can use a MEL statement or expression to learn the UV coordinates where each particle was emitted. You must add parentU and parentV attributes to the emitted particle shape, then turn on Need Parent UV in the emitter.

Work with goals

Create goals

You can make a particle object move towards one or more goal objects with the following procedures. Note that you can create a soft body with its original geometry as its goal (see "Create soft bodies" on page 251).

If you make a trimmed NURBS surface a goal object, all its CVs become the goal, including the CVs where the trimmed area exists. To correct this, convert the NURBS surface to polygons with Modify > Convert > NURBS to Polygons and then add the goal.

To connect a particle object to a single goal object

1. Select the particle object you want to be affected by the goal.

   To select a soft body rather than a conventional particle object, you can select the soft body's original geometry or its child particle object.
2. Shift-select the object you want to become the goal.
3. Select Particles > Goal.
4. To adjust the goal's influence, see "Edit goal attributes" on page 108.
5. Play the animation to see the particles move towards the goal.

> Note To make particles follow the object’s transform rather than its particles, CVs, vertices, or lattice points, you must select Particles > Goal > ❒ to display the Goal options window. Turn on Use Transform as Goal in the Goal Options window before you add the goal.

To connect a particle object to multiple goal objects:

1 Select the particle object or soft body you want to be affected by the goals.
2 In the workspace, shift-select one of the objects you want to become a goal.
3 Select Particles > Goal.
4 Deselect all objects and repeat the previous steps for the next object you want to become a goal.
5 To adjust the goal’s influence, see ”Edit goal attributes” on page 108.
6 Play the animation to see the particles move towards the goal.

Edit goal attributes

The following sections describe how to edit attributes to tune the effect of goals on trailing particle objects.

Setting goal weight on a per object basis

When you add a goal to an object, Maya adds a corresponding goal weight attribute to the trailing particle object. The goal weight sets how much all particles of the trailing object are attracted to the goal. You can set the goal weight before you create the goal using the Goal options window, or you can adjust the goal weight afterwards using the Attribute Editor.

You can set goal weight to a value between 0 and 1. A value of 0 means that the goal’s position has no effect on the trailing particles. A value of 1 moves the trailing particles to the goal object position immediately.

A value between 0 and 1 causes the particles to move toward the goal as if bound to it by an elastic spring. The closer to 1, the faster the trailing particles settle at their final position. The goal weight value is 0.5 by default.

Tips	To make the trailing particle object follow the goal object with less oscillation, set the trailing particle object's Conserve attribute to a value less than 1, for instance, 0.8. A value of 0 prevents oscillation. You can turn the effect of a goal object off or on without setting goal weight. With the trailing particle object attributes displayed in the Attribute Editor, turn off the Goal Active attribute associated with the goal object.

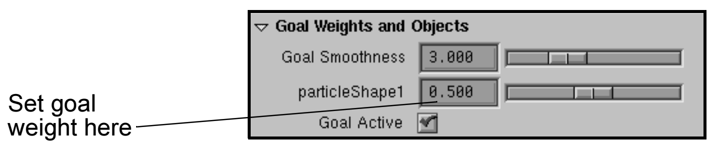
To display the Goal options window:

- Select Particles > Goal > ❐. The attributes in this window affect goals you create after you change an attribute setting.

To change the goal weight in the Attribute Editor

1. Select the particles (or soft body particles) affected by the goal.
2. Display the Attribute Editor.
3. Open the Goal Weights and Objects section.

   The goal weight is next to the name of the goal object.



If the particle object has multiple goal objects, you'll see a goal weight entry box for each object.

Tip	In the Channel Box, the goal weight for the first goal object you added is Goal Weight[0]. The second is Goal Weight[1] and so on.

Set goal weight on a per particle basis

For more control of trailing particle motion, you can tune the goal weight on a per particle basis. The per particle goal weight is controlled by the trailing object's goalPP attribute.

The per particle goalPP attribute is used for *every* goal object connected to the particles. By default, the goalPP attribute has a value of 1 for each particle.

> Tip Using the Paint Soft Body Weights Tool, you can set goal weights on a soft body on a per particle basis by painting on the soft body surface (see "Paint Soft Body Weights Tool" on page 253).

To set per particle goal weight

1. Select the trailing particle object and display the Attribute Editor.
2. Locate the goalPP attribute in the Per Particle (Array) Attributes section of the Attribute Editor. See "Set attributes on a per particle basis" on page 54 for details on how to set per particle attributes.

   The goalPP value is multiplied by the per object goal weight to create the total goal weight for each particle.

   For example, if you have a per object goal weight of 0.5 and you add a per particle goal weight of 0.5 to some of the particles, the total goal weight of those particles is 0.25. If you have a second goal object and the corresponding per object goal weight is 0.1, the total goal weight of the particles is 0.05.

> Note If you create a ramp on goalPP, the ramp is mapped over the lifetime of the particles; it's not mapped over the surface of the goal object. In addition, you cannot texture goals according to emission.

Specify NURBS or polygonal UV positions for goals

When you use the CVs of a NURBS or polygonal surface as a goal, the trailing particles move directly to the CVs. The CVs, however, are usually positioned at distant or undesired positions. You can make the particles move more directly or evenly to the surface by adding and setting the predefined goalU and goalV attributes of the trailing particle object's shape node. These per particle attributes set exact locations on a NURBS or polygonal surface where the particles are attracted.

You can also add and set the predefined goalOffset attribute to more conveniently accomplish similar results (see "Specify an offset to goal positions" on page 111).

If the trailing object has multiple goal objects, the same per particle U and V values are used for each goal object.

If the attributes don't specify a valid position on the object, the closest position on the surface or curve is used as the goal position instead. A position is invalid if it is out of the object's U and V range or it's on a part of the curve or surface that you've trimmed. To ensure the closest point is used, specify a value of -1.

See "Add dynamic attributes" on page 34 for details on adding attributes. You can set the value of these attributes with an expression, MEL script, or other attribute editing technique.

Work with polymesh goal objects

When per-particle goalU/goalV attributes apply to polymesh goal objects, you specify particular points on the goal mesh surface to which particles will be attracted. Animation of goalU/goalV values, as well as the goal object and its UV coordinates, is fully supported.

For each polymesh object, a UV Set control appears to let you decide which UV set to use when determining the goal position of a particle given its goalU/goalV values.

Since UV mappings on polymesh objects are arbitrary, it is possible that a given goalU/goalV pair may map to more than one point on the goal surface. If there are multiple potential goal points for a particle, Maya selects the point that is closest to the particle's current position in world space, and uses that as the goal position. This decision is recomputed at every frame, so if the particle moves closer to one of the other potential goal positions, the particle will be attracted to that goal position instead of the original goal position.

Similarly, it is also possible that no point on the object matches the specified goalU/goalV values for a particular particle. In that case, the mesh vertex that is closest in UV space to the specified goal point is used. This decision is updated at every frame.

Note	For polymesh goal objects, if a particle's goalU/goalV values are not present on the goal surface (that is, they are unmapped), the particle is attracted to the vertex nearest to the specified values. This may lead to the particle hopping between vertices of the mesh if the goal UV values are animated through unmapped space.

Specify an offset to goal positions

If you add the predefined goalOffset attribute to the trailing particle object's shape node, you can set an offset to the world space position of the goal object. For example, if you have particles moving towards a goal that consists of a curve's CVs, the offset directs the particles to each CV plus the goalOffset. The goalOffset attribute is a per particle vector attribute.

See "Add dynamic attributes" on page 34 for details on adding attributes. You can set the value of this attribute with an expression, MEL script, or other attribute editing technique.

Tip	To create an evenly distributed attraction to an object with symmetrically positioned CVs, use a sphrand( ) function in an expression to assign random values to goalOffset.

Animate goal behavior

The techniques for animating goal behavior follow:

- You can animate a goal's influence off and on with a key, expression, or other animation technique.
- You can set keys to animate the value of the per object goal weight.
- You can tune the effect of goal weight animation as the goal weight increases or decreases over time.

To key a goal's influence off (or on)

1. Select the trailing particle object.
2. Display the Attribute Editor and open the Goal Weights and Objects section.

   If the trailing particle object has multiple goal objects, there is one Goal Active entry for each. Each is listed under the goal weight slider named after the goal object.
3. Turn Goal Active off.
4. Key the attribute (see *Animation*).

To animate the value of per object goal weight

1. Set the value of the goal weight as described on page 108.
2. Key the attribute.

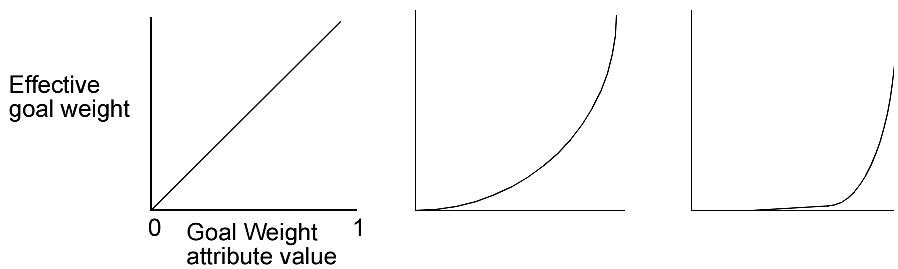
   You cannot set keys for per particle goal weights.

To tune goal weight animation

1. Select the trailing particle object.
2. Display the Attribute Editor and open the Goal Weights and Objects section.

3 Set the Goal Smoothness to a value of 1 or greater, as follows:

With a Goal Smoothness of 1, the goal weight effect increases proportionally as goal weight rises in small increments from 0 to 1. For higher values of Goal Smoothness, the goal weight effect increases slowly as values rise above 0, and quickly as the values approach 1. Objects have a default Goal Smoothness of 3.



The three curves contrast the effective goal weights resulting from Goal Smoothness values of 1, 3, and 10 (from left to right).

Create particle collisions

Make particles collide with a surface

You can make particle objects collide rather than pass through polygonal or NURBS surfaces. Either or both objects can be moving at the moment of collision. Particles cannot collide with other particles.

Note that in the context of this chapter, soft bodies are considered particle objects. To make a pair of surfaces collide, see Chapter 4, “Soft and Rigid Bodies.”

To make particles collide with geometry

You can make a particle object collide with only one geometry object at a time. Repeat the following steps if you need to make the particles collide with multiple objects.

1 Select the particles and Shift-select the geometry in the workspace.

To use a soft body rather than a conventional particle object, you can select the soft body’s original geometry or its child particle object.

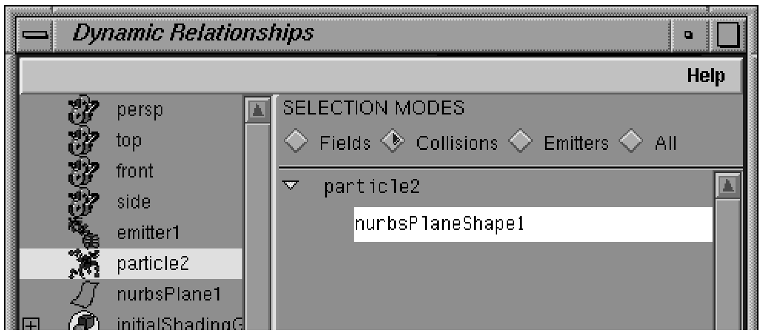
Select Particles > Make Collide.

2 Animate the particles or geometry (or both) to cause a collision.

To disable a collision effect

1. Select the particle object that collides with the geometry.
2. Select Window > Relationship Editors > Dynamic Relationships to display the Dynamic Relationships Editor.
3. In the Selection Modes section of the Dynamic Relationships Editor, turn on Collisions.

   The left column highlights the colliding particle object. The right column highlights the collided geometry object.



This geometry object collides with the particle object highlighted in the left column.

4. In the right column of the Dynamic Relationships Editor, click the highlighted object.

   The item is no longer highlighted, which means it's disconnected. For details on using the Dynamic Relationships Editor, see Chapter 8.

Edit particle collision attributes

After you've made particles collide with geometry, you can use the Attribute Editor to set Friction and Resilience attributes to adjust the bounciness. You can also set an attribute that alters the collision detection sensitivity.

You can set the bounciness on a per geometry basis or on a per particle object basis. If you set the bounciness on a per geometry basis, every particle object that collides with the geometry bounces the same. If you set the bounciness on a per particle object basis, every particle object that collides with the geometry bounces differently.

You can also set the bounciness before you make collisions occur by displaying the options window with Particles > Make Collide > ❐. The Friction and Resilience attributes in the options window set bounciness on a per geometry basis.

Tip	You can speed up scene play in a scene that has particles colliding with complex surfaces. Select the collided surface, display the Attribute Editor, select the geoConnector tab, and set Tessellation Factor to a lower number. Be aware that this lessens collision detection sensitivity.

Set bounciness on a per geometry basis

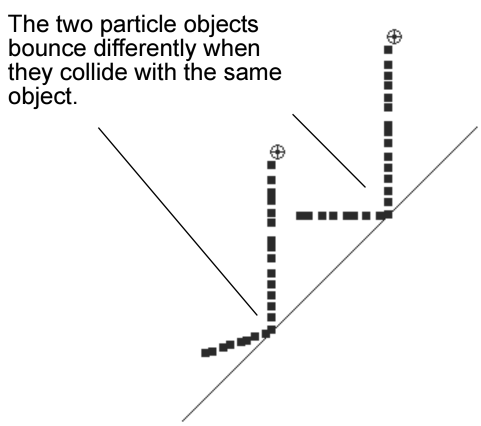
By default, all particle objects that collide with a geometry object bounce the same. The following steps show how to set the per geometry bounciness.

To set the bounciness on a per geometry basis

1. Select the geometry object involved in the collision.
2. Open the Attribute Editor and click the geoConnector tab.
3. Set the attributes.

Set bounciness on a per particle object basis

The following steps show how to make different particle objects bounce differently off the same geometry.



To set the bounciness on a per particle object basis

1. Select the desired particle object.
2. Display the Attribute Editor and open the Collision Attributes section.
3. Right-click the Resilience or Friction attribute to display a pop-up menu, then select Break Connection.

If the particle object collides with two or more geometry objects, you'll see independent Resilience and Friction attributes for each geometry object. The name of the associated geometry object is above the attributes.

4 Enter a new value for Resilience or Friction. See "Set bounciness on a per geometry basis" on page 115 for a description of the attributes.

   This doesn't alter the existing Resilience and Friction of other particle objects that collide with the geometry.

Avoid unexpected particle penetration of geometry

In repeated or numerous collisions, particles might pass through the geometry because of insufficient collision detection sensitivity. You can increase sensitivity to avoid this problem.

To adjust the collision detection sensitivity

1. Select the particle object.
2. In the Attribute Editor's Collision Attributes section, increase Trace Depth.

   The Trace Depth sets the maximum number of collisions Maya can detect for the object in each animation time step. For instance, with a setting of 2, Maya checks twice in a frame. Any more than two collisions are ignored and the subsequent particles penetrate. Increasing the setting increases processing.

Tips You can set Trace Depth to 0 to make a particle object pass through geometry. You can optionally add a traceDepthPP attribute to a particle shape node to set collision detection sensitivity on a per particle basis (see "Work with particle attributes" on page 33). When added, traceDepthPP appears in the Per Particle (Array) Attributes section of the Attribute Editor. The traceDepthPP setting overrides the Trace Depth setting.

Duplicate collision effects

You can duplicate the objects involved in a collision effect. For example, if you make particles collide with a surface, you can duplicate the particles, surface, and collision effect.

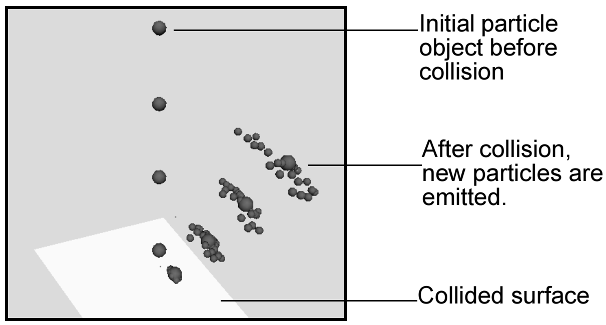
To duplicate the collision effect:

1. Select the particle object and geometry involved in the collision.

2 Select Edit > Duplicate > ❐ to display the Duplicate options window.
3 In the Duplicate options window, turn on Duplicate Input Graph and click Duplicate.
4 Move the objects to a new location as desired.

Emit, kill, or split particles upon contact

You can make particles emit new particles, die, or split into multiple particles when they collide with geometry. You can also execute a MEL script upon collision. The actions that occur after contact are called *events*.



To create a particle collision event:

1. Select the particle object you want to be affected by the event.
2. Make the particles and geometry collide.
3. Select Particles > Particle Collision Events.
4. In the Objects list of the Particle Collision Events Editor, make sure the particle object you want to be affected by the event is highlighted.
5. Set the attributes.
6. Click Create Event.
7. If applicable, set attributes of the split or emitted particles to alter their appearance and behavior.
8. Play the animation and watch the event.

To delete a particle collision event

1. Select Particles > Particle Collision Events.
2. In the Objects list of the Particle Collision Events Editor, select the particle object the event is applied to.
3. Select the event you want to edit in the Events window.
4. Click Delete Event at the bottom of the window.

To create and source a MEL procedure for use with an event

1 Create a MEL script that contains a procedure with this format:

```
global proc myEventProc (string $particleObject,
int $particleId, string $geometryObject)
{
Type MEL statements here;
}
```

After you complete the following steps, when you play the scene and a collision occurs, the event executes and passes values to the three arguments defined in the procedure:

$particleObject—name of the particle object that collides with the geometry.

$particleId—particle id number of the particle that collides.

$geometryObject—name of the geometry.

Be aware of these issues as you create the procedure:

- You do not need to use the contents of the arguments in your procedure, but they must exist in the procedure definition.
- Name the MEL script the same as the procedure so that the procedure is loaded into memory when you source the script.
- Put the MEL script in the `maya/scripts` (Linux, IRIX) or `/Applications/Maya 6.0/Application Support/scripts` (Mac OS X) directory so the script and its procedure are sourced automatically each time you start Maya.

2 In the Script Editor, select File > Source Script to source the script that contains the procedure.

If the procedure creates objects, Maya doesn't delete them when you rewind. Note also that you can source a procedure without sourcing the script that contains it. See *MEL* for more details on working with scripts and procedures.

To edit a particle collision event

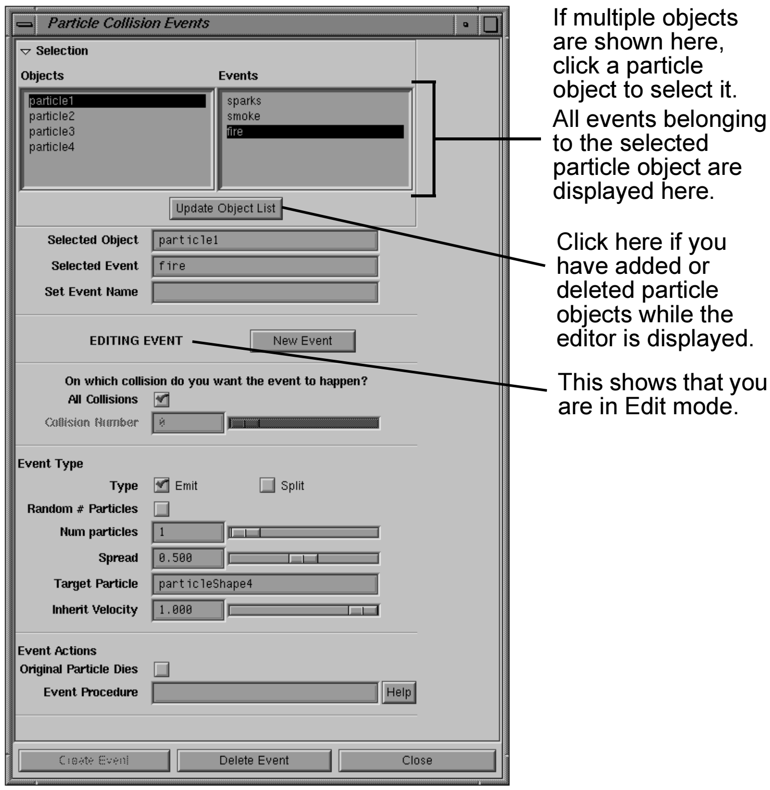
1 Select Particles > Particle Collision Events.

See the figure on the following page for details on window items that help you select events for editing.

2 In the Objects list of the Particle Collision Events Editor, select the particle object that the event is applied to.

3 Select the event you want to edit in the Events window.

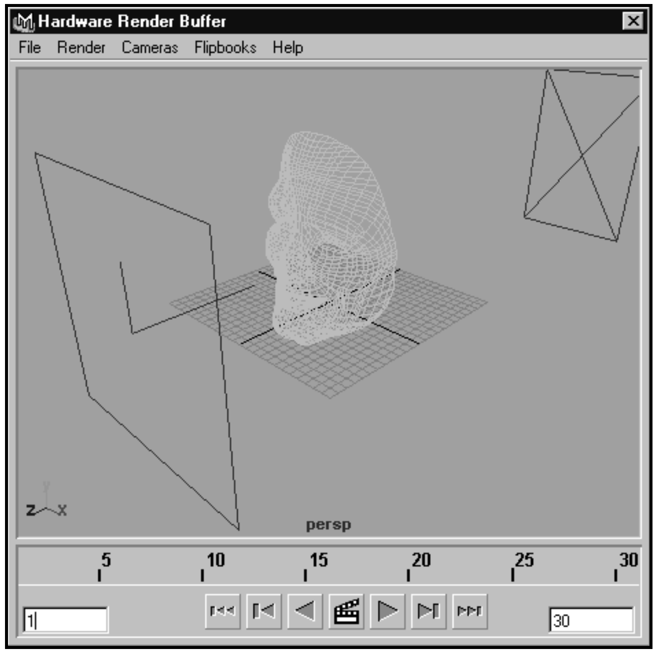
4 Edit the option settings in the window.



Render particles

Before you can preview or render hardware particles, you must set up the Hardware Render Buffer. For complete information on setting up the Hardware Render Buffer, see *Rendering* “Hardware Render Buffer.”

The following procedure describes a typical set up; you may decide to use a slightly different procedure.



To prepare to preview or render hardware particles

1 Select Window > Rendering Editors > Hardware Render Buffer.

The Hardware Render Buffer is displayed.

2 From the Cameras menu, select the camera you want to render from.

Tip	You can adjust the camera's view within the Hardware Render Buffer using the Alt (Windows, Linux, and IRIX) or Option (Mac OS X) key and the mouse buttons.

3 Select Render > Attributes.

The Hardware Render Globals Attribute Editor is displayed.

4 Set the following attributes:

Filename	The base name for all rendered image files.
Extension	The format of the extension(s) added to the base name for all rendered image files.
Start Frame	The first frame to render.

End Frame	The last frame to render.
By Frame	The increment between frames that you want to render.
Image Format	The format for saving rendered image files.

5 Do one of the following:

- To use a standard image resolution, click the Select button to the right of the Resolution attribute and select a preset image resolution.
- To use a custom image resolution, in the Resolution field type:

```
<formatName> <width> <height> <deviceAspectRatio>
```

For example, type:

```
MyFormat 400 200 2
```

6 Do one of the following:

- If you plan to composite the hardware rendered image(s), set Alpha Source to either Hardware Alpha (if your computer has a hardware alpha buffer) or Luminance.
- If you *do not* plan to composite the hardware rendered image(s), set Alpha Source to Off.

7 If you plan to composite the hardware rendered image(s), and will need depth information during compositing (that is, the distance that particles are from the camera), turn on Write ZDepth.

Preview hardware particles

After you have set up the Hardware Render Buffer, you may want to preview particles. The following procedure describes a typical method for previewing particles; you may decide to use a slightly different procedure.

To preview hardware particles

1 If the Hardware Render Buffer is not already open, select Windows > Rendering Editors > Hardware Render Buffer to open it.

2 If the Hardware Render Globals Attribute Editor is not already open, select Render > Attributes to open it.

3 Set Lighting Mode to either Default Light (a directional light), All Lights (up to eight lights in your scene), or Selected Lights (the lights in your scene that you've selected).

4. If you want to preview particles *and* surfaces, set Draw Style according to how you want surfaces to appear (Points, Wireframe, Flat Shaded, or Smooth Shaded).
5. If Draw Style is Wireframe, or if you are rendering particles that have a Particle Render Type of Streak or MultiStreak, turn on Line Smoothing.
6. Turn off Full Image Resolution.
7. Do one of the following:
    - To preview particles only, turn on Geometry Mask.
    - To preview particles *and* surfaces, turn off Geometry Mask.
8. If you want to preview particles *and* surfaces, and you want surfaces to be anti-aliased (that is, to have soft edges instead of sharp, jagged edges), turn on Multi Pass Rendering and Anti Alias Polygons.
9. If you are rendering particles that have a Particle Render Type of MultiStreak or MultiPoint, and you want the particles to be softened or blurred, turn on Multi Pass Rendering and set Render Passes (the higher the value, the more blurry particles will appear).
10. If you want particles to appear motion blurred, increase the Motion Blur value (the higher the value, the more blurred particles will appear), and set the Render Passes value to at least the Motion Blur value minus 1. (For example, if the Motion Blur value is 4, set the Render Passes value to at least 3.)

> Important! To render particles with Motion Blur, you must either turn on the particle's Cache Data attribute or select Solvers > Create Particle Disk Cache. Otherwise, the particle may behave erratically when rendered.

11. If you want icons to appear in the preview, turn on the desired Display Options.
12. If you want to have a colored background, set the Background Color.
13. Close the Hardware Render Globals Attribute Editor.
14. In the Hardware Render Buffer, select a reduced resolution from the Render > Scale Buffer sub-menu.
15. To preview a single frame, select Render > Test Render.

- To preview an animation, select Render > Render Sequence.

  The Hardware Render Buffer displays each frame as it is rendered.

Important Do not move the Hardware Render Buffer or drag another window over it during rendering. Disable any screen saver software before you start rendering. (Maya saves hardware rendered images by taking a screen shot of the Hardware Render Buffer.)

- To cancel rendering, press Esc.
- To display the view of your scene after rendering is completed, click inside the Hardware Render Buffer.
- To view a rendered animation, see "View rendered hardware particles" on page 124.

Render hardware particles at final production quality

After you have set up the Hardware Render Buffer, and perhaps previewed particles, you can render hardware particles at final production quality. The following procedure describes a typical method for rendering hardware particles; you may decide to use a slightly different procedure.

To render hardware particles

1. If the Hardware Render Buffer is not already open, select Window > Rendering Editors > Hardware Render Buffer to open it.
2. If the Hardware Render Globals Attribute Editor is not already open, select Render > Attributes to open it.
3. Set Lighting Mode to either Default Light (a directional light), All Lights (up to eight lights in your scene), or Selected Lights (the lights in your scene that you've selected).
4. If you are rendering particles that have a Particle Render Type of Streak or MultiStreak, turn on Line Smoothing.
5. Turn on Full Image Resolution.
6. Turn on Geometry Mask.
7. If you are rendering particles that have a Particle Render Type of MultiStreak or MultiPoint, and you want the particles to be softened or blurred, turn on Multi Pass Rendering and set the Render Passes value (the higher the value, the more blurred particles will appear).

8 If you want particles to appear motion blurred, increase the Motion Blur value (the higher the value, the more blurred particles will appear), and set the Render Passes value to at least the Motion Blur value minus 1. (For example, if the Motion Blur value is 4, set the Render Passes value to at least 3.)

Important! To render particles with Motion Blur, you must either turn on the particle's Cache Data attribute or select Solvers > Create Particle Disk Cache. Otherwise, the particle may behave erratically when rendered.

9 Make sure all of the Display Options are off, and that the Background Color is black.
10 Close the Hardware Render Globals Attribute Editor.
11 In the Hardware Render Buffer, select Render > Scale Buffer > 100%.
12 Select Render > Render Sequence.

The Hardware Render Buffer displays each frame as it is rendered.

Important Do not move the Hardware Render Buffer or drag another window over it during rendering. Disable any screen saver software before you start rendering. (Maya saves hardware rendered images by taking a screen shot of the Hardware Render Buffer.)

- To cancel rendering, press Esc.
- To display the view of your scene after rendering is completed, click inside the Hardware Render Buffer.
- To view a rendered animation, see "View rendered hardware particles" below.

View rendered hardware particles

After you have rendered hardware particles, you can view them either from within Maya or from a Terminal, UNIX shell or a DOS window.

To view hardware rendered particles from within Maya

1 In the Hardware Render Buffer, select the sequence of images you want to view from the Flipbooks menu.

The hardware rendered particles are displayed in an Fcheck display window. See the online documentation for Fcheck.

The play repeats continuously unless you specify otherwise with an fcheck option. You can specify Fcheck options.

2 To close the Fcheck display window, press Esc (Windows, Linux, and IRIX) or Command+q (Mac OS X).

Notes	Inadequate memory or processing speed might cause play to be choppy or slower than you anticipate for final production. If this happens, scale down the hardware render buffer size before rendering. Smaller images require less memory and give more accurate speed. You can play more than one sequence at time, up to the limitations of your workstation's memory.

To set fcheck options

1 Select Flipbooks > Options > Flipbook Flags. The Flipbook Options window displays.

2 In the Options window, enter fcheck command options.

See the online documentation for "fcheck" in the *Rendering Utilities* online documentation for more information on fcheck options.

Note	Do not set options for the frame rate (`-r`) or for the frame range and increment (`-n`). The existing Maya settings controls these options.

Tip	To improve fcheck play speed with large images, use options `-s4` and `-m4` to shrink and magnify the images. This decreases memory usage and image quality, but displays the same image size.

To view hardware rendered particles from a UNIX shell or DOS window

1 Do one of the following:

- From a UNIX shell or DOS window, type:

```
fcheck <image>
```

- Select Start > Programs > Alias > Maya 6.0 > Fcheck.

For example, type:

```
fcheck fire
```

The hardware rendered particles are displayed in an fcheck display window. See “fcheck” in the *Rendering Utilities* online documentation.

2 To close the fcheck display window, press Esc.

To view hardware rendered particles from a Mac OS X fcheck window

1 Click the Applications icon from the Finder window, open the Maya folder, then double-click the FCheck icon.

The hardware rendered particles are displayed in an fcheck display window. See “fcheck” in the *Rendering Utilities* online documentation.

2 To close the fcheck display window, press Command+q.

Apply shadow casting to particles

Maya supports shadow casting for hardware-rendered particle types Point, MultiPoint, and Sphere. The particles will also cast shadows on themselves (self-shadow). Due to the limited resolution of the depth map, particles may cast rough or pixelated shadows.

Software rendered particles

If your scene contains only software particles—Blobby Surface, Cloud, or Tube, you must software render your scene. See *Rendering,* “Rendering a frame”and “Rendering an animation.”

If you are using Cloud render type, see ”Create raytraced shadows with particles” on page 126.

To raytrace particle shadows, see ”Create raytraced shadows with particles” on page 126.

To control shader attributes on a per-particle basis, or use the rgbPP and opacityPP attributes in software rendering, see ”Particle Sampler Info node” on page 25.

> Important! To render particles in software with Motion Blur, you must either turn on the particle’s Cache Data attribute or select Solvers > Create Particle Disk Cache. Otherwise, the particle may behave erratically when rendered.

Create raytraced shadows with particles

You can render raytraced shadows with software-render type particles.

To create raytraced shadows with particles

1. Create your particles and set the Render Type Cloud, Blobby, or Tube.
2. Add a directional, point, spot, or area light to the scene.
3. Display the Attribute Editor for the light and display the Shadows Attributes.
4. Turn on Use Ray Trace Shadows (under Raytrace Shadow Attributes).
5. In the Render Globals window, display the Raytracing Quality attributes and turn on Raytracing.
6. Do one of the following:
   - If you want particles raytraced in a mirror, display the particle Attribute Editor and turn on Visible in Reflections (under Render Stats).
   - If you want particles refracted in glass, turn on Visible in Refractions (under Render Stats).

Note	Make sure Casts Shadows is turned on (under Render Stats). It is on by default.

Tip	If you get strange looking or missing shadows when rendering cloud or tube particles with raytraced shadows, it may be that the Ray Depth Limit is too low. Increase the Ray Depth Limit for the light that is using the raytraced shadows by an increment of 1 per software-rendered particle shape.

Create a Particle Sampler Info node

This is the general procedure for creating a Particle Sampler Info node. These are general steps—we've included examples of specific tasks in "Examples of using the Particle Sampler Info node" on page 128.

To create a Particle Sampler Info node

1. Create your particles and set the Render Type Cloud, Blobby, or Tube.
2. Determine which particle attribute you want to use to drive a shader attribute.
3. Add that attribute to the particle shape, if necessary.

4 Select the particles and display the Attribute Editor. Click the Attribute Editor tab for the shader, and find the shader attribute you want to drive with the particle attribute.

5 Click the (map) button next to that attribute to display the Create Render Node window.

6 In the Create Render Node window, select the Utilities tab. Under Particle Utilities, click the Particle Sampler Info button.

   This displays the Connection Editor with the Particle Sampler Info node on the From side and the shader on the To side.

   In the case of color, transparency, and incandescence, the Connection Editor is not displayed when you click the Particle Sampler Info button. Instead, Maya creates the new Particle Sampler Info node for you, and automatically makes the following default connections:

- outColor -> color
- outTransparency -> transparency
- outIncandescence -> incandescence

7 Using the Connection editor, connect Particle Sampler Info node attributes to shader node attributes however you wish.

   The connection between the particle shape and the Particle Sampler Info node is implicit— you don't actually make it. The only connection you make is between the Particle Sampler Info node and the shader.

Examples of using the Particle Sampler Info node

To help you use the Particle Sampler Info node: we've included the following examples to help you learn how to use the Particle Sampler Info Node to achieve some commonly used effects. Refer to "Set Particle Sampler Info node attributes" on page 137 for complete information on Particle Sampler Info node attributes.

- "Example: Using the particles' age to color particles" on page 129
- "Example: Using the Particle Sampler Info node with rgbPP" on page 131
- "Example: Creating fading particle" on page 133
- "Example: Creating smoke" on page 135

Use the Particle Sampler Info node to drive a texture placement node

Here is an example of the first strategy—using the Particle Sampler Info node to drive a texture placement node, which textures the shader color. In this example, we'll use the particles' age to color the particles.

Example: Using the particles' age to color particles

1 Make a cone emitter emitting upwards with a speed of 8, set the particles' Lifespan Mode to Constant with a Lifespan of 2, and set their Render Type to cloud.

To do this, you can copy the following Mel commands from the online help and paste them into the Script Editor:

```
emitter -pos 0 0 0 -type volume -r 100 -vsh cone -
alx 8;
particle;
connectDynamic -em emitter1 particle1;
setAttr particleShape1.lifespanMode 1;
setAttr particleShape1.lifespan 2.0;
setAttr particleShape1.particleRenderType 8;
```

2 Select the particle shape and display the Attribute Editor.

3 Select the particleCloud tab, and press the  button next to Life Incandescence to display the Create Render Node window.

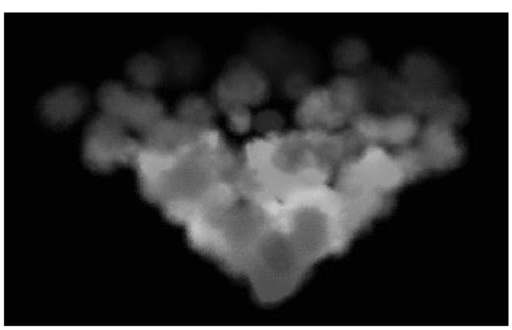
Tip	Incandescence is the same as color but is self illuminating so you don't need to add a light.

4 Click the Ramp button.

This creates a new ramp, a texture placement node, and a Particle Sampler Info node. Life Color isn't actually an attribute at all—it's just a control in the Attribute Editor that tells Maya to set up these connections to color for you.

5 Play the scene back and render.

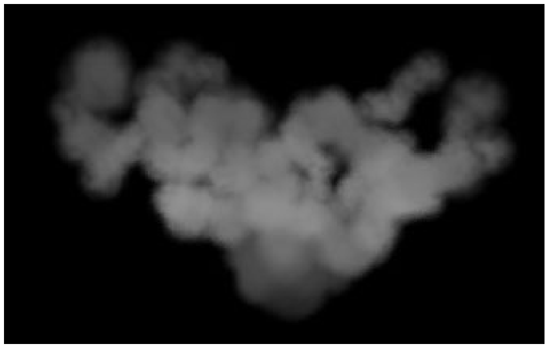
You'll see the particle color textured by the ramp as the particles age.



Now we'll change some of the controls in the Particle Sampler Info node.

6 Select the particle shape and display the Attribute Editor. Click the particleSamplerInfo1 tab.

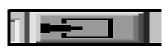
7 Turn on Inverse Out Uv and re-render.



The colors are reversed—instead of the particles changing from red to green to blue, they change from blue to green to red.

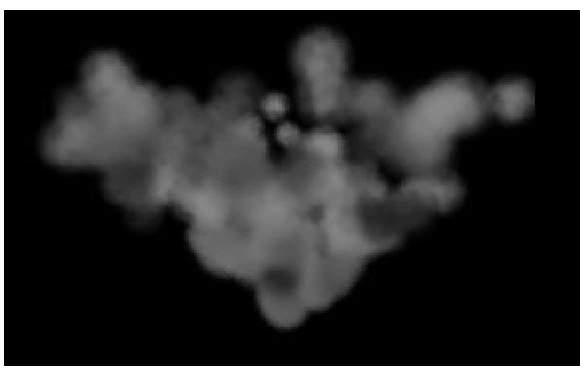
Now we'll change a control on the texture node.

8 In the particle shape Attribute Editor, click the particleCloud tab.

9 Click the Input Connection  button next to Life Incandescence.

10 Click place2DTexture1.

11 Set Noise UV to 1.0 and 1.0 and re-render.



Noise is applied to the placement of the ramp creating randomness in the red, green, and blue particles.

12 Change Noise UV back to 0.0 and 0.0.

13 Click Load Attributes and display the particle shape Attribute Editor.

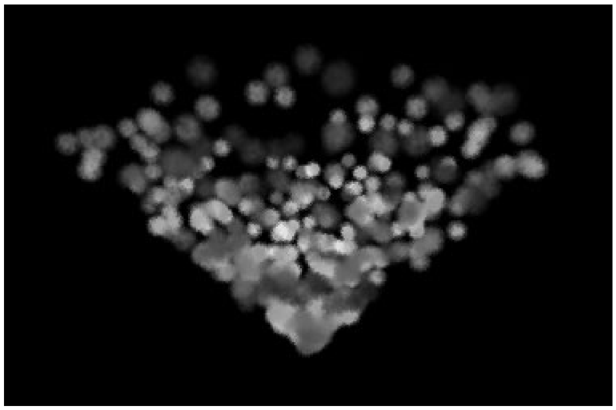
14 Change Lifespan Mode to Live forever.

15 Select the tab for the Particle Sampler Info node.

16 Change the Out Uv Type to Absolute Age and set a normalization value of 1.

17 Rewind, play about 100 frames, and render.

Notice that the particles cycle through the ramp more than once as they age.



Use the Particle Sampler Info node with per-particle attributes

Here is an example of the second general strategy—driving shader color directly with rgbPP. Use this strategy when you want to use expressions or ramps driven by something other than age.

Example: Using the Particle Sampler Info node with rgbPP

1 Make a cone emitter emitting upwards with a speed of 8, set the particles' Lifespan Mode to Constant with a Lifespan of 2, and set their Render Type to cloud.

To do this, you can copy the following Mel commands from the online help and paste them into the Script Editor:

```
emitter -pos 0 0 0 -type volume -r 100 -vsh cone -
alx 8;
particle;
connectDynamic -em emitter1 particle1;
setAttr particleShape1.lifespanMode 1;
setAttr particleShape1.lifespan 2.0;
setAttr particleShape1.particleRenderType 8;
```

2 Select the particle shape and select Fields > Gravity.

3 With the particle shape selected, display the Attribute Editor.

4 Under Add Dynamic Attributes, click Color.

The Particle Color window is displayed.

5. Click Add Per Particle Attribute and click Add Attribute.
6. In the Attribute Editor, click the particleCloud1 tab.
7. Next to the Color attribute, press the  map button to display the Create Render Node window.
8. Click the Utilities tab, and under Particle Utilities, click the Particle Sampler Info button.

   This automatically creates a new Particle Sampler Info node and connects its color output to the color attribute. (If you had already created a Particle Sampler Info node, you could use that and connect it directly, in the connection editor.)
9. Select the particle shape tab in the Attribute Editor. Under Per Particle (Array) Attributes, right click on rgbPP and select Runtime Expression (before or after dynamics calculation).
10. In the Expression Editor, enter the following expression:

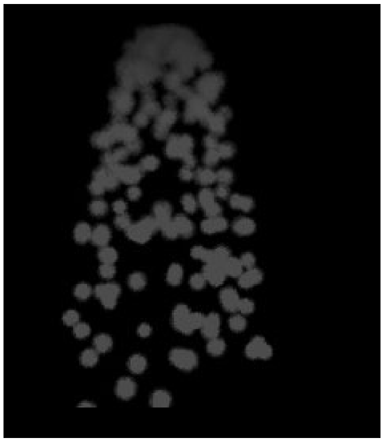
```
particleShape1.rgbPP = <<
mag(particleShape1.velocity) /

10, 0.0, 1 -mag(particleShape1.velocity)/10 >>;
```

   This tells Maya to change the particle color from blue to red as the velocity increases.

Tip	We used an expression in this example to achieve a specific effect. However, you don’t have to use expressions with the Particle Sampler Info node. You can also use a ramp, which will give you a different effect.

11. Add a light to the scene. We recommend a directional light.
12. Play about 75 frames and render.



Notice that the Particle Sampler Info node outIncandescence output automatically uses rgbPP if that attribute is present. If there is no rgbPP, it uses colorRed/colorGreen/colorBlue, if they are present.

The advantage of using rgbPP is that you can set or animate its value in any of the ways available for per-particle attributes, and see the results in software. You will also then be using the same color attribute in both hardware and software rendering.

However, rgbPP can be animated only with expressions and ramps (or the component editor). The advantage of using Life Color is that you get a texture placement node and can make use of its additional controls. In particular, you can use any texture, not just a ramp, since you get both u and v control. You can connect any compatible Particle Sampler Info node attribute you want.

Set transparency with the Particle Sampler Info node

As with color, Life Transparency is still available. It uses the outUvCoord attribute of a Particle Sampler Info node in conjunction with a texture placement node, instead of a particle transparency mapper and age mapper.

As with color, you can use per-particle attributes such as opacityPP directly. You use the outTransparency attribute of the Particle Sampler Info node.

Example: Creating fading particle

1. Make a volume emitter with the shape of a flat cylindrical disk, with Speed Away From Axis equal to 1 and other speeds zero.

   To do this, you can copy the following Mel commands from the online help and paste them into the Script Editor:

```
emitter -pos 0 0 0 -type volume -r 100 -sro 0 -spd
1 -vsh cylinder -afx 1;
scale -r 5 0.5 5 ;
particle;
connectDynamic -em emitter1 particle1;
```

2 Reduce emission rate to 50.

3 Set the particle's render type to cloud. Click Add Attributes for Current Render Type and set the Radius to 1.5.

4 In the Attribute Editor, under Add Dynamic Attributes, click Opacity. Turn on Add Per Particle Attribute and click Add Attribute.

5 Under Per Particle Attributes, right click on opacityPP and select Runtime Expression (before or after dynamics calculation).

6 In the Expression Editor, enter the following expression and click Create:

```
opacityPP = 1 / (0.01 + mag( position ));
```

(We added 0.01 in the denominator to avoid dividing by zero when particle is at the origin.)

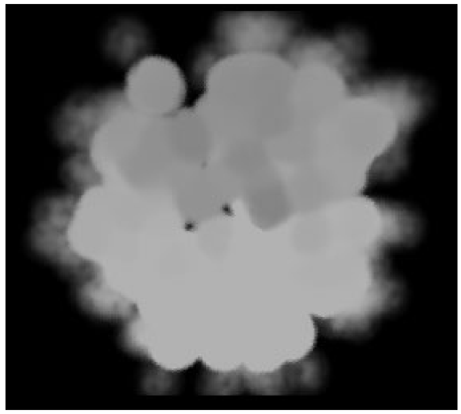
7 Select the particleCloud shader tab and click the map  button for Transparency (not Life Transparency).

8 In the Create Render Node window, click the Utilities tab and click the Particle Sampler Info button (under Particle Utilities).

This create a new Particle Sampler Info node and connects its outTransparency to the shader transparency. As with color and incandescence, the outTransparency output automatically uses opacityPP, if that attribute is present, otherwise it uses opacity.

Opacity and transparency have an inverse relationship: the more opaque something is, the less transparent it is. If you connect the outTransparency output of the Particle Sampler Info node, then the node automatically converts opacity to transparency so that you get the right result.

9 Add a light, play the scene about 50 frames, and render. A camera view from the top, looking down the y-axis, is best. The particles will appear more transparent the further they are from the origin.



Control noise on a per-particle basis

You can connect any per-particle attribute to the attribute in the shader that you want to control. You can then put an expression on the per-particle attribute to control it exactly the way you want.

Example: Creating smoke

1 Make a simple curve emitter emitting particles vertically upward.

To do this, you can copy the following Mel commands from the online help and paste them into the Script Editor:

```
curve -d 3 -p -6.861087 0 7.36723 -p -2.43128 0
2.685247 -p 2.685247 0 -2.43128
-p 6.968851 0 -6.714884 -k 0 -k 0 -k 0 -k 1 -k 1 -k
1 ;
emitter -type curve -r 100 -sro 0 -nuv 0 -cye none
-cyi 5 -spd 6 -srn 0 -nsp 0
-tsp 0 -mxd 0 -mnd 0 -dx 0 -dy 1 -dz 0 -sp 0 ;
particle;

connectDynamic -em emitter1 particle1;
```

2 Set the particles' lifespan mode to random range. Set lifespan to 2 and lifespanRandom to 2.

3 Set the particle rendering type to cloud. Click Add Attributes for Current Render Type, and set Radius to 1.5.

4 Select the particleCloud tab, and set the color to a light gray color, or any other color you like.

Now we'll use any per-particle attribute you aren't using. For example, we have no need of parentU and parentV here, so we'll use that.

Note	For simplicity, we used parentU and parentV in this example because they are easily added and weren't being used. However, you can use any dynamic attribute—even those you define yourself.

5 Select Attributes > Add Attributes. Click the Particle tab in the Add attribute window.

6 Select parentU and click Add. Select parentV and click Add. Click OK.

7 Right-click on either parentU or parentV, select Runtime Expression (before or after dynamics calculation) , and add the following expression:

```
parentU = 0.7 + 2*(age/finalLifespanPP);
parentV = age /(1.5*finalLifespanPP);
```

We want the particles to get less dense and more noisy as they age, the way smoke would.

Tip	We used an expression in this example to achieve a specific effect. However, you don't have to use expressions with the Particle Sampler Info node. You can also use a ramp, which will give you a different effect.

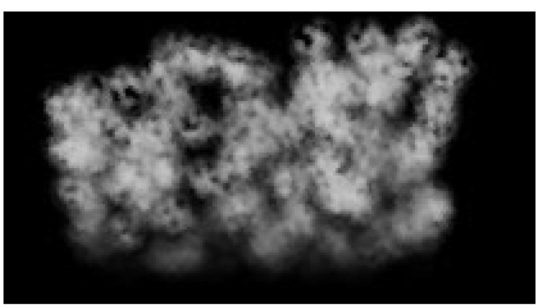
8 In the particleCloud Attribute Editor, click the map  button next to noise.

9 The Create Render Node window, click the Utilities tab and click the Particle Sampler Info button (under Particle Utilities).

The Connection Editor is displayed.

10 Connect the Particle SamplerInfo node's parentU to Noise, and connect parentV to Density.

11 Add a light. Play the scene to about frame 60 and render.



Set Particle Sampler Info node attributes

The Particle Sampler Info node attributes provide all the functionality of the old particle color mapper, transparency mapper, incandescence mapper, and age mapper.

To set the Particle Sampler Info node attributes

1. Select the particle shape and display the Attribute Editor. Click the particleSamplerInfo tab.

   The particle shape Attribute Editor includes a tab for each Particle Sampler Info node associated with that shape. The two sections in the Attribute Editor: Common Read-Only attributes and More read-Only attributes list the attributes that the node knows about by default. You can also use your own user-defined attributes.
2. You can set the attributes.

Re-use a Particle Sampler Info node

In most cases, you need to make a new Particle Sampler Info node only the first time you want to map attributes for a particular shading node. If you decide later that you want to map another attribute, you can re-use the existing Particle Sampler Info node.

By reusing the Particle Sampler Info node, you can reduce clutter. However, if you want different attribute settings for different purposes. you can use more than one Particle Sampler Info node.

To re-use a Particle Sampler Info node

1. Select Window > Rendering Editors > Hypershade.
2. Locate the Particle Sampler Info node on the left side of the Hypershade window.
3. Middle-mouse-drag it on top of the shading node.

   The Connection Editor is displayed.

Tip	If you are not sure which Particle Sampler Info node goes with which shader, put the shader in the workspace and use Graph | Up and Downstream Connections to display the associated utility nodes.

Add user-defined attributes

To use any of these attributes, you must add an attribute of the identical name and type to the particle shape.

You cannot add attributes of arbitrary names and use them with the Particle Sampler Info node. However, you can assign the value of any of your attributes to the one that the particle sampler info node recognizes, using an expression.

To add a user-defined attribute

1. Create an emitter.
2. Display the Attribute Editor and select Attributes > Add Attributes.
3. Click the Particle tab in Add Attribute window.
4. Scroll down the list and select a userScalar#PP or userVector#PP attribute. Click Add.

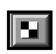
Example 1

Suppose you want to make the particles more noisy the further they are from the origin. To do this you would animate the noise parameter of the shader as a function of the magnitude of the particle's position vector.

1. Create an emitter.
2. Select the particle shape and set the Render Type to Cloud.
3. Display the Attribute Editor and select Attributes > Add Attributes.
4. Click the Particle tab in Add Attribute window.
5. Add the userScalar1PP attribute to the particle shape.
6. Assign it a value with this runtime expression (before or after dynamics calculation):

   ```
   userScalar1PP = mag(position);
   ```

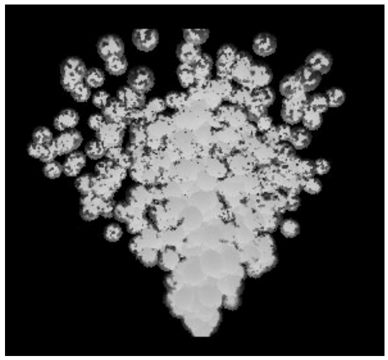
Tip	Although this example uses an expression to achieve its effect. you can use a ramp to animate userScalar1PP.

7. Select the particleCloud1 tab in the particle Attribute Editor.
8. Click the  (map) button next to the Noise attribute.
9. In the Create Render Node window, select the Utilities tab. Under Particle Utilities, click the Particle Sampler Info button.
10. In the left column of the Connection Editor, select UserScalar1 PP and in the right column, select Noise. Click Close.

    This connects the Particle Sampler Info node's User Scalar1 PP output to the shader's Noise input.

The Particle Sampler Info node will fetch the value of userScalar1PP and pass it to noise. In this example, we assigned userScalar1PP a value using an expression, but we could have used a ramp to assign it a value.

11 Add a light, play the scene. Select Window > Rendering Editors > Render View and test render.



Example 2

Suppose you already had an attribute called *myNoise*, which you had already written an expression on or connected to a ramp, and you wanted to pass this value to the Particle Sampler Info node. Then you would do this:

1. Add the userScalar1PP attribute to your particle shape.
2. Add this as the last line of your expression:

```
userScalar1PP = myNoise;
```

3. Select the particleCloud1 tab in the particle Attribute Editor.
4. Click the (map) button next to the Noise attribute.
5. In the Create Render Node window, select the Utilities tab. Under Particle Utilities, click the Particle Sampler button.
6. In the left column of the Connection Editor, select UserScalar1 PP and in the right column, select Noise. Click Close.

   Now the Particle Sampler Info node fetches userScalar1 PP, which gets the value of *myNoise*, as desired. Do this with expressions and do not try to simply connect *myNoise* to userDefined1PP.

Reference Windows and Editors

Particle Collision Event Editor

Particle Collision Event Editor

Lets you create and edit particle collision events.

Related Topics

- "To edit a particle collision event" on page 118

To show the Particle Collision Event Editor, select Particles > Particle Collision Events.

Particles > Particle Collision Events

Objects and Events

Click a particle object in the Objects list to select it. All events belonging to the selected particle object are displayed in the Events window.

Update Object List

Updates the list of objects if you have added or deleted particle objects while the editor is displayed.

Selected Object

Displays the selected object.

Selected Event

Displays the selected event.

Set Event Name

Edit or create an event name.

CREATING/EDITING EVENT

Label changes to indicate if you are creating or editing an event.

New Event

Click this button to create a new event.

All Collisions

Check this box to have the event occur on all collisions.

Collision Number

Adjust this value to have the event occur on a specific collision number.

Event Type

Emit

Check this box to set the particle event type to Emit.

Split

Check this box to set the particle event type to Split

Random # Particles

Check this box to instruct the event to use a random number of particles.

Num Particles

Adjust this value to affec the number of particles involved in the event.

Spread

Adjust this value to affect the spread of the particles involved in the event.

Target Particle

Input a particle into this field to be used as a target for the event.

Inherit Velocity

Adjust this value to affect the percentage of velocity that particles inherit during the event.

Event Actions

Original Particle Dies

Check this box if you want the original particle to die during the event.

Event Procedure

The Event Procedure is a MEL script procedure that will be executed whenever any particle in the particle object that owns the event collides with an object. The procedure must have the following format and argument list:

```
global proc myEventProc(string $particleName, int
$particleID, string $objectName)
```

where `$particleName` is the name of the particle object that owns the event, `$particleID` is the ID number of the particle in that object that has collided, and `$objectName` is the name of the object against which the particle collided.

Create Event

Creates the event

Delete Event

Deletes the event.

Close

Closes the Particle Collision Event Editor.

Sprite Wizard

Sprite Wizard

The particle SpriteWizard simplifies the process for displaying a texture image or image sequences on particles. The particle Sprite Wizard leads you throughthe steps necessary to associate image files with sprites. You can assign a single image or a sequence of images to each particle. The images can be assigned randomly, or using various criteria such as the particleID or a ramp. You can also edit the sprites once you've created them with the Sprite Wizard.

The Sprite Wizard automatically:

- assigns a Lambert material to the particle object
- adds a creation expression and runtime expression that enables all the wizard options
- connects a ramp to the spriteNumRamp attribute
- sets the particle render type to Sprites and adds the attributes for sprites
- assigns the images you specify to the sprites

For more information, see ”To use the Sprite Wizard” on page 41

Menus

Dynamics menu set

Particles >

Particles > Particle Tool

This chooses the particle tool.

Particles > Particle Tool > ❐

These are the descriptions of the items in the Particle Tool options window.

Particle Name

The name helps you identify the object in the Outliner. If you don't enter a name, the particle object receives a default name such as *particle1*.

Conserve

Influences the motion of particles whose velocity and acceleration attributes are controlled by dynamic effects. See "Conserve" on page 183.

Number of Particles

Enter the number of particles you want to create per mouse click

Maximum Radius

If you choose a number greater than 1 for Number of Particles, you can distribute particles randomly in a spherical region where you click. To choose a spherical region, set Maximum Radius to a value greater than 0.

Sketch Particles

When this option is selected, drag the mouse to sketch a continuous stream of particles.

Sketch Interval

This sets the pixel spacing between particles. A value of 0 gives you nearly a solid line of pixels. The higher the value, the more space between the pixels.

Create Particle Grid

Creates a particle grid.

Particle Spacing

Only active when creating a particle grid. Sets the spacing (in units) between particles in the grid.

Placement

Choose with cursor, to set the volume with your cursor, or with text fields, to set the grid co-ordinates manually.

Minimum Corner

The x, y, z co-ordinates of the lower left corner of the 3D particle grid.

Maximum Corner

The x, y, z co-ordinates of the upper right corner of the 3D particle grid.

Particles > Create Emitter

This creates an emitter.

Particles > Create Emitter > ❐

This sets the options when you create an emitter.

Basic Emitter Attributes

Emitter type

Select one of these choices from the pop-up menu:

Omni	Sets the emitter type to an omnidirectional point emitter. Particles emit in all directions.
Directional	Sets the emitter type to a directional point emitter. Particles emit in the direction you specify with the Direction X, Y, and Z attributes.
Surface	Emits particles from randomly distributed positions on or near a NURBS or polygonal surface.
Curve	Emits particles from randomly distributed positions on or near a curve.
Volume	Emits particles from a closed volume. You pick the shape from the Volume Shape pulldown.

Rate

Sets the average rate at which particles are emitted per second. The rate is absolute, unless you turn on “Scale Rate by Object Size”, see below.

Note that emission occurs only if your animation plays two or more consecutive frames with the emitter rate set to a positive value.

If you want to see how many particles have been emitted, select the emitted particle object and examine the Count in the General Control section of the Attribute Editor.

Scale Rate by Object Size

Only available when the current Emitter Type is Surface, Curve, or Volume.

If you turn on this attribute, the size of the object emitting the particles affects the rate of particles emitted per frame. The larger the object, the greater the rate of emission. The attribute is off by default.

For surface emission, the rate is particles per centimeter of area per second. For example, a 2 cm by 2 cm plane has an area of 4 square cm. If the emission rate is 3, the plane emits roughly 12 particles each second. If you use inches or other units, Maya converts the units to centimeters to make the rate calculation.

For curve emission, the rate is particles per cm length per time unit. For example, a 4 cm curve with a rate of 3 emits roughly 12 particles per second.

If this attribute is turned off (the default), the emission rate is absolute, instead of relative to object size, which is how previous versions of Maya behave.

Use Per-Point Rates(Rate PP)

See "Vary emission from different points of point emitters" on page 97.

Need Parent UV

(NURBS *Surface emitters only.*) If you turn this on in the Emitter Options window (before you create the emitter), Maya adds parentU and parentV attributes to the particle shape and sets the needparentUV attribute to on. You can use parent UVs to drive the value of some other parameter such as color or opacity.

If you turn this on in the Attribute Editor or Channel Box (after you create the emitter). Maya sets the needParentUV attribute to on, it does not add the attributes.

Cycle Emission

Cycle Emission lets you restart the random number sequence of the emission. You can use it to create simple cycles for games work.

If you set it to Frame, the sequence is restarted after the number of frames you specify in Cycle Interval.

If you set it to None, the random number generator is not restarted.

Cycle Interval

Defines the interval in frames for restarting the random number sequence when using Cycle Emission. Only available when Frame in selected in the Cycle Emission drop-down menu.

Distance/Direction Attributes

Min Distance

Sets the minimum distance from the emitter at which emission occurs. You can enter a value of 0 or greater.

Particles are emitted randomly and uniformly between the Min Distance and Max Distance.

Max Distance

Sets the maximum distance from the emitter at which emission occurs. You can enter a value of 0 or greater, but it must be greater than Min Distance.

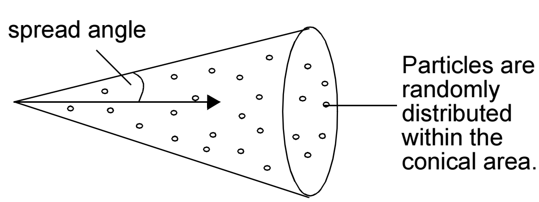
Direction X, Y, Z

Sets the emission direction relative to the emitter's position and orientation. Available only for directional, curve, and volume emitters.

> Note With point emitters, the emitter's position and orientation are set by its Translate and Rotate attributes—Scale and Shear have no effect.
>
> The original speed of particles emitted by a Directional emitter is equal to the Speed attribute of the emitter, which, by default, is 1 unit (cm) per second. The Direction X, Y and Z attributes do not affect the speed.
>
> The original speed of particles emitted by an Omni emitter is 1 unit per second in whatever direction the particles are emitted, unless it's been modified by the "Speed Random" attribute.

Spread

Sets the emission spread angle. This angle defines a conical region where the particles are emitted randomly. You can enter any value between 0 and 1. A value of 0.5 is 90 degrees, a value of 1 is 180 degrees. Available for only Directional and Curve emitters.



Basic Emission Speed Attributes

Speed

Sets a speed multiplier for the original emission speed of the emitted particles. You can enter a value of 0 or more. A value of 1 leaves the speed as is. A value of 0.5 reduces the speed by half. A value of 2 doubles the speed.

Speed doesn't affect velocity resulting from expressions, fields, or other dynamics.

Speed Random

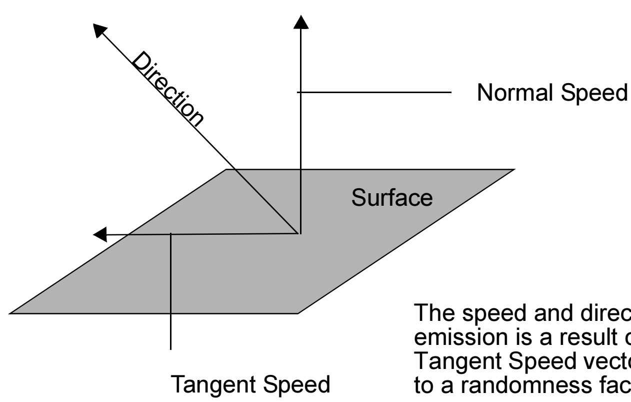
The Speed Random attribute lets you add randomness to your emission speeds without using expressions. If you set Speed Random to a positive value, the emitter generates random speeds for each particle. The Speed value is the mean speed; Speed Random defines the range of the speed variation.

Each particle's speed is a random value between:

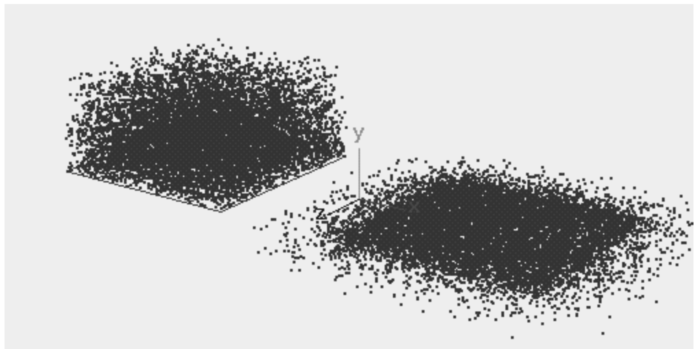
```
Speed - Speed Random/2
Speed + Speed Random/2
```

Tangent Speed

Sets the magnitude of the tangent component of emission speed for surface and curve emission (see the following figures). The default value is 0. You can enter a value of 0 or greater. *Valid for surface and curve emitters only.*



The speed and direction of surface emission is a result of Normal and Tangent Speed vectors (in addition to a randomness factor).



Normal Speed 2
Tangent Speed 0

Normal Speed 0
Tangent Speed 2

Normal Speed

Sets the magnitude of the normal component of emission speed for surface and curve emission (see the following figures). The default value is 1. You can enter a value of 0 or greater. *Valid for surface and curve emitters only.*

Volume Emitter Attributes

In the Attribute Editor and the Create Emitter option window, attributes not applicable to a volume emitter are dimmed. However, the Channel Box does not support dimming. We recommend you use the Attribute Editor instead of the Channel Box to edit volume field attributes until you become familiar with the attributes that apply to each volume shape.

Volume Shape

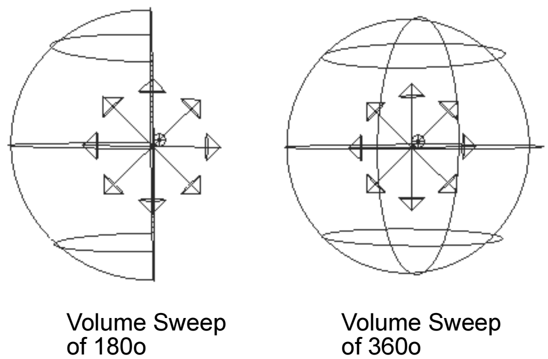
Specifies the shape of the volume that the particles are emitted into. Choose Cube, Sphere, Cylinder, Cone, or Torus.

Volume Offset

Offsets the emitting volume from the location of the emitter. If you rotate the emitter, you also rotate the offset direction because it operates in local space.

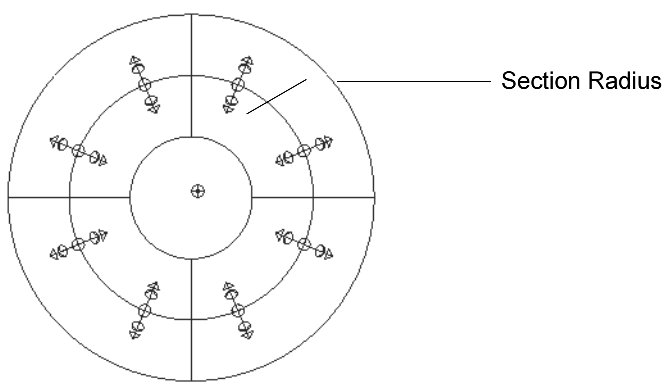
Volume Sweep

Defines the extent of rotation for all volumes except cubes. This can be a value between 0 and 360 degrees.



Section Radius

(*Torus only*) Defines the thickness of the solid portion of the torus, relative to the radius of the torus's central ring. The radius of the central ring is determined by the emitter's scale. If you scale the emitter, the Section Radius will maintain its proportion relative to the central ring.



Die on Emission Volume Exit

If you turn this attribute on, the emitted particles die when they exit the volume. Although this is a particle shape attribute, you can initially set it using the Emitter options window.

If you want to edit this attribute after you create the emitter, display the particle shape attributes in the Attribute Editor Volume Exit (under Emission Attributes).

Volume Speed Attributes

The Volume Speed Attributes apply only to the initial velocity of the particles. To affect particles as they move through the volume, use the Volume Axis field (see "Volume Axis field" on page 223).

Away From Center

Specifies the speed at which particles move away from the center point of cube or sphere volumes.

Away From Axis

Specifies the speed at which particles move away from the central axis of cylinder, cone, or torus volumes.

Along Axis

Specifies the speed at which particles move along the central axis of all volumes. The central axis is defined as positive-Y for cube and sphere volumes.

Around Axis

Specifies the speed at which particles move around the central axis of all volumes.

Random Direction

Adds irregularity to the direction and initial speed of the particle's Volume Speed attributes, somewhat like Spread does for other emitter types.

Directional Speed

Adds speed in the direction specified by the Direction XYZ attributes of all volume emitters.

Scale Speed by Size

If you turn this attribute on, when you increase the size of the volume, the speed of the particles increases.

Display Speed

Turns on the display of arrows indicating speed.

Texture Emission Attributes

See "Use a texture to color emission or scale the rate" on page 100.

> Tips — You can cause emitted particles to inherit the velocity of a moving emitter. Select the emitted particles and display the Attribute Editor. Set Inherit Factor greater than 0 to have the emitted particles inherit a decimal fraction of the emitter's velocity. For example, 0.5 makes the emitted particles inherit half the emitter's velocity.

Particles > Emit from Object

This makes an object into an emitter.

Particles > Emit from Object > ❒

This sets the options when you make an object into an emitter. These options are the same as the Create Emitter options. For more information, ee "Particles > Particle Tool" on page 142.

Particles > Use Selected Emitter

This connects a particle object to a selected emitter. For more information, see "To connect a particle object to an emitter" on page 97.

Particles > Per-Point Emission Rates

This creates an attribute named emitterRatePP for the particle object that emits. This attribute lets youvary emission rates on a per particle basis. For more information, see "To vary emission from particles" on page 98

Particles > Make Collide

Makes particles collide with geometry.

Particles > Make Collide > ❒

Here are the options you can set in the Collisions Options dialog box.

Resilience

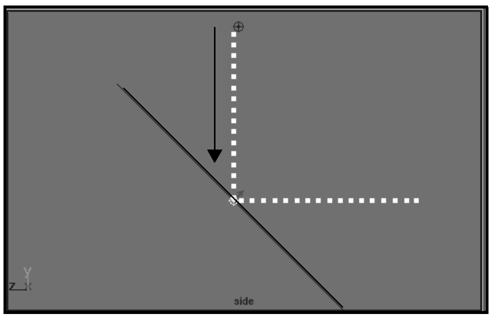
Sets how much rebound occurs. A value of 0 makes particles collide with no bounce. A value of 1 causes particles to rebound fully. A value between 0 and -1 makes the particles pass through the surface with refraction out the back side. Values greater than 1 or less than -1 add speed to the particles.

Friction

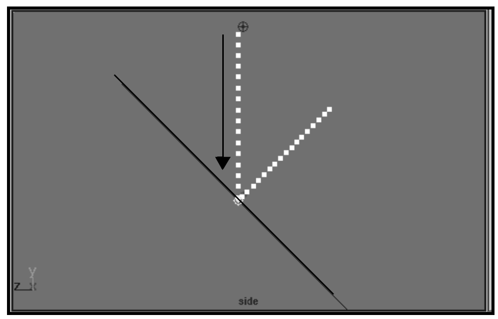
Sets how much the colliding particle's velocity parallel to the surface decreases or increases as it bounces off the collision surface.

A value of 0 means that particles are unaffected by friction. A value of 1 makes particles reflect straight off along the normal of the surface. If Resilience is 0 while Friction is 1, the particles don't bounce.

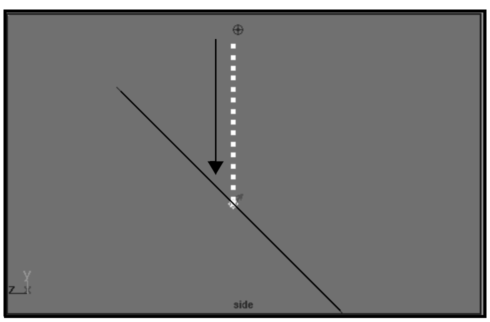
Only values between 0 and 1 correspond to natural friction. Values outside this range exaggerate the response.



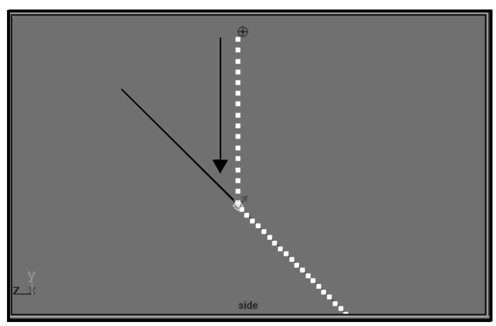
Resilience 1
Friction 0



Resilience 1
Friction 1



Resilience 0
Friction 1



Resilience 0
Friction 0

Particles > Particle Collision Events

Opens the Particle Collision Events Editor. See ”Particle Collision Event Editor” on page 140.

Particles > Goal

Sets a goal for a particle.

Particles > Goal > ❐

Here are the options you can set in the Goal Options dialog box.

Goal Weight

The goal weight sets how much all particles of the trailing object are attracted to the goal. You can set the goal weight before you create the goal using the Goal options window.

You can set goal weight to a value between 0 and 1. A value of 0 means that the goal's position has no effect on the trailing particles. A value of 1 moves the trailing particles to the goal object position immediately.

A value between 0 and 1 causes the particles to move toward the goal as if bound to it by an elastic spring. The closer to 1, the faster the trailing particles settle at their final position. The goal weight value is 0.5 by default.

Tips To make the trailing particle object follow the goal object with less oscillation, set the trailing particle object's Conserve attribute to a value less than 1, for instance, 0.8. A value of 0 prevents oscillation.

You can turn the effect of a goal object off or on without setting goal weight. With the trailing particle object attributes displayed in the Attribute Editor, turn off the Goal Active attribute associated with the goal object.

Use Transform as Goal

Makes particles follow the object's transform rather than its particles, CVs, vertices, or lattice points.

Particles > Instancer (Replacement)

Particles > Instancer (Replacement) > ❐

The options that follow are available when you select Particles > Instancer (Replacement) > ❐. If you want to modify the options after you use the particle instancer, select the instanced particle object and display the Attribute Editor. Most of the options are in the Instancer Attributes section of the particle shape node tab. A few options are in the instancer node, available by clicking the instancer node tab in the Attribute Editor after you select the particle object. You can also select the instancer node in the Outliner.

> Note You can apply the particle instancer two or more times to the same particle object. When you select the particle object and display the Instancer Attributes section of the Attribute Editor, you'll see an Instancer Nodes menu that lets you choose the instancer node associated with the settings you want to modify.

Particle Instancer Name

Optional name of the instancer node. A default name is created if you leave this entry empty.

Rotation Units

If you set the Rotation settings for you particles, this option specifies whether the value is interpreted as degrees or radians. See ”particleShape” on page 156.

Rotation Order

If you set the Rotation settings for your particles, this option sets the precedence order of rotation, for instance, XYZ, XZY, or ZXY. See ”particleShape” on page 156.

Level of Detail

Sets whether the source geometry appears at the particle locations or whether a bounding box or boxes appear instead. Bounding boxes speed up scene play.

Geometry	The source geometry appears at the particle locations.
BoundingBox	Displays a single box for all the objects in the instanced hierarchy.
BoundingBoxes	Displays individual boxes for each object in the instanced hierarchy.

Cycle

None	Instance a single object.
Sequential	Cycle through the objects in the Instanced Objects list.

If you select None and the Instanced Objects list has more than one object listed, only the first object listed will be used as the instanced object.

Note	If you select Sequential, Maya cycles through the objects in the Instanced Objects list in the order shown. You can select a different starting object for each particle with the CycleStartObject (see ”particleShape” on page 156), but the order of the cycling remains the same. For instance, one particle might cycle through objects 0-1-2-3, while another cycles through 2-3-0-1. No particle can cycle through 3-2-1-0. To use an order different than the Instanced Objects list, select None, create a custom attribute, select the custom attribute as input to the Object Index option (see ”particleShape” on page 156), then write a creation and runtime expression to set the custom attribute appropriately as the scene plays.

Cycle Step Unit

If you’re using an object sequence, select whether frames or seconds are used for the Cycle Step Size value.

Cycle Step Size

If you’re using an object sequence, enter the particle age interval at which the next object in the sequence appears. For example, a Cycle Step Size of 2 (seconds) displays the next object in the sequence at the frame where a particle’s age exceeds 2, 4, 6, and so on. See the Age attribute in ”particleShape” on page 156 for details on how to alter the age of particles.

If you use an object sequence, you’ll typically use unanimated objects displayed in sequence. You can use *animated* objects effectively if you set the Cycle Step large enough to portray their animation.

Particles > Sprite Wizard

For more information, see ”Sprite Wizard” on page 142 and ”Use the Sprite Wizard” on page 40.

Particles > Connect To Time

Reconnects the Current Time to the scene time for a particle shape node.

Nodes

Particle nodes

particleShape

All but one of the following options display a menu that lets you choose a custom or static attribute as input. It is most common to use an expression to set the value of a custom attribute being used as input to the options. The options themselves are not attributes.

Not all static attributes in the menus have a practical application. For instance, using worldVelocity as the input to the object's Rotation might not have a useful effect. The validity of each input attribute depends on the context.

Except for the first three of the following options, you can set the input attributes with per particle attribute values. You therefore can set values differently for each particle.

General Control Attributes

Is Dynamic

A value of true toggles dynamics on for the object. A value of false toggles dynamics off for the object.

Dynamics Weight

A value of 0 causes fields, collisions, springs, and goals connected to the particle object to have no effect. A value of 1 provides the full effect. A value less than 1 sets a proportional effect. For example, 0.6 scales the effect to 60% of full strength.

Expressions are unaffected by Dynamics Weight.

Conserve

The Conserve value controls how much of a particle object's velocity is retained from frame to frame. Specifically, Conserve scales a particle's velocity attribute at the beginning of each frame's execution. After scaling the velocity, Maya applies any applicable dynamics to the particles to create the final positioning at the end of the frame.

Conserve doesn't affect motion created by keyframes. Keyframes affect only a particle object's worldVelocity attribute, not its local velocity attribute.

If you set Conserve to 0, none of the velocity attribute value is retained. The velocity is reset to 0 before each frame. At the end of each frame, the velocity is entirely the result of dynamics applied during that frame.

If you set Conserve to 1, the entire velocity attribute value is retained. This is the real-world physical response.

If you set Conserve to a value between 0 and 1, a percentage of the velocity attribute value is retained. For example, if you set Conserve to 0.75, each frame Maya first reduces the velocity attribute 25%, then it calculates any dynamic or expression effects on the object.

For example, suppose you create a particle falling with the acceleration of gravity, 9.8 units per second per second. The following table compares how Conserve values of 1 (default), 0.5, and 0 affect the velocity attribute after several frames execute.

Frame	Velocity with Conserve = 1	Velocity with Conserve = 0.5	Velocity with Conserve = 0
2	<<0,0,0>>	<<0,0,0>>	<<0,0,0>>
3	<<0,-0.41,0>>	<<0,-0.41,0>>	<<0,-0.41,0>>
4	<<0,-0.82,0>>	<<0,-0.61,0>>	<<0,-0.41,0>>
5	<<0,-1.23,0>>	<<0,-0.71,0>>	<<0,-0.41,0>>
6	<<0,-1.63,0>>	<<0,-0.77,0>>	<<0,-0.41,0>>

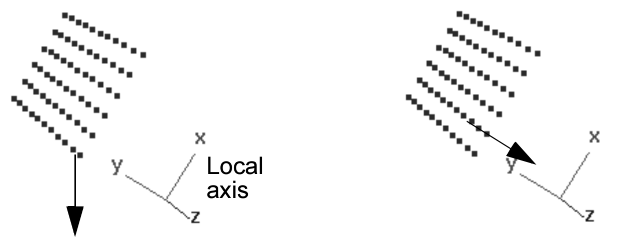
With Conserve set to 1, velocity increases each frame at the exact acceleration rate of gravity.

With Conserve set to 0, velocity stays a constant value—the particles do not accelerate. At the beginning of each frame, velocity is reset to 0. The gravity field's acceleration is then added to the velocity of 0, which results in the same number <<0,-0.41,0>> being used at the end of each frame.

With Conserve set to 0.5, velocity increases each frame at a much slower rate than gravity. At the beginning of each frame, velocity is scaled to 50% of the value it had at the end of the prior frame. The acceleration of gravity is then added to this scaled value to create the slowly increasing velocity used at the end of the frame.

Forces In World

If you prefer that a field affect a particle object in its local space, select the particles, display the Attribute Editor, and turn off Forces In World.



When Forces In World is on, gravity makes the particles fall down in world space regardless of the local axis orientation.

When Forces In World is off, gravity makes the particles move down the local Y axis.

Note that the orientation of the local axes of a particle object matches the orientation of the world space axes unless you rotate the object.

> Tip — If you have not keyed, parented, or otherwise controlled the transform attributes of a particle object, you can turn off Forces In World to speed up dynamic calculations for the object. When Forces In World is on, Maya does extra computations to convert world space to local space coordinates.

See "Apply forces in an object's local space" on page 74.

Cache Data

You can alternatively turn memory caching on or off in the Attribute Editor by turning off the Cache Data attribute. The location of the attribute in the Attribute Editor depends on the type of object. If you turn caching on for a rigid body, Maya also turns caching on for all rigid bodies in that rigid body's solver.

If you cache data in memory for emitted particles and later change the rate or another attribute of the emitter or emitted particles, you must disable the cache to see the effect of the attribute change.

Count

Contains the total number of particles in the object. This is a read-only attribute.

Total Event Count

Contains the total number of collisions which have occurred for this object. All collisions are counted regardless of whether any collision events were actually executed. In other words, totalEventCount is the sum of the per-particle “event” attribute.

Emission Attributes

Max Count

Contains the maximum count of particles this shape will allow. If some particles die off, new particles will again be accepted up to the max count, and so on.

Level Of Detail

This attribute is currently only used to scale the amount of emission to be used for quick motion tests without having to change emitter values. This attribute affects only emitted particles.

Inherit Factor

Contains the fraction of emitter velocity that particles emitted into this object inherit.

Emission In World

This boolean attribute tells the particle object to assume that particles created from emission are in world space, and to transform them into object space before adding them to the particle array. This makes the particles respond as if they were in the same space as the emitter when they are in some non-identity hierarchy.

Die on Emission Volume Exit

When this boolean attribute is set to true, if the particles were emitted from a volume, they die when they exit that volume. By derfault, this attribute is set to false.

Lifespan Attributes

Lifespan Mode

Live forever	All particles live forever, unless killed by collision events or emission volume exit.
Constant	This setting allows you to input a constant lifespan for the particles. The particles will die at the specified time.
Random range	This attribute must be set to enable Lifespan Random (see below).

lifespanPP only	Pre-Maya 3.0 expressions that refer to lifespanPP work correctly as long as you select *lifespanPP only* as the lifespan mode.

Lifespan

This setting allows you to input a lifespan for the particles.

Lifespan Random

This attribute is used only if lifespanMode is set to "Random Range".

The attribute identifies a range of random variation for the lifespan of each particle. If set to a non-zero value, each particle's lifespan varies randomly up to plus or minus lifespanRandom/2, with the "lifespan" attribute as the mean (the average lifespan). For example, lifespan 5 and lifespanRandom 2 will make the lifespans vary between 4 and 6.

In Constant or Random Range Mode, the finalLifespanPP attribute stores the values generated from lifespan and lifespanRandom.

Note Changes in the values of lifespan and lifespanRandom affect only new particles, not particles that already exist. For example, if you key the value of lifespan to be 2 up until frame 50 and 5 thereafter, then particles generated from frame 1 to 50 will have finalLifespanPP 2 and particles generated after frame 50 will have finalLifespanPP 5. The finalLifespanPP values of particles born prior to frame 50 will not change.

General Seed

This attribute represents the seed for random number generation. It is independent of all other random number streams.

Time Attributes

Start Frame

This attribute represents the frame after which dynamics will be solved. No dynamics will play back for this object prior to startFrame.

Current Time

This attribute represents the current time in the timeline.

Collision Attributes

Trace Depth

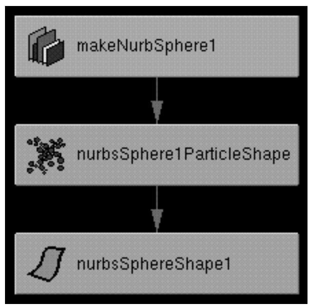
This attribute represents the maximum number of consecutive collisions that are detected within a frame for each particle. Particles may collide fewer times, of course.

Soft Body Attributes

For more information, see "Soft bodies" on page 245.

Input Geo Space

This attribute lets you choose which coordinate space Maya uses to position point data provided by the input geometry to the particle shape. The original geometry is the geometry being converted to a soft body. The *input geometry* refers to the node in Maya that creates the original geometry. An example of input geometry for a NURBS sphere is the Make Sphere node:



Input geometry
Soft body particles
Target geometry (soft body)

Geometry Local — Provides the point positioning data from the input geometry's local space. No world space transformation is applied to the positions.

World — Provides the point positioning data from the input geometry's world space. The world space transformation is applied to the positions.

Particle Local — Provides the point positioning data from the particle object's local space. The point positioning data from the geometry are transformed into world space as above, then into the particle object's local space using the inverse of the particle object's world space transformation. This puts the points in the same space as the particle object's position attribute.

Target Geo Space

This attribute lets you choose the coordinate space Maya uses to position point data provided by the particle shape to the target geometry.

Geometry Local — Takes positions from the particle object's position attribute and transforms them into the target geometry's local space. It uses both the particle object's world space transformation and the target geometry's inverse world space transformation

	before setting them into the target geometry. Wherever the particle object or target geometry is transformed in the scene, the points in the target geometry will have the same world space positions as the particles.
World	Takes positions from the particle object's position attribute and transforms them into world space. It uses the inverse of the particle object's world space transformation before setting them into the target geometry.
Particle Local	Sets position attribute values directly in the target geometry without transforming them in any coordinate space.

Enforce Count From History

If this attribute is turned on and you change the original geometry's construction history in a way that alters the number of CVs, vertices, or lattice points, Maya updates the corresponding number of particles of the soft body.

For example, if you add spans or sections to a NURBS surface soft body, Maya adds CVs to the surface and corresponding particles to the soft body. The updating of particles ensures the soft body deforms correctly according to the new points added or removed.

In some instances, you won't want this updating to occur. For instance, if you connect a soft body's particles to an emitter, you can emit the soft body into view.

Goal Weights and Objects

Goal Smoothness

This value is used to control the "smoothness" of the change in the goal forces as the weight changes from 0.0 to 1.0. This is purely an aesthetic effect, with no scientific basis. The higher the number, the smoother the change.

Instancer (Geometry Replacement)

Instancer Nodes

Selects which instancer is used for the instanced objects. *Available in Attribute Editor only.*

Allow All Data Types

When on, this expands the list of attributes you can choose as input to the options in the following pages. The expanded list includes attributes whose data type differs from the option's data type.

If the data type of an input attribute differs from the receiving option, Maya converts the data type to the receiving option's data type. (See *Expressions* for details on data types.) For example, if you select an integer attribute as input to a vector array option, Maya uses the integer value in each of the three vector components for each element of the array.

When Allow All Data Types is off, only attributes whose data type is the same as the receiving option's are included as possible selections.

Particle ObjectTo Instance

The particle object to which the geometry is applied. This option is available in the Particle Instancer Options window only.

General Options

Position

Position of the instanced objects. The default setting is worldPosition. Maya interprets the values of the selected attribute in the local space of the Instancer node, not in world space. So if you move the Instancer node, you move the instances also.

If you are an API developer, note that you can write an API to send point data to the Instancer.

Scale

Scale of the instanced objects. The default setting is None, which use the value of 1, 1, 1.

Shear

Shear of the instanced objects. The default value setting is None, which use the value of 0, 0, 0.

Visibility

Sets whether display of each instanced object is on or off. The default setting is None, which turns on the display of instances of all particles.

ObjectIndex

If you set the Cycle option of the Instancer to None, this option sets which object from the Instanced Objects list is instanced for each particle. If you set Cycle to Sequence, ObjectIndex is ignored.

There are a few common techniques for using this option. In each case you select a custom attribute as input to ObjectIndex, then write expressions to control the attribute.

One technique is to use a creation expression to assign different values for each particle to the custom attribute. Each value selects a different object in the Instanced Objects list. The value 0 selects the first object in the list, 1 selects the second, 2 selects the third, and so on.

Another expression-writing technique is to assign specific numbers from the Instanced Objects list or use a random number function such as rand( ) to assign each particle a different random object from the list.

You can alternatively use a creation expression and runtime expression to increment a custom attribute through an unusual object sequence, for instance, 2-4-6-8.

The default setting is None, which sets the value to 0, the first object on the list.

Rotation Options

RotationType

For the instanced objects, you can choose one of three methods to set their orientation: Rotation, AimDirection, and AimPosition. Though typically you'll select the same method for all instanced objects, you can select a different method for each object by using a per particle expression to set the RotationType attribute.

To set the RotationType attribute, use a value of 0 for Rotation, 1 for AimDirection, or 2 for AimPosition.

If you don't provide a number, Maya uses whichever of the Rotation, AimDirection, and AimPosition options that has an attribute input selected. For instance, if you select an attribute input for AimPosition, Maya uses AimPosition as the default.

If you select an attribute input for two or three of the RotationTypes, Maya uses the first from this list: Rotation, AimDirection, and AimPosition. For instance, if you select an input to both AimPosition and AimDirection, Maya uses AimDirection. If you select an input to Rotation, AimPosition, and AimDirection, Maya uses Rotation.

If you select no attribute inputs to Rotation, AimPosition, and AimDirection, Maya uses Rotation.

Rotation

Sets the orientation of the instanced objects relative to their initial orientation. See RotationType for details. Also see Rotation Units and Rotation Order in the prior section.

AimDirection

Sets the orientation of the instanced objects by specifying the direction along which each instanced object points relative to the original position of its local origin. The default setting is None, which uses the value of 1, 0, 0. Also see RotationType.

Note that you can make an object point in the direction the particles are moving by selecting velocity. See "Aim instanced geometry" on page 70 for more information.

AimPosition

Sets the orientation of the instanced objects by specifying the location where each instanced object points relative to the original position of its local origin. The default setting is None, which uses the value of 0, 0, 0. See RotationType for important details.

AimAxis

Specifies the object axis that points directly at the AimDirection or AimPosition.

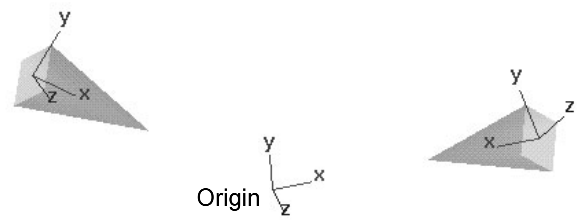
AimAxis is a vector attribute that works only with AimDirection and AimPosition rotation types. The default setting is None, which uses the value of 1, 0, 0.

AimUpAxis

Specifies the object axis that points up (as much as possible) relative to how the AimAxis points at the AimDirection or AimPosition.

In this context, *up* is the direction the world up-axis points (see AimWorldUp). AimUpAxis is a vector attribute that works only with AimDirection and AimPosition rotation types. The default setting is None, which uses the value of 0, 1, 0.

For example, suppose the AimPosition is set to the origin of the workspace, and the world space Y-axis (0, 1, 0) is the AimWorldUp value. An instanced object's X-axis (1, 0, 0) is the AimAxis value, and the object's Y-axis (0, 1, 0) is the AimUpAxis value. As the instanced object moves from left to right in the workspace, the object has the following orientation:



The object's X-axis points directly at the origin wherever the object moves in the workspace. The object is rolled around its X-axis as necessary to keep its Y-axis aiming up as much as possible. Up is defined by the AimWorldUp setting, which in this case is set to the world's Y-axis.

AimWorldUp

Sets, in world coordinates, the axis that indicates the up direction used by the AimUpAxis. AimWorldUp is a vector attribute that works only with AimDirection and AimPosition rotation types. The default setting is None, which uses the value of 0, 1, 0. This attribute is unaffected by the setting for the World Coordinate System's Up Axis found in the Window > Settings/Preferences > Preferences > Settings window.

Cycle Options

CycleStartObject

If you set the Cycle option of the Instancer to Sequence, CycleStartObject specifies the cycle's starting object from the Instanced Objects list. For example, suppose the list has four objects. If the attribute that provides input to CycleStartObject is set to 3 for each particle, each particle cycles through objects 3-0-1-2 repeatedly. The default value is 0.

Age

If you set the Cycle option of the Instancer to Sequence, Age works with the Instancer's Cycle Step setting to set how often Maya changes from one object to another.

For example, you can create an attribute named myAge, select it as input to the Age option, then write a runtime expression to control myAge as follows:

```
if (particleId == 0)
myAge = age;
if (particleId == 1)
```

```
myAge = age * 2;
if (particleId == 2)
myAge = age * 4;
```

This causes the particle with particleId 1 to cycle through the objects twice as quickly as particleId 0. The particle with particleId 2 cycles four times as fast as particleId 0.

The default value is the particle age attribute setting.

Emission Random Stream Seeds

For details on working with this option, see ”Work with emission randomness” on page 105.

Render Attributes

Depth Sort

This boolean attribute toggles depth sorting of particles for rendering on or off. By default, it is set to false (off).

Particle Render Type

This attribute specifies the hardware rendering method for the particles.

Render Stats

Render Stats are accessible from any object’s Attribute Editor.

For more information, search the online help for “Render Stats”.

Per Particle (Array) Attributes

For more information, search the online help for Understand per particle and per object attributes.

Details on particle expressions can be found in the MEL and Expressions book.

Add Dynamic Attributes

For more information, search the online help for Add dynamic attributes

Details on particle expressions can be found in the MEL and Expressions book and in the Tutorials.

Clip Effects Attributes

These attributes are present only when a clip effect (such as Fireworks) is being created or modified.

For more information, search the online help for "clip effect".

Sprite Attributes

For more information, see "Sprite attributes" on page 188.

Extra Attributes

For information on the ghosting attributes, see the Ghost Selected options window section of the *Animation* guide.

Emitter

Many of the emitter node attributes are the same as the options available in the Emitter Options window. See "Particles > Create Emitter" on page 144.

Basic Emitter Attributes

Use Per-Point Rates (ratePP)

This attribute determines if the node uses RatePP or Rate. By default, this node uses Rate.

Need Parent UV (NURBS/Poly surfaces only)

Only available when the current Emitter Type is Surface.

If you add a surface emitter to a NURBS or polygonal surface, you can use a MEL statement or expression to learn the UV coordinates where each particle was emitted. You must add parentU and parentV attributes to the emitted particle shape, then turn on Need Parent UV in the emitter.

Texture Emission Attributes (Nurbs Surface only)

Particle Color

Inherit Color	Particles emit from the texture with the same color as the texture.
Inherit Opacity	Particles emit from the texture with the same opacity as the texture.
Use Luminance	Particles emit from the texture with the same luminance as the texture.
Invert Opacity	Particles emit from the texture with the opposite opacity as the texture.

Texture Rate

Enable Texture Rate	Allows you to connect textures to emitters to set how many particles are emitted.
Emit From Dark	When Emit From Dark is off, fully luminous parts of the texture emit at 100% of the specified emitter rate. Parts of the texture that lack luminance don't emit at all. Other parts emit between 0 and 100% in proportion to the luminance. In simple terms, bright colors emit more than dark colors. When Emit From Dark is turned on, the opposite luminance scheme applies. Parts of the texture that lack luminance emit at 100% of the specified rate, while fully luminous parts don't emit at all.

geoConnector

Many of the geoConnector attributes are the same os the options in the Collision Options window. See ”Particles > Particle Tool” on page 142.

Geo Connector Attributes

Tessellation Factor

This attribute represents the approximate number of polygons in the tesselated surface.

List of particle attributes

The following table gives a summary of the particle shape node attributes you can set or examine in expressions or MEL. The attributes affect the particle object within which they exist. For more details on the attributes, use this book’s Index. Note these issues:

- The table shows attribute spellings required in expressions and MEL. User interface spellings typically use capital letters and spaces between words.
- Attributes with a data type of vector array or float array are per particle attributes. All other attributes are per object attributes. For details on data types, see *Expressions* and the online MEL documentation.
- Attributes marked by an asterisk (*) are dynamic attributes that exist only if you or Maya add them to the object. Attributes that list the attribute’s short name in parentheses exist in each particle object by default.

- If you are an API developer, be aware that there are more attributes described in the online documentation. Use your internet browser to go to the main Maya documentation page, then go to the Dependency Graph Nodes page, and finally to the Particle page.
- The table omits the compound attributes centroid and worldCentroid. Compound attributes consist of two or more component attributes. The centroid attribute consists of centroidX, centroidY, and centroidZ attributes. The worldCentroid attribute consists of worldCentroidX, worldCentroidY, and worldCentroidZ You can use compound attributes with MEL commands such as setAttr. For details, see the online MEL documentation.

Attribute long name (and short name)	**Description**	**Data Type**
acceleration (acc)	Sets the rate of change of velocity on a per particle basis.	vector array
age (ag)	Contains the number of seconds each particle in the object has existed since the first frame. This is a read-only attribute.	float array
attributeName*	Specifies the name of the attribute whose values are to be displayed at particle positions. By default, particle id numbers are displayed. Valid for Numeric render type.	string
betterIllumination*	Provides smoother lighting and shadowing at the expense of increased processing time. Valid for Cloud render type.	boolean
birthPosition	Stores the position at which each particle was born in the particle's local space.	vector array
birthTime (bt)	Contains the Current Time value at which each particle in the object was created. This is a read-only attribute.	float array
cacheData (chd)	Turns on or off dynamic state caching for the object.	boolean

Attribute long name (and short name)	**Description**	**Data Type**
castsShadows (rsh)	Turns on or off the object's ability to cast shadows in software rendered images. Valid for Cloud, Blobby Surface, and Tube render types.	boolean
centroidX, centroidY, centroidZ (ctdx, ctdy, ctdz)	Contains the X, Y, and Z elements of the average position of its particles. These are read-only attributes.	float
collisionFriction (cfr)	Sets how much the colliding particle's velocity parallel to the surface decreases or increases as it bounces off the collision surface. This attribute is displayed as Friction in the user interface. It works on a per geometry basis.	float (multi)
collisionResilience (crs)	Sets how much rebound occurs when particles collide with a surface. This attribute is listed as Resilience in the user interface. It works on a per geometry basis.	float (multi)
collisionU, collisionV*	U and V positions of the NURBS surface where a particle collided in the current frame. For polygonal surfaces, the values are always 0. The values are reset to -1 at the start of each frame. Values change only in frames where collision occurs. These are read-only attributes.	float

Attribute long name (and short name)	Description	Data Type
colorAccum*	Adds the RGB components of overlapping particles. Also adds opacity values of overlapping particles. Generally, colors become brighter and more opaque as they overlap. To see the effect of Color Accum, you must add an opacity attribute to particles displayed as Points. Valid for MultiPoint, MultiStreak, Points, and Streak render types.	boolean
colorBlue*	Sets blue component of RGB color. Valid for all render types except Numeric and Tube.	float
colorGreen*	Sets green component of RGB color. Valid for all render types except Numeric and Tube.	float
colorRed*	Sets red component of RGB color. Valid for all render types except Numeric and Tube.	float
conserve (con)	Sets how much of a particle object’s velocity attribute value is retained from frame to frame.	float
count (cnt)	Contains the total number of particles in the object. This is a read-only attribute.	integer
currentTime (cti)	Sets the time value for the particle object’s independent clock.	time
depthSort (ds)	Turns on or off depth sorting of particles for rendering. This prevents unexpected colors when you hardware render overlapping colored, transparent particles. Valid for MultiPoint, MultiStreak, Points, Streak, and Sprites render types.	boolean

Attribute long name (and short name)	Description	Data Type
dynamicsWeight (dw)	Scales the effect of fields, collisions, springs, goals, and emission on particles.	float
emissionInWorld (eiw)	When on, emission occurs in the world coordinate system. This is the default setting. When off, emission occurs in the emitted particle object's local space.	boolean
emitterRatePP*	Sets the per particle emission rate.	float array
enforceCountFromHistory (ecfh)	In a soft body, if you change the original geometry's construction history in a way that alters the number of CVs, vertices, or lattice points, Maya updates the number of particles of the soft body correspondingly.	boolean
event*	Contains the number of times each particle in the object has hit something. This is a read-only attribute.	float array
expressionsAfterDynamics (ead)	Sets whether expressions are evaluated before or after other dynamics.	boolean
force (frc)	Contains the accumulation of all forces acting on the particle object. This is a read-only attribute. If you use this attribute in an expression, first turn on expressionsAfterDynamics.	vector array
forcesInWorld (fiw)	Sets whether forces are applied to the object in world space or in its local space.	boolean

Attribute long name (and short name)	Description	Data Type
goalActive (ga)	For a goal object, turns each goal on or off. It has the same effect as setting the corresponding goalWeight to 0, except the animation processing is more efficient. This attribute works on a per object basis.	boolean (multi)
goalOffset*	Sets an offset to the world space position of the goal object.	vector array
goalPP*	Sets how much the particles try to follow the goal on a per particle basis.	float array
goalSmoothness (gsm)	Sets how smoothly goal forces change as the goal weight changes from 0 to 1. The higher the number, the smoother the change.	float
goalU, goalV*	Set the exact locations on a NURBS surface where the particles are attracted.	float array
goalWeight (gw)	Sets how much all particles of the object are attracted to the goal.	float (multi)
incandescencePP*	Sets glow color in conjunction with a software rendering Particle Incand Mapper Node (see *Rendering*.) Valid for Cloud and Tube render types.	vector array
inheritFactor (inh)	Sets the (decimal) fraction of velocity an emitted particle object inherits from an emitter.	float
inputGeometrySpace (igs)	For a soft body, this sets the coordinate space Maya uses to position point data provided by the input geometry to the particle shape.	integer

Attribute long name (and short name)	Description	Data Type
isDynamic (isd)	Turns on or off dynamic animation of the object.	boolean
isFull (ifl)	Contains 1 if an emitted particle shape is full, or 0 if not full. An emitted particle shape is full when the number of emitted particles equals the maxCount. This is a read-only attribute.	boolean
lifespan*	Sets when all particles in the object die.	float
lifespanPP*	Sets when particles die on a per particle basis.	float array
levelOfDetail (lod)	Scales the number of particles that can be emitted into the emitted particle object.	float
lineWidth *	Sets the width of streaking particles. Valid for MultiStreak and Streak render types.	float
mass (mas)	Specifies the physical mass of particles. Mass values affect the results of dynamic calculations. By default, each particle of a particle object has a mass of 1.	float array
mass0 (mas0)	Initial state counterpart to mass.	float array
maxCount (mxc)	Sets a limit on the number of particles the emitted particle shape accepts from an emitter.	int
multiCount*	Sets number of points you want displayed for each particle in the object. Valid for MultiPoint and Points render types.	float

Attribute long name (and short name)	**Description**	**Data Type**
multiRadius*	Sets radius of spherical region in which particles are randomly distributed. Valid for MultiPoint and MultiStreak render types.	float
needParentUV*	Turns on or off the ability to read the parentU and parentV attributes. If you add a surface emitter to a NURBS or polygonal surface, parentU and parentV contain the UV coordinates where each particle was emitted. You can use these attributes in expressions and MEL scripts.	boolean
normalDir*	Sets direction of normal for particles. Use with useLighting. Valid for MultiPoint, MultiStreak, Points, and Streak render types.	integer (1-3)
opacity*	Sets amount of transparency for all particles in the object. Valid for all render types except Numeric and Tube.	float
opacityPP*	Sets amount of transparency on a per particle basis. Valid for all render types except Numeric, Tube, and Blobby Surface.	float array
particleId (id)	Contains the id number of each particle. Valid for Numeric render type. This is a read-only attribute.	float array

Attribute long name (and short name)	Description	Data Type
parentId*	If you emit from a particle object, this contains the id of all particles that emit the particles. You can use the id to query the emitting object's attribute values, for example, acceleration, velocity, and lifespanPP. This is a read-only attribute. Note that if you use the MEL emit command to create the particles that emit, the parentId attribute of those emitted particles is always 0.	float array
parentU, parentV*	If you add a surface emitter to a NURBS or polygonal surface, these attributes contain the UV coordinates where each particle was emitted. To use these read-only attributes, you must turn on Need Parent UV in the emitter. You can use these attributes in expressions and MEL scripts.	float array
particleRenderType*	Sets render display type of particles, for example, Streak.	integer
pointSize*	Sets how large particles are displayed. Valid for MultiPoint, Numeric, and Points render types.	float
position (pos)	Sets the object position in local space coordinates on a per particle basis.	vector array
position0 (pos0)	Initial state counterpart to position.	vector array
radius*	Sets radius size of all particles. Valid for Blobby Surface, Cloud, and Sphere render types.	float

Attribute long name (and short name)	**Description**	**Data Type**
radius0*	Sets starting point radius for Tube render type.	float
radius1*	Sets ending point radius for Tube render type.	float
radiusPP*	Sets radius size on a per particle basis. Valid for Blobby Surface, Cloud, and Sphere, render types.	float array
rampAcceleration (rac)	Controls acceleration with a ramp. Any other animation of acceleration is added to the ramp-controlled acceleration.	vector array
rampPosition (rps)	Controls position with a ramp. Any other animation of position is added to the ramp-controlled position.	vector array
rampVelocity (rvl)	Controls velocity with a ramp. Any other animation of velocity is added to the ramp-controlled velocity.	vector array
rgbPP*	Sets color on a per particle basis. Valid for MultiPoint, MultiStreak, Points, Spheres, Sprites, and Streak render types.	vector array
seed (sd)	Sets the id of the random number generator of the associated emitter. This attribute works on a per object basis.	float (multi)

Attribute long name (and short name)	**Description**	**Data Type**
sceneTimeStepSize (sts)	Contains the value of the time difference between the last displayed frame and current frame. This contains 1 if you're simply playing the animation or clicking the frame forward or backward button. If you click widely separated frames in the Time Slider, the attribute contains the difference in time between the two frames. This is a read-only attribute.	time (in current units)
selectedOnly*	Turns on or off display of id numbers only for selected particles. Valid for Numeric render type.	boolean
spriteNum*	Sets the image number index for a Sprite image sequence.	integer
spriteNumPP*	Sets the image number index for a Sprite image sequence on a per particle basis.	integer array
spriteScaleX, spriteScaleY*	Sets the Sprite X- and Y-axis image scale.	float
spriteScaleXPP, spriteScaleYPP*	Sets the Sprite X- and Y-axis image scale on a per particle basis.	float array
spriteTwist*	Sets the Sprite image's rotation angle.	float
spriteTwistPP*	Sets the Sprite image's rotation angle on a per particle basis.	float array
startFrame (stf)	Sets the animation frame after which dynamics (including emission) are computed for the object.	float

Attribute long name (and short name)	**Description**	**Data Type**
surfaceShading*	Sets how sharply the spheres of Cloud render type are displayed. Use a value between 0 and 1. A value of 1 displays spheres more distinctly; a value of 0 creates a cloudier effect.	float
tailFade*	Sets the opacity of tail fade. Valid for MultiStreak and Streak render types.	float
tailSize*	Sets the length of the tails for MultiStreak, Streak, and Tube render types.	float
targetGeometrySpace (tgs)	For a soft body, sets the coordinate space Maya uses to position point data provided by the particle shape to the target geometry.	integer
threshold*	Controls surface blending between Cloud or Blobby surface spheres. This is a read-only attribute.	float
timeStepSize (tss)	Contains the animation frame increment in current units. For example, if your animation is set to Film (24 fps), timeStepSize has a value of 1 (frame). Keying or otherwise setting the Current Time value alters the timeStepSize. For instance, with a frame rate of 24 frames per second, suppose you set the Current Time to 0 at frame 0, and to 100 at frame 50. Because you're compressing twice as much time between frames 0 and 50, the timeStepSize is twice as large, in other words, 2. This is a read-only attribute.	time

Attribute long name (and short name)	**Description**	**Data Type**
totalEventCount (evc)	Contains total events that have occurred for all particles of the object. This is a read-only attribute.	integer
traceDepth (trd)	Sets the maximum number of collisions Maya can detect for the object in each animation time step.	integer
traceDepthPP*	Sets the trace depth on a per particle basis.	float array
UseCurrentUVSet	Selects the UV set on the poly emitter that is used for textured emission effects. The same UV set is used for all textures attached to all textured emission effects (color/opacity and emission rate). If the UseCurrentUVSet option is selected, then the active UV set on the surface emitter is used.	boolean
useLighting*	Turns on or off whether scene lighting lights up particles. Valid for MultiPoint, MultiStreak, Points, Sprites, and Streak render types.	boolean
userScalar1PP userScalar2PP userScalar3PP userScalar4PP userScalar5PP	Predefined outputs for user-defined attributes used in Particle Sampler Info node.	float array
userVector1PP userVector2PP userVector3PP userVector4PP userVector5PP	Predefined outputs for user-defined attributes used in Particle Sampler Info node.	vector array
velocity (vel)	Sets speed and direction on a per particle basis.	vector array
velocity0 (vel0)	Initial state counterpart to velocity	vector array

Attribute long name (and short name)	Description	Data Type
visibleInReflections (rrl)	Turns on or off whether the object is visible in reflections when software rendered. Valid for Cloud, Blobby Surface, and Tube render types.	boolean
visibleInRefractions (rrr)	Turns on or off whether the object is visible in refractions when software rendered. Valid for Cloud, Blobby Surface, and Tube render types	boolean
worldBirthPosition	Stores the position at which each particle was born in world space.	vector array
worldCentroidX, worldCentroidY, worldCentroidZ (wctx, wcty, wctz)	Contains the world space X, Y, and Z elements of the average position of its particles. These attributes are a read-only attributes.	float
worldPosition (wps)	Contains the world space counterpart to position. This is a read-only attribute.	vector array
worldVelocity (wvl)	Contains the world space counterpart to velocity. This is a read-only attribute.	vector array
worldVelocityInObjectSpace (wvo)	Contains the local space equivalent to the object's world space velocity. This is a read-only attribute.	vector array

Section of Attribute Editor	**Contents**
General Control Attributes	Attributes that set various operation characteristics of the entire particle object. These attributes are described in various places: Is Dynamic See "Create rigid bodies" on page 259. Dynamics Weight See "Scale the effect of dynamics" on page 18. Conserve See "Adjust frame-to-frame velocity conservation" on page 74. Forces In World See "Apply forces in an object's local space" on page 74. Expressions After Dynamics See "Control execution time of particle dynamics" on page 76. Cache Data See "Memory caching" on page 378. Count See "Attribute long name (and short name)" on page 170. Total Event Count See "Attribute long name (and short name)" on page 170
Emission Attributes	Attributes that determine some of the behavior of particles that are emitted. See "Edit particle attributes" on page 36. Also see the Emitter tab in the Attribute Editor of more attributes that affect emitted particle behavior.
Lifespan Attributes	Attributes that determine the particle lifespan, which can make particles disappear from the scene after they reach a specified age. Lifespan is typically used with emitted particles. See "Set particle lifespan" on page 51.

Section of Attribute Editor	Contents
Time Attributes	Attributes that set the start frame and timing of dynamic influences on particles. See ”Change the start frame for a particle object” on page 76 and ”Control the timing of particle dynamics” on page 77.
Collision Attributes	Attributes that set how precisely collision detection occurs. See ”Avoid unexpected particle penetration of geometry” on page 116.
Soft Body Attributes	Attributes for advanced users who want to tune soft body effects (see Chapter 4, “Soft and Rigid Bodies”).
Goal Weights and Objects	Attributes you can set if you add a goal to an object and connect it to a particle object or soft body. See ”Goals” on page 21.
Instancer (Geometry Replacement)	Attributes that affect geometry instanced to animated particles. See ”Instance geometry to particles (single and animated)” on page 67.
Particle Seeds	Attributes that alter the random number stream used by Maya to create the random positions of emitted particles. See ”To make particles live forever” on page 52.
Render Attributes	Attributes that affect how particles are displayed and rendered. See ”Choose how particles render” on page 37.
Render Stats	Attributes that set aspects of particle software rendering. See ”Add reflections, refractions, and shadows” on page 46.
Per Particle (Array) Attributes	Attributes you can set on an individual particle basis for several commonly used motion and display features. See ”Per particle and per object attributes” on page 36 and ”Set attributes on a per particle basis” on page 54.
Add Dynamic Attributes	Buttons for adding attributes that control commonly desired features (see ”Add dynamic attributes” on page 34).
Extra Attributes	Any custom *per object* attributes you’ve added. See ”Per particle and per object attributes” on page 36.

Add Attributes For Current Render Type button

The added attributes follow:

Color Accum

Adds the RGB components of overlapping particles. Also adds opacity values of overlapping particles. Generally, colors become brighter and more opaque as they overlap. To see the effect of Color Accum, you must add an opacity attribute to particles displayed as Points. See ”Set particle opacity” on page 50.

Normal Dir

Lets you tune the illumination of moving particles lit by a light added to the scene. See ″Use lights with moving particles″ on page 47 for more details.

Point Size

Sets the size of the particle points. Point size is unaffected by camera distance. Both far and near particles appear the same size in the workspace.

Use Lighting

Illuminates particles using lights added to the scene. See ″Use lights with moving particles″ on page 47 for details.

> Note After you add attributes for a particle render type, you can change the render type and add its default Particle Render Type attributes. Any attributes that don't apply to the new render type are dimmed.

Add Attributes For Current Render Type button

The added attributes follow:

Color Accum

Adds the RGB components of overlapping particles. Also adds opacity values of overlapping particles. Generally, colors become brighter and more opaque as they overlap. To see the effect of Color Accum, you must add an opacity attribute to particles displayed as MultiPoint. See ″Set particle opacity″ on page 50.

Multi Count

Sets the number of points you want displayed per particle.

Multi Radius

Sets the radius of the spherical region within which the particles are randomly distributed.

Normal Dir

Lets you tune the illumination of moving particles lit by a light added to the scene. See ″Use lights with moving particles″ on page 47 for details.

Point Size

Sets the size of the particle points. Point size is unaffected by camera distance. Both far and near particles appear the same size in the workspace.

Use Lighting

Illuminates particles using lights added to the scene. See "Use lights with moving particles" on page 47 for details.

Add Attributes For Current Render Type button

The added attributes follow:

Color Accum

Adds the RGB components of overlapping particles. Also adds opacity values of overlapping particles. Generally, colors become brighter and more opaque as they overlap.

Line Width

Sets the width of each streak.

Normal Dir

Lets you tune the illumination of moving particles lit by a light added to the scene. See "Use lights with moving particles" on page 47 for details.

Tail Fade

Sets the opacity of the tail fade, from 0 to 1. A value of 1 makes the tail completely opaque; a value of 0 makes the tail transparent.

Tail Size

Scales the length of the tail. A value of 1 gives the default length. Values less than 1 shorten the tail; values greater than 1 lengthen the tail.

Use Lighting

Illuminates particles using lights added to the scene. See "Use lights with moving particles" on page 47 for details.

Add Attributes For Current Render Type button

The added attributes follow:

Color Accum

Adds the RGB components of overlapping particles. Also adds opacity values of overlapping particles. Generally, colors become brighter and more opaque as they overlap.

Line Width

Sets the width of each streak.

Multi Count

Sets the number of points you want displayed per particle.

Multi Radius

Sets the radius of the spherical region within which the particles are randomly distributed.

Normal Dir

Lets you tune the illumination of moving particles lit by a light added to the scene. See ”Use lights with moving particles” on page 47 for details.

Tail Fade

Sets the opacity of the tail fade, from 0 to 1. A value of 1 makes the tail completely opaque; a value of 0 makes the tail transparent.

Tail Size

Scales the length of the tail. A value of 1 gives the default length. Values less than 1 shorten the tail; values greater than 1 lengthen the tail.

Use Lighting

Illuminates particles using lights added to the scene. See ”Use lights with moving particles” on page 47 for details.

Render attributes

Sprite Num

Sets the filename extension number of the texture file to be displayed as part of a sequence of images. When you use the Sprite Wizard, the Sprite Num attribute is ignored.The Sprite Wizard adds the per-particle attribute SpriteNumPP, which overrides the Sprite Num attribute.

Sprite Scale X

Scales the image along the X-axis by a percentage. For example, a value of 0.5 scales the image by 50%; a value of 2 scales the image by 200%.

Sprite Scale Y

Scales the image along the Y-axis by a percentage. For example, a value of 0.5 scales the image by 50%; a value of 2 scales the image by 200%.

Sprite Twist

Sets the rotation angle of the image's position in the scene. For example, a value of 45 rotates the image 45 degrees counter-clockwise about the Z axis.

Use Lighting

Illuminates particles using lights added to the scene. See ”Use lights with moving particles” on page 47 for details.

Tips

To set the transparency of the entire sprite, add an opacity or opacityPP attribute to the particle shape, then set the attribute value between 0 and 1. For example, the value 0.25 creates 25% opacity. See ”Set particle opacity” on page 50.

To set the scale and twist of sprites on a per particle basis, add the spriteScaleXPP, spriteScaleYPP, and spriteTwistPP attributes, then set their values. See ”Add dynamic attributes” on page 34 and ”Set attributes on a per particle basis” on page 54.

If you need to select the particle object but it disappears from the workspace after you add these attributes, select the object in the Outliner. If you need to see the particles to select individual particles and set their per particle attribute values, temporarily select Points as the Particle Render Type.

Sprite attributes

Sprite Animation

Specifies how images are assigned to the sprites. Select one of the following:

off - Each particle uses some fixed image throughout its lifetime.

on - Each particle displays a sequence of images in turn.

Sprite Start Option

Selects the technique used to assign the initial sprite to each particle. Select one of the following:

first - Every particle uses the first image in the sequence.

particleid - The particle uses its particleId value to pick an image. If the particleId is greater than the number of images in the sequence, it starts over with the first image.

random - Each particle uses a random image from the sequence.

custom - The particle uses its spriteNumPP value to select the image. You must set the spriteNumPP values by editing the particle creation expression. See ”To customize the selection of the initial sprite image” on page 42.

ramp - Connects a ramp to the spriteNumRamp attribute of the particle object. The ramp determines the image selection. You can modify this ramp to determine which sprite receives each image.

Sprite Cycle Pattern

Specifies how the images are cycled for each particle. Select one of the following:

linearUp - Plays through the image sequence at an even speed.

easeUp - Eases in and eases out the rate that the image sequence is played. The first and last images in the sequence will be used for longer times.

linearUpDown - Plays through the image sequence at an even speed. When the end of the sequence is reached, it plays in reverse.

easeUpDown - Eases in and eases out the rate that the image sequence is played. When the end of the sequence is reached, it plays in reverse.

custom - The particle uses its spriteNumPP value to select the image. You must set the spriteNumPP values by editing the particle runtime expression. See ”To customize sprite cycling” on page 43.

ramp - Connects a ramp to the spriteNumRamp attribute of the particle object. The ramp determines the image selection. You can modify this ramp to determine which sprite receives each image.

Sprite Cycle Length

Controls the length of each animation cycle. this is valid only if the Cycle only once during lifespan option is disabled.

Sprite Invert Cycle

Invert the meaning of the ramp's values. Higher values use images closer to the start of the sequence.

Add Attributes For Current Render Type button

The added attribute follows:

Radius

Sets the radius of the spheres. To set the radius on a per particle basis, add the radiusPP attribute, then set its values. See "Add dynamic attributes" on page 34 and "Set attributes on a per particle basis" on page 54.

Numeric

Add Attributes For Current Render Type button

The added attributes follow:

Attribute Name

Name of the attribute whose values are to be displayed. You can use static, dynamic, or per particle attributes. See "Work with particle attributes" on page 33 for details.

Point Size

Sets the size of the particle points. Point size is unaffected by camera distance. Both far and near particles appear the same size in the workspace.

Selected Only

Displays the attribute value of currently selected particles only.

Click  (the Select by Component Type icon) to select individual particles.

Blobby Surface

Add Attributes For Current Render Type button.

The added attributes follow:

Radius

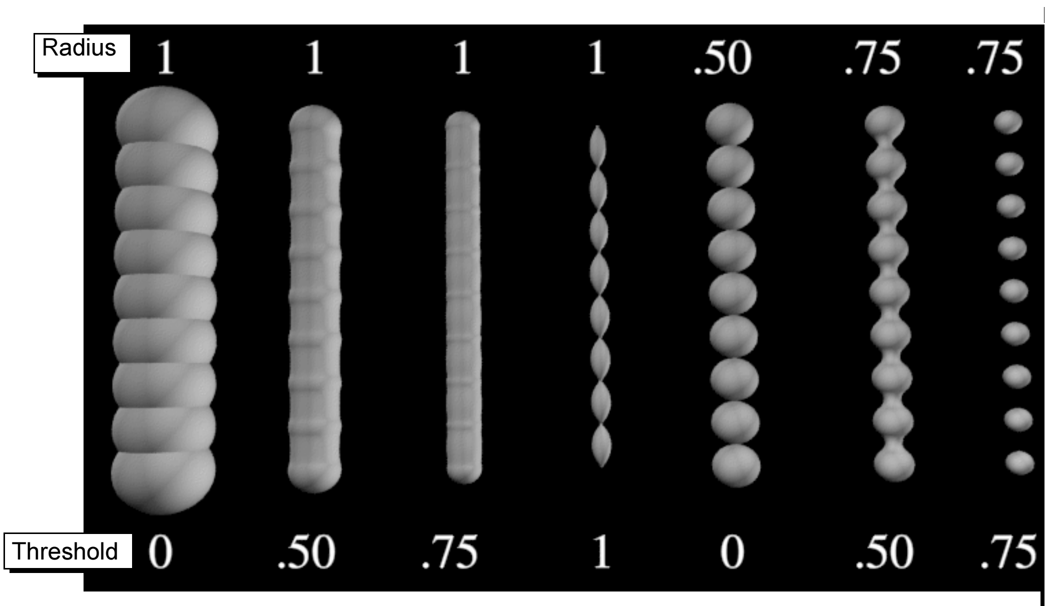
Sets the radius of the Blobby Surfaces.

To set the radius on a per particle basis, add the radiusPP attribute, then set its values. See "Add dynamic attributes" on page 34 and "Set attributes on a per particle basis" on page 54.

Threshold

Controls surface blending between the particles. A value of 0 creates no blending. Generally, blending increases as the value approaches 1, then decreases again above 1. The Blobby Surfaces disappear if you set the value too high.

Increasing Threshold displays the Blobby Surfaces smaller. Increase the Radius if necessary. An example of the relationship between these attribute values follows:



Cloud

Add Attributes For Current Render Type button

The added attributes follow:

Better Illumination

Provides smoother lighting and shadowing at the expense of increased processing time. See "Use lights, reflections, refractions, and shadows" on page 46 for details on using lights and shadows.

Radius

Sets the radius of the Clouds.

To set the radius on a per particle basis, add the radiusPP attribute, then set its values. See ”Add dynamic attributes” on page 34 and ”Set attributes on a per particle basis” on page 54.

Surface Shading

Sets how sharply the Clouds are displayed. Enter a value between 0 and 1. A value of 1 displays the Clouds more distinctly; a value of 0 creates a cloudier effect.

Threshold

Controls surface blending between the Clouds. A value of 0 creates no blending; a value of 1 creates maximum blending.

Increasing the Threshold displays the Clouds smaller, so you might need to increase the Radius as you increase Threshold.

Maya assigns the default particle cloud shader to the particle object. We recommend that you create and assign a unique particle cloud shader. Using default shading groups can cause problems if you duplicate the object. See ”Create raytraced shadows with particles” on page 126 for information on setting particle cloud shader attributes.

Tube

Add Attributes For Current Render Type button

The added attributes follow:

Radius0

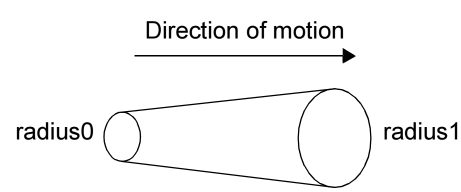
Sets a radius for the tube at its starting point. See the figure that follows.

Radius1

Sets a radius for the tube at its ending point.

Tail Size

Scales the length of the tube. This value is multiplied by the particle's velocity to set the tube length. The faster the particle moves, the longer the tube.



Maya assigns the default Cloud particle material to the particle object. This material is part of the default initialParticle shading group. You can create and assign a different shading group with your own custom material.

The predefined vector array attributes are listed below:
ramp for a vector array (per particle) attribute

Attribute	To add this attribute:
rgbPP (color)	See "Set particle color" on page 48.
rampPosition, rampVelocity, or rampAcceleration	No action necessary. A particle object has these attributes by default.
goalOffset	See "Create goals" on page 107 and "Add dynamic attributes" on page 34.

a ramp for a float array (per particle) attribute

The predefined float array attributes are listed below:

Attribute	To add this attribute:
opacityPP (opacity)	See "Set particle opacity" on page 50.
radiusPP (radius of Blobby Surfaces, Clouds, or Spheres)	See "Add dynamic attributes" on page 34.
*emitterName*RatePP (emission rate of a point emitter)	See "Vary emission from different points of point emitters" on page 97.
goalPP (goal weight)	See "Setting goal weight on a per object basis" on page 108.
goalU and goalV (NURBS U and V position for goals)	See "Add dynamic attributes" on page 34.

Attribute	To add this attribute:
mass	No action necessary. A particle object has this attribute by default.
spriteNumPP, spriteScaleXPP, spriteScaleYPP, and spriteTwistPP (filename extension, X-axis image scale, Y-axis image scale, and image rotation angle of Sprites)	See ”Add dynamic attributes” on page 34.
traceDepthPP (collision detection sensitivity)	See ”Add dynamic attributes” on page 34.

Input U or Input V menu

None

No attribute affects the ramp component. This is the default setting for Input U.

Particle’s Age

The particle object’s ageNormalized attribute. This attribute contains the particle’s age, normalized to values between 0 and 1 from birth to death. This is the default setting for Input V.

> Note   Maya normalizes only the particle’s age (ageNormalized) values from 0 to 1. All other attributes control the ramp component with their literal values. If the attribute exceeds the range 0 to 1, the values in the ramp repeat. For example, a value of 1.5 represents the same position in the ramp as a value of 0.5.

*attribute*UPP or *attribute*VPP

Creates a custom attribute that controls the ramp component. The attribute is named *attribute*UPP or *attribute*VPP, where *attribute* is the name of the attribute that appears in the menu. You can set this attribute with an expression, ramp, or the Component Editor.

Note that you can also select *attribute*UPP or *attribute*VPP from the Attribute Editor. Simply right-click the attribute box of the per particle attribute that's connected to a ramp, slide the pointer to the arrow to the right, and select *attributeUorV*PP > arrayMapper2.outValuePP.

Other attributes

The Input U and Input V menus also lists all float array attributes in the particle object. You can select one of the listed attributes to control the ramp component.

per particle attribute that's connected to a ramp

Break Connection

Breaks the array mapper's input to the particle shape. This menu item doesn't delete the ramp node or array mapper. Use this menu item if you don't want to remove the ramp when you break the connection.

Delete Array Mapper

Breaks the array mapper's input to the particle shape by deleting the array mapper. The ramp will also be deleted if it has no other connections. Use this menu item if you no longer need the ramp in the scene.

emitted particle object

Max Count	Sets a limit on the number of particles the emitted particle object accepts from an emitter. For example, if you set Max Count to 20, the particle object accepts new particles until it has 20 particles, then the emission turns off. If particles die and reduce the count below 20, more particles emit until the count reaches 20. The default value of -1 means there is no limit.
Level Of Detail	Scales the number of emitted particles. You can enter a value between 0 and 1 to scale the emission rate between 0 and 100%. Scaling down the emission rate speeds scene play.

Inherit Opacity	Gives the particles the texture's transparency using the alpha information—Alpha Gain, Alpha Offset, and Contrast (if present). If you turn on Inherit Opacity, you can use the following attributes:
Use Luminance	Uses the texture's luminance rather than alpha information as the source of transparency.
Invert Opacity	Gives the particles a transparency opposite the texture's transparency, whether from alpha or luminance.

geoConnector attributes

Resilience

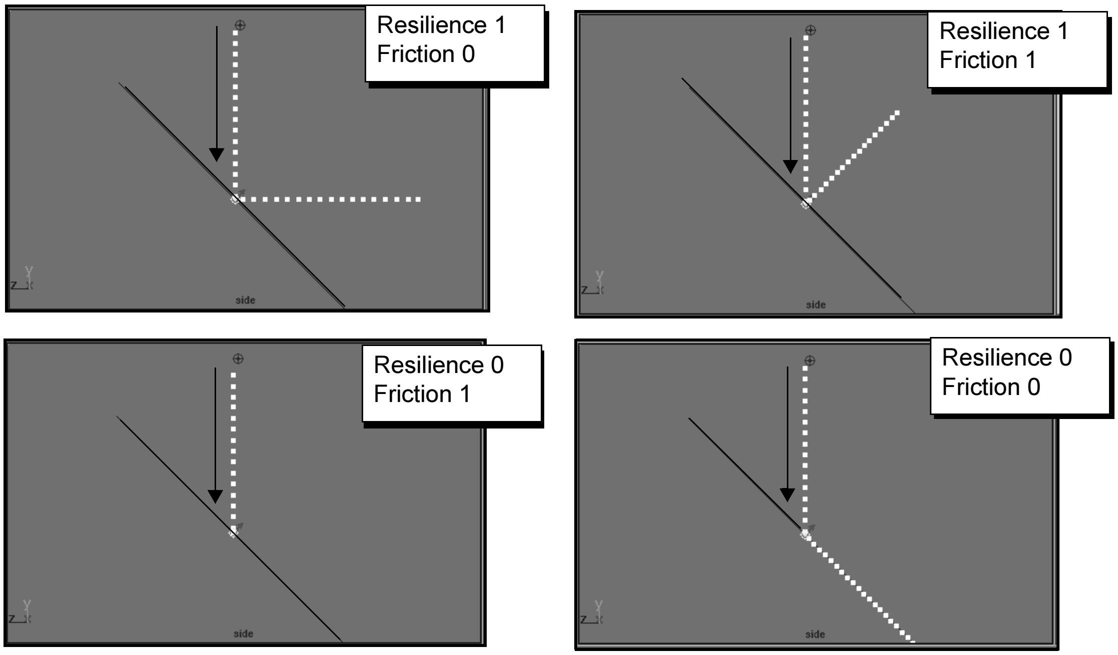
Sets how much rebound occurs. A value of 0 makes particles collide with no bounce. A value of 1 causes particles to rebound fully. A value between 0 and -1 makes the particles pass through the surface with refraction out the back side. Values greater than 1 or less than -1 add speed to the particles.

Friction

Sets how much the colliding particle's velocity parallel to the surface decreases or increases as it bounces off the collision surface.

A value of 0 means that particles are unaffected by friction. A value of 1 makes particles reflect straight off along the normal of the surface. If Resilience is 0 while Friction is 1, the particles don't bounce.

Only values between 0 and 1 correspond to natural friction. Values outside this range exaggerate the response.



Particle Collision Event

Set Event Name

Specifies the name of the event so you can identify it for editing later.

All Collisions

By default, the event occurs each time a particle in the object collides with the geometry, whether or not it's a repeat collision. If you want the event to occur only on a specific collision, for instance, on a second (repeat) collision, turn off All Collisions and set the Collision Number to the desired value. For example, if you set the Collision Number to 2, the event happens on each particle's second collision.

Emit

When on, colliding particles emit new particles. The number of particles is specified in the Num particles box. The new particles start with an age of 0 and have the point render type by default.

Split

When on, colliding particles split into new particles. The number of particles is specified in the Num particles box. The new particles inherit the current age of the particles that split. The new particles have the Points render type by default.

When you turn on Emit or Split, you can set the following attributes.

Random # Particles

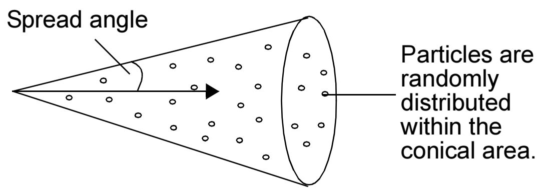
When off, the particles will split or emit the number of particles in Num particles. When on, a random number between 1 and the number specified in Num particles is used.

Num particles

Number of particles that are split or emitted.

Spread

Sets the angle of a conical region where the particles are emitted randomly. You can enter a value between 0 and 1. A value of 1 means 180 degrees.



Target Particle

Specifies the particle object whose attributes are used by the new particles. (The new particles become part of this object.) The target can be an existing particle object or a new particle object. If you don't specify a target, a default name is assigned. To use the original particle object's render type, specify that object as the target.

Inherit Velocity

Sets how much velocity of the original particle is inherited by the new particles after collision.

A value of 0 means the new particles do not inherit the velocity of the original particle and do not bounce. A value of 1 means the new particles inherit all the velocity of the original particles, so the particles bounce. Values between 0 and 1 create a diminished bounce.

To make the new particles collide with the same surface, you must do the steps in "Make particles collide with a surface" on page 113.

Note	You can use the following attributes to execute a MEL script or make particles die regardless of whether you are emitting or splitting particles. You can also apply multiple events to the same particle object to create various simultaneous effects.

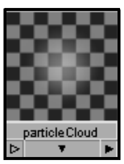
Original Particle Dies

Makes particles disappear upon collision.

Event Procedure

Lets you specify the name of a MEL procedure to be executed upon particle collision. Do not use a .mel extension in the name. You're specifying a procedure, not a script name. You must create and source a MEL procedure before the collision event occurs (see page 118).

Particle Cloud attributes



Common Material Attribute

You can set these attributes to a fixed value, or texture map them over the particles' lifetime. If mapped, the V values of the map are mapped onto the particles' lifetime. You can also control them with per-particle attributes using the Particle Sampler Info node.

Color

The basic color of the particle cloud. The default color is a green-blue.

Transparency

Controls how much you can see through the particle cloud. It is a color, so that you can control the transparency of the red, green, and blue channels separately. To make the cloud more opaque, set transparency to be darker. To make the cloud more transparent, set transparency to be brighter (transparency = 1-opacity).

Incandescence

Use Incandescence to make the particle cloud brighter, as though it were a light source. By default, Incandescence is black, meaning no glow will be added.

Note	When Incandescence is turned on, although the particle cloud will glow, it will not cast light on other objects in the scene.

Life Color

Determines the color at a particular time in the life of the particle. You can use the Particle Sampler Info node to animate this parameter over a particle’s lifetime (see ”Particle Sampler Info node” on page 25).

Life Transparency

Determines the transparency at a particular time in the life of the particle. You can use the Particle Sampler Info node to animate this parameter over a particle’s lifetime (see ”Particle Sampler Info node” on page 25).

Life Incandescence

Determines the incandescence at a particular time in the life of the particle. You can use the Particle Sampler Info node to animate this parameter over a particle’s lifetime (see ”Particle Sampler Info node” on page 25).

Glow Intensity

Controls how much of a halo-like glow effect will be added to the particle cloud. This glow effect is added as a post-process, after the rendering is completed. Glow Intensity is zero by default, meaning that no glow is added.

Transparency

Density

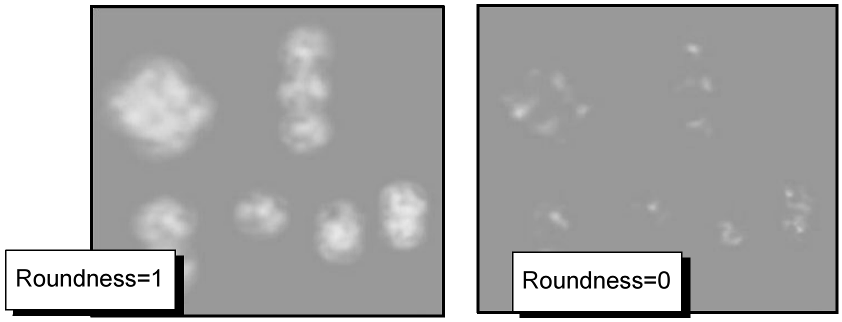
Similar to transparency; it controls how dense the cloud of particles appears to be, and therefore how much of the background can be seen through it. Increase this value to make the cloud more dense.

Blob Map

Specifies a scaling factor applied to the transparency of the particle cloud. You can connect a 3d texture to it in order to give some internal texture or shape to the cloud beyond what it gets from the particles.

Roundness

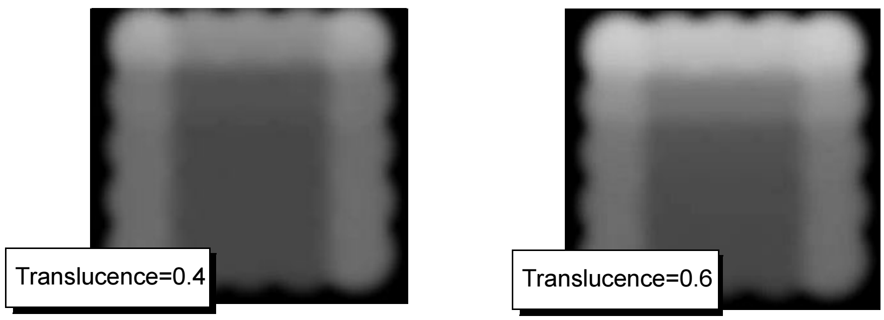
Controls the noise’s irregularity. The smaller the value, the less rounded the shape.



Translucence

Specifies a scaling factor for density that is used to compute shadows only. The larger the translucence value, the more light penetrates. The formula is:

```
density * (1 - translucence)
```



Built-in Noise

Noise

Controls the jitteriness within the particle cloud. If it is set to zero, the cloud will look very smooth and uniform throughout. As the amount of noise increases, the cloud will appear *noisier*, like static on a television screen. Noise is set to 0.75, by default.

Noise Freq

Determines the size of the noise artifacts when Noise is turned on. Higher values of Noise Frequency produce smaller, finer artifacts, and lower values produce larger, coarser artifacts. If Noise Frequency is set to zero, that is the same as turning Noise off.

Noise Aspect

Controls the distribution of the noise (when Noise is turned on). It is zero by default, meaning that the noise is equally distributed in X and Y. Positive values make the noise run perpendicular to the particle's path. Negative values make the noise run more parallel to the path.

Noise Anim Rate

Specifies a scaling factor that controls the rate of built-in noise changes during an animation.

Solid Core Size

Determines the size of the core, which is the area where the particle is opaque.

Surface Shading Properties

Diffuse Coeff

Controls how much of the light in the scene is reflected from the particles. Most materials absorb some of the light falling on them, and scatter the rest.

The default value is 0.0. If you set this to 1.0, all the light falling on the material is reflected. Use a high value when you are creating dense clouds. If you set this to 0.0 (the minimum), no light is reflected and no surface shading occurs.

The surface color is modulated by the transparency. This value can be greater than 1.0, so the surface property can still appear even when the material is transparent.

Surface Color

Specifies the basic color of the particle cloud surface (as opposed to the inside of the cloud). Diffuse Coeff must be set to a value greater than 0 to enable this option.

Bump Mapping

Makes the surface appear rough or bumpy by altering surface normals (during rendering) according to the intensity of the pixels in the bump map texture. Diffuse Coeff must be set to a value greater than 0 to enable this option.

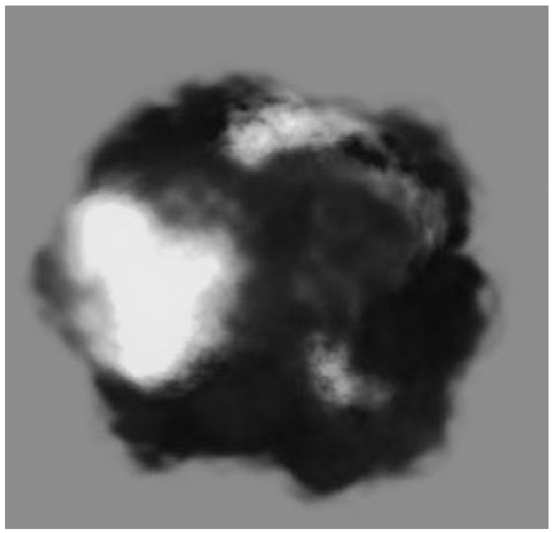
A bump map does not actually alter the surface. A silhouette of the surface will appear smooth.

Translucence Coeff

Simulates the way light diffusely penetrates through translucent objects. This means that when light shines on one side of the object, the other side is partially illuminated. You can use this to create effects such as clouds, fur, hair, marble, jade, wax, paper, leaves, etc. If you set Translucence Coeff to 0 (the default), no light shows through the object. If you set Translucence Coeff to 1, all the light shows through. Diffuse Coeff must be set to a value greater than 0 to enable this option.

Surface Shading Shadow

Determines if the surface shading is combined with the pre-illumination, which contains shadows, if enabled (see the "Filter Radius" attribute). Diffuse Coeff must be set to a value greater than 0 to enable this option.



This cloud shader uses Surface color and Surface Shading Shadow to create an explosion.

Pre-illumination Controls

Filter Radius

Volumetric particles use pre-illumination, which evaluates the lighting at each particle's center by default. This can sometimes cause popping if the illumination changes too fast in an animation, and is especially noticeable if Surface Shading Shadow is on.

Filter radius lets you filter the pre-illumination results so the value at each particle's center will be the average of all the pre-illumination results within the filter radius. Higher values increase render time but produce smoother images.

Particle Sample Info Node

Out Uv Coord

By default, Out U Coord and Out V Coord give you the normalized age of the particles. The outUv Coord attribute is comprised of Out U Coord and Out V Coord.They are provided in this form so that you can use this attribute to control texture placement nodes.

Out Uv Type

Specifies how Out Uv Coord is computed. You have four choices:

Normalized age

The particle's age divided by its finalLifespanPP. The finalLifespanPP attribute is used for the particle's age. This works in all Lifespan Modes of the particle shape. But if Lifespan Mode is set to Live forever, all lifespans are effectively infinite, so normalized age is always zero.

Out UV Type set to Normalized age is the default; you don't need to set a use relative age attribute as you did in previous releases.

You can use this output of the Particle Sampler Info node regardless of whether the particle shape has an ageNormalized attribute.

Absolute age

Uses the Normalization Value attribute of the Particle Sampler Info node, in place of finalLifespanPP. This is useful if you want to have the particles live forever, or if you want to cycle through a texture a number of times during the lifespan.

Parent UV and Collision UV

Use these attributes of the particle shape as the u and v values. This gives you more power than you had before, because in addition to age, you can use these attributes to define your own texture coordinates for the particles.

Normalization Method

(*Applies only if the Out UV Type is set to Absolute age.*) Specifies how to apply Absolute Age and particle age to compute the Out UV Coord. The age ranges from 0 to the value specified by the Normalization Value.

Oscillate

As the particle's actual age increases, Maya cycles through the normalization range repeatedly. For example, if the normalization value is 2, as the particle age, Maya uses the following values:

Actual particle age	**Value Maya uses**
0	0
0.1	0.1
1	1
2	2
2.1	0.1
3	1
4	2

Clamp

If the actual age exceeds the normalization value, Maya uses the normalization value. For example, if the normalization value is 2, as the particle age, Maya uses the following values:

actual age	**value to use**
0	0
0.1	0.1
1	1
2	2
2.1	2
3	2
4	2

Normalization value

(*Applies only if the Out UV Type is set to Absolute age.*) Specifies upper value of the age range. For example, if you use a value of 2, the particles' ages range from 0 to 2.

Inverse Out Uv

Uses 1 - Out Uv Coord in the final computation (instead of Out Uv Coord). For example, if you have out Uv Type set so that the value is 1 when the particle is born and 0 when it dies. This applies to all Out UV types.

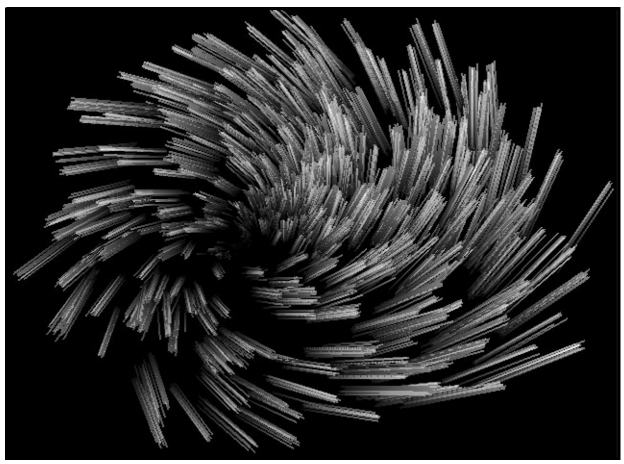
3 Fields

About Dynamics

Fields

Fields

You can simulate the motion of natural forces with dynamic fields. For example, you can connect a vortex field to emitted particles to create swirling motion.



Vortex field connected to emitted particles

Fields are forces that you use to animate the motion of particles, soft bodies, and rigid bodies. A soft body is a polygonal surface, NURBS surface, NURBS curve, lattice, wire, or wrap deformer that you convert as described in ”Soft bodies” on page 245. A rigid body is a polygonal or NURBS surface that you convert as described in ”Rigid bodies” on page 246. You can create a field on an object or as a stand-alone force.

To combine the influence of fields with the keyed translation of particles or soft bodies, see page 18. To combine the influence of fields with the keyed translation of rigid bodies, see page 262.

Stand-alone fields

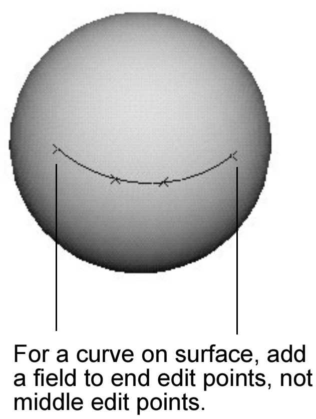
Stand-alone fields influence objects from a stationary or moving position in the workspace. A stand-alone field is not owned by geometry. It is represented by an icon in the workspace and by an independent node in the Outliner.

Object fields

Object fields are owned by an object and exert influence from the object. An object field is represented by an icon on or near the object in the workspace, and by an entry under the owning object in the Outliner. An object can own multiple fields.

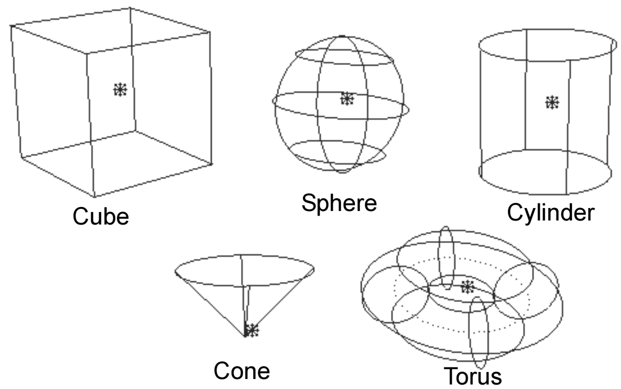
You can add fields to polygons, NURBS curves or surfaces, particle objects, lattices, or curves on surfaces. You can make the field have influence from some or all CVs, edit points, vertices, or lattice points. Alternatively, you can make the influence occur from the average position of the points.

You can add a field to all particles of a particle object, not just to some of the particles. For a curve on surface, you can add fields to the entire curve or to specific edit points, not to specific CVs. If you use edit points, add a field only to the end edit points for best results.



Volume fields

You can select a volume to define the region in space in which particles or rigid bodies are affected by any field in Maya. The volume shapes you can use are cube, sphere, cylinder, cone, and torus.



For example, if you create a Turbulence field and set its volume shape to cube, only particles or rigid bodies within the region of the cube are affected by the turbulence.

This works with all fields in Maya, as well as with the “Volume Axis field”.

How do I? Simulate dynamic effects

Work with fields

Create fields and connect objects to them

When you create a field, you typically also connect it to the objects you want the field to influence. You can alternatively connect objects to the field’s influence after creation (see Chapter 8, “Dynamic Relationships Editor”). As long as a connected object lies within the influence of the field, it will be affected by the field.

Although each field has a different effect, the technique for creating all fields is similar. For information on the effect of each field and how to adjust those effects, see ”Edit field attributes” on page 212.

The following applies to all fields:

- you can select more than one field at once
- selection order is not important
- all selected objects are affected by all selected fields.

To create a stand-alone field and connect objects to the field

1. Select the objects you want the field to affect.

   You can select geometry, particles, rigid bodies, or soft bodies.

Tip	By default, Maya converts a NURBS or polygonal surface to an active rigid body when you connect it to a field's influence. The option that controls this feature is Auto Create Rigid Body, located in Window > Settings/Preferences > Preferences > Dynamics.

2. Select Fields > *fieldname*.

   The field is created at the origin and everything you've selected is connected to it. You can move the field icon away from the origin to change the range and direction of influence. To change the field settings, see "Edit field attributes" on page 212.
3. Click the play button to see the field affect the connected objects.

To add a field to an object and connect objects to the field

1. Select Fields > *fieldname* to create the field. If it's already created, select the field.
2. Shift-select the object (or objects) to which you want to add the field.
3. Select Fields > Use Selected as Source of Field to add the field to the object.

   By default, most fields are added to all CVs or vertices of the object. The field emanates from each point with equal strength. See the field's Apply Per Vertex attribute description for more details.
4. In the Outliner, select the objects you want the field to influence and Ctrl-select or Command-select the field itself.

   The field is indented under the owning object.
5. Select Fields > Affect Selected Object.
6. Click the play button to see the field influence the connected objects.

Tips If you create a field before selecting the objects to be influenced, no objects will be connected to the field. To connect the objects to the field, use the Dynamic Relationships Editor (see Chapter 8).

The Dynamic Relationships Editor is also useful if you've added multiple fields to an object and want to connect (or disconnect) objects to a selected field or fields.

To connect to all fields owned by an object, you can simply select the objects to be influenced, move the pointer over the owning object, then right-click and select Connect Field from the pop-up menu.

To create a volume field

1. Select the objects you want the field to affect.

   You can select particles, rigid bodies, or soft bodies.
2. Select Fields *fieldname* > ❒.
3. In the field's options window, from the Volume Shape pulldown, select a shape (other than None).

   None means the field affects all the particles to which is connected.
4. Set the volume field attributes as desired. See "Set volume control attributes" on page 213.
5. Click Create.

   The implicit shape of the volume field is displayed on the screen. You can move, rotate, scale, or shear the field.

Note You can't deform the volume or use an arbitrary volume.

To add a field to selected points and connect objects to the field

You can add a field to selected CVs, edit points, vertices, or lattice points of an object in the scene (not a referenced scene).

1. Select Fields > *fieldname* to create the field. If it's already created, select the field.
2. Shift-select the desired points on the object.

   For example, select several CVs of the object. See *Basics* for details on selecting components.
3. Select Fields > Use Selected as Source of Field.

This adds the field to the selected points of the object. It also creates a set of the points named *fieldname*Set. You can edit set membership to change the field points. See *Basics*.

4. In the Outliner, select the objects you want the field to influence and Ctrl-select (Windows, Linux, and IRIX) or Command-select (Mac OS X) the field.

   The field is indented under the owning object.
5. Select Fields > Affect Selected Object.
6. Click the play button to see the field influence the connected objects.

To delete a field

1. Select the field in the Outliner or workspace.

   In the Outliner, an object field is indented under the owning object.
2. Press the Backspace (Windows, Linux, and IRIX) or Delete (Mac OS X) key.

To add additional fields to an object

1. Select Fields > *fieldname* to create the field. If it's already created, select the field.
2. In the Outliner, Ctrl-select (Windows, Linux, and IRIX) or Command-select (Mac OS X) the object to which you want to add the field.
3. Select Fields > Use Selected as Source of Field.

Edit field attributes

The type of field and its attributes specify the kind of effect it has. You can set the field attributes before you create the field, or you can create a default field and set the attributes after creation.

To set field attributes before creating the field

1. Select Fields > *fieldname* > ❐ to display the options window for the field.
2. Set the attributes.

   To enter a value that exceeds a default slider range, type the value in the text box.
3. Click the Create to create the field.

To edit field attributes after creating the field

1. Select the field you want to edit.

Tip	You can select the particle object connected to the field and the Attribute Editor displays a tab for the field.

2 Do one of the following:

- Use the Attribute Editor or Channel Box to set the attributes. The attributes are described in the field descriptions in this chapter.
- Use the workspace manipulators to edit commonly used field attributes (see ”Set field attributes with workspace manipulators” on page 215).

Tip	You can animate field effects by setting the field attribute values with keys or expressions. To turn off the effect of a field at some frame, you can key the value of its Magnitude attribute to 0. Make sure the key Tangents are Stepped if you want the Magnitude to change abruptly to 0 rather than with a gradual interpolation.

Set volume control attributes

In the Attribute Editor and the Create Emitter option window, attributes not applicable to a volume shape are dimmed. However, the Channel Box does not support dimming. We recommend you use the Attribute Editor instead of the Channel Box to edit volume field attributes until you become familiar with the attributes that apply to each volume shape.

To set volume field attributes

1 Select the field and display the Attribute Editor.

Tip	You can select the particle object connected to the field and the Attribute Editor displays a tab for the field.

2 Enter values for the volume shape attributes. xref

Tips on using volume fields

- The action of a vortex field is still controlled by its Axis attribute, even for a Torus field. If you want a circular motion around the section axis of the torus, use the Volume Axis field (see ”Volume Axis field” on page 223).

- You cannot deform or tweak the field volume. You are restricted to the volume types we provide—you can't use general volumes.
- The field Max Distance attribute still applies with volumes.
- There is no API support for volume fields in 3.0 but there is some support for the Volume Axis field.
- There are no manipulators for the field attributes.

Work with per-particle field attributes

You can manipulate field attributes on a per-particle level. This functionality works for the case where particleShape is used as the source of the field, as well as the case where the particleShape is affected by a field. In the case where the attribute is affected by both, the value from the particleShape acting as the source of the field will take precedence.

If the particleShape is acting as the source of the field, this feature allows each field to behave its own way. If the particleShape is simply affected by a field, this feature allows each particle to be affected by a unique set of field attributes.

To apply per-particle field attributes

1. Apply a field to a set of particles.
2. In the particleShape Attribute Editor, expand the Add Dynamic Attributes section, if necessary.
3. Click the General button. The Add Attribute dialog box is displayed.
4. Create a per-particle attribute of float or vector type on the particleShape with the name *fieldName_attributeLongNameOnTheField* or *fieldName_attributeShortNameOnTheField.* The type for the per-particle attribute you use must match the attribute type on the field.

   For example, create airField1_magnitude to control the magnitude on airField1.
5. Write an expression to control this attribute.
6. In the field's Attribute Editor, go to the Special Effects tab, and turn on Apply Per Vertex.

Keep particles inside the volume

You can keep particles inside a volume by using a radial field that affects only particles outside the volume you are interested in, and pushes the particles back in. You can also use this procedure to keep particles inside a volume emitter.

To keep particles inside the volume

1. Create the particles (or emitter) and the field whose volume you want the particles to stay inside.
2. Select the particles and select Fields > Radial > ❐.
3. Set the volume shape of the radial field to be the same as for the field you created in step 1.
4. Turn on Volume Exclusion.
5. Set the radial field magnitude to a negative value.

   Try a value which is about one-half the magnitude value of your first field, but is negative. For example, if the first field has magnitude of 10 or -10, use a radial field magnitude of -5.
6. Parent the radial field to the first field, and reset the radial field's transform to be the identity.

   This makes the radial field have the same volume as the original field.

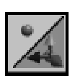
   Whenever particles move outside the original field volume, the radial field pushes them back in. You may want to tune the radial field magnitude and/or transform values to get the exact look you want.

Set field attributes with workspace manipulators

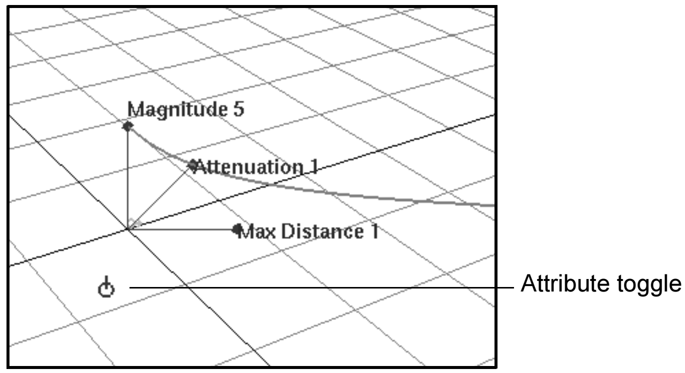
You can use manipulators in the workspace to edit the most commonly used attributes of fields. The manipulators offer an interactive alternative to typing entries in the Attribute Editor.

To use a manipulator on an attribute

1. Select the field.
2. Select the Show Manipulator Tool from the Tool Box.



   Attribute manipulators appear next to the field icon (see following figure). The manipulators show names and values for Magnitude, Attenuation, and Max Distance. Also displayed are curves and lines that graphically illustrate the values. The names and graphs always face directly out for easy visibility.



An attribute toggle also appears near the field. If necessary, dolly towards the toggle to get a clearer view. For most fields, the attribute toggle switches between the different attributes you can manipulate. (Some fields have only one group of attributes you can manipulate, so the toggle has no effect.)

3 Drag the dot next to an attribute to change its value.

4 Click the attribute toggle to display a different attribute.

Each time you click the attribute toggle, you display different attributes. The manipulators are icons you typically drag to change an attribute value. Details specific to attributes are in ”Use manipulator icons” on page 216.

The color of the manipulators and attribute toggle indicates which is active. Yellow is active, blue is inactive.

After clicking the attribute toggle a number of times, you'll see a display mode where all the manipulator icons are displayed without the attribute names. When you click an icon in this mode, the attribute name appears next to it. You can then manipulate the attribute. Click the attribute one more time and you'll see the first manipulators displayed in the cycle: Magnitude, Attenuation, and Max Distance.

Use manipulator icons

Different fields have different attributes you can manipulate. Specifically, you can work with:

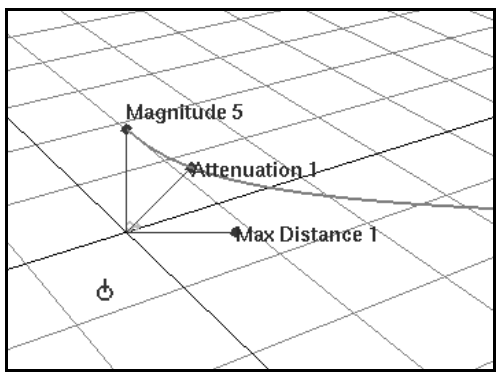
- Magnitude, Max Distance and Attenuation for all fields.
- Direction (X, Y, Z) for air, drag, gravity, and uniform fields.
- Axis (X, Y, Z) for the vortex field.
- Speed and Spread for the air field.
- Min Distance for the Newton field.
- Phase and Frequency for the turbulence field.

Most attribute manipulators work the same way. Drag the manipulator icon away from the field icon to increase the value. Drag toward the field icon to decrease the value. Attribute manipulators that require different techniques follow.

Note In some instances, the line that represents an attribute is a relative value. For example, if you drag a Magnitude value over 10, the line snaps to a fixed position. This occurs so that you can drag the value as high as desired without losing sight of the manipulator in the workspace. Some lines and curves represent actual measurements, for example, Max Distance and Attenuation.

Attenuation

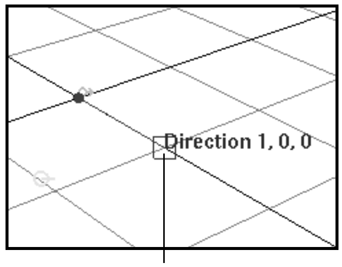
Attenuation is represented by a curve that shows how quickly the field's strength falls off from the field's position to the end of Max Distance. If Max Distance is infinite, the curve is drawn 100 units long.



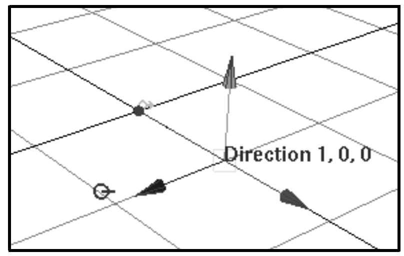
Higher attenuation values steepen the curve.

Direction X, Y, Z or Axis X, Y, Z

Click inside the blue box to display a manipulator that's the same as the Move tool. Drag the center to move in all directions, or drag one of the arrows to move with directional constraint.

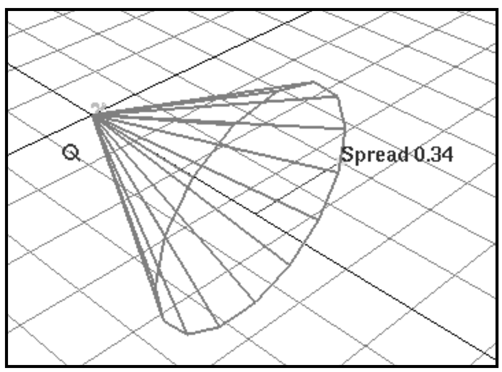


Click inside the blue box to display a Move Tool manipulator.



Air field’s Spread

Drag the dot icon in various directions until you see a cone that indicates the Spread angle. When you begin to drag the dot, a line appears in the workspace indicating the direction you can drag.



Duplicate fields

You can duplicate a an object connected to a field or an object that owns a field. When you duplicate, you get a duplicate object and field.

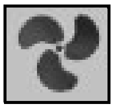
To duplicate fields and objects

1. Select the object connected to the field or the object that owns the field.
2. Select Edit > Duplicate > ❒ to display the Duplicate options window.
3. In the Duplicate Options window, turn on Duplicate Input Connections.
4. Click Duplicate.

   This duplicates the object and field. The field has the same attribute values as the original.

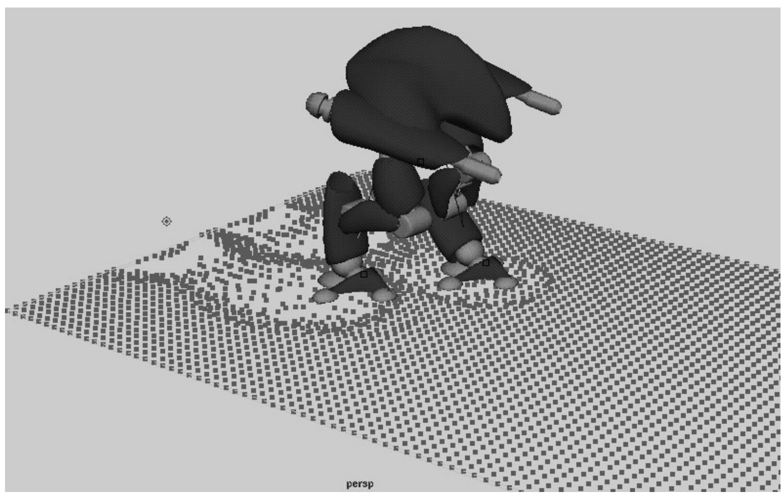
Types of fields

Air field



An air field simulates the effects of moving air. The objects you connect to the air field accelerate or decelerate so their velocities match that of the air as the animation plays.

You can parent an air field to a moving part of an object to simulate a wake of air from the moving part. For example, if you have a character walking through leaves or dust on the ground, you can parent an air field to the foot.

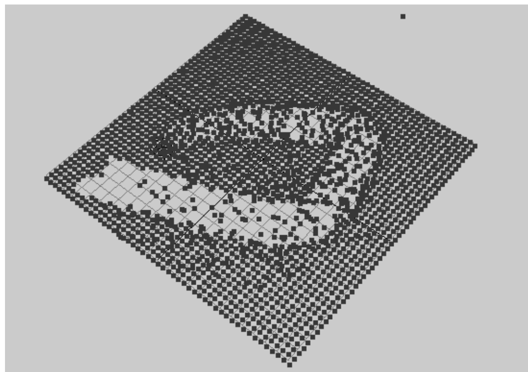


Wake air fields parented to Gunbot's feet displace particles as he walks.

Gunbot walking through particle field created by Corban Gosset

You can use the following buttons and attributes in air fields. Note that the Wind, Wake, and Fan buttons set the Air field attributes to default settings suited to each effect. If you click one of these buttons, you can still tune settings as desired.

Example: Creating an air wake from a moving object

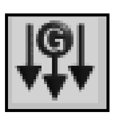


1 Create a grid of particles.
2 Create and select a sphere.
3 Keyframe the sphere to move through the grid of particles (on the XZ plane).
4 Select Fields > Air > ❒ to display the options window.
5 Click Wake.
6 Click Create.
7 Select the particle grid. Move the pointer over the sphere. Right-click the mouse and select Connect Field.
8 Play the animation.

Drag field

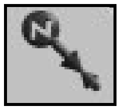
A drag field exerts a friction or braking force on an object that's animated with dynamic motion. See "Fields > Drag" on page 230.

Gravity field



A gravity field simulates the Earth's gravitational force. It accelerates objects in a fixed direction. See "Fields > Gravity" on page 232.

Newton field



A newton field pulls objects towards it. This lets you create effects such as orbiting planets or tethered, colliding balls. This field is based on the principle that a mutual attractive force exists between any two objects in the universe, proportional to the product of their masses. As the distance between the objects increases, the force of the pull decreases.

You can use a newton field to:

- attract objects to a stand-alone newton field.
- attract objects to a NURBS, polygonal, or particle object that owns a newton field.

The newton field is influenced by the mass value of the connected objects. See ”Mass values of objects” on page 221. Alos, see ”Fields > Newton” on page 233.

Mass values of objects

All objects that own or are connected to a newton field have a default mass value. A rigid body has a mass attribute of 1 by default. You can alter the value by displaying the rigid body in the Attribute Editor.

If you add a newton field to a NURBS or polygonal object and the object is not a rigid body, the object has a non-attribute mass equal to its total number of CVs or vertices. For example, a 10-CV object has a mass of 10.

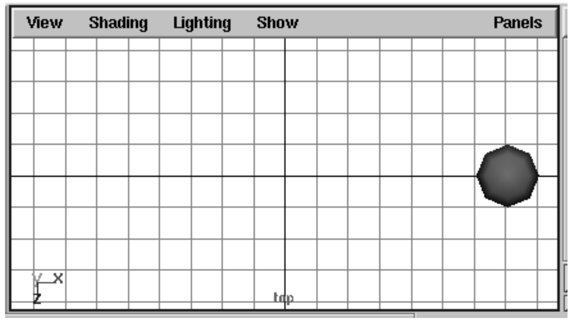
A particle object’s per particle mass attribute controls its mass value. Each particle of the object has a mass value of 1, by default.

A stand-alone newton field lacks a mass attribute but has a default mass value of 1. You cannot alter this value.

Example: Creating an orbiting object

To create an orbiting object

1. Select Window > Settings/Preferences > Preferences and click Dynamics in the Categories list. Make sure Auto Create Rigid Body is turned on.
2. Create a sphere and position it away from the origin as follows.



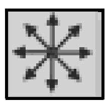
3 With the sphere selected, choose Fields > Newton > ❐.

4 Select Edit > Reset Settings to use default values, then click Create.

5 In the Attribute Editor, set the Attenuation to 0.

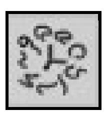
6 Select the sphere, then click the rigid body tab in the Attribute Editor. In the Initial Settings section, set Initial Velocity Z to 5.

Radial field

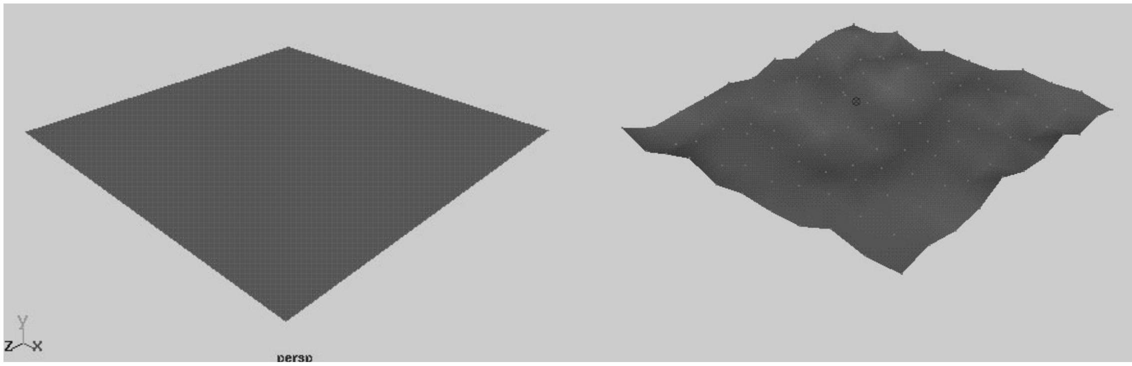


A radial field pushes objects away or pulls them toward itself, like a magnet. See ”Fields > Radial” on page 234.

Turbulence field



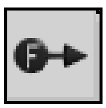
A turbulence field causes irregularities in the motion of affected objects. These irregularities are also called *noise* or *jitter*. You can combine turbulence with other fields to mimic the random motion in fluid or gaseous mediums such as water and air. See ”Fields > Turbulence” on page 236.



Soft body before being connected to turbulence field

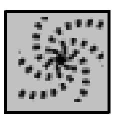
Soft body after being connected to turbulence field

Uniform field



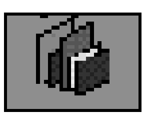
A uniform field pushes objects in a uniform direction. See "Fields > Uniform" on page 237.

Vortex field



A vortex field pulls objects in a circular or spiraling direction. You can use this field with particles to create effects such as whirlpools or tornados. See "Fields > Vortex" on page 239.

Volume Axis field



The Volume Axis field lets you move particles in various directions in a volume. The motion imparted on the object is relative to the axis of the volume.

You can use the Volume Axis field to create effects such as particles flowing around obstacles, solar flares, mushroom clouds, explosions, tornadoes, and rocket exhaust.

For example, you can specify the speed at which particles move around the central axis of a volume. If you use a a cylinder volume shape, this can create swirling gaseous effects.

The Volume Axis field provides most of the volume speed controls of the volume emitter in the form of a field, and its attributes work just like the corresponding volume emitter attributes. You can also invert the attenuation (see the Invert Attenuation attribute on page 242).

Like the volume emitter, the volume field draws arrows to illustrate the speed values you have set. These are not manipulators. As with the emitter, you can turn them off, if you wish. See "Fields > Volume Axis" on page 240.

Reference

Menus

Dynamics menu set

Fields

Fields > Air

This creates an Air field.

Fields > Air > ❐

This sets the options when you create an Air field.

Air Field Predefined Settings

Wind

Sets the Air Field attributes to default settings that approximate the effects of wind. The connected objects increase their velocity until they match 5 units per frame in an X-axis direction.

Wake

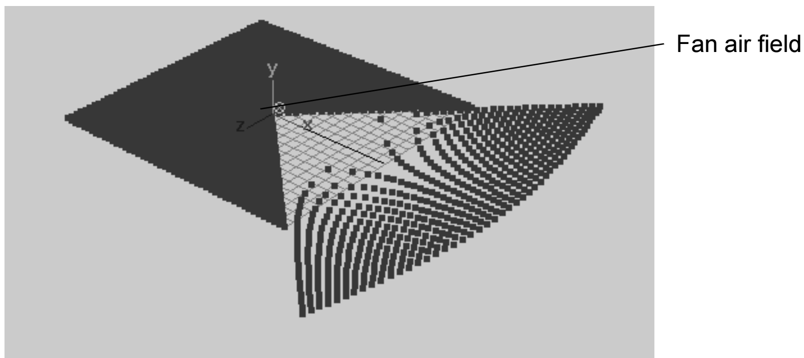
Sets the Air Field attributes to default settings that approximate the movement of air disrupted and pulled along by a moving object. The default settings for wake have no effect on the object unless the air field is moving.

You can animate the motion of the air field or the object that owns the air field, or you can parent the air field to a moving object (see ”Example: Creating an air wake from a moving object” on page 220). If you want the effect of the wake with a stationary air field, set Speed to a value greater than 0.

Fan

Sets the Air Field attributes to default settings that approximate a local fan effect. The connected objects move in a 45 degree spread along the X-axis at 5 units per frame.

By default, the fan air field creates an air pattern similar to this:



Magnitude

Sets the strength of the air field, which sets the speed along the direction the air is moving. The Magnitude and Direction X, Y, and Z attributes set the wind velocity.

The larger the number, the stronger the force. You can use a negative number to reverse the direction of the force.

Attenuation

Sets how much the strength of the field diminishes as distance to the affected object increases. The rate of change is exponential with distance; the Attenuation is the exponent. If you set Attenuation to 0, the force remains constant over distance. Negative numbers are not valid.

> Note The strength of the air field drops quickly to zero when an object approaches the Max Distance from the field. For a graphical representation of how Attenuation affects the strength of a field with distance, see ”Set field attributes with workspace manipulators” on page 215.

Direction X, Y, Z

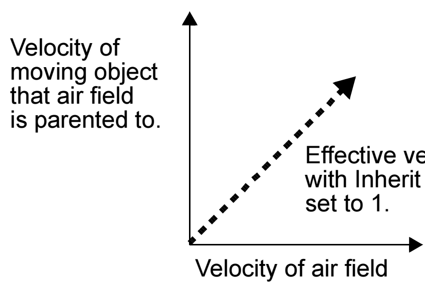
Specifies the direction the air blows.

Speed

Controls how quickly the connected objects match the velocity of the air field. If you set Speed to 0, the objects do not move. If you set Speed to 1, the objects match the velocity of the air field almost instantaneously.

Inherit Velocity (or Inherit)

When an air field is moving or parented to a moving object, Inherit Velocity specifies how much the moving air field’s velocity that is added to the Direction and Magnitude. Use a value from 0 to 1.



Inherit Rotation

If you turn on Inherit Rotation and the air field is rotating or parented to a rotating object, the air flow undergoes that same rotation. Any changes in the rotation of the air field changes the direction that the air field points. This is similar to a rotating fan or air coming from the mouth of a character whose head is moving. Turning on Inherit Rotation affects the effective velocity of the air field.

Inherit Rotation is turned on if you turn on the Wind or Fan buttons.

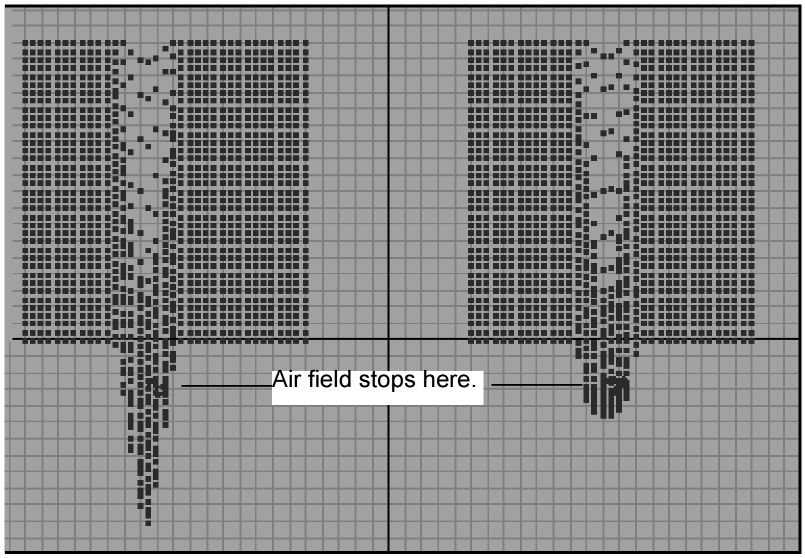
Component Only

If Component Only is off, the air field applies whatever force is necessary to make the affected object's velocity match the air field's velocity.

If Component Only is on, the air field applies force only in the direction specified by the combination of its Direction, Speed, and Inherit Velocity attributes. Also, no force is applied to slow an object along that direction; force is applied only to increase the speed.

Only objects moving slower than the air field are affected. Objects moving faster than the air field continue at that speed.

Component Only is turned on when you turn on the Wake button. The option has effect only when Inherit Velocity has a nonzero value.



Component Only On
The particles continue in the original direction after the air (wake) field stops.

Component Only Off
The particles slow down significantly after the air field stops.

Spread

Enable Spread

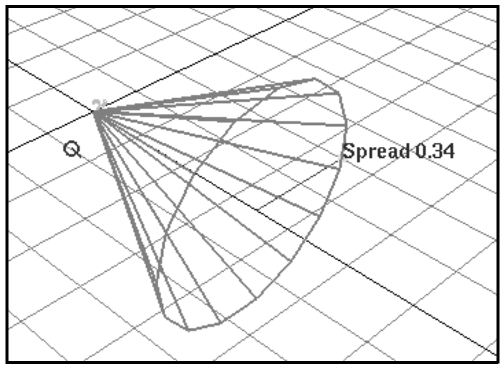
Specifies whether to use the Spread angle. If Enable Spread is turned on, only connected objects within the area specified by the Spread setting are affected by the air field. The motion is spread radially outward in a cone-like shape. If Enable Spread is off, all connected objects within the Max Distance setting are affected by the air field. The motion is uniform in direction.

If you set the Magnitude attribute to a negative value when Enable Spread is turned on, the particles are attracted toward the field center. You can use this to create an effect where the particles get “sucked in.”

Spread

Represents the angle from the Direction settings within which objects are affected by the air field. When Enable Spread is on, the air field pushes from its point of origin in a cone-like shape.

With a value of 1, any object in front of the air field are blown 180 degrees along the direction vector. With a value of 0, only objects *exactly* in front of the air field are blown along the direction vector. Values between 0 and 1 blow objects in cone-like shape along the direction vector.



Air (wind) field with
Enable Spread turned on
and Spread set to 0.34.

Tip	If you set Spread to 0 and you don't see your particles move, the particles are not exactly in front of the air field. Use a Spread greater than 0.

Distance

Use Max Distance

If you turn on Use Max Distance, connected objects within the area defined by the Max Distance setting are affected by the air field. Any connected objects outside the Max Distance are not affected by the air field.

If you turn off Use Max Distance, all connected objects are affected by the air field regardless of distance.

Max Distance

Sets the maximum distance from the air field at which the field is exerted. You must also turn on Use Max Distance for Max Distance to take effect.

> **Note** Max Distance limits a field's range of influence. However, if a field moves an object outside the range of influence, the object typically continues moving because of momentum.

Special Effects

Apply Per Vertex

Sets where the field emanates from the object. If you turn on Apply Per Vertex, each individual point (CV, particle, vertex) of the chosen object exerts the field equally at full strength. If you turn off Apply Per Vertex, the field is exerted only from the average position of the specified points.

If you are using the Attribute Editor, open the Special Effects section to display the Apply Per Vertex attribute. *Available for object fields only.*

Volume shape

Volume determines the region where the field affects particles/rigid bodies.

Volume

Choose one of None, Cube, Sphere, Cylinder, Cone or Torus.

Volume Exclusion

When Volume Exclusion is turned on, the volume defines the region in space where the field has no effect on particles or rigid bodies.

Volume Offset X, Y, Z

Offsets the volume from the location of the field. If you rotate the field, you also rotate the offset direction because it operates in local space.

> **Note** Offsetting the volume changes only the volume's location (and therefore, which particles the field affects). It does not change the actual field location for purposes of computing field force, attenuation, etc.

Volume Sweep

Defines the extent of rotation for all volumes except cubes. This can be a value from 0 to 360 degrees.

Section Radius

Defines the thickness of the solid portion of the torus, relative to the radius of the torus's central ring. The radius of the central ring is determined by the field's scale. If you scale the field, the Section Radius will maintain its proportion relative to the central ring.

Fields > Drag

This creates a Drag field.

Fields > Drag > ❐

This sets the options when you create a Drag field.

Magnitude

Sets the strength of the drag field. The greater the magnitude, the greater the braking force on the moving object.

Attenuation

Sets how much the strength of the field diminishes as distance to the affected object increases. The rate of change is exponential with distance; the Attenuation is the exponent. If you set Attenuation to 0, the force remains constant over distance. *Available in Attribute Editor only.*

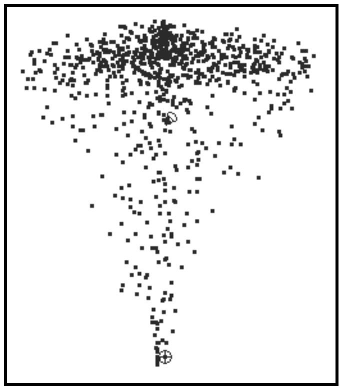
Use Direction

Specifies that the braking force is exerted only against the object's velocity that lies along Direction X, Y, Z. *Available in Attribute Editor only.*

For example, if you turn on Use Direction and your object is moving in the same direction as the drag force, the full braking force of the drag field is applied to it. If the object is moving perpendicular to the drag force, no braking force is applied. If the object is moving in the opposite direction, it speeds up.

Direction X, Y, Z

Sets the direction of the drag force's influence along the X, Y, and Z axes. You must turn on Use Direction for Direction X, Y, and Z to take effect.



Drag field on an emitter
with Use Direction on
and Y Direction set to 1.

Distance

Use Max Distance

If you turn on Use Max Distance, connected objects within the area defined by the Max Distance setting are affected by the drag field. Any connected objects outside the Max Distance are not affected by the drag field. *Available in Attribute Editor only.*

If you turn off Use Max Distance, all connected objects are affected by the drag field, no matter how far away they are from the drag field.

Max Distance

Sets the maximum distance from the drag field at which the field is exerted. You must also turn on Use Max Distance for Max Distance to take effect. *Available in Attribute Editor only.*

Volume shape

❖ See "Volume shape" on page 229.

Special Effects

Available in the Attribute Editor for object fields only.

Apply Per Vertex

Sets where the field emanates from the object. If you turn on Apply Per Vertex, each individual point (CV, particle, vertex) of the chosen object exerts the field equally at full strength. If you turn off Apply Per Vertex, the field is exerted only from the average position of the specified points.

If you are using the Attribute Editor, open the Special Effects section to display the Apply Per Vertex attribute. *Available in the Attribute Editor for object fields only.*

Fields > Gravity

This creates a Gravity field.

Fields > Gravity > ❐

This sets the options when you create a Gravity field.

Magnitude

Sets the strength of the gravity field. The greater the magnitude, the faster the objects will accelerate in the direction of the gravitational force.

Attenuation

Sets how much the strength of the field diminishes as distance to the affected object increases. The rate of change is exponential with distance; the Attenuation is the exponent. If you set Attenuation to 0, the force remains constant over distance.

Direction X, Y, Z

Sets the direction of the gravitational force.

Distance

Use Max Distance

If you turn on Use Max Distance, connected objects within the area defined by the Max Distance setting are affected by the gravity field. Any connected objects outside the Max Distance are not affected by the gravity field.

If you turn off Use Max Distance, all connected objects are affected by the gravity field, no matter how far away they are from the gravity field.

Max Distance

Sets the maximum distance from the gravity field at which the field is exerted. You must also turn on Use Max Distance for Max Distance to take effect. *Available in Attribute Editor only.*

> Note By default, the gravity field and most other fields have Max Distance set to a value of -1 and Use Max Distance turned off. The -1 value for Max Distance has no effect because the Use Max Distance setting is turned off. If you turn on Use Max Distance, the -1 value has the same effect as a value of 0—no objects are affected by the field at any distance.
>
> The -1 value is included for compatibility with previous releases of Maya. If you turn on Use Max Distance and change the value of Max Distance, you can enter a positive number only.

Volume shape

See ”Volume shape” on page 229.

Special Effects

Available in the Attribute Editor for object fields only.

Apply Per Vertex

Sets where the field emanates from the object. If you turn on Apply Per Vertex, each individual point (CV, particle, vertex) of the chosen object exerts the field equally at full strength. If you turn off Apply Per Vertex, the field is exerted only from the average position of the specified points.

If you are using the Attribute Editor, open the Special Effects section to display the Apply Per Vertex attribute.

Fields > Newton

This creates a Newton field.

Fields > Newton > ❐

This sets the options when you create a Newton field.

Magnitude

Sets the strength of the newton field. The larger the number, the stronger the force. A positive number pulls objects toward the field. A negative number pushes objects away.

Attenuation

Sets how much the strength of the field diminishes as distance to the affected object increases. The rate of change is exponential with distance; the Attenuation is the exponent. If you set Attenuation to 0, the force remains constant over distance.

Min Distance

Sets the minimum distance from the newton field at which the field is exerted.

Distance

Use Max Distance

If you turn on Use Max Distance, connected objects within the area defined by the Max Distance setting are affected by the newton field. If you turn off Use Max Distance, connected objects are affected by the newton field no matter how far away.

Max Distance

Sets the maximum distance from the newton field at which the field is exerted. You must turn on Use Max Distance for Max Distance to take effect.

Volume shape

See ”Volume shape” on page 229.

Special Effects

Available in the Attribute Editor for object fields only.

Apply Per Vertex

Sets where the field emanates from the object. If you turn on Apply Per Vertex, each individual point (CV, particle, vertex) of the chosen object exerts the field equally at full strength. If you turn off Apply Per Vertex, the field is exerted only from the average position of the specified points.

If you are using the Attribute Editor, open the Special Effects section to display the Apply Per Vertex attribute.

Fields > Radial

This creates a Radial field.

Fields > Radial > ❐

Magnitude

Sets the strength of the radial field. The larger the number, the stronger the force. A positive number pushes objects away. A negative number pulls objects toward the field.

Attenuation

Sets how much the strength of the field diminishes as distance to the affected object increases.The rate of change is exponential with distance; the Attenuation is the exponent. If you set Attenuation to 0, the force remains constant over distance. Negative numbers are not valid.

Radial Type

Specifies how the radial field's effect diminishes with Attenuation. A value of 1 causes the effect of the radial field to drop quickly to zero when an object approaches the Max Distance from the field.

A value of 0 causes the effect of the radial field to gradually approach (but never reach) 0 as it approaches the Max Distance from the field. Specifically, the force at any position is determined by the ratio of the distance and Max Distance. If you use a value between 0 and 1, Maya uses a linear blend between the two attenuation effects.

Distance

Use Max Distance

If you turn on Use Max Distance, connected objects within the area defined by the Max Distance setting are affected by the radial field.

If you turn off Use Max Distance, connected objects are affected by the radial field no matter how far away.

Max Distance

Sets the maximum distance from the radial field that the field is exerted. You must turn on Use Max Distance for Max Distance to take effect.

Volume shape

See "Volume shape" on page 229.

Special Effects

Apply Per Vertex

Sets where the field emanates from the object. If you turn on Apply Per Vertex, each individual point (CV, particle, vertex) of the chosen object exerts the field equally at full strength. If you turn off Apply Per Vertex, the field is exerted only from the average position of the specified points.

If you are using the Attribute Editor, open the Special Effects section to display the Apply Per Vertex attribute.

Fields > Turbulence

This creates a Newton field.

Fields > Turbulence > ❐

Magnitude

Sets the strength of the turbulence field. The larger the number, the stronger the force. You can use positive or negative values to move the influenced objects in random directions.

Attenuation

Sets how much the strength of the field diminishes as distance to the affected object increases. The rate of change is exponential with distance; the Attenuation is the exponent. If you set Attenuation to 0, the force remains constant over distance.

Frequency

Sets the frequency of the turbulence field. Higher values cause more frequent irregularities in the motion.

Phase X, Y, Z

Sets the phase shift of the turbulence field. This determines the direction of the disruption.

Interpolation Type

Linear specifies a linear interpolation between values in the noise table. This can create noticeable breaks along lines of force. Quadratic interpolation creates a smoother look but requires noticeably more execution time.

Noise Level

The greater the value, the more irregular the turbulence. The Noise Level attribute specifies the number of additional lookups you want done in the noise table. A value of 0 does only one lookup and is equivalent to previous versions of Maya. The total turbulence value is a weighted average of the lookups.

Noise Ratio

Specifies the weighting of successive lookups. The weights are cumulative. For example, if you set Noise Ratio to 0.5, then successive lookups are weighted 0.5, 0.25, and so on. Noise ratio has no effect if the Noise Level is set to 0.

Distance

Use Max Distance

If you turn on Use Max Distance, connected objects within the area defined by the Max Distance setting are affected by the turbulence field.

If you turn off Use Max Distance, all connected objects are affected by the turbulence field, no matter how far away they are from the turbulence field.

Max Distance

Sets the maximum distance from the turbulence field that the field is exerted. You must turn on Use Max Distance for Max Distance to take effect.

Volume shape

See "Volume shape" on page 229.

Special Effects

Available in the Attribute Editor for object fields only.

Apply Per Vertex

Sets where the field emanates from the object. If you turn on Apply Per Vertex, each individual point (CV, particle, vertex) of the chosen object exerts the field equally at full strength. If you turn off Apply Per Vertex, the field is exerted only from the average position of the specified points.

Fields > Uniform

This creates a Uniform field.

Fields > Uniform > ❐

Magnitude

Sets the strength of the uniform field. The larger the number, the stronger the force. A positive number pushes the influenced object away. A negative number pulls the object toward the field.

Attenuation

Sets how much the strength of the field diminishes as distance to the affected object increases. The rate of change is exponential with distance; the Attenuation is the exponent. If you set Attenuation to 0, the force remains constant over distance.

Direction X, Y, Z

Specifies the direction the uniform field pushes objects.

Distance

Use Max Distance

If you turn on Use Max Distance, connected objects within the area defined by the Max Distance setting are affected by the uniform field. Any connected objects outside the Max Distance are not affected by the uniform field.

If you turn off Use Max Distance, all connected objects are affected by the uniform field, no matter how far away they are from the uniform field.

Max Distance

Sets the maximum distance from the uniform field that the field is exerted. You must turn on Use Max Distance for Max Distance to take effect.

Special Effects

Apply Per Vertex

Sets where the field emanates from the object. If you turn on Apply Per Vertex, each individual point (CV, particle, vertex) of the chosen object exerts the field equally at full strength. If you turn off Apply Per Vertex, the field is exerted only from the average position of the specified points.

If you are using the Attribute Editor, open the Special Effects section to display the Apply Per Vertex attribute.

Fields > Vortex

This creates a Vortex field.

Fields > Vortex > ❐

Magnitude

Sets the strength of the vortex field. The larger the number, the stronger the force. A positive number moves the influenced object counterclockwise. A negative number moves the object clockwise.

Attenuation

Sets how much the strength of the field diminishes as distance to the affected object increases. The rate of change is exponential with distance; the Attenuation is the exponent. If you set Attenuation to 0, the force remains constant over distance.

Axis X, Y, Z

Specifies the axis around which the vortex field exerts its force.

Distance

Use Max Distance

If you turn on Use Max Distance, connected objects within the area defined by the Max Distance setting are affected by the vortex field. Any connected objects outside the Max Distance are not affected by the vortex field.

If you turn off Use Max Distance, all connected objects are affected by the vortex field, no matter how far away they are from the vortex field.

Max Distance

Sets the maximum distance from the vortex field that the field is exerted. You must turn on Use Max Distance for Max Distance to take effect.

Volume shape

See ”Volume shape” on page 229.

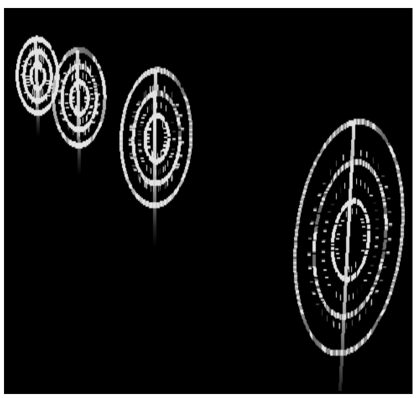
Special Effects

Apply Per Vertex

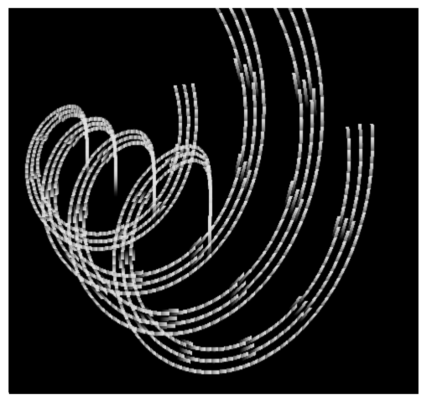
Sets where the field emanates from the object. If you turn on Apply Per Vertex, each individual point (CV, particle, vertex) of the chosen object exerts the field equally at full strength. If you turn off Apply Per Vertex, the field is exerted only from the average position of the specified points.

If you are using the Attribute Editor, open the Special Effects section to display the Apply Per Vertex attribute. *Available for object fields only.*

Tip	If you use a vortex field with a particle object, you can set the Conserve attribute of the particle object to influence of the motion. If you set Conserve to 0, the particle object moves in a circular motion. If you set Conserve greater than 0, you get a spiraling motion.



Vortex field applied to emitted particles with Conserve set to 0. The particles move in a circle.



Vortex field applied to emitted particles with Conserve set to 0.25. The particles move in a spiral.

Fields > Volume Axis

This creates a Volume Axis field.

Fields > Volume Axis > ❐

Magnitude

Specifies the strength of the volume axis field.

Attenuation

Sets how much the strength of the field diminishes from the central axis of the volume axis field.

If you set Attenuation to 1, the strength of the field diminishes in a linear manner from full strength at the central axis to zero strength at the edge of the volume axis field. Values greater than 1 make the strength of the field diminish to zero in an exponential manner—the greater the value, the more rapidly the strength decreases.

A value of 0 makes the strength remain constant from the central axis of the volume axis field to its edge (see also the "Invert Attenuation" attribute on page 242).

Distance

Max Distance

Sets the maximum distance at which the field has an effect. You must also turn on Use Max Distance for Max Distance to take effect.

Note The Max Distance and Attenuation attributes operate in a special way for the Volume Axis field. For all volume axis shapes except sphere, distance (for both the Max Distance and Attenuation attributes) is defined as the distance from the volume's central axis to the point.

For cylinder, cone, and cube, the central axis is the positive Y axis. For torus, the central axis is the ring in the center of the solid part of the torus. The sphere volume is an exception— it uses the center point, not the center axis (it works like the other fields).

This special definition of distance applies only to the Volume Axis field. It does not apply to other fields even when they are using volumes.

Volume Control Attributes

Volume Shape

Specifies the closed volume that defines the area in which the volume axis field affects particles and rigid bodies. You can choose from five volume shapes: cube, sphere, cylinder, cone, and torus.

The implicit shape of the volume axis field is displayed on the screen. You can move, rotate, scale, or shear the field.

Note	You can't deform the volume or use an arbitrary volume.

Volume Offset X, Y, Z

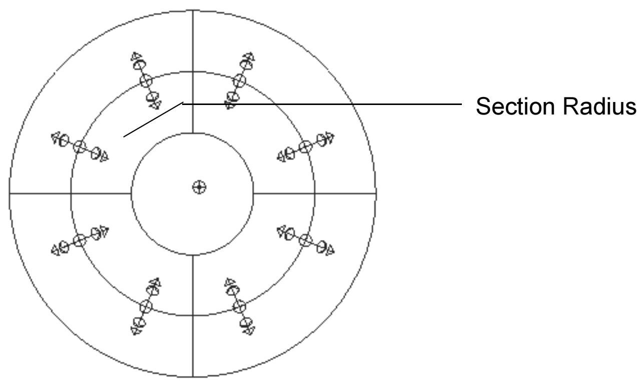
Offsets the volume axis field from the actual location of the field.

Volume Sweep

Defines the extent of the rotation for all volume shapes except cube. This can be a value from 0 to 360 degrees.

Section Radius

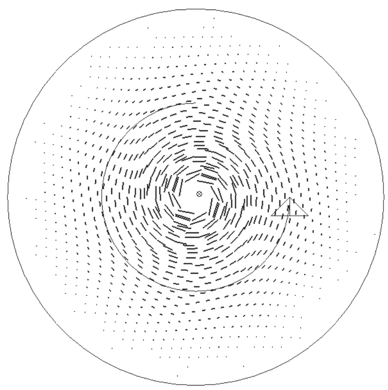
Defines the thickness of the solid portion of a torus volume shape.



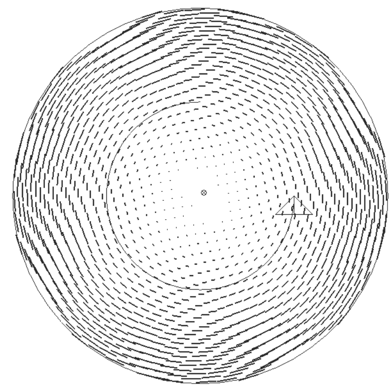
Volume Speed Attributes

Invert Attenuation

When you turn on Invert Attenuation and you have Attenuation set to a value greater than 0, the strength of the volume axis field is strongest at the edge of the volume and diminishes to 0 at the central axis of the volume axis field (see also the Attenuation attribute on page 241).



Cylinder volume axis field with the Attenuation set to 2 and Invert Attenuation turned off. Notice the particles are moving faster in the center of the cylinder.



Cylinder volume axis field with the Attenuation set to 2 and Invert Attenuation turned on. Notice the particles are moving faster along the edge of the cylinder.

Away From Center

Specifies the speed at which particles move away from the center point of cube or sphere volumes. You can use this attribute to create explosive effects.

Away From Axis

Specifies the speed at which particles move away from the central axis of cylinder, cone, or torus volumes. For torus, the central axis is the ring in the center of the solid part of the torus.

Along Axis

Specifies the speed at which particles move along the central axis of all volumes.

Around Axis

Specifies the speed at which particles move around the central axis of all volumes. When used in conjunction with a cylinder volume shape, this attribute can create swirling gaseous effects.

Directional Speed

Adds speed in the direction specified by the Direction XYZ attributes of all volumes.

Direction X, Y, Z

Moves particles in the directions specified by the X, Y, and Z axes.

Tip	When working with volumes, it’s sometimes useful to set the Conserve attribute to 0 in the particle shape. This keeps the particles within the volume. Particles will not exit the volume unless other forces act on them to make them do so. Also, see ”Keep particles inside the volume” on page 214.

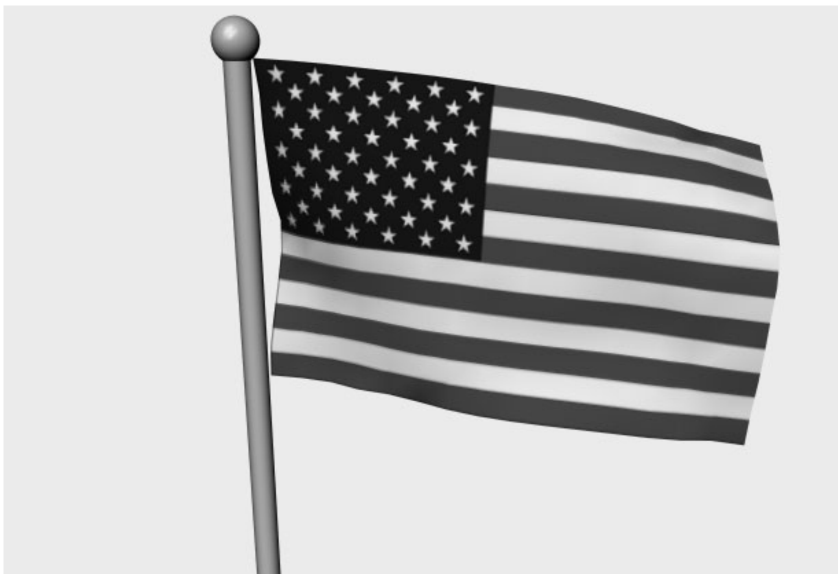
4 Soft and Rigid Bodies

About Dynamics

Soft bodies

Soft bodies

You can recreate a geometric object as a flexible object called a soft body. You can use various animation techniques to make the soft body bend, ripple, and bulge like soft objects in nature.



This flag is a soft body with the influence of turbulence, gravity, and per particle goal weights.

Soft body flag created by Rob Tesdahl

When you make a soft body from geometry or a lattice, Maya creates a corresponding particle object. The particle object is indented under the geometry in the Outliner. The combination of the geometry and particles is a soft body.

The particle object has one particle for each CV or vertex in the geometry. For a polygonal object, the particles exist at the vertices. For a NURBS object, the particles exist at the CVs and are visible in the workspace.

When a field, collision, or expression moves the particles, the corresponding CVs or vertices move in response to the movement of the counterpart particles. You do not render the particles; they exist only so you can influence the geometry with dynamics. You render the geometry (see ”Render soft bodies with motion blur” on page 252 for additional information).

Because a soft body includes a particle object, it has the same static and dynamic attributes as a particle object. You can set these attributes the same as for other particle objects.

You can make the soft body trail or move towards a goal object made from the original or duplicate geometry. Using the Paint Soft Body Weights Tool, you can set goal weights on a soft body on a per particle basis by painting on the soft body surface (see ”Paint Soft Body Weights Tool” on page 253).

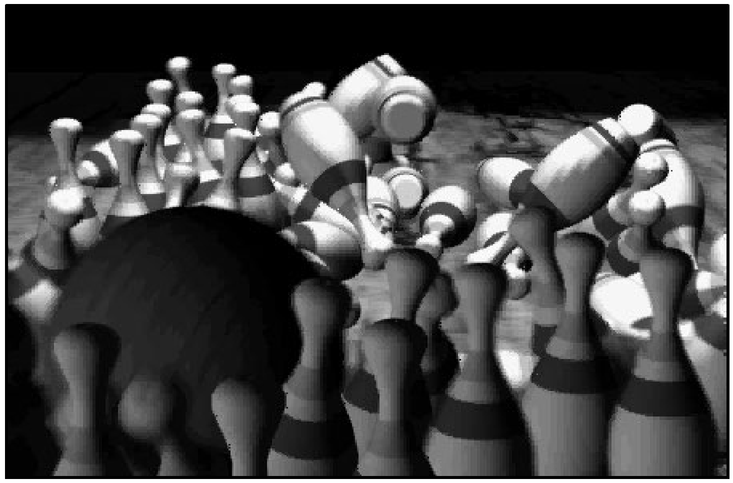
You can create springs on a soft body to alter its deformations and resilience. If you select only the soft body and create the springs, the springs exist between its particles. You can also create springs between the soft body particles and other particle objects or geometry. See ”Springs” on page 249.

Note also that you can emit particles from the soft body’s surface or particles. Select the appropriate item before you use Particles > Create Emitter. Emitting from the particles is the same as emitting from its CVs. See ”Emitters” on page 19.

Rigid bodies

Rigid bodies

A rigid body is a polygonal or NURBS surface converted to an unyielding shape. Unlike conventional surfaces, rigid bodies collide rather than pass through each other during animation. To animate rigid body motion, you use fields, keys, expressions, rigid body constraints, or collisions with particles.



Rigid body collision created by Chris Ford

Maya has two kinds of rigid bodies—active and passive. An *active* rigid body reacts to dynamics—fields, collisions, and springs—not to keys. A *passive* rigid body can have active rigid bodies collide with it. You can key its Translate and Rotate attributes, but dynamics have no effect on it.

For instance, to bounce a ball on a floor, you would make the ball an active rigid body because it needs to fall with gravity and rebound after colliding with the floor. You would make the floor a passive rigid body so it doesn't careen away from the ball when the ball bounces off it.

Dynamic animation of rigid bodies is controlled by a Maya component called a rigid body solver. Dynamic animation of rigid bodies is motion created by fields and collisions.

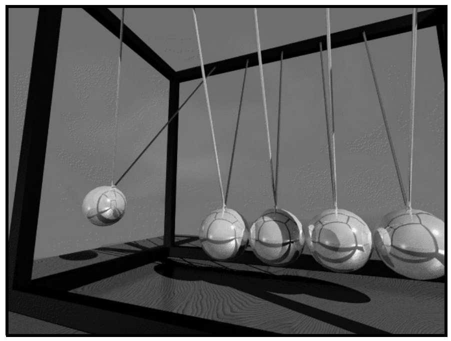
> Note
> - You can create only one rigid body per object or object group.
> - Only one side of a rigid body surface can collide—the side with normals facing outward. Error messages occur if rigid bodies penetrate each other.
> - Do not animate the scale or other deformations of a rigid body.
> - A curve cannot be a rigid body.
> - A subdivision surface cannot be a rigid body.
> - You can make a rigid body collide inside another. Place the object inside the other object, reverse the normals or surface of the outer object, then animate the inner object in an outward direction.
> - When you key a passive rigid body’s Translate and Rotate attributes, the Time Slider doesn’t show key markers. Use the Graph Editor to see the keys. You cannot remove keys on these attributes with the usual techniques. See ”Warning when you delete rigid body connections” on page 284.
> - With rigid bodies, avoid using the Edit > Duplicate menu item with Duplicate Input Connections or Duplicate Input Graph. Use the Dynamics Relationship Editor to connect the duplicate rigid body and fields.

Rigid body constraints

Rigid body constraints

Rigid body constraints restrict the motion of rigid bodies. The constraints simulate the behavior of real-world items you’re familiar with: pins, nails, barriers, hinges, and springs.

You can constrain rigid bodies to a position in your scene or to other rigid bodies. If you create a constraint on an object, Maya automatically makes the object a rigid body.



Newton's pendulum by Tristan Ikuta

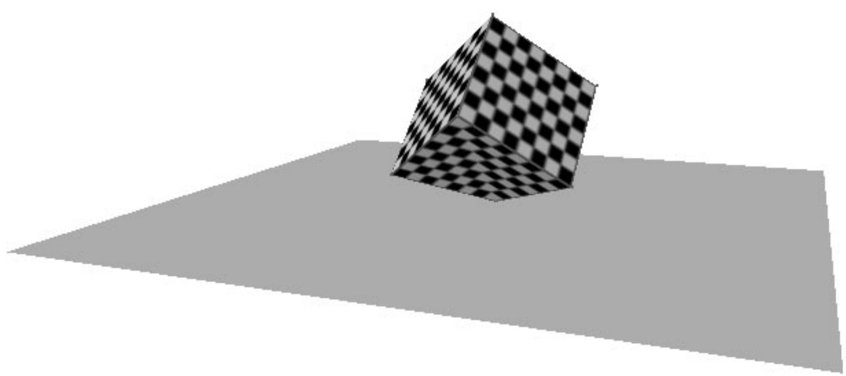
Rigid body constraints restrict the motion of the balls.

This chapter describes how to work with rigid body constraints. For details on error messages that may occur as you work with rigid body constraints, see "Fix rigid body problems" on page 282.

Springs

Springs

You can add springs to a soft body's particles to give the soft body internal structure and improve your deformation control. You can also add springs to regular particles to give them reactive, interconnected motion. The number of springs and their stiffness alters the effect of the springs.



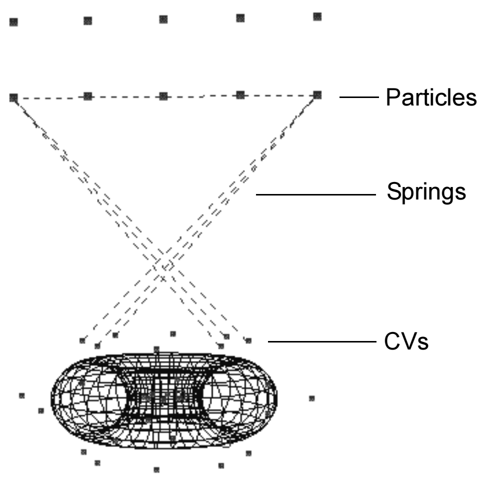
Springs on the soft body cube cause it to flex when it collides with the floor.

You can create springs between particles and:

- CVs or vertices of soft bodies

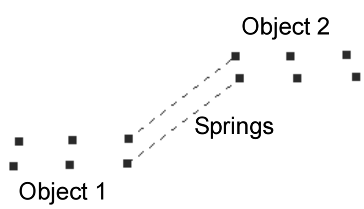
- CVs of curves or surfaces
- vertices of polygonal objects
- lattice points
- other particles

At least one of the endpoints of each spring must be a particle (or soft body particle). If one of the endpoints is not a particle, the spring does not affect the endpoint's motion, but the endpoint's motion affects the spring.

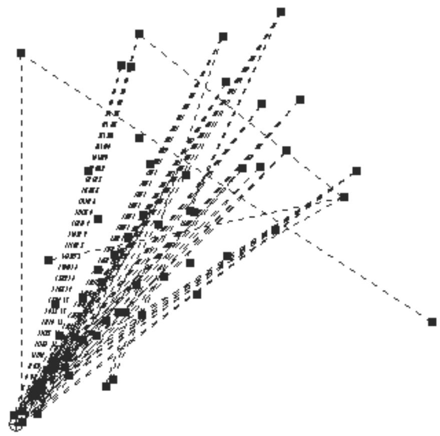


These springs connect selected particles of a particle grid to selected CVs of a torus.

You can create springs on specific object components. For example, you can select a pair of particles from different objects and create springs on just those particles.



You can also create springs on emitted particles. This provides a cohesive mass of particles, for example, a gas cloud.



The springs on these emitted particles
give them a cohesive, interactive motion.

How do I? Create dynamic effects

Create soft bodies

Create soft bodies

You can create soft bodies from:

- polygonal surfaces
- NURBS curves and surfaces, including curves for IK splines and wires
- lattices

You *cannot* make soft bodies from:

- IK skeletons
- anything in the underworld of a shape, such as a curve-on-surface
- trimmed NURBS surfaces, unless you convert them to polygons first

Tips	The more CVs or vertices the original geometry has, the more deformations that occur when you apply dynamics to the resulting soft body. For instance, applying a field to a polygonal cube with eight vertices per face causes less bulge and wiggle than one with 16 vertices per face.

To create a soft body

1. Select the object you want to make a soft body.
2. Select Soft/Rigid Bodies > Create Soft Body > ❐ to display the options window.
3. Set the options in the Soft Body Options window. See ”Soft/Rigid Bodies > Create Soft Body” on page 297.

Duplicate soft bodies

You can duplicate a soft body to create a new soft body with the same attributes as the original.

To duplicate a soft body

1. Select the soft body.
2. Select Edit > Duplicate > ❐ to display the options window.
3. In the options window, turn on Duplicate Input Graph and click the Duplicate button.

Render soft bodies with motion blur

Because a soft body includes a particle object, special circumstances arise when you render the soft bodies with motion blur. See *Rendering* for more information on rendering.

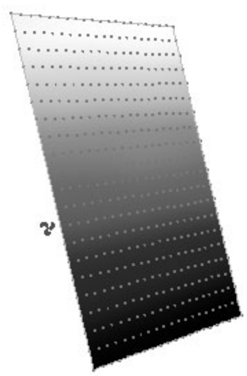
To render soft bodies with motion blur

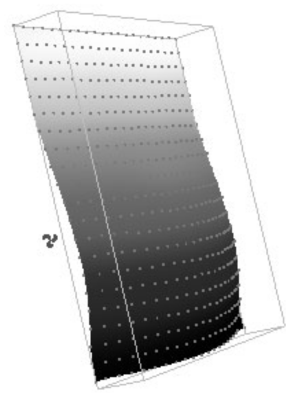
1. In the Attribute Editor for the soft body geometry, turn on Motion Blur in the Render Stats.
2. In the Render Globals Window, turn on Motion Blur.
3. Do one of the following:
   - Set the Blur by Frame in the Render Globals window to 1.
   - Select Solvers > Create Particle Disk Cache to create a particle disk cache before rendering (see ”Use particle disk caching” on page 379.

Paint Soft Body Weights Tool

Paint particle goal weights on soft bodies

Using the Paint Soft Body Weights Tool, you can set goal weights on a soft body on a per particle basis by painting on the soft body surface. The tool provides color feedback so you know which parts of the soft body have particles with different goal weights. Weights display as a range of grayscale values, with a weight of 1 displaying as white and 0 as black.





To paint goal weights on a soft body

1. Select the soft body you want to paint goal weights on. The soft body must have been created with goal weights. For information on creating soft bodies with goal weights, see the option description for "Make Non-Soft a Goal" on page 300.
2. Select the Paint Soft Body Weights Tool and open the Tool Settings editor (Soft/Rigid Bodies > Paint Soft Body Weights Tool > ❐).

   The Paint Soft Body Weights Tool automatically detects the soft body and goalPP attributes and selects the soft body for painting.

Tip	You can select the soft body for painting without opening the Tool Settings window by right-clicking the soft body and selecting the particle node from the Paint command submenu.

3. Check that Color Feedback is turned on in the Display section of the Tool Settings editor. Color feedback helps you identify the weights on the surface by representing them as grayscale values (smaller values are darker, larger values are lighter).

Tip	You can use the default hotkey Alt + c (Linux, IRIX, and Windows) or Option-c (Mac OS X) to turn Color Feedback on and off outside the Tools Settings Editor.

4 Select a brush, paint operation, and value and define other settings as required. See ee ”Soft/Rigid Bodies > Paint Soft Body Weights Tool” on page 303.

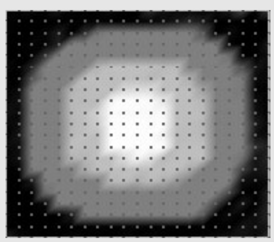
5 Drag the brush across the soft body.

Flooding soft body goal weights

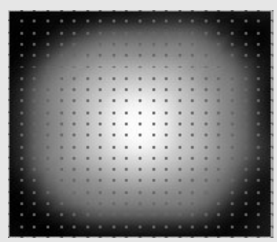
Flooding soft body goal weights is like taking a huge brush and applying its settings to all the particles on the soft body. When you flood a soft body, Maya changes the goal weight of each particle on the soft body according to the value and operation set for the Paint Soft Body Weights Tool.

To flood a soft body, follow the steps under ”To paint goal weights on a soft body” on page 253, but instead of painting in step 5, click the Flood button or use the hotkey Alt + f (Windows, IRIX, and Linux) or Option+f (Mac OS X).

Tip To smooth the transition between goal weights on a soft body, select the Smooth paint operation and flood the soft body.



Before Smooth Flood



After Smooth Flood

Map soft body goal weights

Using the Paint Soft Body Weights Tool you can map particle attribute values onto a surface. The tool applies the mapped values to the soft body particle goal weights. For details on mapping, see *Map attributes* in the Paint Effects, Artisan, and 3D Paint guide.

Paint soft body goal weight values on restricted areas

When you paint goal weights over selected vertices on a soft body, your strokes are applied only to the goal weights for the particles corresponding with the selected vertices. In effect, the unselected vertices act as a mask, where only selected vertices are affected by any weight painting or flooding you do.

Before defining the restricted area you must first create the soft body. For details on restricting the paint area surfaces, see *"Restrict an area for painting"* in the Paint Effects, Artisan, and 3D Paint guide.

Paint Soft Body Weights Tool settings

See "Soft/Rigid Bodies > Paint Soft Body Weights Tool" on page 303. For details on Brush, Stroke, Stylus Pressure, Attribute Maps, and Display settings, see *"Common Artisan Brush Tool Settings"* in the Paint Effects, Artisan, and 3D Paint guide.

Tip	You can define hotkey combinations to change most of the settings without opening the Tool Settings editor. For details on setting hotkey combinations, see *Define Artisan hotkeys* in the Paint Effects, Artisan, and 3D Paint guide.

Special uses of soft bodies

This section suggests ways to use soft bodies with other Maya features. For details on the features themselves, see the applicable documentation.

Make a skin a soft body

A skin is geometry that deforms with the movement of a skeleton. You can make the skin a soft body to cause it to jiggle with the movement of the skeleton.

To make a skin a soft body

1. Select the skin.
2. Select Soft/Rigid Bodies > Create Soft Body > ❐ to display the options window.
3. From the Creation Options menu, select Duplicate, Make Copy Soft.
4. Turn on Hide Non-Soft Object and Make Non-Soft a Goal.
5. Set Weight less than 1.

   The lower the Weight, the more the skin jiggles.

6 Click the Create button.

7 To increase jiggle in certain areas, add per particle goal weights to the soft body and decrease the goal weights in those areas. See "Setting goal weight on a per object basis" on page 108 for details.

Make a lattice a soft body

To add an overall jiggle to an object, you can add a lattice to it and make the lattice a soft body. When the lattice is a soft body, it is affected by fields. The lattice deforms the object according to the field affecting it.

By making the lattice a soft body rather than the object itself, Maya calculates dynamics faster. Also, it is easier for you to add springs to a lattice than to complex geometry.

Tip	If you are going to use the lattice as a collision object, match the size of the lattice closely to the size of the object's bounding box.

To make a lattice a soft body

1 Select the lattice.

2 Select Soft/Rigid Bodies > Create Soft Body > ❐ to display the options window.

3 From the Creation Options menu, select Make Soft.

4 Click the Create button.

5 Connect the soft body to the field you want to use to deform the object.

Make a wire a soft body

A wire is a curve that deforms an object. If you make a wire a soft body, you can connect it to fields to deform it.

To make a wire a soft body

1 Select the wire.

2 Select Soft/Rigid Bodies > Create Soft Body > ❐ to display the options window.

3 From the Creation Options menu, select Make Soft.

4 Click the Create button.

5 Connect the soft body to the field you want to use to deform the object.

Make an IK spline curve a soft body

An IK spline curve is a curve that controls a skeleton. If you make an IK spline curve a soft body, you can use fields to affect its motion.

To make an IK spline curve a soft body

1. Select the IK spline curve.
2. Select Soft/Rigid Bodies > Create Soft Body > ❐ to display the options window.
3. From the Creation Options menu, select Make Soft.
4. Click the Create button.
5. Connect the soft body to the field you want to use to deform the motion.

Make a motion path a soft body

A motion path is a curve that controls the motion of an object. If you make a motion path a soft body, you can connect it to fields to change the motion.

To make a motion path a soft body

1. Select the motion path.
2. Select Soft/Rigid Bodies > Create Soft Body > ❐ to display the options window.
3. From the Creation Options menu, select Make Soft.
4. Click the Create button.
5. Connect the soft body to the field you want to use to deform the motion.

Make a model from a soft body

If you make an object a soft body, you can use fields or expressions to deform the object, then duplicate the deformed object at a frame to create a new object.

To make a model from a soft body:

1. Select the object.
2. Select Soft/Rigid Bodies > Create Soft Body > ❐ to display the options window.
3. From the Creation Options menu, select Make Soft.
4. Use a field or expression to deform the soft body.

5 Play the animation and stop it when the soft body is deformed as desired.
6 Select the soft body.
7 Select Edit > Duplicate > ❐ to display the options window.
8 In the options window, turn off Duplicate Input Graph and turn off Duplicate Input Connections, then click the Duplicate button.

   The copy is a snapshot of the soft body at the current frame.
9 In the Outliner, delete the particle object indented under the copy.

   You can continue to adjust the field or expression to make additional copies.

Use attributes for advanced applications

The following soft body attributes are for API developers and other advanced users:

- Enforce Count From History
- Input Geom Space
- Target Geom Space

The attributes are available in the Soft Body Attributes section of the Attribute Editor when you select the particle object of the soft body.

To emit a soft body into view

1 Set the Lifespan Mode to Constant and the lifespan of the emitted particles to 0.
2 Play the animation to remove the particles and soft body geometry.
3 Select the particles and choose Solvers > Initial State > Set For Selected.
4 Turn off Enforce Count From History.
5 Set the lifespan of the particles to some positive value so they don’t die immediately.

   When Maya updates the number of particles, any springs that you previously applied to the particles will no longer work as before. You’ll need to create new springs. If you wrote an expression that uses the value of the particle object’s particleId attribute, you’ll need to rewrite the expression.

   To avoid unexpected results, do not animate the original geometry’s construction history in a way that alters the number of CVs, vertices, or lattice points over time. You can edit, but not animate, the construction history.

   Enforce Count From History is turned on by default.

Note	In the following explanations, the term *world space transformation* refers to the values Maya uses internally to calculate an object's translate, rotate, scale, and shear in world space. The term *points* refers to CVs, vertices, or lattice points.

Create rigid bodies

Create rigid bodies

You can create a passive or active rigid body from an object or hierarchy of objects. To have multiple objects react as one rigid body, you must create a rigid body from a hierarchy of objects. By default, Maya automatically makes an object an active rigid body when you connect it to a field.

To create a rigid body from a single object

1. Select the object.
2. Do one of the following:
   - Select Soft/Rigid Bodies > Create Active Rigid Body.
   - Select Soft/Rigid Bodies > Create Passive Rigid Body.

     A rigid body is created with default settings. See "Edit attributes of a rigid body" on page 260.

To create a rigid body from several objects

1. Group the NURBS or polygonal objects under a parent node.

   For instance, select the objects and select Edit > Group.
2. Select the group node.
3. Do one of the following:
   - Select Soft/Rigid Bodies > Create Active Rigid Body.
   - Select Soft/Rigid Bodies > Create Passive Rigid Body.

     A single rigid body is created for the group with default settings. See "Edit attributes of a rigid body" on page 260.

To turn off automatic creation of rigid bodies

1. Select Window > Settings/Preferences > Preferences.
2. Click Dynamics in the Categories list.

3 Turn off Auto Create Rigid Body.

To delete a rigid body

1 Select the rigid body (or parent group node).
2 Select Edit > Delete by Type > Rigid Bodies.

To delete all rigid bodies in a scene

- Select Edit > Delete All by Type > Rigid Bodies.

Create rigid bodies from trimmed surfaces

If you've trimmed a NURBS surface and made it a rigid body, unwanted collisions might occur on the trimmed part. To prevent this, you can either convert the NURBS surface to polygons or delete the construction history before making it a rigid body. For information on how to convert NURBS to polygons, see the *Polygon Modeling* guide.

Create rigid bodies from objects with deformer, skin or flexor

If you create a rigid body from an object that has a deformer, skin, or flexor applied to it, Maya applies the rigid body to the original, undeformed object that is hidden when you apply the deformer. When the rigid body collides, it might seem to do so at the wrong time or wrong part of its surface.

For example, suppose you animate an active rigid body sphere bulging from a lattice deformation. The bulging area penetrates rather than collides with a nearby passive rigid body cube. You won't see a collision until the original, undeformed part of the sphere moves against the cube.

Edit attributes of a rigid body

You can set rigid body attributes before or after you create the rigid body.

To set rigid body options before you create the emitter

1 Select Soft/Rigid Bodies > Create Active Rigid Body > ❐ or Create Passive Rigid Body > ❐ to display the options window.

The same options window is displayed for both menu selections.

2 Set the attributes described in the following pages.
3 Select the objects you want to make rigid and click the Create button.

To edit attributes after you create the rigid body

1 Select the rigid body.

To select the rigid body in the workspace, select the x-shaped icon that exists near the surface. To select the rigid body in the Outliner, select Display > Shapes in the Outliner, open the surface node, and select the indented rigid body node.

2 Open the Attribute Editor and set the attributes.

Notes	For many attributes, you can enter values that exceed the default slider range by typing the value in the text box. If you change attributes, rewind to ensure correct animation.

Edit attributes of a rigid body solver

Dynamic animation of rigid bodies and rigid body constraints is controlled by a Maya component called a rigid body solver. Dynamic animation of rigid bodies refers to motion created by fields and collisions. You can change attributes of the rigid body solver to adjust aspects of the dynamic animation.

To edit the rigid body solver attributes

1 Select Solvers > Rigid Body Solver.

Note	If you’ve created multiple solvers, first select the solver from the list in Solvers > Current Rigid Solver >. For details on multiple solvers, see ”Segregate collisions with multiple solvers” on page 265.

The Attribute Editor is displayed with the rigid body solver attributes.

2 Set the Rigid Solver attributes.

Control complex motion and forces

The following sections describe techniques for controlling complex motion and forces, including,

- ”Keying impulses to rigid bodies” on page 261
- ”Combine rigid body dynamics and keys” on page 262
- ”Get data on velocity, forces, and collisions” on page 264

Keying impulses to rigid bodies

You can key an impulse on and off to instantaneously change a rigid body’s velocity, for instance, when a bat hits a baseball.

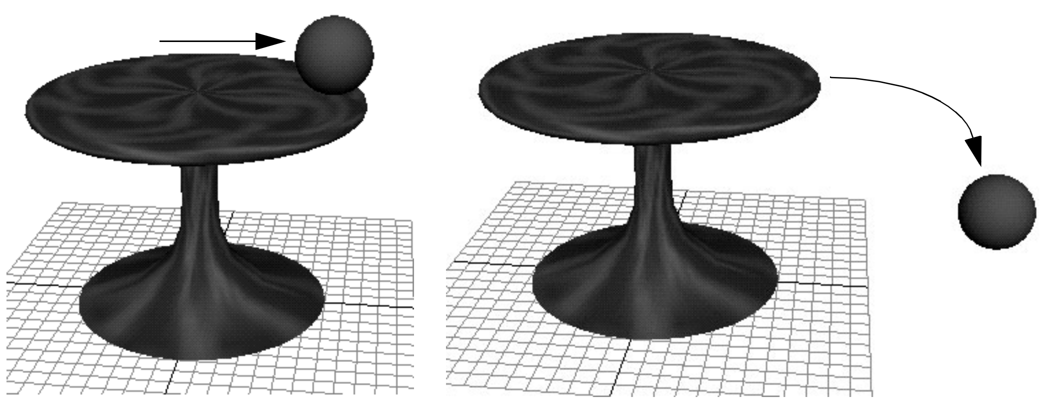
To key the impulse

1 Select the rigid body.
2 At the first frame of the animation, use the Attribute Editor to key the Impulse X, Y, Z and Spin Impulse X, Y, Z to 0, 0, 0.
3 Go to the frame where you want the impulse.
4 Key new values for the Impulse X, Y, Z and Spin Impulse X, Y, Z.
5 Go to the frame after the one where you keyed the attributes.
6 Key Impulse X, Y, Z and Spin Impulse X, Y, Z to 0, 0, 0.
7 Set the tangents of all keys to stepped in the Graph Editor or Time Slider.

   For example, select a keyframe in the Time Slider, select the attributes in the Channel Box, then right-click the Time Slider and select Tangents > Stepped. Repeat this for each keyframe.

Combine rigid body dynamics and keys

When a rigid body is passive, you can animate it by keying its Translate and Rotate attributes. When the rigid body is active, you can animate it with dynamics, specifically fields and collisions. Sometimes you'll want to use a combination of keys and dynamics. For instance, you can roll a ball off a table with keys then cause the ball to fall with gravity.



Motion from keys

Motion from keys plus gravity. The ball retains the rolling motion it had immediately before control switched from keys to dynamics.

To use dynamics after keys

1 Select the object.

2 Key the object's Translate and Rotate attributes to give the object motion.

3 At the first frame of keyed motion, select Soft/Rigid Bodies > Set Passive Key.

This keys the Active attribute to the off setting and keys the object's current Translate and Rotate attribute values.

4 At the first frame where you want the dynamics to have control, select Soft/Rigid Bodies > Set Active Key.

This keys the Active attribute to the on setting and keys the object's current Translate and Rotate attribute values.

5 Apply the dynamics to the object.

At a subsequent frame, you can switch control from dynamics to Translate and Rotate keys by selecting Soft/Rigid Bodies > Set Passive Key.

To use keys after dynamics

1 Select the object.

2 Make the object a rigid body and use dynamics to control its motion.

3 At the first frame where dynamics control the motion, select Soft/Rigid Bodies > Set Active Key.

This keys the Active attribute to the on setting and keys the object's current Translate and Rotate attribute values.

4 At the first frame where you want the Translate and Rotate keys to have control, select Soft/Rigid Bodies > Set Passive Key.

This keys the Active attribute to the off setting and keys the object's current Translate and Rotate attribute values.

5 Set keys to control the motion.

At a subsequent frame, you can switch control from Translate and Rotate keys to dynamics by selecting Soft/Rigid Bodies > Set Active Key.

Tip When you animate rigid body motion with dynamics after keys, the velocity that exists immediately before the switch to dynamics is added to the dynamic motion. When you animate rigid body motion with dynamics followed by keys, the rigid body adopts the keyed position immediately when you switch to keys. The prior dynamic velocity is not added to the keyed motion.

Get data on velocity, forces, and collisions

A rigid body has read-only attributes that contain its current velocity and any forces and collisions applied to it. This section describes these attributes.

Force, velocity, spin, and torque attributes

When you select a rigid body, the Attribute Editor shows the current X, Y, and Z component values for Velocity, Spin, Force, and Torque. You can also query these attributes in MEL commands or expressions by using the attribute names in the following table.

Example runtime expression

```
print (velocityX + "\n");
```

This displays the contents of velocityX in the Script Editor each frame.

Contact attributes

A rigid body has read-only attributes that contain data on the contact made with it during collisions. You can query the attributes in MEL commands or expressions by using the attribute names in the table that follows.

To accumulate data in these attributes, select the solver that controls the rigid body, display the Rigid Solver States section of the Attribute Editor, then turn on Contact Data attribute. The attribute values are zeroed out each time you turn on Contact Data. Turning on Contact Data slows playback.

Example runtime expression

```
print (rigidBody1.contactCount +"\n");
```

This displays the number of contacts each frame.

Convert rigid body animation to keys

You can convert an active rigid body's dynamic animation to keys, a process known as baking the simulation. This is useful if you want to adjust an active rigid body's motion at certain frames by altering keyed values.

To convert an active rigid body's dynamic animation to keys

1. Create the rigid body's dynamic animation.
2. Select the active rigid bodies for which you want to bake the simulation. If a rigid body is a hierarchy of objects made into a single rigid body, select the parent transform node of the hierarchy.

3 Select Edit > Keys > Bake Simulation > ❐ to display the options window.

4 If you are baking rigid bodies that exist in a hierarchy, turn on the Below setting for Hierarchy. If the rigid bodies are not in a hierarchy, turn on Selected.

5 For the Channels setting, turn on From Channel Box.

This creates keys only for the attributes you select in the Channel Box. You'll select the attributes in a subsequent step.

6 For Time Range, turn on one of the following:

- Time Slider to convert the entire frame range.
- Start/End to specify a Start Time and End Time.

7 For the Sample By option, enter the frame increment by which you want to create keys.

For example, if you leave it set to the default setting of 1, a key is created at every frame.

8 Turn on Disable Implicit Control.

9 Select the Translate and Rotate attributes in the Channel Box.

This selects the Translate and Rotate attributes of all selected rigid bodies.

10 Click Bake.

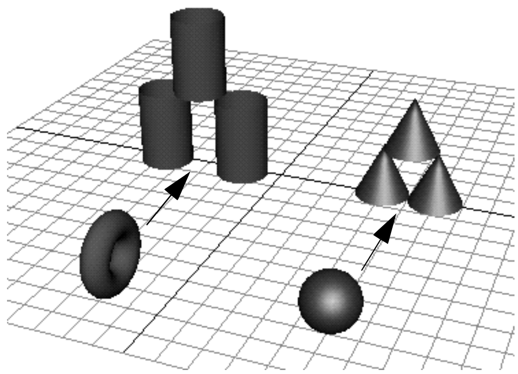
Maya creates the keys.

Tips Use the Graph Editor to simplify the animation curves created when you bake the simulation. See *Animation* for details. Simplifying the curves reduces the number of keys, which eases your ability to edit the animation. If you simplify the curve too much, you might alter the motion of the rigid body.

If you bake the simulation for a rigid body, you no longer need the object's rigid body node. You can remove such nodes to speed scene processing.

Segregate collisions with multiple solvers

You can create multiple solvers, one for each collection of colliding objects. By using the same solver only for objects that collide with each other, each rigid body solver avoids tracking the presence of unnecessary objects and therefore uses less processing time.



The ball and cones collide together, but they don't collide with the torus and cylinders. Processing time is saved by using two solvers, one for the ball and cones, and one for the torus and cylinders.

When you create the first rigid body in a scene, Maya creates a rigid body solver that controls that rigid body and all others you create thereafter unless you create a different solver. If you create a different solver, it controls rigid bodies you create thereafter. Each solver has its own independent attributes that affect the behavior of the rigid bodies it controls. Note that you can connect a field to any rigid bodies, regardless of the solver.

To create a new solver

- Select Solvers > Create Rigid Body Solver.

  The Solver receives a default name such as rigidSolver1. The name appears at the bottom of the list displayed when you select Solvers > Current Rigid Solver > *rigidSolver#*.

To assign a rigid body to a solver

1. Select the solver from the list in Solvers > Current Rigid Solver > *rigidSolver#*.
2. Create the rigid body.

   The rigid body is controlled by the solver selected in the prior step. For example, if you select rigidSolver3 then make a sphere an active rigid body, the sphere is controlled by rigidSolver3.

To move rigid bodies to a different solver

1. Select the rigid bodies.
2. In the Command Line or Script Editor, enter this command:

```
rigidBody -edit -solver solverName
```

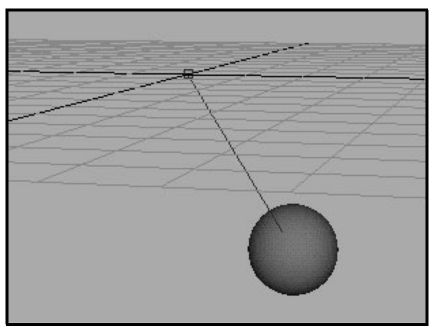
where *solverName* is the name of the solver to be used by the rigid bodies.

Work with rigid body constraints

Create a Nail constraint

A Nail constraint nails a single active rigid body to a position in the workspace. It works as if a solid bar connects the rigid body to the constraint position. You cannot use a Nail constraint on a passive rigid body.

If you animate an active rigid body with a field such as gravity, you can use a Nail constraint to create effects such as a ball swinging from a cord.



To create a Nail constraint

1. Select the active rigid body you want to constrain.
2. Select Soft/Rigid Bodies > Create Constraint > ❐ to display the options window.
3. For the Constraint Type, select Nail.
4. Turn on Set Initial Position and enter X, Y, and Z values for the nail position.

   If you don't turn on Set Initial Position, Maya puts the constraint at the rigid body's center of mass.
5. Click Create in the Constraint Options window.

   To alter the constraint after creation, see "Edit constraints" on page 273.

Create a Pin constraint

A Pin constraint links two rigid bodies at a specified position. It works as if the two objects are connected by a metal pin with a ball joint between its ends. You can use a Pin constraint to create effects such as a link in a chain or a robotic arm. You can pin together two active rigid bodies or an active and a passive rigid body

To create a Pin constraint

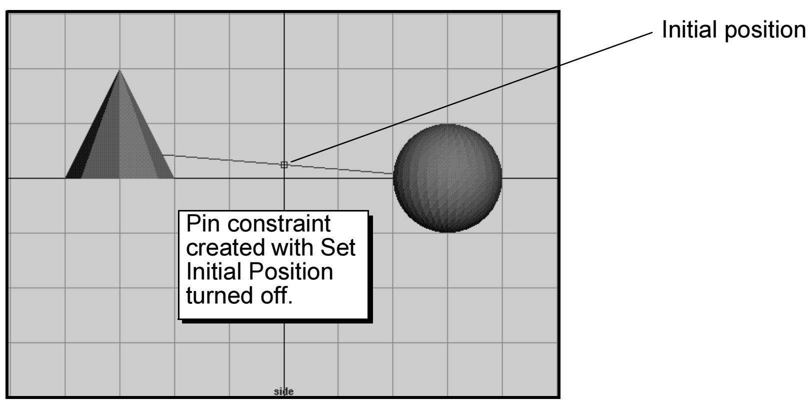
1. Select the two rigid bodies you want to constrain.
2. Select Soft/Rigid Bodies > Create Constraint > ❐ to display the options window.
3. For the Constraint Type, select Pin.
4. If you want the rigid bodies to penetrate each other rather than collide upon contact, turn on Interpenetrate.

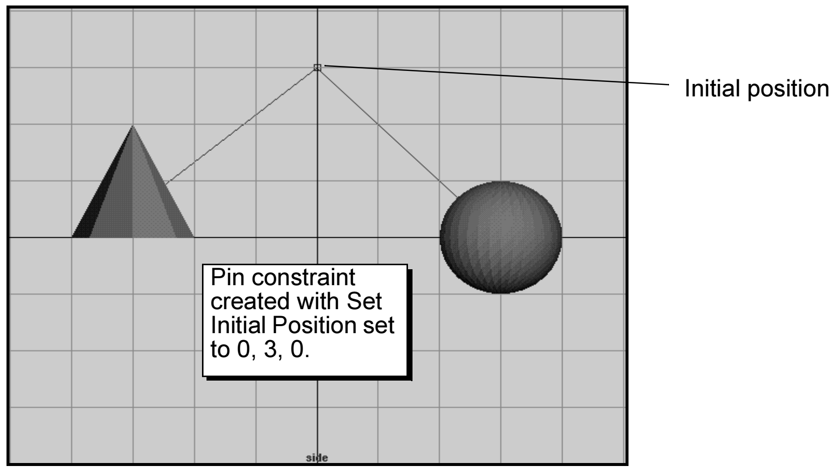
If you turn on Interpenetrate, animation playback is faster.

5. If you want to specify the constraint's initial position, turn on Set Initial Position and enter X, Y, and Z values for the position.

   If you don't turn on Set Initial Position, a Pin constraint is created at the midpoint between the two rigid bodies. It extends from one rigid body's center of mass to the other's center of mass.
6. Click Create in the Constraint Options window.

   To alter the constraint after creation, see "Edit constraints" on page 273.





Create a Hinge constraint

A Hinge constraint constrains rigid bodies along a specified axis by a hinge. You can use a Hinge constraint to create effects such as a door on a hinge, a link connecting train cars, or a pendulum of a clock. You can create a hinge constraint between:

- one active or passive rigid body and a position in the workspace.
- two active rigid bodies.
- an active and passive rigid body.

The Hinge constraint is able to react to forces. Its axis rotates when torque is applied, which gives it a realistic look when your hinge is flying through the air or bouncing against other rigid bodies.

To create a Hinge constraint

1. Select the one or two rigid bodies you want to constrain.
2. Select Soft/Rigid Bodies > Create Constraint > ❐ to display the options window.
3. For the Constraint Type, select Hinge.
4. If you are constraining two rigid bodies and want them to penetrate each other rather than collide upon contact, turn on Interpenetrate.

   If you turn on Interpenetrate, playback of rigid body animation is faster.
5. Turn on Set Initial Position and enter X, Y, and Z values for the position.

If you don’t turn on Set Initial Position when you constrain a single rigid body to a position, the Hinge constraint is created at the rigid body’s center of mass.

If you don’t turn on Set Initial Position when you constrain a pair of rigid bodies, the constraint is created at the midpoint between their centers of mass.

Though a passive rigid body has a center of mass that Maya uses for default positioning of Hinge constraints, Maya doesn’t use the center of mass in dynamic calculations of passive rigid bodies.

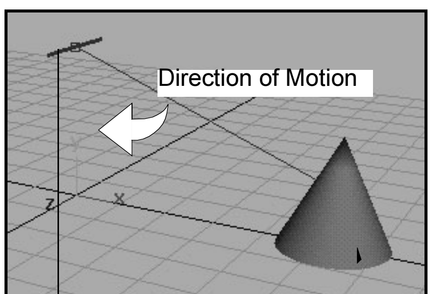
Tip	After you create the constraint, you can use the Move and Rotate tools to change its position and orientation conveniently in the workspace.

6 To set the initial orientation of the constraint axis, enter X, Y, and Z values in degrees for the Initial Orientation.

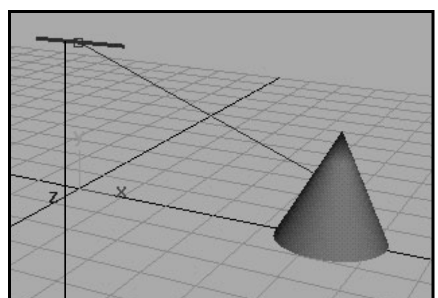
By default, the hinge constraint lies parallel to the world space Z-axis.

7 Click Create in the Constraint Options window.

To alter the constraint after creation, see ”Edit constraints” on page 273.



The Initial Orientation of this Hinge constraint is along the X-axis. This orients the hinge parallel to the direction of the motion of the rigid body. The rigid body does not swing.



The Initial Orientation of this Hinge constraint is along the Z-axis. The rigid body can swing like a pendulum around the Z-axis.

Tip	You can make a directional hinge, which always maintains the direction of its axis. This is how the original hinge constraint worked prior to Maya 4.5. After you create the hinge constraint, set the Constraint type to directionalHinge in the Attribute Editor or Channel Box.

Create a Spring constraint

A Spring constraint simulates an elastic cord. You can use a Spring constraint to create effects such as a man bungee-jumping off a building. You can create a spring constraint between:

- one active or passive rigid body and a position in the workspace.
- two active rigid bodies.
- an active and passive rigid body.

To create a Spring constraint

1. Select the one or two rigid bodies you want to constrain.
2. Select Soft/Rigid Bodies > Create Constraint > ❐ to display the options window.
3. For the Constraint Type, select Spring.
4. If you are constraining two rigid bodies and want them to penetrate each other rather than collide upon contact, turn on Interpenetrate.

   If you turn on Interpenetrate, playback of rigid body animation is faster.
5. If you're using a Spring constraint on a single rigid body, turn on Set Initial Position and enter X, Y, and Z values for the constraint's position.

   If you don't turn on Set Initial Position when you constrain a single rigid body to a position, the Spring is created at the rigid body's center of mass.

   If you create a Spring constraint for two rigid bodies, the constraint will extend from one center of mass to the other. The Set Initial Position option is invalid for Spring constraints between two rigid bodies.

   Though a passive rigid body has a center of mass that Maya uses for default positioning of Spring constraints, Maya doesn't use the center of mass in dynamic calculations of passive rigid bodies.
6. To set the attributes of the Spring, open the Spring Attributes section of the Constraint Options window.
7. Click Create in the Constraint Options window.

   To alter the constraint after creation, see "Edit constraints" on page 273.

Create a Barrier constraint

A Barrier constraint creates an infinite barrier plane beyond which the center of mass of the rigid body will not move. You can use a Barrier constraint to create objects that block other objects, for instance, a wall or floor.

You can save processing time by using this type of constraint in place of a collision effect. However, objects will deflect but not bounce off the plane. To constrain multiple objects, you must create a Barrier constraint for each object.

This constraint works only with a single active rigid body; it does not constrain a passive rigid body.

To create a Barrier constraint

1. Select the rigid body you want to constrain.
2. Select Soft/Rigid Bodies > Create Constraint > ❐ to display the options window.
3. For the Constraint Type, select Barrier.
4. Turn on Set Initial Position and enter X, Y, and Z values for the Initial Position of any position on the plane. The position is irrelevant because the Barrier is an infinite plane.

   If you don’t turn on Set Initial Position, the Barrier plane is created at the origin and connected to the center of mass of the selected rigid body. The barrier plane is oriented on the XZ plane.
5. If you want to set the initial orientation of the barrier plane, enter X, Y, and Z values (in degrees, by default) for Initial Orientation.

For more information, search the online help for By default, the Barrier constraint is oriented on the XZ plane.

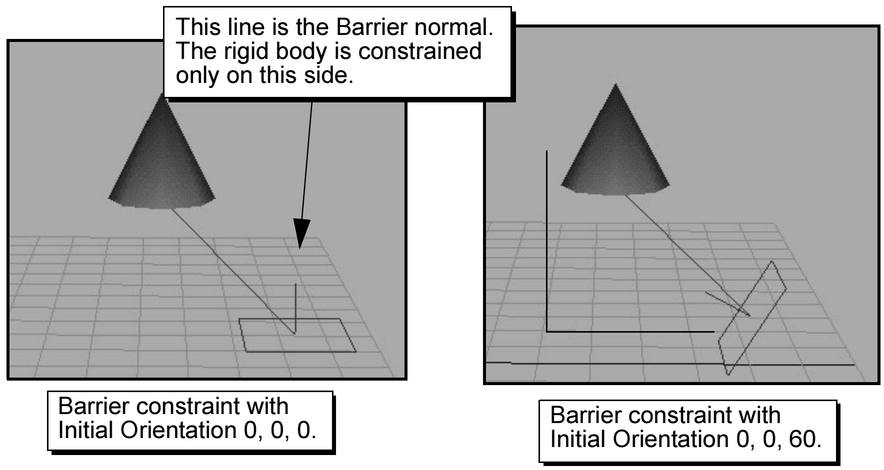
Tip	After you create the constraint, you can use the Move and Rotate tools to change its position and orientation conveniently in the workspace.

6. Click Create in the Constraint Options window.

   An square icon appears in the workspace at the Barrier plane position.

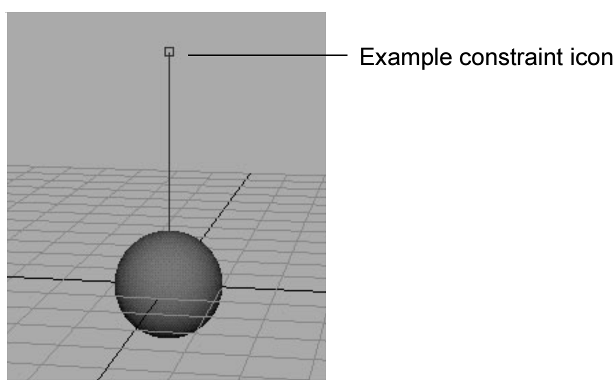
   A perpendicular line sticking up from the Barrier plane icon indicates the normal direction of the plane surface. Only this side of the surface constrains the rigid body.

   To alter the constraint after creation, see ”Edit constraints” on page 273.



Edit constraints

After you create a constraint, Maya creates a constraint named for the type of constraint. For example, rigidNailConstraint1 is the default name given to a Nail constraint. To adjust a constraint's attributes in the Attribute Editor, you must first select the constraint in the Outliner or the constraint icon in the workspace.



Note The Rigid Constraint Attributes section of the Attribute Editor displays read-only Force and Solver Id attributes. The Force shows the X, Y, and Z components of the current force pulling towards one or both objects connected to the constraint. The Solver Id is the constraint index number the solver uses to calculate dynamics for this constraint. You can query the value of these attributes in expressions and MEL commands.

Adjust the initial position and orientation

You can adjust the initial position of all rigid body constraints. You can adjust the initial orientation of only Hinge and Barrier constraints.

To adjust a constraint's position

- Do one of the following:
  - Use the Move tool to drag the constraint icon.
  - In the Rigid Constraint Attributes section of the Attribute Editor, enter X, Y, and Z values for Initial Position.

Tip	If you move a constraint in a frame other than the starting frame, rewind to make the constraint animate properly.

To adjust a Hinge or Barrier constraint's initial orientation:

- Do one of the following
  - Use the Rotate tool to rotate the initial orientation icon.
  - In the Rigid Constraint Attributes section of the Attribute Editor, enter X, Y, and Z values for Initial Orientation.

Change or turning off the constraint type

You can change the constraint type or turn a constraint off or on after you create it. Before you change the constraint type, be aware of these issues:

- You cannot change a single-body constraint to a dual-body constraint.
- If you change the constraint type, the constraint name doesn't change.
- A Nail or Barrier constraint cannot constrain a passive rigid body.
- You cannot constrain two passive rigid bodies to each other.

To change the constraint type

- In the Rigid Constraint Attributes section of the Attribute Editor, select the constraint type from the Constraint Type menu.

To turn a constraint off (or on)

- In the Constraint Control section of the Attribute Editor, set these attributes:

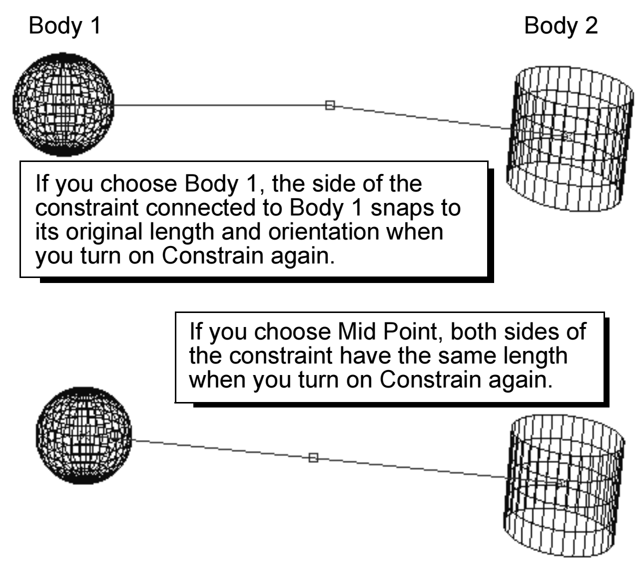
Constrain

Turns a constraint on or off.

Relative To

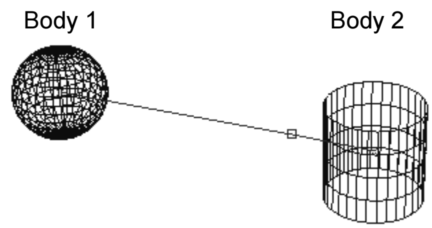
For a constraint between a pair of rigid bodies, this lets you choose where Maya recreates a constraint after you turn it on again. Your choices follow:

Body 1	Recreates the constraint at the original distance and orientation from the first of the two rigid bodies you selected when you created the constraint.
Body 2	Recreates the constraint at the original distance and orientation from the second of the two rigid bodies you selected when you created the constraint.
Mid Point	Recreates the constraint at the midpoint between the two rigid bodies.
User Defined	Lets you specify the position in the User Defined Position X, Y, and Z boxes. These attributes are dim unless you turn on User Defined.



Example

Suppose you create a pin constraint between two objects. The pin is positioned as follows. The small square represents the pin's location.



Before making the pin constraint, the sphere was selected first and the cylinder second. In the context of the Relative To attribute, the sphere is therefore Body 1 and the cylinder is Body 2.

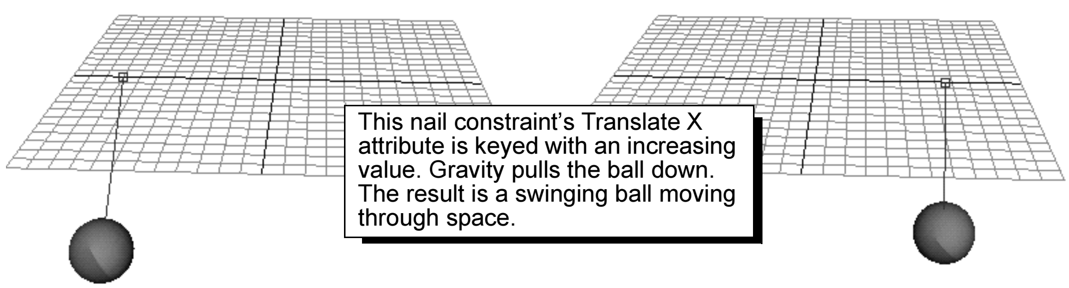
You then animate the movement of the cylinder in a direction perpendicular to the constraint line icon. The constraint forces the sphere to swing around the cylinder. If you key Constrain off for several frames, the sphere is no longer constrained to the cylinder. It moves in its own direction using the force previously applied to it by the constraint.

When you key Constrain on again in a later frame, the sphere will once again be tied to the cylinder. You can set the location and orientation of the pin by setting the Relative To attribute. This influences how the objects interact with each other's movement.

Key and parent constraints

You can parent the following constraints or key their positions:

- nail
- spring (single-body only)
- hinge (single-body only)



Create springs

Create springs

The following procedures describe how to create springs within or between objects, and how to create springs with emitted particles.

To create springs within an object or between two objects

1. Select the objects or components to which you want to add springs.
2. Select Soft/Rigid Bodies > Create Springs > ❐ to display the Spring options window.
3. Set options as described on ”Set spring attributes” on page 278.
4. Click Create.

Tips	If you use springs on a polygonal soft body or a NURBS soft body with many vertices (and particles), you can improve playback efficiency and spring response by putting springs on a simplified proxy object. See page 289. To cancel spring creation after you click Create, press the Esc key. Springs will still be created, but only those that were created before you cancelled the process.

To create springs on emitted particles

1. Play the animation until you see the emitted particles.
2. Select the emitted particle object.

   You can optionally select specific particles to receive the springs.
3. Select Soft/Rigid Bodies > Create Springs > ❐ to display the Spring options window.
4. Set options as described on ”Set spring attributes” on page 278.
5. Click Create.

   When you rewind, the particles and springs disappear. When you play the animation, the springs appear with the particles and influence their motion. Springs won’t be created on any particles that were not visible when you created the springs.

   You can edit the emitter after creating the springs. The springs appear wherever the particles are emitted.

Edit spring operation

The techniques for editing spring operation are described in these topics:

- ”Set spring attributes” on page 278
- ”Add or remove springs after creation” on page 279
- ”Edit per spring rest length, stiffness, and damping” on page 281

Set spring attributes

You can set all spring attributes in the Spring options window before you create the spring. You can change many settings in the Attribute Editor after you create the spring.

Settings available in the Attribute Editor

To use the Attribute Editor, select the spring object that’s created when you create the springs. The following attributes are also available in the Spring options window.

Per-spring Stiffness,
Per-spring Damping,
Per-spring Rest Length

Lets you set stiffness, damping, and rest length for individual springs. See ”Edit per spring rest length, stiffness, and damping” on page 281.

If you use the Attribute Editor to change the setting of Per-spring Stiffness, Per-spring Damping, and Per-spring Rest Length from off to on *after* you create springs, Maya uses the Stiffness, Damping, and Rest Length attribute values that apply to all springs in the spring object.

Stiffness

Sets the rigidity of the springs. If you increase the stiffness of springs too much, the springs might stretch and contract excessively (see ”Fix playback problems” on page 286).

Note that you can adjust the bounciness of a colliding soft body by changing the stiffness.

Damping

Mutes the spring action. A high value causes spring length to change more slowly. A low value causes the length to change quickly. You can use a low Damping value and a high Stiffness value to create a jiggle effect.

Rest Length

Sets the length for the springs when they are at rest. Every spring in the spring object is set to the same Rest Length value.

The springs try to reach their rest length when you play the scene. If you set it to a value smaller than the length of the spring, the spring contracts when you play the scene. If you set it to a value that is larger than the length of the spring, the spring expands.

End1 Weight

Sets the amount of the spring's force applied to the spring's starting point. You can enter a value between 0 and 1. A value of 0 means the point is not affected by the spring forces; a value of 1 means the point is fully affected.

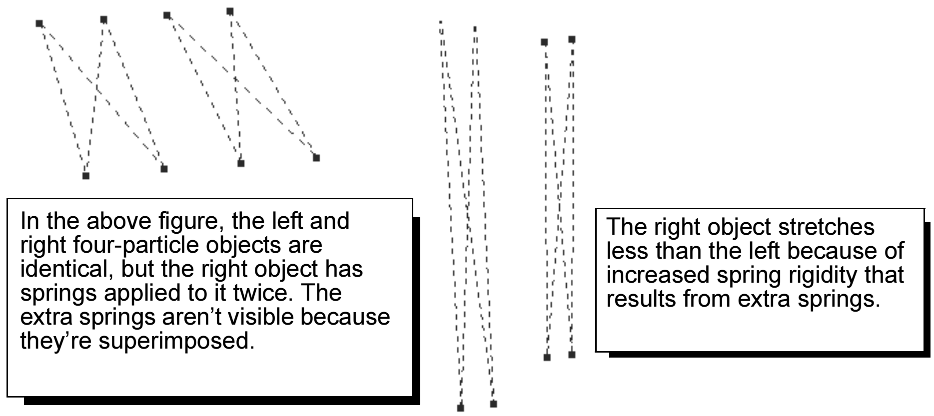
End2 Weight

Sets the amount of the spring's force applied to the spring's ending point. You can enter a value between 0 and 1. A value of 0 means the point is not affected by the spring forces; a value of 1 means the point is fully affected.

Note The order of particle creation establishes which spring points are the starting and ending points. Because it's hard to know which points are the starting and ending points for a particle object with many particles, End Weights are easiest to use with springs between small numbers of particles.

Add or remove springs after creation

After you create springs on an object, you can add or remove selected springs to tune the spring effect. This is useful if you need to add a few more springs to an object between points previously without springs. You can also add two springs to the same pair of points to increase rigidity. A single stiff spring has an attractive force that differs from two springs set less stiffly.



To add more springs to an object

1. Select the objects or components to receive the additional springs.
2. If the associated spring object isn't already selected, Shift-select it.
3. In the Spring Option window, turn on Add to Existing Spring, then click the Create button.

To remove selected springs from an object

1. Turn on Select by Component Type: Lines, preferably with all line types other than Springs turned off.
2. In the workspace, select the springs you want to remove.
3. Press the Backspace (Linux, IRIX, and Windows) or Delete (Mac OS X) Backspace key.

To remove all springs from an object

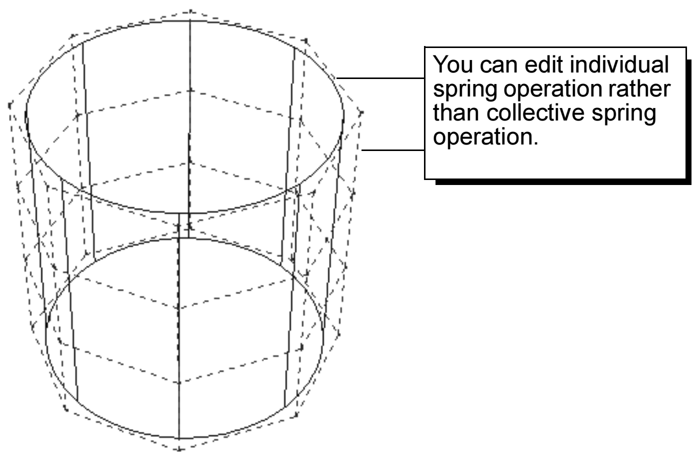
1. Turn on Select by Object Type mode, then select the spring object in the workspace.

   Be careful not to select the particle object or geometry with the spring object. You can alternatively select the spring object in the Outliner.
2. Press the Backspace (Linux, IRIX, and Windows) or Delete (Mac OS X) Backspace key.

Tips	If you've animated spring attributes and don't want to lose the animation, simply remove all selected springs on the object and add new ones. The new ones receive the prior attribute values.

Edit per spring rest length, stiffness, and damping

If you turn on Per-spring Stiffness, Per-spring Damping, or Per-spring Rest Length in the spring object, you can edit stiffness, damping, and rest length for individual springs.



Note	You do not add per-spring attributes, like you add per-particle attributes to particles. Per-spring Stiffness, Per-spring Damping, or Per-spring Rest Length are built- in and those are the only ones you can edit.

To edit the per spring attributes

1. Select the spring object.
2. Click (the Select by Component Type icon), then limit selection to springs. See *Basics* for details.
3. Select the individual springs that have the attribute values you want to change.
4. Select Window > General Editors > Component Editor.

   The Springs tab of the Component Editor displays rest length, stiffness and damping values for the selected springs. The attribute names end with PS, which indicates you can set the attributes on a per spring basis.
5. Set new values for the desired springs.

   See *Basics* for more details on the Component Editor.

What went wrong?

Dynamics

Fix rigid body problems

As you work with rigid bodies and rigid body constraints, you might see warning or error messages in response to your actions. Common messages and solutions follow.

Errors when you create a rigid body

```
Error: Attempting to create a rigid body from the shape or
hierarchy 'object' which already contains a rigid body.
```

This message means you tried to create a rigid body from an object or object group that already has a rigid body created for it. You can create only one rigid body for an object or object group.

Delete the rigid body of the object and recreate it.

```
Warning: The surface of rigid body 'rigidBody1' may be
reversed. Reversing the surface may avoid interpenetration
errors.
```

This message occurs if the rigid body solver calculates that the surface of a rigid body is facing the wrong way. Only one side of an rigid body's surface can collide with another rigid body's surface—the side with normals pointing outward.

Reverse the NURBS surface or polygonal object's normals.

```
Error: The rigid body cannot be created because the
selected shape contains polygons which do not have unique
vertices. Use the polygon tools to determine bad polygons.
```

This message means you tried to create a rigid body from an object that has at least one polygon with two or more coincident vertices. For example, the object might contain a triangle with four vertices.

Use polygonal modeling tools to fix the polygon.

```
Warning: Selected item "rigidBody1" is not deformable.
```

This message occurs when you create a rigid body from an object that has a deformer, skin, or flexor applied to it. Because rigid bodies cannot deform, Maya applies the rigid body to the original, undeformed object that is hidden when you apply the deformer to it. The resulting rigid body might seem to collide at the wrong time.

For example, suppose you animate an active rigid body sphere bulging from a lattice deformation. The bulging area penetrates rather than collides with a nearby passive rigid body cube. You won't see a collision until the original, undeformed part of the sphere moves against the cube.

None. This message is for your awareness only.

Warning when you set the center of mass

```
 Warning: The center of mass is locked. To edit the center
of mass, set the 'lock center of mass' attribute to false.
```

If you turn on an object's Lock Center of Mass attribute and try to change the Center of Mass, the above message occurs.

If you want to change the Center of Mass, select the rigid body, display the Attribute Editor, and turn off Lock Center of Mass.

Error when rigid bodies fail to collide

```
 Warning: Rigid Body interpenetration occurred between
'rigidBody1' and 'rigidBody2'.
```

Only one side of an rigid body's surface can collide with another rigid body's surface—the side with normals pointing outward. Interpenetration occurs when two or more rigid bodies pass through each other. This typically occurs when one body passes through the wrong side of another.

Though it's valid and often desirable for one passive rigid body to penetrate another, it's invalid for an active rigid body to penetrate another rigid body, whether passive or active.

If you're unsure which objects interpenetrate, look in the Outliner. Maya selects and highlights the objects after the warning message occurs.

- Make sure no rigid bodies overlap at rewind.
- Reverse the NURBS surface or polygonal normals of one of the interpenetrating rigid bodies. If this doesn't fix the problem, undo the reversal and try reversing the other object's surface or normals.

Warning when you cache rigid bodies

```
 Warning: The rigid body is currently cached. Changing an
attribute will have no affect until the cache is manually
deleted via the Solvers->SceneCaching->Delete menu.
```

This message occurs if you've selected Solvers > Memory Caching > Enable for a rigid body, then you change attributes on the rigid body that affect the rigid body's translation or rotation. Because the rigid body is cached, it gets the translation and rotation information for the rigid body from the cache. You won't see any effect on the rigid body until you delete the cache.

Select the rigid body and select Solvers > Memory Caching > Delete.

Error when a solver computes rigid body dynamics

```
 Error: Error detected in rigid solver computation. Solution
may be invalid.
```

An invalid internal computation occurred, possibly because of conflicting rigid body constraint forces. This message typically occurs with a message warning of an incorrectly positioned constraint.

If rigid body constraints are forcing a rigid body to interpenetrate with another rigid body, reposition the constraints or rigid bodies.

Warning when you delete rigid body connections

```
 Warning: Cannot disconnect: 'rigidBody1.choice' from
'rigidBody_tx.selector'. To force the disconnection, set the
'Allow Disconnection' attribute in the attribute editor.
```

In the dependency graph, a rigid body node controls the transform node of the rigid body object with connections through several choice nodes. The choice nodes let you key passive rigid body Translate and Rotate attributes so you can switch from keys to dynamic influences and vice versa.

Maya usually lets you break the connections to a transform node's Translate and Rotate attributes, for example, by selecting the attribute in the Channel Box then selecting Channels > Break Connections. For rigid body transform nodes, breaking such connections would cause the rigid body to lose control of the transform node. Because of this, Maya prevents you from breaking the connections in the usual way.

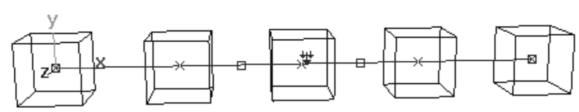
- To remove the keyframe connection to the rigid body, select Soft/ Rigid Bodies > Break Rigid Body Connections. This removes any connection to the rigid body not specifically used by the rigid body, for instance, connections to keyframes.

To break the connections from the rigid body to the choice nodes or rigid body solver, you must first select the rigid body or rigid body solver, display the Attribute Editor, and turn on Allow Disconnection. Breaking these connections might cause unexpected results.

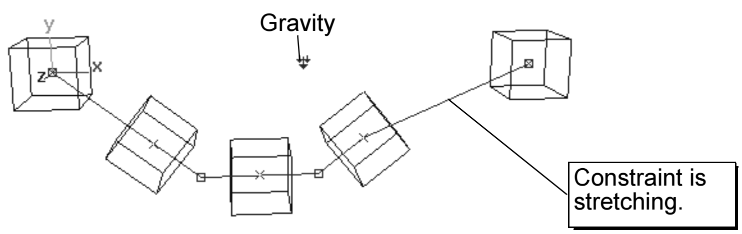
Fixing problems with constraints

When you use multiple constraints on several rigid bodies, you might create conflicting forces on the rigid bodies. The conflicting forces generate unexpected results. A conflicting force is a force applied perpendicular to constraints such that the correct solution would be for the constraints to stay exactly where they started.

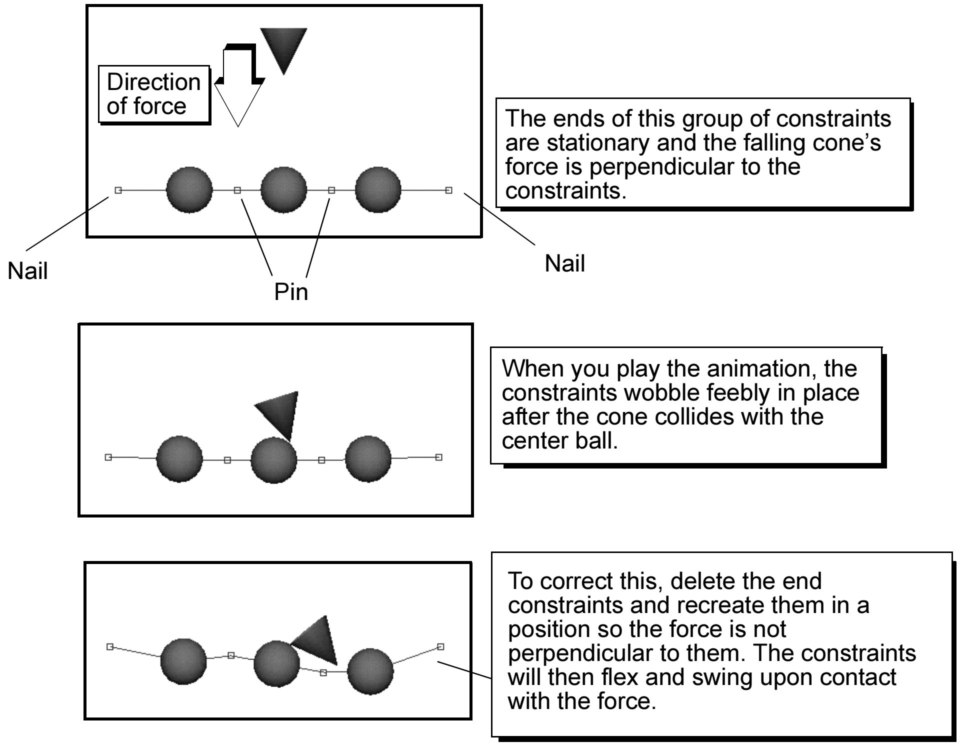
For example, suppose you use a sequence of horizontally-aligned constraints and the end constraints are stationary (see the following figure).



If you apply gravity or if another rigid body collides with it, a perpendicular force is applied to the constraints. The constraints may stretch or wobble feebly.



To avoid this situation, delete the end constraints and recreate them so they are *not* perpendicular to the direction of the force.



Fix playback problems

This section explains solutions to a few problems you might have when using springs:

The springs expand and contract excessively

You can often fix excessive spring expansion and contraction by increasing oversampling of dynamic calculations. Increasing the oversampling slows down playback but increases the stability of the springs. Do these steps to increase oversampling:

1. Select Solvers > Edit Oversampling or Cache Settings.

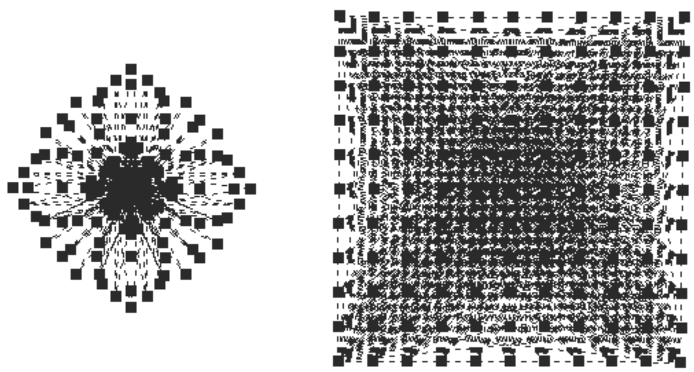
   This displays the Attribute Editor.
2. Enter a value for Over Samples.

Tip Increasing the oversampling intensifies the attractive force of goals. If you increase the oversampling in a scene that contains goals, you can compensate by decreasing the goal weights, if necessary.

In some instances, Maya uses an Over Samples rate that is higher than the value you enter. This is necessary for internal computations.

Springs don't move the particles, CVs, vertices, or points

Either no force is acting on the particles or the Per-spring Rest Length is turned on but you didn't set any values for it. By default, each spring's rest length equals its original length. It won't move unless an external force is applied to it.



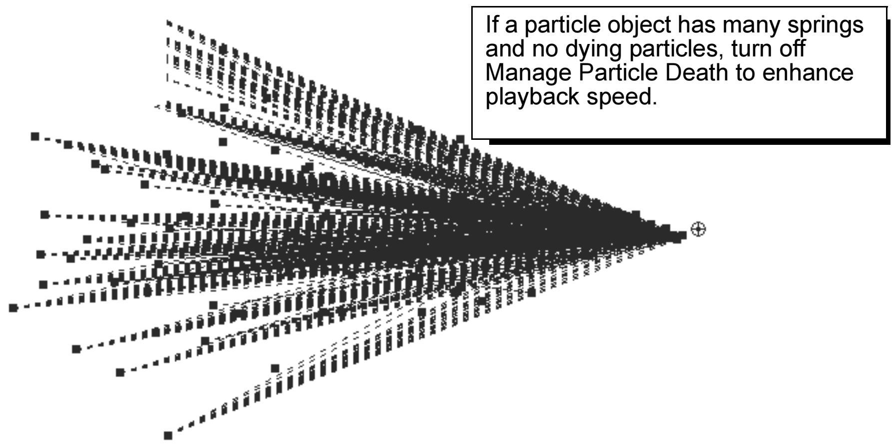
A pair of identical particle grids with springs are shown here after several frames of play. The particles on the right fail to move because Per-spring Rest Length is turned on while the default values are in use.

To make the springs take effect, do the appropriate action that follows:

1. Animate the object, for example, with a field.
2. Use the Component Editor to alter Per-spring Rest Length. See "Edit per spring rest length, stiffness, and damping" on page 281.
3. Change to per object rest length.
4. If the springs are on particles and you broke the default connection from time to the particle shape node's Current Time attribute, reconnect the input to Current Time. A simple way to reconnect the time input is to select the particle object, select the Current Time attribute in the Channel Box, then select Particles > Connect to Time.

An animation with many springs plays slowly

If the age of any or all particles exceeds the lifespan, Maya removes the particles from the object. For brevity, we describe such particles as dying particles. If a scene has a particle object with many springs and no dying particles, the animation plays needlessly slow because of complex calculations. You can turn off the attribute Manage Particle Death to eliminate the calculations and enhance the playback efficiency.



In short, it's helpful to turn off Manage Particle Death for the spring object associated with the following objects if they have undying particles:

- soft bodies
- objects made with the Particle Tool
- emitted particles

Do not turn off Manage Particle Death if any particles in the object die. Springs attached to the particles won't work correctly. This is true if particles have *ever* died in the object. If you set current attribute values of the particles for initial state operation at a point where no more particles die, you still must not turn off Manage Particle Death.

To turn off Manage Particle Death, select the spring object and display the Attribute Editor. Turn off the Manage Particle Death attribute in the Spring Attributes section.

You edit per spring attributes but the editing doesn't take effect

You might have turned off Per-spring Stiffness, Per-spring Damping, or Per-spring Rest Length, which means Maya uses the per object counterpart attribute instead. Turn on the appropriate Use attribute.

A polygonal or NURBS soft body with springs slows playback and has unrealistic spring behavior

If you put springs on a polygonal soft body or a NURBS soft body with many vertices (and particles), your animation will play slowly and the springs might contract and expand unrealistically. To avoid this problem, put springs on a low-resolution copy of the object as follows:

1. Duplicate the polygonal (or NURBS object) with default options.

   Note that copy must exist at the same locations as the original for the entire animation.

2. Decrease the surface resolution of the object to roughly 10-50% of its original resolution.

   For a polygonal object, decrease the Subdivisions X and Y values. For a NURBS object, decrease the Spans and Sections values.

   In the remaining steps, the original object is called the low-resolution object, and the copy is called the high-resolution object.

3. Make the low-resolution object a soft body with default options.
4. Select the soft body’s particles and connect them to the desired dynamics.

   For example, connect the particles to gravity and make them collide with another object.

5. Create springs on the soft body particles with default options.
6. Tune the appearance of the soft body animation.

   You’ll likely need to set spring Stiffness to over 100. You’ll also likely need to select Solvers > Oversampling or Cache Settings and set Over Samples to between 2 and 4. Use the lowest value that gives a desirable appearance to the soft body deformations.

7. Using the Outliner, select the transform node of the high-resolution object, thenCtrl-select (Linux, IRIX, and Windows) or Command-select (Mac OS X) the low-resolution object.
8. From the Animation menu set, select Deform > Create Wrap.

   This forces the high-resolution object to deform the same as the low-resolution object when you play the animation.

9. Hide the low-resolution object.

Reference Menus

Dynamics menu set

Soft/Rigid Bodies

Soft/Rigid Bodies > Create Active Rigid Body

Creates an active rigid body.

Soft/Rigid Bodies > Create Active Rigid Body > ❐

Sets the options when creating an active rigid body. Here are the options:

Rigid Body Name

Lets you name the rigid body for easy identification.

Active

Makes the rigid body an active rigid body. If off, the rigid body is passive.

Particle Collision

If you've made particles collide with the surface and the surface is an active rigid body, you can turn Particle Collision on or off to set whether the rigid body reacts to the collision force.

Allow Disconnection

By default, you cannot break connections from the rigid body to the rigid body solver that handles its dynamic animation. You can turn on Allow Disconnection so that you can break the connections. See "Warning when you delete rigid body connections" on page 284.
Available in Attribute Editor only.

Mass

Sets the mass of an active rigid body. The greater the mass, the greater the influence on objects it collides with. Maya ignores the mass attribute of passive rigid bodies.

Note	The Rigid options window includes checkboxes for Set Center of Mass, Set Initial Position, and Set Initial Velocity. These checkboxes make the X, Y, and Z settings for Center of Mass, Initial Position, and Initial Velocity available (or unavailable) for editing. The Set Center of Mass applies only to active rigid bodies.

Center of Mass X, Y, Z

Specifies the position of an active rigid body's center of mass in local space coordinates. An x-shaped icon represents the center of mass. It's easiest to see in wireframe mode.

The center of mass affects how an active rigid body bounces. For example, suppose you place a sphere's center of mass below and to the side of the sphere's surface. If you make the sphere fall with gravity and collide with a passive rigid body NURBS plane, the sphere bounces and bobbles around the center of mass.

The center of mass also sets the point about which an active rigid body rotates when you set the Initial Spin (described later). For example, if you set the center of mass within an active rigid body sphere, the rigid body spins about itself. If you set it outside the sphere, the sphere rotates about the center of mass.

By default, a polygonal object's center of mass is the centroid of its bounding box. A NURBS object's default center of mass might be slightly away from the centroid.

Maya doesn't use the center of mass in dynamic calculations of passive rigid bodies.

Lock Center of Mass

By default, Maya recalculates a rigid body's center of mass when you alter the object's surface during modeling. If you turn on this attribute, Maya doesn't change the center of mass. *Available in Attribute Editor only.*

Static Friction

Sets how much a rigid body resists moving from resting contact with another rigid body. For instance, if you place a ball on a sloped plane, Static Friction sets how easily the ball begins its initial slide or roll down the plane. Static friction has little or no effect after an object is moving.

A value of 0 lets the rigid body move freely. A value of 1 diminishes movement.

Dynamic Friction

Sets how much a moving rigid body resists movement against another rigid body's surface.

A value of 0 lets the rigid body move freely. A value of 1 diminishes movement.

> **Tip** When two rigid bodies are in contact, the Static and Dynamic Friction of each contributes to the motion. To tune sliding and rolling of rigid bodies in contact, try various values for Static and Dynamic Friction. For example, to simulate a hockey puck sliding across wet ice, you must set Dynamic Friction to nearly 0 on the puck *and* ice.

Bounciness

Sets the resilience of the rigid body.

Damping

Sets an opposing force against the rigid body's movement. This attribute is similar to drag; it affects object movement before, during, and after contact with another object. A positive value diminishes movement. A negative value increases movement.

Impulse X, Y, Z

Creates an instantaneous force, with magnitude and direction, on the rigid body at the local space position specified in Impulse Position X, Y, Z. The higher the number, the greater the magnitude of force. For more details, see "Keying impulses to rigid bodies" on page 261.

Impulse Position X, Y, Z

Specifies the position in the rigid body's local space where the impulse strikes. If the impulse strikes a point other than the center of mass, the rigid body rotates about its center of mass in addition to moving with the change to its velocity.

If you specify a position outside the object's surface boundaries, you'll still see rotation and velocity. Note that the 0, 0, 0 position of an object's local space is at the center of its bounding box.

Spin Impulse X, Y, Z

Applies an instantaneous rotational force (torque) on the rigid body's center of mass in the direction you specify by the X, Y, and Z values. These values set magnitude and direction. The higher the number, the greater the magnitude of the rotational force.

> **Note** The Attribute Editor displays the current Velocity, Spin, Force, and Torque of a rigid body. See "Get data on velocity, forces, and collisions" on page 264 for details on these read-only attributes.

Solver Id

The read-only rigid body index number the solver uses to calculate dynamics for this rigid body. You can use the Solver Id in MEL scripts and expressions to identify a specific rigid body in the solver.

Initial Settings attributes

Initial Spin X, Y, Z

Sets the initial angular velocity of the rigid body. This spins the rigid body.

Initial Position X, Y, Z

Sets the initial position of the rigid body in world space.

Initial Orientation X, Y, Z

Sets the initial local space orientation of the rigid body. For example, an Initial Orientation X value of 90 means the object has an orientation of 90 degrees rotation about its X-axis. This assumes you are using the degrees rather than radians as your working units.

Initial Velocity X, Y, Z

Sets the initial speed and direction of the rigid body.

Performance Attributes

Stand In

Displays a menu that lets you select a simple internal cube or sphere as a stand-in for rigid body calculations. The original object remains visible in the scene. If you use a stand-in sphere or cube, playback speed improves but collision reactions differ from the actual object. To use the actual geometry, select none.

Apply Force At

Displays a menu that lets you set whether forces affect a rigid body at its center of mass, the corners of its bounding box, or its CVs or vertices

Center of Mass

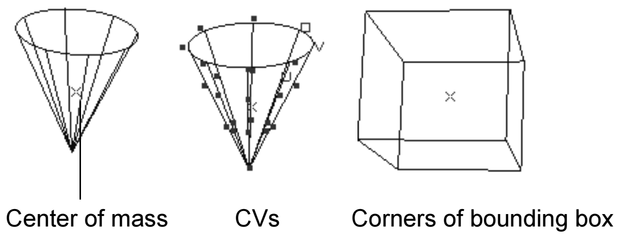
Applies force to a single position at the center of mass. No torque is imparted to the rigid body.

Bounding Box

Applies force to the eight corners of the object’s bounding box. This is the default setting. To see the object’s bounding box, select Shading > Bounding Box from the menus above the workspace pane.

Vertices or CVs

Applies force at each vertex of polygonal objects, or at each CV of NURBS surfaces. This is the slowest but most accurate choice.



Tessellation Factor

Maya internally converts NURBS objects to polygons before it animates rigid body dynamics. The Tessellation Factor sets the approximate number of polygons created during the conversion. Lower numbers create coarser geometry and lessen animation accuracy, but increase the playback speed.

Increase the Tessellation Factor if you are bouncing an object on a bumpy, irregular surface. Experiment with various values until you see the desired result.

If you change the Tessellation Factor, the internal conversion occurs once, immediately after you change its value. This takes some time for complex NURBS surfaces. Increasing the Tessellation Factor increases the time for Maya to detect rigid body collisions.

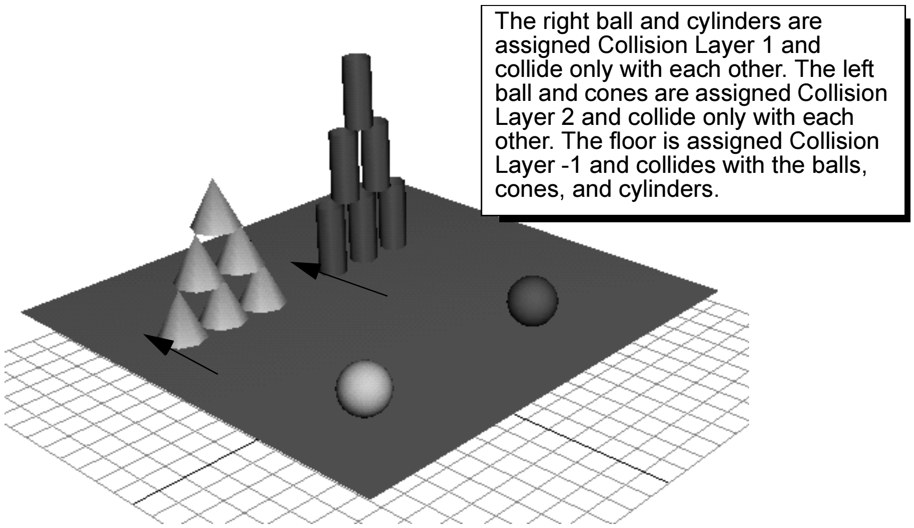
Collision Layer

You can use collision layers to create exclusive groups of objects that collide with each other. Only rigid bodies with the same collision layer number can collide with each other.

For example, suppose you have four objects moving towards each other, but you want one of the objects to pass through without colliding. You can assign the non-colliding object a different collision layer number.

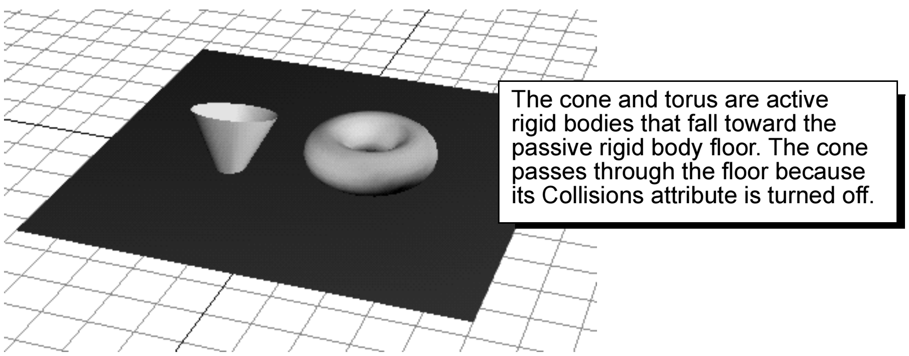
By putting rigid bodies that don’t collide in different collision layers, you can lessen collision processing time.

A rigid body in collision layer -1 (minus 1) will collide with all rigid bodies in the solver, regardless of their collision layer numbers.



Collisions

When off, the rigid body doesn't collide with any objects in a scene.



Tip	You can turn on the Interpenetrate attribute of two or more rigid bodies to prevent only those objects from colliding with each other. To do this, select the objects then select Solvers > Set Rigid Body Interpenetration. To turn off the Interpenetrate attribute later, select the objects then select Solvers > Set Rigid Body Collision.

Ignore

When on, this turns off the influence of fields, collisions, and any other rigid body effects. This is useful if you have a scene with many computation-intensive rigid bodies and want to disable several to speed up animation play.

Soft/Rigid Bodies > Create Passive Rigid Body

Creates a passive rigid body.

Soft/Rigid Bodies > Create Passive Rigid Body > ❐

Sets the options when creating a passive rigid body. These are the same as the options when creating an active rigid body, except that the Active check box is unchecked.

See "Soft/Rigid Bodies > Create Soft Body" on page 297.

Soft/Rigid Bodies > Create Constraint

Creates a constraint.

Soft/Rigid Bodies > Create Constraint > ❐

Sets the options when creating a constraint. Here are the options:

Constraint Name

Name of the constraint.

Constraint Type

Choose one of: Nail, Pin, Spring, Hinge, Barrier.

Interpenetrate

If you want the rigid bodies to penetrate each other rather than collide upon contact, turn on Interpenetrate.

Set Initial Position

If you want to specify the constraint's initial position, turn on Set Initial Position and enter X, Y, and Z values for the position.

If you don't turn on Set Initial Position, a constraint is created either at the midpoint between the two rigid bodies (extending from one rigid body's center of mass to the other's center of mass), or at the center of mass of the rigid body, depending on the type of constraint.

Spring Attributes

Stiffness

Sets the rigidity of the spring constraint. The greater the value, the greater the force the spring exerts on the body for the same displacement.

Damping

Mutes the spring action. A high value brings the rigid body to rest faster. A low value brings the rigid body to rest slower. A negative value increases the spring's force on the rigid body. A spring with a zero or negative damping value never comes to rest.

Set Spring Rest Length

Enables you to set the Rest Length.

Rest Length

Sets the length the springs try to reach when you play the scene. If you don't turn on Set Spring Rest Length, the Rest Length is set to the same length as the constraint.

Soft/Rigid Bodies > Set Active Key

???

Soft/Rigid Bodies > Set Passive Key

???

Soft/Rigid Bodies > Break Rigid Body Connections

???

Soft/Rigid Bodies > Create Soft Body

Creates a soft body.

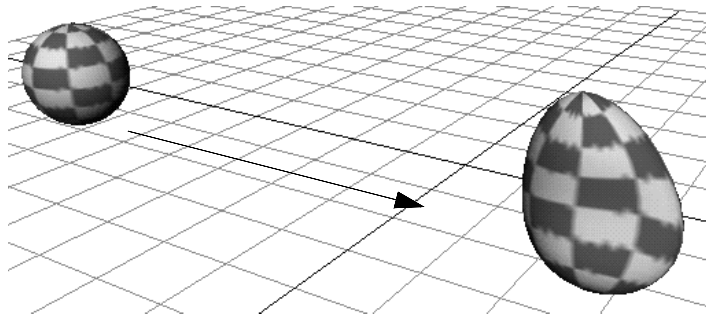
Soft/Rigid Bodies > Create Soft Body > ❐

Sets the options when creating a soft body. Here are the options:

Make Soft

Converts the object to a soft body. Select this option if you haven't animated the object and will animate it with dynamics (for instance, a field). Also use this option if you've used nondynamic animation on the object and want to keep the animation after you create the soft body. Nondynamic animation includes keys, motion paths, non-particle expressions, and deformers.

Dynamics that affect the soft body's particles are not affected by momentum from the nondynamic animation. The dynamics occur in the particle object's local space, which are uninfluenced by other animation on the object.



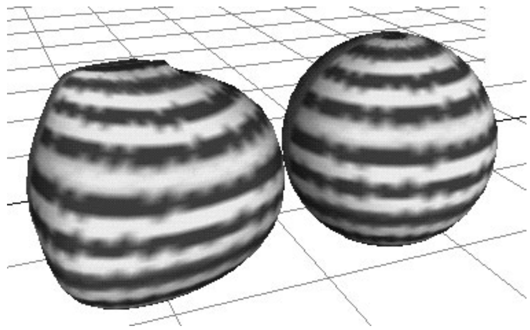
The ball was animated with keyed translate attributes, made a soft body with Make Soft, then connected to a turbulence field.

Duplicate, Make Copy Soft

Makes a copy of the object a soft body without altering the original object. If you use this option, you can turn on Make Non-Soft a Goal (described later) to make the original object a goal object for the soft body. The soft body follows the animated goal object. You can edit the goal weight of the soft body particles to create rubbery or jiggling motion.

For instance, suppose you make a sphere a soft body with Duplicate, Make Copy Soft and Make Non-Soft a Goal selected. Next, you set keys to animate the motion of the goal object, which is the original sphere. You then give the particles of the soft body copy various goalPP values. When you play the animation, the soft body copy deforms like jiggling gelatin as it follows the original sphere. Note that you'll typically hide the original sphere by turning on Hide Non-Soft Object.

See "Goals" on page 21 for details on goals and goal weights.



The soft body copy (left) with various goalPP weights jiggles as it follows the original sphere.

Another way to use Duplicate, Make Copy Soft is to use it repeatedly while Make Non-Soft a Goal is off. This creates identical soft body copies of an object. You can animate each copy independently of the original.

Duplicate, Make Original Soft

This option works like Duplicate, Make Copy Soft, except it makes the *original* object a soft body, and makes a copy of the original object.

Use Duplicate, Make Original Soft when the object's downstream construction history requires you to use the original rather than the copy as the soft body.

For example, because an IK spline requires you to use the original curve rather than a copy to control its joint chain, you must use Duplicate, Make Original Soft to make an IK spline a soft body. If you also turn on Make Non-Soft a Goal, you can animate the curve's copy (the goal) to control the motion of the soft body IK spline.

Duplicate Input Graph

Duplicates upstream nodes when you create the soft body with either of the Duplicate options. Turn this on if the original object has dependency graph inputs you want to be able to use and edit in the copy.

Hide Non-Soft Object

When you duplicate an object as you create a soft body, one of the objects becomes a soft body. If you turn on this option, the object that's not a soft body is hidden.

If you later need to display a hidden non-soft object, be aware that you can select the object in the Outliner and select Display > Show > Show > Selection.

Make Non-Soft a Goal

Turn this option on to make the soft body trail or move towards the goal object made from the original or duplicate geometry. Using the Paint Soft Body Weights Tool, you can set goal weights on a soft body on a per particle basis by painting on the soft body surface (see ”Paint Soft Body Weights Tool” on page 253).

To edit the goal weights of the soft body particles after you create the soft body, see ”Edit goal attributes” on page 108. A goal weight of 0 makes the soft body bend and deform freely. A value of 1 makes the soft body stiff; it follows the contours of its goal object exactly.

Note that if you create a soft body with Make Non-Soft a Goal turned off, you can still create a goal for the particles. Select the soft body particles, shift-select an object to become the goal, and select Particles > Add Goal.

Weight

Sets how closely the soft body follows the goal object made from the original or duplicate geometry. A value of 0 makes the soft body bend and deform freely. A value of 1 makes the soft body stiff. Values between 0 and 1 have intermediate stiffness.

Tips You’ll typically apply a field to the soft body but not the non-soft body. If you don’t turn on Hide Non-Soft Object, use the Outliner to select the soft body without selecting the non-soft body. If you accidentally apply a field to the non-soft body, it becomes a rigid body that’s affected by the field by default. This is usually undesirable. You’ll more likely want to key its motion or leave it alone.

Collisions between soft bodies—including a soft body colliding with itself—can have unexpected results, especially if the soft bodies move toward each other after colliding.

Soft/Rigid Bodies > Create Springs

Creates a spring.

Soft/Rigid Bodies > Create Springs > ❐

Sets the options when creating a spring. Here are the options:

Spring Name

Names the spring object for easy identification in the Outliner.

Add to Existing Spring

Adds springs to an existing spring object rather than a new one. When you turn on this option, other options in the Spring options window work the same as if you're creating new springs on an object.

Don't Duplicate Springs

Avoids creating a spring between two points if a spring already exists there. This option has effect only when you turn on Add to Existing Spring.

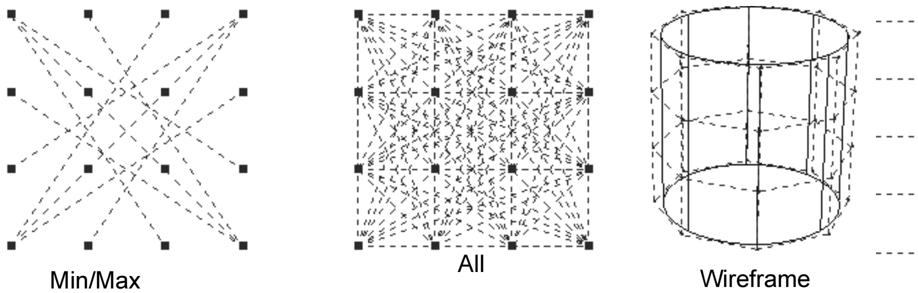
Set Exclusive

When more than one object is selected, points from the selected objects are linked with springs to points in every *other* object, based on the average length between the points. The software determines an appropriate minimum and maximum distance range based on the objects. The points within each object are not linked with springs.

Creation Method

Set one of the following

MinMax	Creates springs only within the range of Min and Max Distance options. You can use the MinMax option to cause selected particles to adhere to each other, for example, to simulate a puddle of fluid.
All	Creates springs between all pairs of selected points. Choose this option when you want an object to have a uniform spring structure throughout its shape, for example, a bouncing ball or block.
Wireframe	Creates springs between all particles on the outer edges of a soft body. This option is useful for soft bodies made from curves, for example, a rope. It's also useful for soft bodies with irregular surfaces, or for objects that you want to have a gel or liquid-like adhesion on its outer edges. See Wire Walk Length for more details.



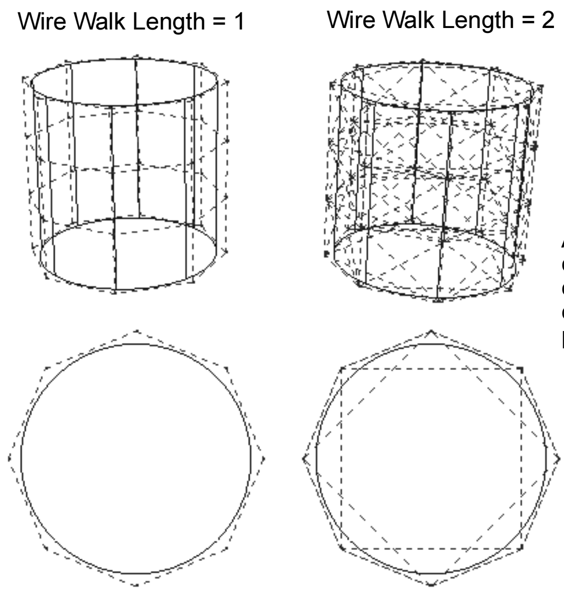
The left and center objects are particle grids. The right object is a soft body cylinder.

Min and Max Distance

See "Creation Method" on page 301.

Wire Walk Length

This option works with the Wireframe option. It sets how many springs are created between edge particles. For instance, for each edge particle, the value 2 creates springs to all edge particles within a distance of two particles. The higher the wire walk length, the stronger the structure of the soft body.



A Wire Walk Length of 2 connects a spring between each edge particle and the edge particles within two particles' distance.

Soft/Rigid Bodies > Paint Soft Body Weights Tool

Selects the Paint Soft Body Weights Tool. For more information, see "Paint Soft Body Weights Tool" on page 253. See also *"How Artisan brush tools work"* in the Paint Effects, Artisan, and 3D Paint guide.

Soft/Rigid Bodies > Paint Soft Body Weights Tool > ❐

Lets you specify the settings for the Paint Soft Body Weights Tool in the Tool Settings editor. There are attributes unique to the Paint Soft Body Weights Tool in the Paint Attributes section. These unique attributes are described below. For descriptions of all other attributes in all other sections, see *"Common Artisan Brush Tool Settings"* in the Paint Effects, Artisan, and 3D Paint guide.

Paint Attributes

These are descriptions of the attributes in the Paint Attributes section.

copyOf*modealname*ParticleShape.goalPP

Displays the name of the particle node selected to paint and the attribute you are painting (goalPP weights). To select another particle node to paint, click this button and select the appropriate particle node goalPP weights name. By default, the tool selects the first particle node it detects.

When you select the soft body, the Paint Soft Body Weight Tool automatically detects the particle node and goalPP attribute on the soft body. The name of the node and attribute displays on the top button.

Filter: particle

Sets a filter so that only particle nodes display on the menu for the button above this one. You are painting particle goalPP weights with the Paint Soft Body Weights Tool, so you do not need to change this filter unless you want to paint other types of attributes.

Paint Operation

Select an operation to define how you want painted goalPP values to be affected.

Replace	Replaces the goalPP values for the soft body particles you paint over with the specified Value and Opacity.

Add	Adds the specified Value and Opacity to the current goalPP values you paint over. If the value is negative (possible if you set Min or Max to a negative value), the value actually decreases.
Scale	Scales the current goalPP values you paint over by the Value and Opacity factors.
Smooth	Changes the goalPP values to be the average values of the surrounding goalPP values.

Value

Set the value to apply when you perform any of the painting operations.

Min/Max Value

Set the minimum and maximum possible paint values. By default, you can paint values between 0 and 1. Setting the Min/Max Values you can extend or narrow the range of values. Negative values are useful for subtracting weight. For example, if you set Min Value to -1, Value to -0.5, and select Add for the operation, you would subtract 0.5 from the weight of vertices you paint. Positive values are added.

> Tip To help you differentiate paint values when you paint with ranges greater than 0 to 1 (for example, -5 to 5), and to maximize the range of values that display when you paint values with ranges between 0 to 1 (for example, 0.2 to 0.8), set Min Color and Max Color (in the Display section) to correspond with the Min/Max values.

Clamp

Select whether you want to clamp the values within a specified range when you paint, regardless of the Value set.

Lower	Turn this on to clamp the lower value to the Clamp Value specified below. For example, if you clamp Lower and set the lower Clamp Value to 0.5, the values you paint will never be less than 0.5, even if you set the Value to 0.25.
Upper	Turn this on to clamp the upper value to the Clamp Value specified below. For example, if you clamp Upper, set the upper Clamp Value to 0.75, and set Value to 1, the values you paint will never be greater than 0.75.

Clamp Values

Set the Lower and Upper values for clamping.

Flood

Click Flood to apply the brush settings to all the particle goals on the selected soft body. The result depends on the brush settings defined when you perform the flood.

Vector Index

If you are painting a three channel attribute (RGB or XYZ), select the channel you want to paint. Soft body goalPP weight is a single channel attribute, therefore you do not need to change this setting.

Nodes

rigidBody

The table doesn't list the compound vector attributes force (for), velocity (vel), and torque (tor). These attributes consist of their X, Y, and Z components.

Long name	Short name	Description	Data type
forceX	fx	Force on the rigid body in the X direction.	float
forceY	fy	Force on the rigid body in the Y direction.	float
forceZ	fz	Force on the rigid body in the Z direction.	float
velocityX	vx	Velocity of the rigid body in the X direction.	float
velocityY	vy	Velocity of the rigid body in the Y direction.	float
velocityZ	vz	Velocity of the rigid body in the Z direction.	float
torqueX	trx	Torque on the rigid body in the X direction.	float
torqueY	try	Torque on the rigid body in the Y direction.	float

Long name	Short name	Description	Data type
torqueZ	trz	Torque on the rigid body in the Z direction.	float

Contact attributes

A rigid body has read-only attributes that contain data on the contact made with it during collisions. You can query the attributes in MEL commands or expressions by using the attribute names in the table that follows.

Long name	Short name	Description	Data type
contactCount	cct	Number of contacts in the current frame.	integer
contactPosition	cnp	Position of each point contact in the current frame. The length of the array is equal to contactCount.	vector array
contactName	cnn	List of rigid body names that are in contact with the rigid body. The array is of length contactCount.	string array

rigidSolver

Rigid Solver Attributes

Step Size

Sets how often within a frame the rigid body calculations occur. For example, if each frame of the animation is 0.1 seconds and the Step Size is 0.033 seconds, the solver calculates rigid body animation three times in the frame.

Generally, decreasing the Step Size value improves rigid body animation accuracy but slows scene play. If you have fast-moving rigid bodies that don’t collide as expected, decrease the Step Size. *This is a nonkeyable attribute available in the Attribute Editor only.*

Collision Tolerance

Sets how accurately and quickly a rigid body solver detects collisions. Generally, smaller Collision Tolerance values increase the calculation time and collision accuracy. A small Collision Tolerance is often necessary to create accurate collisions involving tiny or thin objects. *This is a nonkeyable attribute available in the Attribute Editor only.*

Scale Velocity

Scale Velocity is used with the Display Velocity attribute (see page 309). If you turn on Display Velocity, a moving rigid body displays a velocity arrow icon that represents the magnitude and direction of the rigid body motion. You can change the Scale Velocity to scale the arrow.

Start Time

Sets the Time Slider frame when the solver starts animating dynamics for the rigid bodies it controls.

Current Time

Lets you speed up or slow down dynamic animation for all rigid bodies connected to the solver. Current Time works the same for rigid bodies as it does for particle objects. See ”Control the timing of particle dynamics” on page 77.

Rigid Solver Methods

Solver Method

Provides a pop-up menu for choosing the accuracy and speed of rigid body calculations

Midpoint

Calculates faster with less accuracy.

Runge-Kutta

Calculates at medium speed and accuracy.

Runge-Kutta Adaptive

Calculates slower but with most accuracy. This is the default setting.

Rigid Solver States

State

Turns off or on the effect of fields, collisions, and rigid body constraints for the rigid bodies it controls. If you want to speed up playback of your animation and temporarily are not concerned with rigid body effects, turn off the State.

Friction

Sets whether the rigid bodies stick or slide after collision. If Friction is on, the rigid bodies stick; if Friction is off, they slide.

If the contact between rigid bodies is limited to instantaneous collisions, the Friction setting does not affect it much. Turning off Friction might speed playback.

Bounciness

Turns bounciness on or off. When off, objects won’t bounce against each other, but rigid body animation plays faster.

Contact Motion

When on, Maya simulates Newtonian physics for its rigid body dynamics. When off, Maya simulates a damped environment without inertia. More specifically, collision forces such as Bounciness and Friction do not affect the rigid bodies. Fields affect the rigid bodies, but not initial spin, initial velocity, or impulses.

Contact Data

Accumulates data about contact between rigid bodies in the scene (see ”Contact attributes” on page 264).

Allow Disconnection

By default, you cannot break connections from a rigid body to the rigid body solver that handles its dynamic animation. You can turn on Allow Disconnection so that you can break the connections. See ”Warning when you delete rigid body connections” on page 284.
Available in the Attribute Editor only.

Cache Data

If you turn this on, Maya caches in memory the dynamic state of all rigid bodies connected to this solver. You can thereafter examine the animation of the rigid bodies by scrubbing the Time Slider or playing in reverse. See ”Memory caching” on page 378.

Delete Cache

Deletes the cached dynamic state of all rigid bodies connected to this solver.

Rigid Solver Display Options

Display Constraint

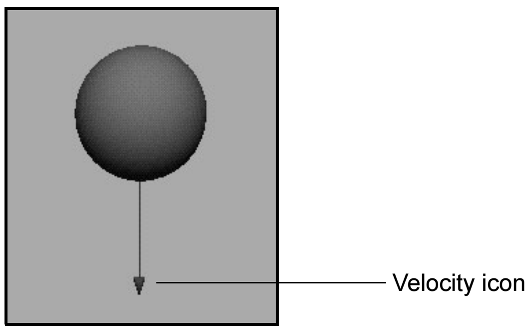
Displays icons for rigid body constraints. See "Edit attributes of a rigid body solver" on page 261

Display Center of Mass

Displays an icon for the center of mass of each rigid body.

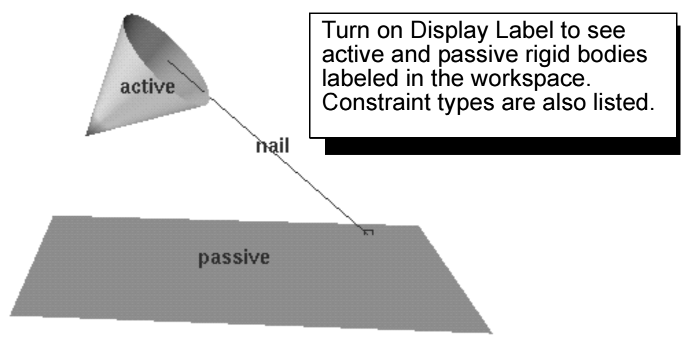
Display Velocity

Displays arrow icons that indicate the velocity magnitude and direction of rigid bodies. See "Scale Velocity" on page 307 to tune the display of the arrow.



Display Label

Labels rigid bodies as active or passive. Also shows the type of constraint.



rigidConstraint

This node contains the attributes for the rigid constraints. These constraints can only be used with rigid bodies. The constraints include nail, pin, hinge, spring, and barrier.

Constraint Control

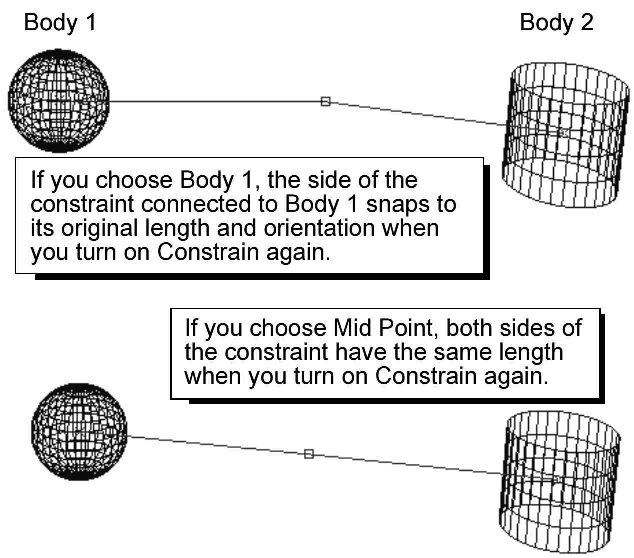
Constrain

Turns a constraint on or off.

Relative To

For a constraint between a pair of rigid bodies, this lets you choose where Maya recreates a constraint after you turn it on again. Your choices follow:

Body 1	Recreates the constraint at the original distance and orientation from the first of the two rigid bodies you selected when you created the constraint.
Body 2	Recreates the constraint at the original distance and orientation from the second of the two rigid bodies you selected when you created the constraint.
Mid Point	Recreates the constraint at the midpoint between the two rigid bodies.
User Defined	Lets you specify the position in the User Defined Position X, Y, and Z boxes. These attributes are dim unless you turn on User Defined.



springShape

Many of the spring node attributes are the same as the options available in the Spring Options window. See "Soft/Rigid Bodies > Create Springs" on page 300.

Spring Attributes

Count

The number of springs in the object.

Manage Particle Death

This attribute determines if the spring node needs to do ID/Index mapping. The default value is true for this attribute. If users know that no particles died, it can be set to false. With the false value, the spring node will not do particle ID/Index mapping and speed up the evaluation.

Per Spring (Array) Attributes

Component Editor

Opens the Component Editor for the current spring.

5 Effects

About Dynamics

Effects

Effects

Maya effects are built-in programs that make it easy for you to create complex animation effects such as smoke and fire. Each Maya effect offers many options and attributes for tuning the results.

Fire

You can use the Fire effect to emit fire from these objects:

- particle object
- NURBS or polygonal surface
- NURBS curve
- number of CVs, vertices, edit points of an object
- lattice points

❖ "Create Fire" on page 318

Use smoke effects

The Smoke effect emits smoke from a position in the workspace or from a selected particle object or geometry object. The effect uses a series of smoke images (sprites) included with the Maya software. You can alternatively use your own images. You must hardware render the resulting smoke.

❖ "Create Smoke" on page 320

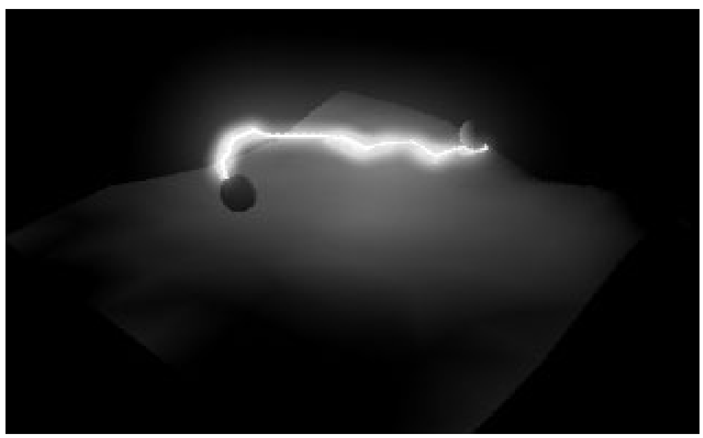
Fireworks

You use the Fireworks effect to make fireworks displays quickly and easily. The effect creates a number of rockets that fly up, leave a trail as they fly, and burst. Gravity fields are also created for all particles. Initially, the effect randomly chooses colors and positions for rocket launches and bursts. You can edit these defaults later. The fireworks are rendered in software (not hardware) as streaks.

❖ "Create fireworks" on page 323

Lightning

The Lightning clip effect creates a bolt of lightning between two or more objects or locators.

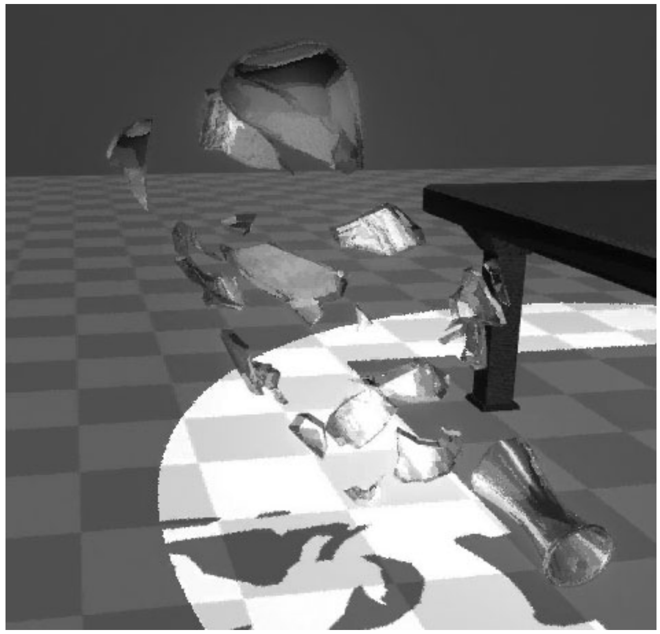


The lightning bolt is made up of soft body curves with extruded surfaces, which are rendered. The motion of the lightning is derived from an expression on the particles that make up the soft body curve.

❖ "Create lightning" on page 331

Shatter

Shatter duplicates an object and breaks the duplicate into multiple pieces called shards. Depending on the type of shard you choose to create, you can move the shards with dynamic forces or keyframe their movement.



Shatter provides three methods of breaking the object:

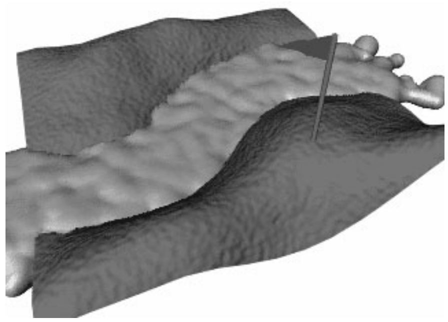
- "Surface shatter"
- "Solid shatter"
- "Crack shatter"

❖ "Create shatter" on page 336

Curve Flow

The Curve Flow effect emits particles that flow along a curve you create. You can use this effect to create flowing fluids, gases, or instanced objects. The emitted particles start at the first CV and end at the last CV you drew when you created the curve.

For example, to make water flow down a ravine, you would first create a NURBS or polygonal surface as the hill with a ravine. You then create a curve in the middle of the ravine from beginning to end. You use the Curve Flow effect on the curve and adjust the Curve Flow attributes. Finally, you select an appropriate render type such as Blobby Surface for the emitted particles and adjust their appearance.

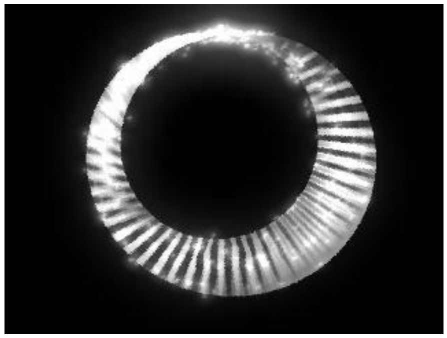


Tips	You can use the Curve Flow effect on a curve or surface as a convenient way to guide the flow along deformed surfaces. It's often helpful to duplicate the curve or surface, raise the copy above the surface, then apply the Curve Flow effect to the copy. To keep particles flowing within a planar boundary, for instance, water flowing down a river, you can sandwich the curve between invisible planes and then make the particles collide with the planes. The particles bounce within the two invisible planes. It's often useful to use the Curve Flow effect on the same curve more than once. By setting option settings differently with each usage, you can create a complex look not possible with a single usage.

❖ "Create curve flow" on page 345

Surface Flow

The Surface Flow effect emits particles that flow along the surface of any NURBS model. You can use this effect to create flowing fluids, gases, or instanced objects. The flow effect automatically adjusts to any changes in the shape of the surface.



❖ "Create a surface flow" on page 349

How do I? Simulate dynamic effects

Work with effects

Create Fire

Prepare to use the Fire effect

Before you use the Fire effect, be aware of these issues:

- Use the Fire effect on only one object at a time.

  To apply Fire to a group of NURBS surfaces, first convert each surface to polygons and combine the surfaces. From the Modeling menu set, select Modify > Convert > NURBS to Polygons, then choose Polygons > Combine (from the Modeling menu set).

  To apply Fire to a group of polygonal surfaces, simply combine the surfaces with Polygons > Combine.
- If you emit from a NURBS or polygonal object, the size and shape of the object affects the quality of the fire. You'll need to use an object large enough to generate an adequate flame area. If you emit fire from a curve, avoid using a curve with abrupt changes in direction.
- It's often useful to use the Fire effect on the same geometry more than once. By setting options differently with each usage, you can create a complex look not possible with a single usage.

- If you want to animate the movement of the fire around the workspace, consider using the Fire effect on a particle object. You can work with per particle expressions on particle objects, so you have more flexibility in altering the fire's motion.
- You'll often need to emit flames from part of an object rather than from its entire geometry. In some instances, the part of the object where you want the fire won't have geometry present. A common technique in such cases is to emit from an invisible geometric object in that area.

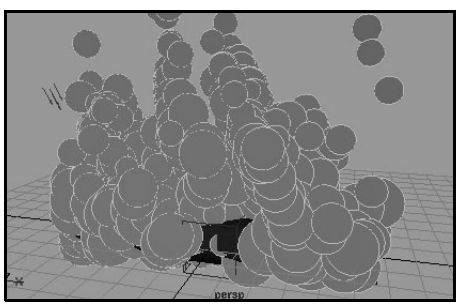
To use the Fire effect

1. Do one of the following:
   - Select the object or CVs, edit points, vertices, or particles that you want to emit fire.
   - To create a positional emitter, deselect all objects.
2. Select Effects > Create Fire > ❐.
3. Set attributes in the Create Fire Effect Options window (see page 320) and click Create.

   The Fire effect creates an emitter, emitted particle object, expressions, ramps, textures, and several fields.
4. Play the animation.

   Emitted particles appear as circles in the workspace because the particles are displayed as the Cloud render type.

   Here's an example with Shading > Smooth Shade All turned on.



5. Add a light to the scene and software render to see the fire.

   From the Rendering menu set, for example, select Render > Render Current Frame.

Edit attributes of the Fire effect

The following pages explain attributes you can set to tune the Fire effect. The Fire effect creates several custom attributes in the emitted particle object it creates. The custom attributes control a combination of field or emitter attributes to lessen the settings you would otherwise need to make to tune the fire.

Attributes in the Create Fire Effect Options window

These attributes appear in the Create Fire Effect Options window when you select Effects > Create Fire > ❒. Changes you make to the options window affect only fire you create after you make the changes.

You can edit most of these attributes after you use the Fire effect by selecting the emitted particle object and opening the Extra Attributes section of the Attribute Editor. Exceptions are noted in the text.

❖ ”Effects > Create Fire” on page 359

Tune the fire’s appearance

- Adjust any other attributes that are controlled by expressions, ramps, and textures to tune the effect. To learn which expressions, ramps, and textures are created by the Fire effect, apply Fire to an object in an otherwise empty scene. Use the Attribute Editor, Expression Editor, and Hypershade to see the additions.
- Edit attributes of the Cloud particle render type.
- Turn off the turbulence field by disconnecting it using the Dynamics Relationship Editor.
- Animate the emitter to move the fire in your scene.

Create Smoke

Prepare to use the Smoke effect

Before you use the Smoke effect, be aware of these issues:

- Use the Smoke effect on at most one object at a time.

  To apply Smoke to a group of NURBS surfaces, first convert each surface to polygons and combine the surfaces. From the Modeling menu set, select Modify > Convert > NURBS to Polygons, then choose Polygons > Combine.

  To apply Smoke to a group of polygonal surfaces, simply combine the surfaces with Edit Polygons > Combine.
- If you emit from a NURBS or polygonal object, the size and shape of the object affects the quality of the smoke. You'll need to use an object large enough to generate an adequate smoke area. If you emit smoke from a curve, avoid using a curve with abrupt changes in direction.
- It's often useful to use the Smoke effect on the same geometry more than once. By setting options differently with each usage, you can create a complex look not possible with a single usage. For instance, by using two emitters with different emission rates and different sprite sequences, you can create a combination of burning smoke and steaming smoke.
- If you want to animate the movement of the smoke around the workspace, consider using the Smoke effect on a particle object. You can work with per particle expressions on particle objects, so you have more flexibility in altering the smoke's motion.
- You'll often need to emit smoke from part of an object rather than from its entire geometry. In some instances, the part of the object where you want the smoke won't have geometry present. A common technique in such cases is to emit from an invisible geometric object in that area.

  For example, suppose you want to blow smoke out of the end of a hollow exhaust pipe, but the pipe has no object where you can conveniently emit the smoke. You can create a disk of the correct size and shape, position it inside the pipe, and apply an emitter to the disk. Make the disk invisible by selecting Display > Hide > Hide Selection.
- If you create your own smoke images, for instance, with paint software such as VizPaint or StudioPaint, remember to save the alpha channel (transparency data) when you create the images.

To use the Smoke effect

1. To use smoke images supplied with Maya, copy the images `Smoke.0` through `Smoke.50` from the following directory on the Documentation, Lessons, and Extras CD to the `/sourceimages` directory of your current project:

   `Gifts/smoke/`

To use your own images, put them in the `/sourceimages` directory of your current project.

2 Do one of the following:

- Select the object or CVs, edit points, vertices, or particles that you want to emit smoke.
- To create a positional emitter, deselect all objects.

3 Select Effects > Create Smoke > ❐.

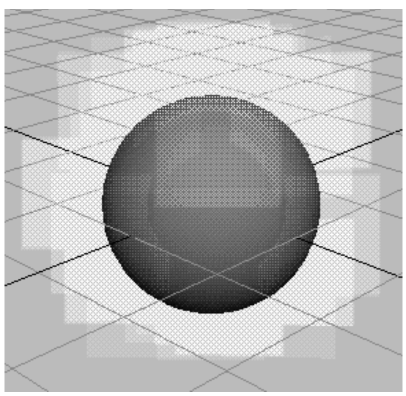
4 Set attributes in the Create Smoke Effect Options window (see page 322) and click Create.

The Smoke effect creates an emitter, emitted particle object, expressions, turbulence field, and other fields needed to make the smoke.

5 Play the animation.

Emitted particles appear as squares in the workspace because the particles are displayed as the Sprite render type.

Here’s an example with Shading > Smooth Shade All turned on.



6 Hardware render the scene to see the smoke.

See Rendering for details.

Edit attributes of the Smoke effect

The following pages explain attributes for tuning the Smoke effect. The Smoke effect creates several custom attributes in the emitted particle object it creates. The custom attributes control a combination of field and emitter attributes to lessen the settings you would otherwise need to make to tune the smoke.

Attributes in the Create Smoke Effect Options window

The following attributes appear in the Create Smoke Effect Options window when you select Effects > Create Smoke > ❐. Changes you make to the options window affect smoke you create after you make the changes.

You can edit most of these attributes after you use the Smoke effect by selecting the emitted particle object and opening the Extra Attributes section of the Attribute Editor.

❖ ”Effects > Create Smoke” on page 361

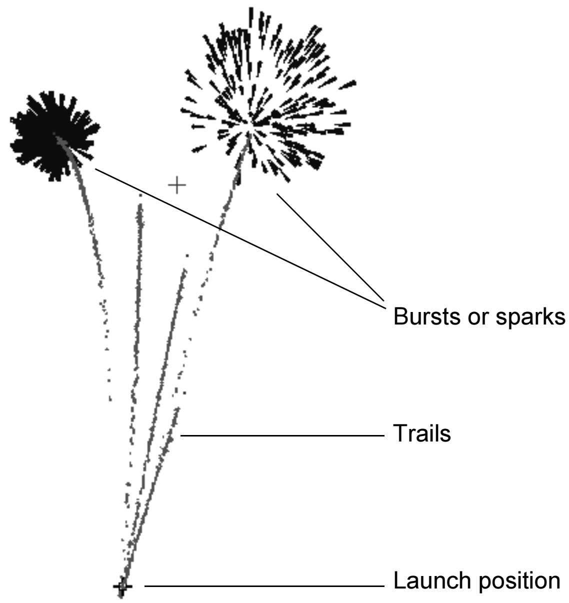
Additional tips

You can do the following additional steps to tune the smoke’s appearance:

- Change the size and orientation of the smoke by altering the Scale and Rotate values of the emitted particle object.
- Key a change in the value of Scale Y to make the particles appear to move faster or slower.
- Edit attributes of the sprites. See ”Sprites” on page 40.
- Edit any expressions created by the Smoke effect. To learn which expressions are created by the effect, apply Smoke to an object in an otherwise empty scene. Use the Expression Editor to see the additions.
- Turn off the turbulence field by disconnecting it using the Dynamics Relationship Editor.
- Animate the emitter to move the smoke in your scene.

Create fireworks

Before you create fireworks, it is helpful to review the terminology used in the Fireworks effect.



To create fireworks

1. Choose Effects > Create Fireworks > ❐.
2. Set controls in the option window as needed.

   For controls related to creation, see ”Set Fireworks creation controls” on page 325.

   For controls that you can edit later, see ”Edit general Fireworks attributes” on page 325.
3. Click Create in the options window or choose Effects > Create Fireworks.

   The fireworks group and rockets selection handles appear.
4. Click the play button.

   If you make changes to the fireworks attributes, you may need to replay the scene from frame one.
5. If you want faster playback, turn off the Display Geometry attribute.

   Display Geometry is near the bottom of the fireworks group attributes. When turned off, the particles appear as streaks rather than instanced cones. For details, see the description on page 326.
6. To render the fireworks, make sure Display Geometry is turned on and use software rendering.

Set Fireworks creation controls

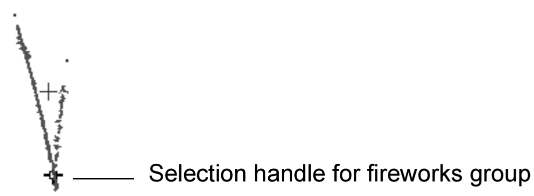
Of the numerous controls in the Create Fireworks Effect Options window, the following are described here because they relate to creation. The other controls are described in ”Edit general Fireworks attributes” on page 325.

❖ ”Effects > Create Fireworks” on page 363

Edit general Fireworks attributes

After you create the fireworks, you can adjust general, group-related attributes by editing the fireworks attributes. You can do this using the Channel Box or the Attribute Editor.

Start by selecting the fireworks group selection handle.



Max Burst Speed

Affects how fast all rockets burst, and, consequently, how wide the bursts appear. Each particle in the burst has a random speed, and this speed is the maximum.

Max Sparks Count, Min Sparks Count

Each burst consists of a number of streaks randomly distributed between these two values.

Sparks Color Spread

Determines the number of colors to use per burst. The colors are selected from the palette of colors established by the Num Spark Colors option. The spread refers to the number of colors on either side of the main color. For example, a spread of one means a total of three colors: the main color, plus one color on either side of the color palette.

Rocket Gravity

Sets the magnitude of the gravity field affecting rocket trajectories. This option does not affect the gravity fields for rocket trails and burst sparks. To edit those fields, open their gravity nodes in the Attribute Editor.

Attaching any fields to the rockets other than the created gravity will make the burst position unpredictable.

Show All Burst Positions

Shows you the rocket burst positions by displaying the rocket particle ID number in the view panel.

Show All Launch Positions

Shows you the rocket launch positions by displaying the rocket particle ID number in the view panel.

Trail Emit Rate

Sets the rate that rockets emit trails. An Trail Emit rate of 0 (zero) means no rocket trail appears.

Trail Emit Speed

Sets the Trail Emit speed of the rocket trail particles. You can enter a value of 0 or more. A value of 1 leaves the speed as is. A value of 0.5 reduces the speed by half. A value of 2 doubles the speed.

Trail Emit Spread

Sets the emission spread angle. This angle defines a conical region into which the rocket trail particles are emitted. You can enter any value between 0 and 1. A value of 0.5 is 90 degrees, a value of 1 is 180 degrees.

Trail Min Tail Size, Trail Max Tail Size

Sets the size range for particles within the rocket trails.

Trail Glow

Sets the amount of glow from the rocket trail shaders.

Trail Incandescence

Sets the amount of incandescence from the rocket trail shaders.

Sparks Min Tail Size, Sparks Max Tail Size

Sets the size range for particles within rocket bursts.

Sparks Glow

Sets the amount of glow from the burst sparks shaders.

Sparks Incandescence

Sets the amount of incandescence from the burst sparks shaders.

Display Geometry

Switches between displaying particles only and displaying the cones instanced to each particle. In order to create lighting and glow when you render fireworks, there is a cone instanced to each particle, and a shader connected to each cone. During playback, the cones display more slowly than the particles. To increase the playback speed, turn off Display Geometry so that only particles appear. Before you render, turn Display Geometry back on.

Edit rocket positions, timing, and colors

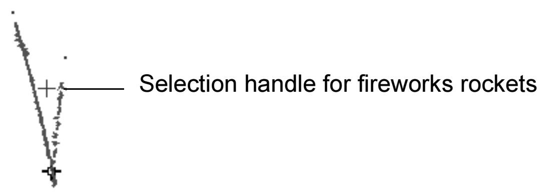
You can edit the following on a per rocket basis:

- positions of the burst and launch
- frames when launch and bursts occur
- colors of the trails and bursts

To make these edits, you must first select the rocket attributes in the Attribute Editor, FireworksRocketShape node.

To select rocket attributes

1. Select the rockets selection handle, placed in the center of the fireworks rockets.

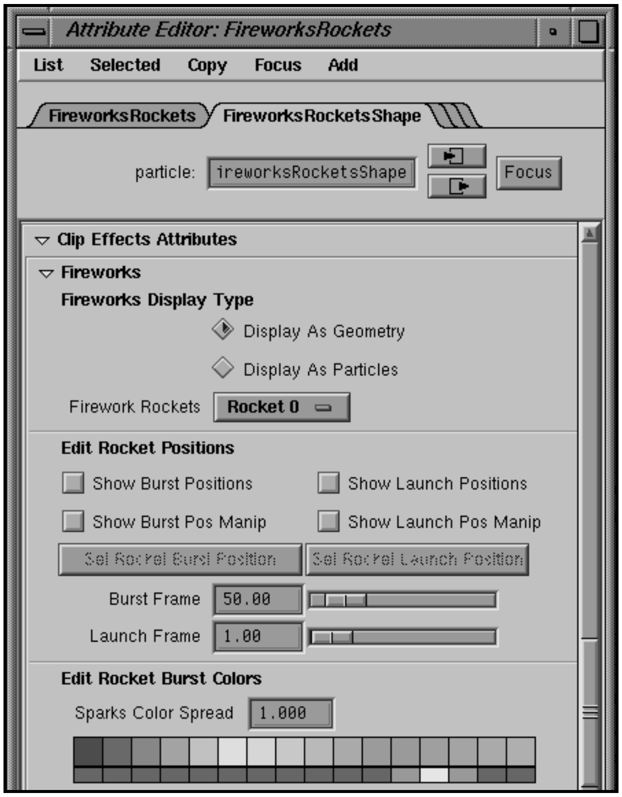


   If the rockets selection handle becomes hidden behind the rocket trails, try drawing a selection box in the center of the Fireworks effect.

2. Open the Attribute Editor.
3. Turn off Auto Load Selected Attributes, under the List menu.

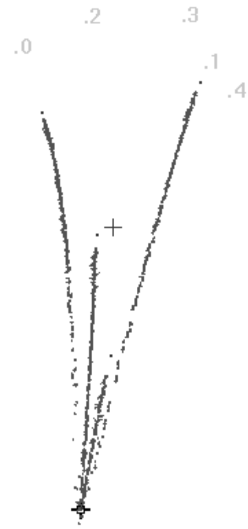
   The auto load option must be off to prevent the wrong node from displaying.

4. Click the FireworksRocketsShape node in the Attribute Editor.
5. Navigate to the Clip Effects Attributes, Fireworks section.



6 Select the rocket you want to edit in the Firework Rockets pull-down list.

You can see the number of each rocket by turning on the Show Burst Positions or Show Launch Positions checkbox.



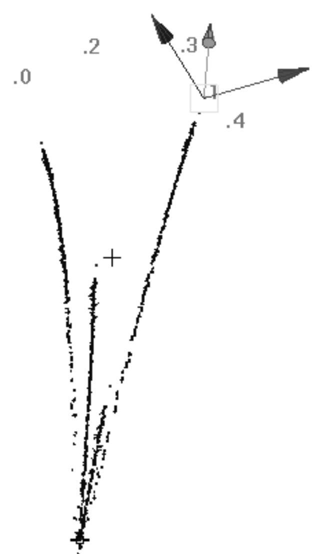
You can see rocket numbers by turning on Show Burst Positions

To edit rocket burst or launch positions

1 Select the rocket attributes as explained in "To select rocket attributes" on page 327.

   Remember to click Load Attributes if the Attribute Editor is not refreshed.

2 Turn on Show Burst Pos Manip or Show Launch Pos Manip.



Note	If the node in the Attribute Editor changes, you must turn off Auto Load Selected Attributes, under the List menu. To redisplay the FireworksRocketShape node, select the rocket group and click Load Attributes.

3 Move the burst or launch manipulator as needed.

4 Click Set Rocket Burst Position or Set Rocket Launch Position once the manipulator is in position.

5 If the rockets have already been drawn, replay the animation from frame one to see your changes.

To edit rocket burst or launch times

1 Select the rocket attributes as explained in ”To select rocket attributes” on page 327.

Remember to click Load Attributes if the Attribute Editor is not refreshed.

2 Edit the Burst Frame or Launch Frame attributes.

Maya tries to satisfy the burst time as close as possible by computing an appropriate velocity and trajectory for the rocket.

To edit rocket colors

1 Select the rocket attributes as explained in ”To select rocket attributes” on page 327.

Remember to click Load Attributes if the Attribute Editor is not refreshed.

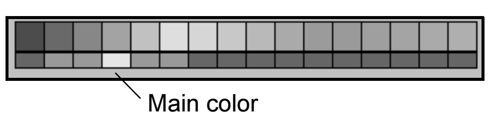
2 Edit the color attributes described below.

Sparks Color Spread

Sets the number of colors used in the rocket’s burst sparks. The spread appears on either side of the main color you choose in the sparks color palette.

Sparks color palette

Enables you to select a spread of colors for the rocket’s burst sparks. Click the rectangle under the main color you want, and the other colors in the spread are chosen from either side of it.



By default, the palette consists of colors chosen in consecutive order around the color wheel. You can change the palette one color at a time by clicking the color square, which opens the Color Chooser. Or, you can use the Remake Color Palette option.

Remake Color Palette

If you want to replace the entire sparks color palette with a custom palette, you can specify a custom MEL procedure name. Maya looks for the custom procedure by searching your user scripts directory for a MEL script with the same name.

Your procedure should include the following syntax:

```
global proc vector[] myFireworksColors( int
$numColors )
```

In this syntax, the argument $numColors specifies the total number of colors requested. The return value should be an array of vectors with the new colors in it.

If you made individual color changes, you can return to the custom or default color palette by clicking Reset from *colorProcedureName*.

Edit Rocket Trail Colors

Shows the spread of colors for the rocket's trail. You can change the palette one color at a time by clicking the color square, which opens the Color Chooser. The number of colors is set in the fireworks options at the time of creation. Trail colors apply to all rockets, not just the rocket you selected.

3 If the rockets have already been drawn, replay the animation from frame one to see your changes.

Create lightning

When you create lightning, you set the *Lightning Creation Controls* in the Create Lightning Effect Options window to determine certain aspects of the lightning. You *cannot* change these once the lightning is created. If you want to change them, you must delete the lightning and recreate it.

You can also set the *Lightning Attributes* in the options window. These attributes can be edited *after* you create the lightning.

To create lightning

1 Add the objects to the scene that you want the lightning to extend between.

2 Select the objects you want the lightning to extend between.

By default, lightning will be created between all the selected objects. If you want to change the Lightning Creation Controls, see ”Set lightning creation controls” on page 332. The order of selection determines the direction of the lightning, depending on the Creation Option you selected.

3 Select Effects > Create Lightning.

A lightning node is created between each selected object.

- To change the look of the lightning, see ”Edit lightning attributes” on page 332.
- To adjust the lightning position, see ”Position lightning” on page 335.

4 Press the play button to see the lightning in motion.

Set lightning creation controls

The Lightning Creation Controls determine certain aspects of the lightning. You set these before you create the lightning.

To set the lightning creation controls

1 Select Effects > Create Lightning > ❐.

The Create Lightning Effect Options window is displayed.

2 Set the following attributes:

❖ ”Effects > Create Lightning” on page 365

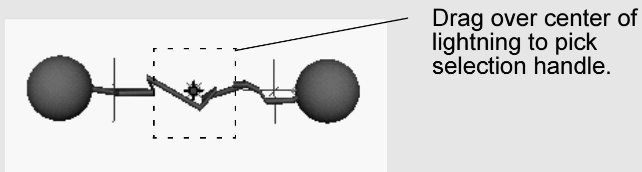
Edit lightning attributes

After you create the lightning, you can adjust the appearance of the lightning by editing the lightning attributes. You can do this using the Channel Box or the Attribute Editor.

To edit lightning attributes

1 Select the lightning you want to edit.

Tip You can select lightning in the Outliner or in the viewport. To select it in the viewport, you must pick the selection handle, which is located under the light icon in the middle of the lightning.



2 In the Channel Box, enter values for the following attributes.

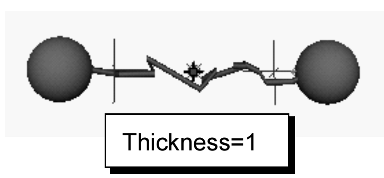
or

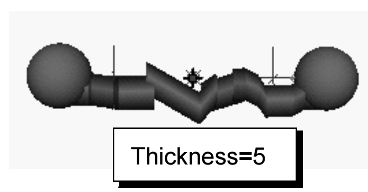
Display the Attribute Editor and click Extra Attributes to display the lightning attributes.

Note In the Attribute editor, you'll notice ten uneditable attributes under Extra Attributes. These display information about the lightning, such as the objects it extends between. You can't edit these; however, you can click the > button to display the attributes for the nodes.

Thickness

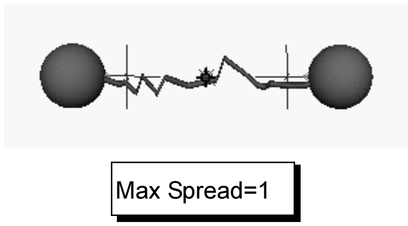
The Thickness attribute determines how fat the lightning is. It specifies the radius of the circle that gets extruded along the curve that makes up the lightning.

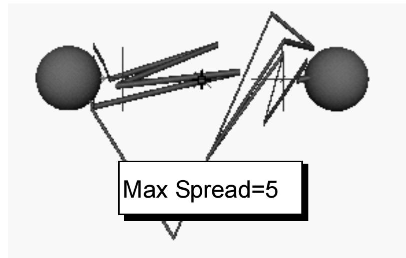




Max Spread

Controls the amount of jitter in the lightning. The greater the value, the more the lightning jitters.

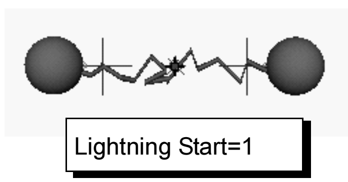


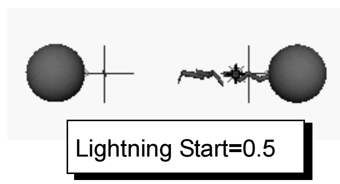


Lightning Start

When you create lightning, the first object you select is the start object and the second object you select is the end object. The Lightning Start attribute determines the point between the start and end objects where the lightning starts.

If you set Lightning Start to 0, the lightning starts at the start object (the first object you selected when you created the lightning). If you set Lightning Start to 0.5, the lightning starts halfway between the start object and the end object. If you Lightning Start to 1, the lightning starts at the end object.





Lightning End

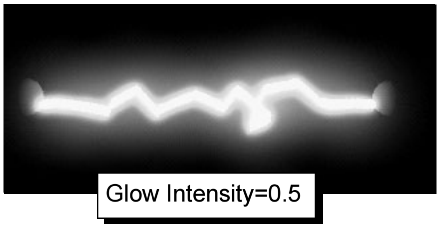
The Lightning End attribute determines the point between the start and end objects where the lightning ends (see “Lightning Start” above).

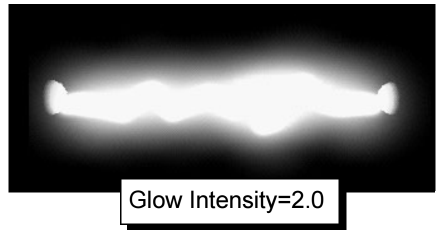
If you set Lightning End to 1, the lightning ends at the end object. If you set Lightning End to 0.5, the lightning ends halfway between the start object and the end object. If you set Lightning End to 0, the lightning ends at the start object.

Tip	You can animate the Lightning Start and Lightning End values to make lightning strike an object, or move from one object to another.

Glow Intensity

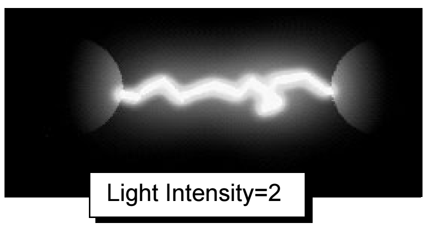
Specifies the brightness of the lightning when rendered.

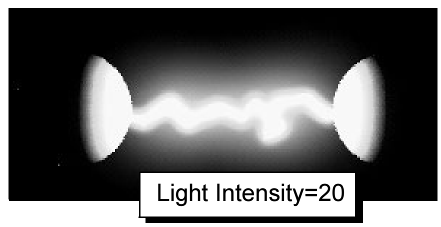




Light Intensity

When you create lightning, a light at the center of the lightning is also created, which casts light on surrounding objects. The default intensity of the light is based on the Glow Intensity attribute, and the difference between the Lightning Start and the Lightning End attributes. The Light Intensity attribute multiplies this default intensity. Increase the value of Light Intensity to increase the lighting of the surrounding objects.





Color R/G/B

Sets the RGB values for the lightning. Adjust the Color RGB attributes to adjust the color of the lightning. The default is 0.5, 1, 1.

Tip	You see the color of the lightning in the RenderView window. If you want to quickly preview the color without using the RenderView window, turn on Lighting > Use All Lights to see the color of the lightning reflected on surrounding surfaces.

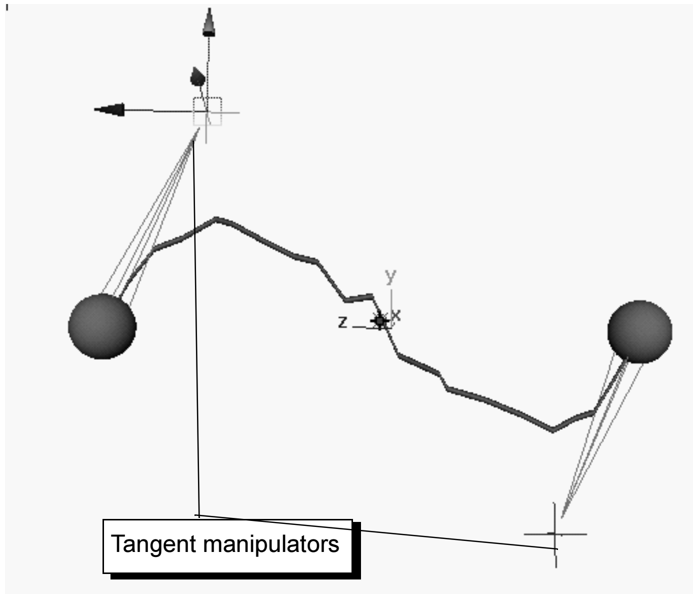
Position lightning

You can adjust the direction the lightning moves away from the start object or toward the end object using the tangent manipulators.

To position the lightning

1 Pick the tangent manipulator for the end of the lightning you want to move.

2 Using the Move tool, drag the tangent manipulator to position the lightning.



Adjust the lightning shader

In the lightning shader, Color, Incandescence, and Glow Intensity are set by the lightning attributes and cannot be changed in the shader. You can adjust the other shader attributes.

Create shatter

Shatter provides three methods of breaking the object:

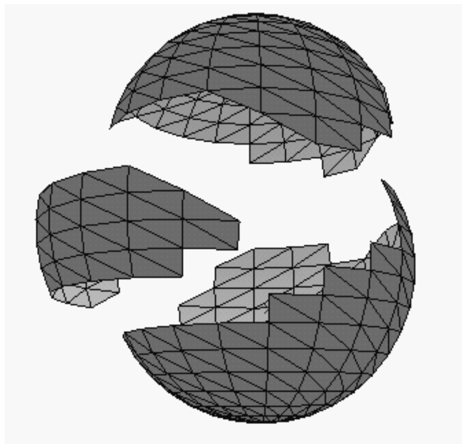
- "Surface shatter"
- "Solid shatter"
- "Crack shatter"

Note If the shards or their parent(s) are moved their values are saved as the initial state for the dynamics simulation. If this is done at a frame other than the start frame the shards may save the results of the dynamics simulation as their intial state.

Rewind the simulation to the start before adjusting the positions of the objects.

Surface shatter

Surface shatter breaks the selected object along polygonal boundaries. You can use surface shatter to break an object into individual polygons.



To create a surface shatter

1 Select the object you want to shatter.

You can select a NURBS or polygonal object. It can be an open or closed surface.

Note	NURBS surfaces are tessellated before the object is shattered. The result is a polygonal object.

2 Select Effects > Create Shatter > ❐.

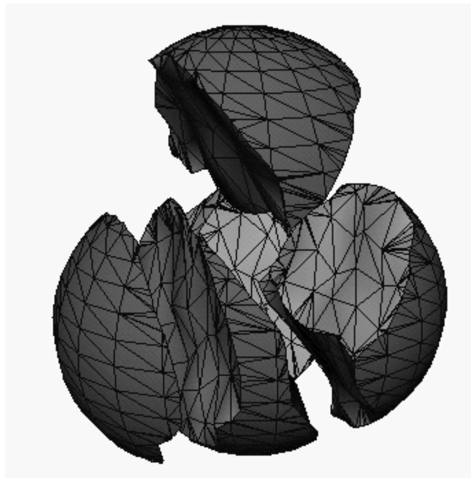
3 In the Create Shatter Effect Options window, click the Surface Shatter tab.

4 Set the options as described in ”Set shatter options” on page 339.

5 Click the Create button.

Solid shatter

Solid shatter breaks the surface of an object but keeps the interior polygons and creates solid pieces. It does not break the object along polygonal boundaries so the edges of the shattered pieces are more realistic.



To create a solid shatter

1 Select the object you want to shatter.

   You can select a NURBS or polygonal object. It must be a closed surface.

2 Select Effects > Create Shatter > ❐.

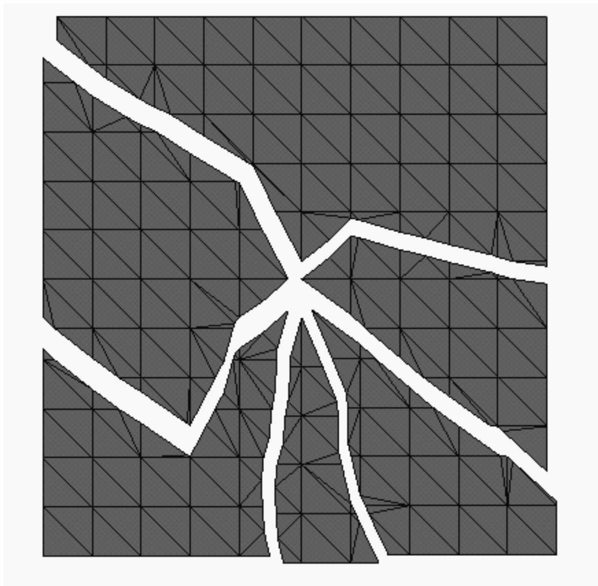
3 In the Create Shatter Effect Options window, click the Solid Shatter tab.

4 Set the options as described in ”Set shatter options” on page 339.

5 Click the Create button.

Crack shatter

Crack shatter creates cracks that radiate from selected points. You use crack shatter on an open polygonal object.



To create a crack shatter

1 Select the object you want to shatter.

You must select polygonal object, and it must be an open surface.

2 Position the pointer over the object, click the right mouse button, and select Vertex from the pop-up menu.

3 Drag to select a vertex.

The cracks will radiate from the selected vertex.

4 Select Effects > Create Shatter > ❐.

5 In the Create Shatter Effect Options window, click the Crack Shatter tab.

6 Set the options as described in "Set shatter options" on page 339.

7 Click the Create button.

Set shatter options

You set the shatter options before creating the shatter. The options vary slightly according to the type of shatter you are creating.

❖ "Effects > Create Shatter" on page 367

Connect shards to fields

Once you've shattered your object, you can move the shards with the move tool and keyframe the motion, or you can connect the shards to a Dynamics field and let the field move the shards.

> Note If you add a field to shards that are shapes, the software automatically converts the shapes to active rigid bodies with collision on. This will cause interpenetration problems with the rigid bodies because the shard's edges overlap each other. Turning off collisions for the rigid bodies solves this problem.

To connect rigid body shards to fields

1. In the Outliner, expand the shatter node.
2. Highlight the shards you want affected by the field.
3. From the Dynamics menu, select Fields > *fieldname*.

   The field is created and connected to the shards.

To connect soft body shards to fields

1. In the Outliner, expand the shatter node.
2. Expand the shard you want affected by the field and select the shard particle shape (shard#Particle).

   If you want all the shards to be affected by the field, you must select the particle shape for each shard.
3. From the Dynamics menu, select Fields > *fieldname*.

   The field is created and connected to the shards.

Link original surface to shards

When you turn on the Link to Shards option, Maya makes several connections from the original surface to the shards. This option lets you control the visibility of the original surface and the shards with one attribute on the original surface's transform node.

Shatter attribute

When you use the Link To Shard option, a Shatter attribute is added to the transform of the original object. The Shatter attribute toggles the visibility of the shattered surface with the visibility of the original surface. When the Shatter attribute is on, the shattered object is visible. When the Shatter attribute is off, the original object is visible. Shatter is turned on by default when you create the shatter so you can see the shattered object.

Rigid bodies

If you have selected rigid bodies with collisions off as the post-operation, and turned on Link To Shards, the shatter node is parented to the original object and the Shatter attribute is connected to the active/passive attribute of the rigid body.

If you've turned on the Make Original Surface Rigid option, the Shatter attribute is also linked to the Ignore attribute of the original rigid body. When the Shatter attribute is turned on, the Ignore attribute of the original object is also turned on. This causes the original surface to be ignored in the rigid solver and prevents the original surface from influencing any other rigid body once the shatter object is visible.

When the Shatter attribute is turned off, the shards become passive rigid bodies. When shatter is on, the shards become active rigid bodies so fields will affect them. You can use this to create an object that shatters when it hits a surface.

To shatter a rigid body on impact

1. Create a surface for the object to hit and make it a passive rigid body.
2. Set the collision layer of the passive rigid body to -1.

   This sets it to the universal collision layer that allows it to collide with every rigid body.
3. Create an object to shatter.
4. Select Effects > Create Shatter > ❐.
5. Select the type of shatter you want.
6. Set the Post Operation to rigid bodies with collisions off.
7. Turn on Link To Shards.
8. Turn on Original Surface Rigid.
9. Click Create.
10. In the Outliner, expand the object you shattered to display the shatter node. Expand the shatter node. Select the shards and Ctrl-click (Linux, IRIX, and Windows) or Command-click (Mac OS X) to select the original object.
11. From the Dynamics menu, select Fields > Gravity.
12. Select the original object. At the start of the animation, keyframe the Shatter attribute to off.

    The original object becomes visible and an active rigid body.
13. Play the animation until the object is about to hit the passive rigid body surface.
14. Keyframe the Shatter attribute on.

The shattered object becomes visible and the shards become active rigid bodies.

15 Rewind and play back.

Soft bodies

If you have selected a soft body post operation, and turned on Link To Shards, the shatter node is *not* parented to the original object. The Shatter attribute is added to the original object's transform node and controls the visibility. When the Shatter attribute is turned on, the shattered object is visible. When the Shatter attribute is turned off, the original object is visible.

A goal weight attribute, which controls the goal weights of the soft body particle shapes, is also added to the transform of the original object. This "control" goal weight is in *addition* to the goal weight that is added to the particle shape of the shard.

When the Shatter attribute is turned off, the goal weight of the particle shape of the shards is set to 1.This causes the shattered object to follow the goal object, which is a copy of the original surface, when it is not visible. When the shatter attribute is turned on, the shattered object becomes visible and the goal weight of each of the particle shapes is set to the value of the "control" goal weight attribute of the original surface. These attributes are connected with an expression.

You can use this to create a soft body that explodes.

To explode a soft body

1 Create an object to shatter.

2 Select Effects > Create Shatter > ❐.

3 Select the type of shatter you want.

4 Set the Post Operation to soft bodies with lattices and goals.

5 Turn on Link To Shards.

6 Click Create.

7 In the Outliner, expand the shatter node, then expand the softBodyLattices# node.Ctrl-click (Linux, IRIX, and Windows) or Command-click (Mac OS X) To select each lattice of each shard (shatter#Lattice).

8 From the Dynamics menu, select Fields > Radial.

9 Select the original object. At the start of the animation, keyframe the Shatter attribute to off.

The original object becomes visible.

10 Select a frame at which you would like to explode the object.

11 Keyframe the Shatter attribute on.

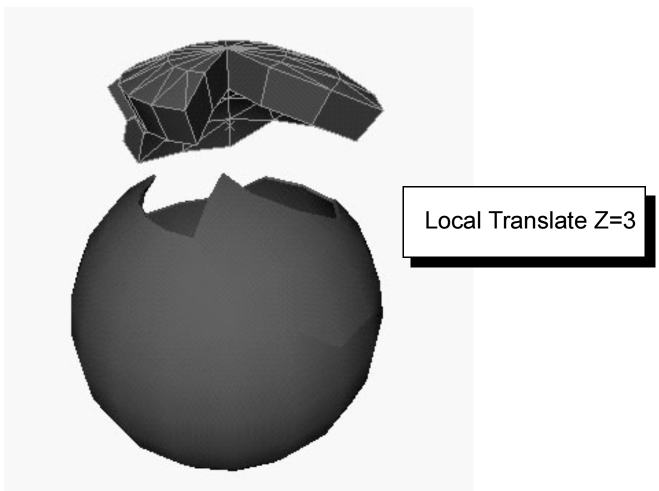
The shattered object becomes visible and the shards become active rigid bodies.

12 Rewind and play back.

Change the extrusion value of a shard

Once you've created a shatter with an extrusion, you can change the extrusion value for all the shards or for individual shards using the Local Translate Z attribute of the shard shape.

If you haven't created the shatter using an extrusion, an extrusion node is not connected to the shards.



To change the extrusion value of a shard

1. Select the shard.
2. If the shard is a rigid body or a shape, In the Channel Box, select shardShape#.
3. In the Channel Box under INPUTS, select polyExtrudeFacet#.
4. Scroll down in the Channel Box to locate the Local Translate Z attribute.
5. Enter a value to change the extrusion.

Set goal weights on soft-body shards

The goal weight influences how much a soft body follows the goal object. You can set the goal weight to a value between 0 and 1. A value of 0 means that the goal's position has no effect on the soft body. A value of 1 moves the soft body to the goal object position immediately.

Values between 0 and 1 cause the soft body to move toward the goal as if bound to it by an elastic spring. The closer to 1, the greater the influence the goal object has on the position of the shards. The goal weight value is 0 by default, which means the goal's position has no effect.

When you select a post-operation option that creates a goal weight but you don't turn on Link to Shards, a goal weight attribute is *not* created on the original shape. The goal weight is an attribute of the particle shape so you must select the particle shape in each shard of the shatter node to display the goal weight attributes.

To set the goal weight on soft-body shards with goals

1. In the Outliner, expand the shatter node.
2. Expand the shard whose goal weight attributes you want to set.
3. Select shard#Particle.
4. Do one of the following:
   - In the Attribute Editor, display the Goal Weights and Objects attributes.
   - Scroll down the Channel box to locate Goal Smoothness, Goal Weight[0], and Goal Active[0].
5. Enter values for the goal weight attributes. For complete information on setting goal weight attributes, see "Goals" on page 21.

To set the goal weight on shards with soft-body lattices and goals

1. In the Outliner, expand the shatter node.
2. Expand the softBodyLattices# node.
3. Expand the shatter#Lattice whose goal weight attribute you want to set.
4. Select shatter#LatticeParticle.
5. Do one of the following:
   - In the Attribute Editor, display the Goal Weights and Objects attributes.
   - Scroll down the Channel box to locate Goal Smoothness, Goal Weight[0], and Goal Active[0].

6 Enter values for the goal weight attributes. For complete information on setting goal weight attributes, see ”Goals” on page 21.

Assign shaders to sets

If you selected sets as the post-operation option, you can assign shaders to the sets.

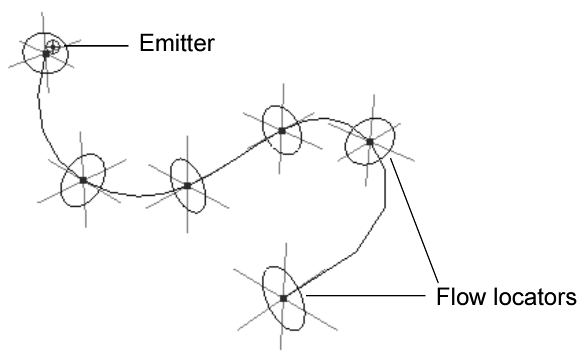
To assign shaders to sets

1 Create the shaders.
2 In the Outliner, turn on Display > Set Members.
3 Expand the *name*ShatterShard# node to display the set members.
4 Select the set members.
5 From the Rendering menu, select Lighting/ Shading > Assign Existing Material > *shadername.*

Create curve flow

To use the Curve Flow effect

1 Select the curve.
2 Select Effects > Create Curve Flow > ❐.
3 Set attributes in the Create Curve Flow Effect Options window (see page 346), then click Create.



An emitter and flow locators appear on the curve. The flow locators are visual aids that show the maximum spread of the particles during animation. For details on altering the speed and spread of the particles, see ”Work with flow locators” on page 346.

4 Play the animation.

Emitted particles flow along the curve. You can move the curve or its CVs to change the direction of flow.

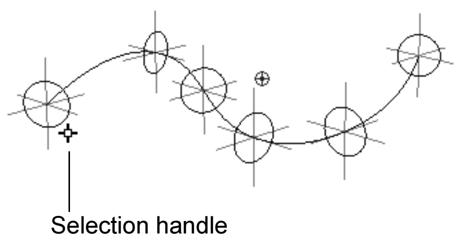
5 Select the emitted particles, and use the Attribute Editor to choose the desired render type, color, opacity, lifespan, and so on.
6 Render the scene with software or hardware rendering, whichever is appropriate for the render type of the particles.

Edit attributes of the Curve Flow effect

The following attributes appear in the Create Fire Effect Options window when you select Effects > Create Curve Flow > ❐. You can edit many of the attributes with the Attribute Editor after you use the Curve Flow effect. Exceptions are noted in the text.

❖ ”Effects > Create Curve Flow” on page 372

To display the attributes in the Attribute Editor, first select the Curve Flow node in the Outliner or workspace (see Curve Flow Group Name below). To select the Curve Flow node in the workspace, select the selection handle near the first flow locator.



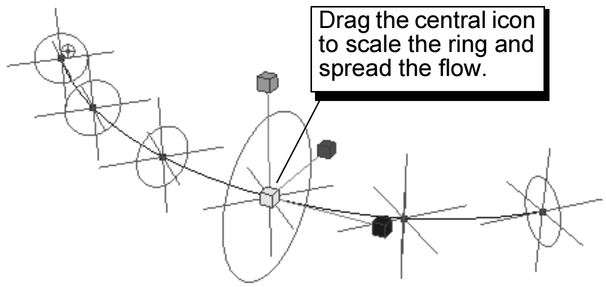
If you use the Curve Flow effect two or more times on the same curve, you’ll see multiple selection handles near the first flow locator.

Work with flow locators

Flow locators control the speed and diameter of the flow in different parts of the curve. You can scale the flow locator rings to expand or shrink the flow diameter in that area. You can move the rings along the curve to speed or slow the flow in that area. This is useful when you’re simulating fluid flow down uneven surfaces.

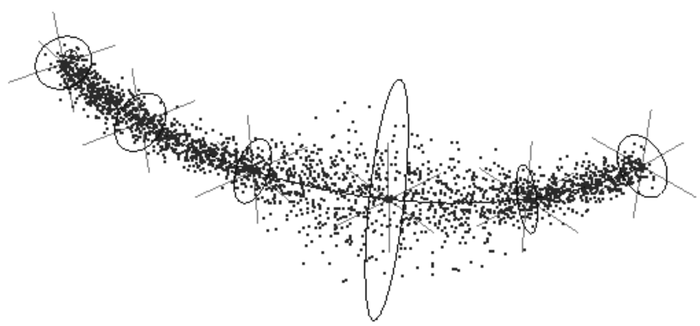
To expand or shrink flow diameter at part of a curve

1 Select the Curve Flow node.
2 Select the flow locator ring without selecting the curve.
3 Select the Scale tool and drag the yellow scale icon to scale the ring along all three axes.



You cannot scale along a single axis. Emission always occurs in a circular region. The ring represents the outer boundary of emission at that point. Maya interpolates the emission diameter between neighboring rings. In other words, the emission diameter increases or decreases smoothly between neighboring rings of different diameters.

An example follows:



To speed or slow flow at an area of a curve

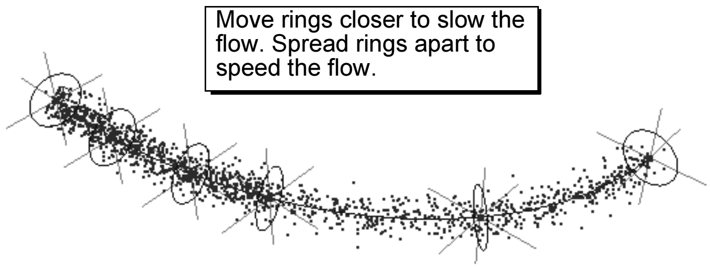
The flow is slower where rings are closer together. The flow is faster where rings are further apart. You can therefore space the rings to adjust the flow speed.

1. Select the Curve Flow node.
2. In the Extra Attributes section of the Attribute Editor, edit the value of the desired locator. The locators are listed in the Attribute Editor in the order of their position on the curve.

   Try different numbers for a locator until you see it move to the desired position.

or

Click the locator name in the Channel Box then slowly middle-drag the mouse back and forth in the workspace until you see the locator move to the desired position.



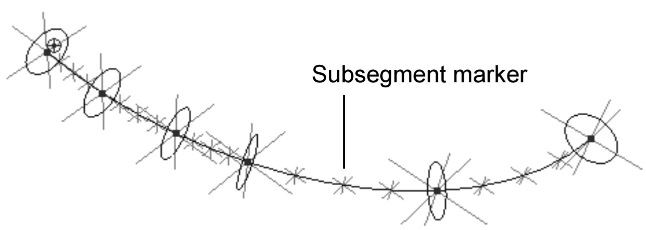
To enhance icon display

1 Select the Curve Flow node.

2 In the Extra Attributes section of the Attribute Editor, set these attributes:

Display Subsegments

When on, displays a marker at each control subsegment for better visual representation of their locations. You can’t change their location directly. You can do so indirectly by moving the flow locators.

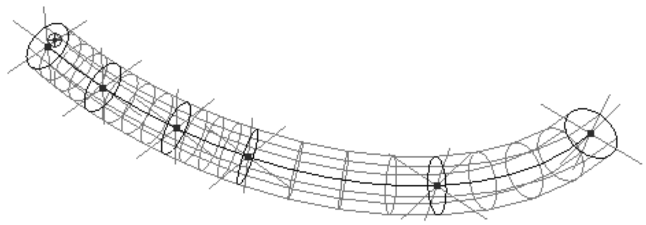


Display All Circles

Displays circles for the control subsegments. This simply makes it easier to see the subsegments.

Display Thickness

Displays circles for control segments and subsegments. Also shows a wireframe that represents the cylindrical region of flow. This gives you a more detailed view of the region of flow.



Create a surface flow

To create a surface flow

1 Select one or more NURBS surfaces.

2 Choose Effects > Create Surface Flow > ❒.

3 Set controls in the option window as needed.

For controls related to creation, see ”Set surface flow creation controls” on page 349.

For controls that you can edit later, see ”Editing a surface flow” on page 349.

4 Click Create in the Create Surface Flow Effect Options window.

An emitter and flow manipulators appear on the surface.

Note	It may take a long time to create a surface flow. Creation time is affected by the number of objects in the scene, by the number of other flows already existing, and by the number of flow manipulators you specify.

5 Play the animation.

6 Render the scene with software or hardware rendering, whichever is appropriate for the render type of the particles.

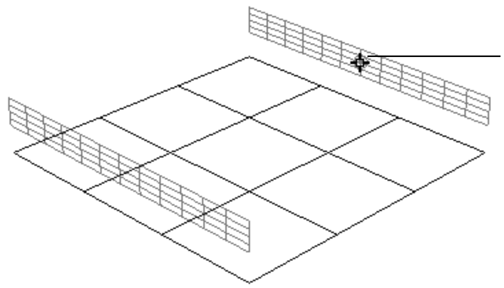
Set surface flow creation controls

The creation controls apply only at the time of creation. *You cannot modify these controls later*.

❖ ”Effects > Create Surface Flow” on page 373

Editing a surface flow

You can edit these attributes after you create the Surface Flow effect by selecting the surface flow selection handle and opening the Channel Box or the Attribute Editor, Extra Attributes section.



> Note You cannot directly rotate or translate manipulators, because they automatically adjust to any changes in the shape of the surface. However, you can edit the Location attributes to move them along the surface.

Display Sub Manips

Turns visibility on or off for the flow sub manipulators.

ShowTrue ManipShape

By default, the manipulators are displayed as smooth bands that wrap around the base surface in exactly the shape that you specify. The *actual* shape that the effect uses depends on the Manipulator Resolution value that you set at the time you created the surface flow.

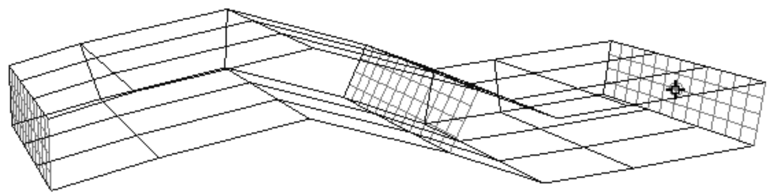
To see that shape, turn on the ShowTrueManipShape attribute. This does not affect the way that the surface flow moves the particles. It simply lets you see how accurate the simulation is trying to be.

Smooth Sub Manips

You can use this option to smooth the transition of sub manipulators between flow manipulators that are spread apart or moved together. If you haven't changed the location of flow manipulators, this option has no effect, because the default positions are smooth already.

Display Min Loft, Display Max Loft, Display Edge Loft

Turns visibility on or off for the flow loft. The loft shows the edges of the flow.



Emitter Rate

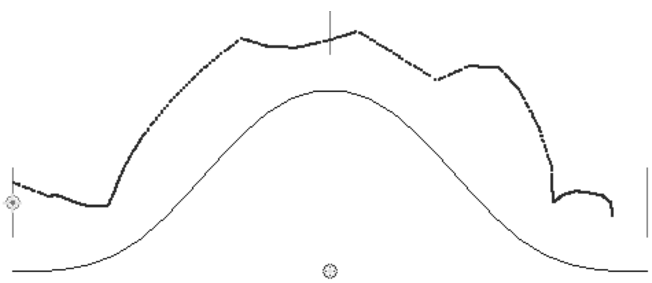
Sets the degree of particle emission in the flow. The emission is applied to the surface area of the first flow manipulator. An emission rate of 0 (zero) means no particles appear.

Random Speed

Sets the amount of random changes in the particle speed. Use this attribute to create a more natural effect.

Random Radius

Sets the amount of random changes in the particle emission radius, so that particles spread beyond the flow manipulator loft. Use this attribute to create a more natural effect.

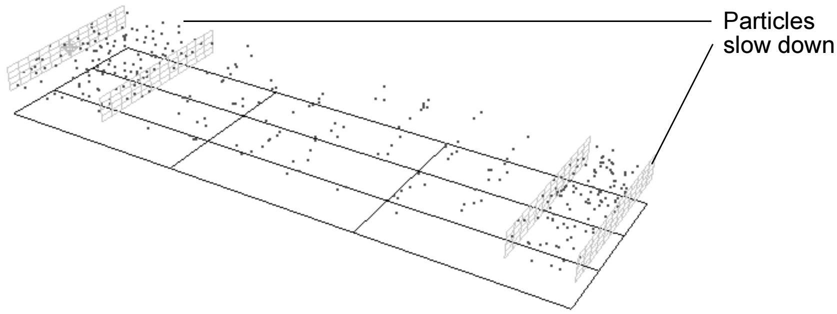


Particles following a random radius.

U or V Location

Controls the position of each flow manipulator, measured in U or V. You can move manipulators anywhere along the surface in any order.

Typically, you adjust the manipulator locations to affect the particle speed. Particles slow down between manipulators that are close and speed up between manipulators that are distant.



You may also need to adjust the location of a manipulator if it is placed at the end of converging threads. For example, a manipulator at the end of a cone would be invisible because the surface at that point is zero.

These values are always between 0 and 1, regardless of the parameterization of the original surface. Therefore, a value of 0.5 will be approximately in the center of the surface.

Min U or V, Max U or V

Controls the beginning and end positions of each flow manipulator, measured in U or V. Adjust these values if you want the flow to cover only part of the surface. For example, you can break up a surface into two adjacent flows by ending one flow (Max setting) at the same point where the second flow begins (Min setting).

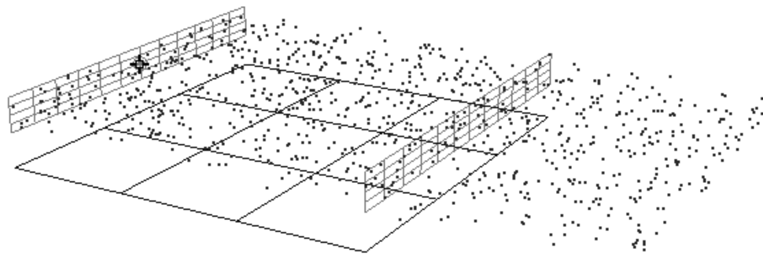
These values are always between 0 and 1, regardless of the parameterization of the original surface. Therefore, a value of 0.5 will be approximately in the center of the surface.

Min Distance, Max Distance

Controls the distance between the surface and the top and bottom of each flow manipulator. By adjusting these values, you can affect not only the proximity to the surface, but also the thickness of the flow.

Min Age Ratio, Max Age Ratio

Specifies what part of the total particle lifespan occurs in the flow—a ratio of flow lifespan to total lifespan. For example, if you set the Max Age Ratio to 0.5, the particles flow along the surface for half their lifespan and then continue beyond the last flow manipulator.



The age ratio also indirectly affects particle speed. In the above example, the particles only have half their lifespan to reach the end of the flow, so they speed up.

Goal Weight

Sets how much all particles in the flow are attracted to the flow manipulators. Values can range from 0 (no weight) to 1 (full weight).

Edit particle lifespan

To edit the particle lifespan, you must select the particle node, not the flow group node, because there may be several particles per flow.

Particle Lifespan

Sets when all particles in the flow die. This option controls the total lifespan, but the Min and Max Age Ratio control the lifespan within the flow. As a result, Particle Lifespan may affect the particle speed in order to meet the requirements of the Min and Max Age Ratio settings.

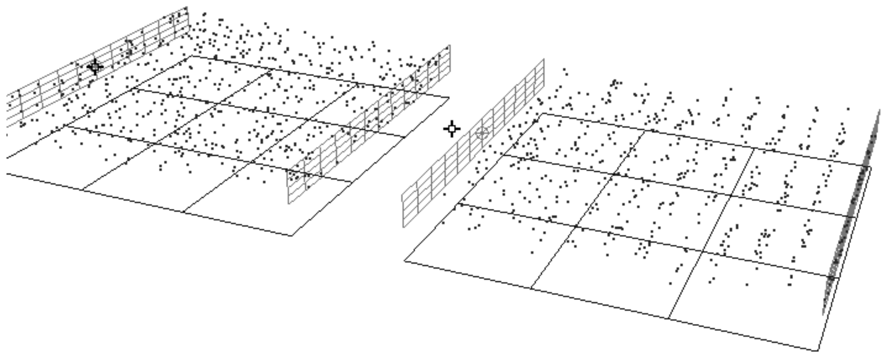
Connect flows on separate surfaces

You can connect flows from separate surfaces so they share the same particles. The workflow is different depending on whether you create flows at the same time or start with an existing flow.

To create connected flows

1. Select the surfaces in the order you want the particles to flow.
2. Choose Effects > Create Surface Flow > ❐.
3. In the options window, turn off Create Particles Per Flow.
4. In the Control Resolution option, specify the number of flow manipulators *per surface*.
5. Click Create.

   The effect creates one flow per surface, with the particles flowing in the order of selection.

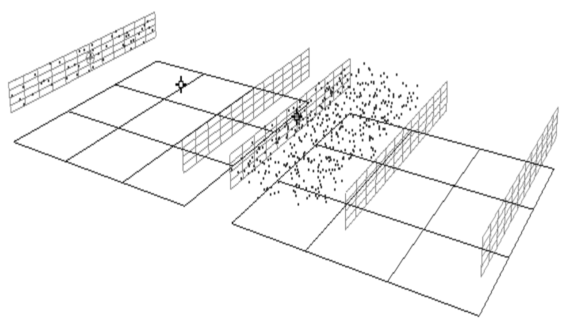


> Note By default, the Min and Max Age Ratio settings are evenly divided among the flows. As a result, a gap appears if the surfaces are separated. Edit the Min and Max Age Ratio settings if you want the particles to fill the gap.

To connect an existing flow

1. Select the particle from the existing flow, along with the surfaces or flows you want to connect. (The selection order doesn't matter.)
2. Choose Effects > Create Surface Flow.

   There will be one flow per surface, but the particles may not flow correctly. Correct the flow of particles using the following steps.



3. For all flows after the starting flow, change the Emitter Rate to 0 (zero).
4. In all the flows, adjust the Min and Max Age Ratio settings so the particle life span extends evenly among all flows.

   The following table gives example settings for two flows.

Flow	**Min Age Ratio**	**Max Age Ratio**
A	0	.5
B	.5 or higher	1

Delete flows

Because surface flows create expressions in the affiliated particle objects, you should not delete flows using the Delete key or Edit > Delete option. Follow the instructions below.

To delete a flow

1. Select the flow or flows you want to delete.
2. Choose Effects > Delete Surface Flow > ❐.
3. Edit the Delete Surface Flow Effect Options as needed.

❖ "Effects > Delete Surface Flow" on page 376

4. Click Delete.

To resolve incorrect deletion

If you accidentally delete a flow with the Delete key or Edit > Delete option, follow these instructions to clean up the nodes in your scene.

1. Delete the particles associated with the flow.
2. Remove the shaders for the particles by opening the HyperShade or Hypergraph and choosing Delete Unused.

Surface Flow procedures

There are several global procedures provided with the Surface Flow effect that allow you to query information about the surface flows in your scene.

int isSurfaceFlow(string $flow)

Returns 1 if the flow object you specify is the root object of a surface flow effect. Otherwise, it returns 0.

string[] surfaceFlowParticles(string $flow)

Returns the names of all of the particle objects that are affected by the surface flow you specify.

int isParticleInSurfaceFlow(string $flow, string $particle)

Returns 1 if the particle object you specify is affected by the surface flow you specify. Otherwise, it returns 0.

string[] surfaceFlowsDrivingParticle(string $particle)

Returns a string array containing the list of all surface flows in your scene that affect the particle object you specify.

string[] selectedSurfaceFlows()

Returns all of the surface flow objects from the selection list in the order they were selected.

string[] selectedSurfaceFlowsAndSurfaces()

Returns all of the surface flow objects and NURBS surfaces from the selection list in the order they were selected.

string surfaceFlowActualSurface(string $flow)

Returns the name of the NURBS surface associated with the surface flow you specify. This is the surface selected at the time the surface flow was created.

string surfaceFlowReferenceSurface(string $flow)

Returns the name of the reference NURBS surface associated with the surface flow you specify. At the time of creation, the surface flow effect makes an exact copy of the selected surface to use as a reference. All objects created for the effect fall under the reference object rather than cluttering the actual surface node.

string[] surfaceFlowLofts(string $flow)

Returns the names of the edge, min, and max lofts used to define the boundary for the surface flow you specify.

string[] surfaceFlowEdgeCurves(string $flow)

Returns all of the curves used to calculate the edge loft for the surface flow you specify.

string[] surfaceFlowMaxCurves(string $flow)

Returns all of the curves used to calculate the max loft for the surface flow you specify.

string[] surfaceFlowMinCurves(string $flow)

Returns all of the curves used to calculate the min loft for the surface flow you specify.

string[] surfaceFlowCurves(string $flow)

Returns all of the curves used to calculate all of the lofts for the surface flow you specify.

string surfaceFlowEmitter(string $flow)

Returns the name of the emitter that is attached to the first manipulator of the surface flow you specify.

string surfaceFlowGoal(string $flow)

Returns the name of the goal object being used to control the motion of the particles associated with the surface flow you specify.

string[] surfaceFlowManips(string $flow)

Returns the names of all of the manipulators for the surface flow you specify.

string[] surfaceFlowRamps(string $flow)

The surface flow effect is achieved through the manipulation of ramp textures and goal objects. This procedure returns the names of all of the ramp textures used by the surface flow you specify.

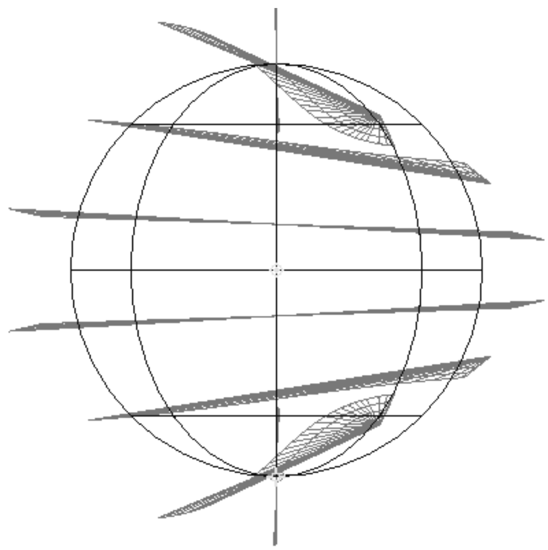
The number of ramps created depends on the values of the three resolution options used to create the effect.The ramps are used as a part of the construction history for the animation of the effect not as a part of a rendering network. The ramps don't appear in the Texture sections in the Visor, HyperShade, or the MultiLister because they can easily outnumber the rendering textures in the scene. The ramps do appear in the networks drawn in the main windows of the HyperGraph and HyperShade because they are useful parts of those networks.

What went wrong?

Dynamics

Avoid twists in the flow manipulators

If your flow moves up or down a periodic surface, which wraps around 360 degrees, the flow manipulators may twist or pop into the surface. An example appears in the following illustration.



To avoid the twisting effect

The following steps may help reduce the twisting effect.

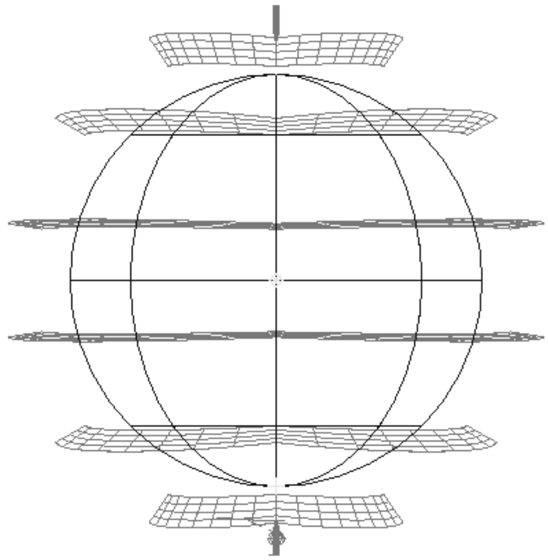
1. Change the sweep used to create the surface from 360 to 359.9 degrees.

   This change prevents the manipulators from popping into the surface.

2. Use several, shorter flows around the surface instead of one large flow.

   For example, the sphere in the following illustration has four flows, with the Min and Max flow values adjusted according to the following table.

Flow	Min U or V	Max U or V
A	0	.25
B	.25	.5
C	.5	.75
D	.75	1



Reference Menus

Dynamics menu set

Effects

Effects > Create Fire

Creates the fire effect.

Effects > Create Fire > ❐

Sets the options when creating a fire effect. Here are the options:

Object On Fire

Name of the object to which the Fire effect will be applied. Omit this entry if you've selected an object in the workspace.

Fire Particle Name

Name of the emitted particle object. If you provide no name, Maya uses a default name.

Fire Emitter Type

Click the pull-down menu to select Surface, Omni-directional Point, Directional Point, or Curve emission. *You cannot change this option after creation.*

To emit from CVs, vertices, edit points, particles, or a position not associated with an object, you must select Omni-directional Point or Directional Point. You can select Surface only for NURBS or polygonal surfaces. You can select Curve only for NURBS curves. See ”Change emitter attributes with workspace manipulators” on page 90 for more details.

Fire Density

Sets how much flame appears in relation to transparent space. This affects the brightness of the flames.

Flame Start and End Radius

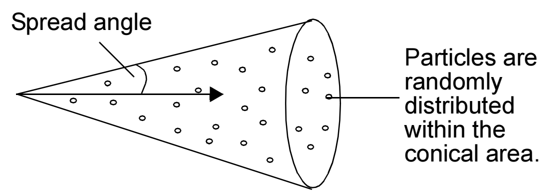
The Fire effect displays emitted particles as the Cloud particle render type. These attributes set how large the radius of each particle cloud is at the start and end of its lifespan. This controls the size of the flame and how the size diminishes during its lifetime.

Fire Intensity

Sets the brightness of the fire.

Fire Spread

Sets the emission spread angle. This angle defines a conical region where the particles are emitted randomly. You can enter a value between 0 and 1. A value of 1 means 180 degrees. *Valid for Directional Point and Curve emitters only*.



Fire Speed

Sets how quickly the flames move.

Fire Direction X, Y, Z

Sets the direction the flames move. It also controls the direction of the directional point emitter.

Fire Turbulence

Sets how much the flame speed and direction are perturbed.

Fire Scale

Scales the Fire Density, Flame Start and End Radius, Fire Speed, Fire Turbulence, and Fire Lifespan.

Fire Lifespan

Sets how many seconds each emitted fire particle lives. *Available in the Attribute Editor only (with emitted particle object selected).*

Additional tips

You can alter the fire's appearance by tuning the values of the following attributes of the emitted particle object.

lifespanPP

Controls the per particle lifespan. A creation expression controls this attribute. To edit the expression, select the emitted particles, right-click the lifespanPP attribute box in the Attribute Editor's Per Particle (Array) Attributes section, then select Creation Expression.

radiusPP

Sets the radius of the particle cloud from birth to death using a ramp. To edit the ramp, select the emitted particles, right-click the radiusPP attribute box in the Attribute Editor's Per Particle (Array) Attributes section, then select arrayMapper1.outValuePP > Edit Ramp.

Effects > Create Smoke

Creates the smoke effect.

Effects > Create Smoke > ❐

Sets the options when creating a smoke effect. Here are the options:

Sprite Image Name

Identifies the filename (with extension) of the first image of the series to be used for the smoke. To use the default images supplied with Maya, enter the name Smoke.0.

To use files from a directory other than the sourceimages directory in your current project, you can specify the complete path as in this example:

(Mac OS X) `/Users/username/Documents/maya/projects/tripper/sourceimages/haze.0`

(Linux, IRIX) `/usr/maya6.0/projects/tripper/sourceimages/haze.0`

The images must have filenames in the format *name*.1, *name*.2, *name*.3, and so on.

To choose a different filename after you use the Smoke effect, display textures in the Hypershade and double-click the file1 texture to display the Attribute Editor. The Attribute Editor has an attribute named Image Name that specifies the filename and path.

Smoke Particle Name

Names the emitted particle object. If you do not provide a name, Maya gives a default name to the object.

Cycle Images (Cycle Enabled)

If you turn on Cycle Images, each emitted particle cycles through the series of images over the course of its lifetime. If you turn off Cycle Images, each particle picks one image and uses that throughout. You can turn this attribute on or off in the emitted particle shape node.

Start and End Image

Specifies the numerical file extension of the starting and ending image of the series. The extension numbers in the series must be continuous.

Smoke Sprite Min and Max Lifespan

The particles live a random time uniformly distributed between the Smoke Sprite Min and Max Lifespan values. For example, if Min Lifespan is 3 and Max Lifespan is 7, each particle lives from 3 to 7 seconds. To change the height of the smoke without affecting speed, adjust these attributes.

Smoke Threshold

When each particle is emitted, it has an opacity of 0. The opacity gradually increases, hits a peak, then tapers off to 0 again. The Smoke Threshold sets the moment the opacity hits its peak, specified as a fraction of the particle's lifespan. For example, if Smoke Threshold is 0.25, each particle's opacity peaks at one-quarter of its lifetime.

Smoke Opacity

Scales the entire smoke opacity from 0 to 1. The closer to 0, the lighter the smoke. The closer to 1, the denser the smoke. The expression created by the Smoke effect controls opacity on a per particle basis; it uses the value of this attribute.

Effects > Create Fireworks

Creates the fireworks effect.

Effects > Create Fireworks > ❐

Sets the options when creating a fire effect. Here are the options:

Fireworks Name

Specifies the name of the fireworks object.

Rocket Attributes

Num Rockets

Specifies the number of rocket particles that fly up and burst. You cannot add or remove rockets once the Fireworks effect is created. You need to choose Create Fireworks again if you want more or less rockets.

Launch Position X, Y, Z

Specify the launch coordinates used to create all the firework rockets. These parameters are used only at creation time; you can later specify different launch positions for each rocket.

Burst Position Center X,Y,Z

Specify the coordinates of the center position around which all of the rockets burst. These parameters are used only at creation time; you can move the burst positions later.

Burst Position Extents X,Y,Z

Specify the size of the rectangular volume containing the random burst positions. Each extent is the distance from the Burst Position Center setting. These parameters are used only at creation time; you can individually move the burst positions later.

First Launch Frame

Sets the frame when the first rocket launches. Subsequent launch times are determined by the Launch Rate (Per Frame) field. This parameter is used only at creation time; you can specify individual rocket launch times later.

Launch Rate (Per Frame)

Sets the rate of rocket launches after the first launch. The rate formula is one divided by the number of frames per rocket. For example, 0.1 means one rocket launches every 10 frames. This parameter is used only at creation time; you can specify individual rocket launch times later.

Min, Max Flight Time (Frames)

Set the time range between the launch and burst of each rocket. These parameters are used only at creation time to randomly distribute the length of each rocket's flight. Later, you can specify each rocket's launch and burst frame.

Max Burst Speed

Sets how fast all rockets burst, and, consequently, how wide the bursts appear. Each particle in the burst has a random speed, and this speed is the maximum. This parameter is typically set at creation time, but you can edit it later.

Rocket Trail Attributes

Set Color Creation Proc

Turn this checkbox on to specify your own MEL procedure that creates the colors used for all rocket trails. Leave this checkbox off to use the default color creation procedure. Using your own color creation procedure can save you time editing it later in the Attribute Editor. *You cannot change this option after creation.*

Color Creation Proc

If you turned on Set Color Creation Proc, this parameter lets you specify a custom MEL procedure name to create a custom trail color palette. Maya looks for the custom procedure by searching your user scripts directory for a MEL script with the same name. *You cannot change this option after creation,* although you can individually edit the trail colors.

Your procedure should include the following syntax:

```
global proc vector[] myFireworksColors( int $numColors )
```

In this syntax, the argument $numColors specifies the total number of colors requested. The return value should be an array of vectors with the new colors in it.

Num Trail Colors

Sets the number of different colors for rocket trails. By default, trail colors are chosen in the yellow to orange spectrum. You can change the colors used, *but you cannot change the number of colors* after the firework is created.

Rocket Sparks creation controls

Min Sparks Count, Max Sparks Count

Each burst consists of a number of streaks randomly distributed between these two values.

Num Spark Colors

Sets the number of different colors in your burst sparks palette. The *actual* number of colors used is set by the Sparks Color Spread option (see "Sparks Color Spread" on page 325 for details). By default, the burst sparks palette colors are chosen in consecutive order around the color wheel. You can change the colors used, *but you cannot change the number of colors available* after the firework is created.

> Note For information on the Color Creation Proc controls, see the Remake Color Palette attribute on page 331.

Effects > Create Lightning

Creates the lightning effect.

Effects > Create lightning > ❐

Sets the options when creating a lightning effect. Here are the options:

Group Lightnings

When Group Lightning is turned on, Maya creates a group node and places the newly created lightning into it. No special controls are associated with this group. However, grouping can help you organize your lightning effects.

Creation Options

Specifies how you want the lightning created. Click to turn on one of the following options:

All

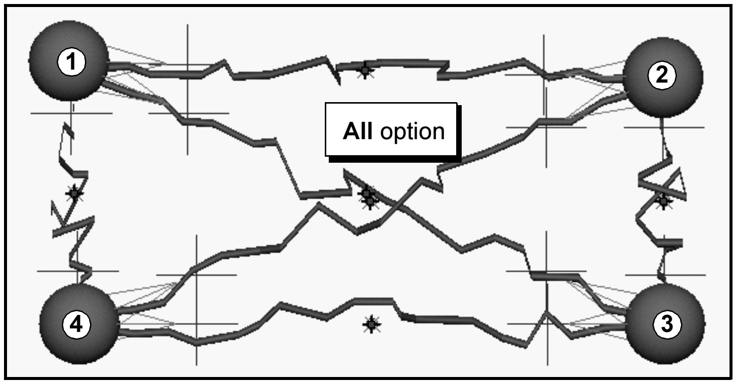
Creates lightning between all the selected objects.

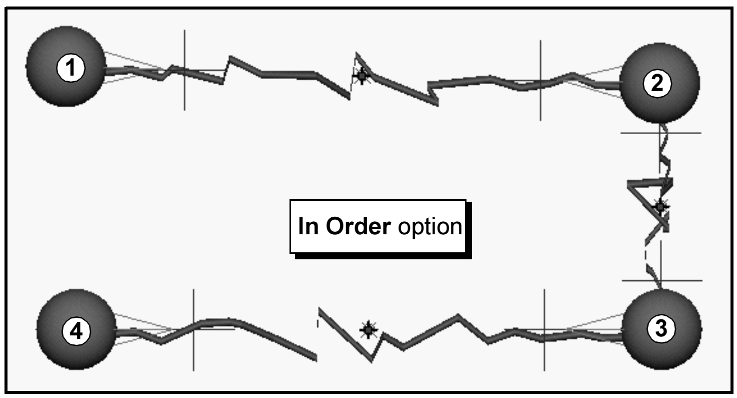
In Order

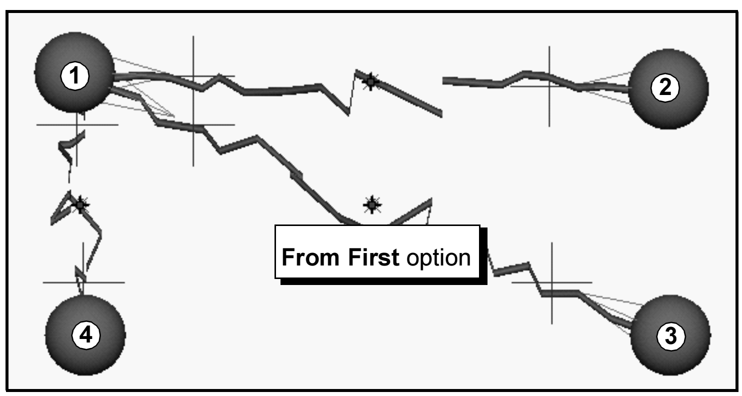
Creates lightning from the first selected object to the other selected objects in the order of their selection.

From First

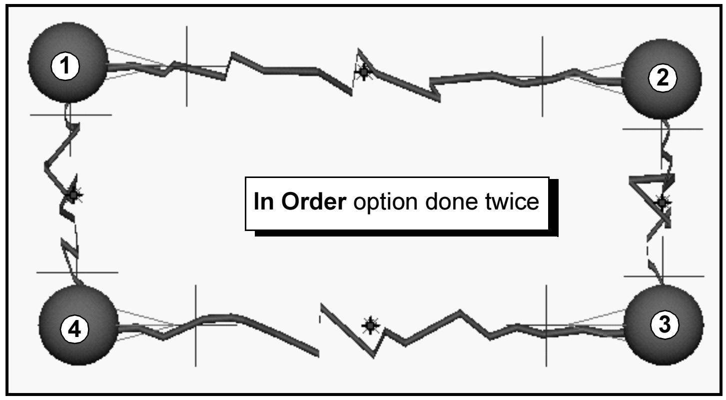
Creates lightning from the first object you select to all the other selected objects.





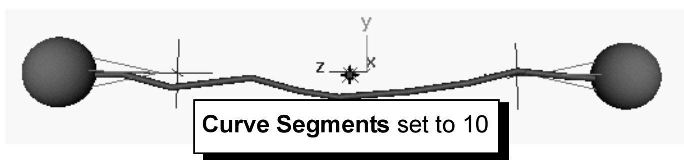


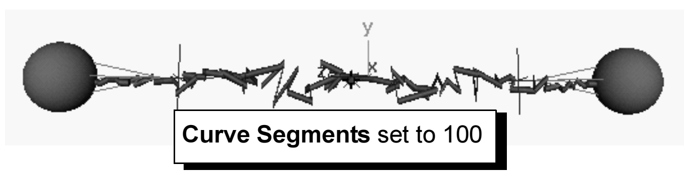
Tip	Notice in the illustration for the In Order option above, the lightning does not extend from object 4 back to object 1. If you want to do this, you need to add another lightning effect between objects 4 and 1.



Curve Segments

Lightning is made up of soft body curves with extruded surfaces. The Curve Segments value sets the number of segments in the lightning.





Effects > Create Shatter

Creates the shatter effect.

Effects > Create Shatter > ❐

Sets the options when creating a shatter effect. Here are the options:

Shard Count

Specifies the number of pieces the object is broken into. For surface shatters, if the Shard Count is greater than the number of polygons in the object, Maya creates a shard out of each individual polygon. High Shard Count values may not create the exact number of shards.

Crack Count

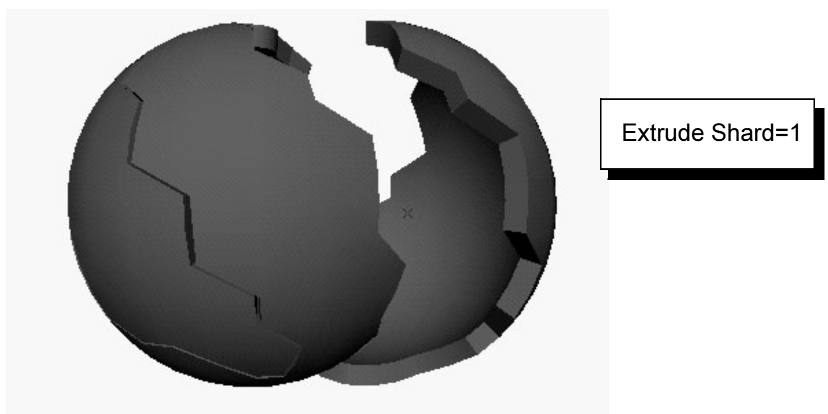
(*Available for Crack Shatter only.*) Specifies the number of cracks created in the shatter.

Crack Length

(*Available for Crack Shatter only.*) Specifies the length of the crack. If the Crack Length exceeds the distance from the selected vertices to the edge of the surface, the crack terminates at the edge.

Extrude Shards

Specifies a thickness for the shards. A positive value pushes the surface outward to create thickness; a negative value pushes the surface inward. You can edit the extrusion amount after you create the shatter, see ”Change the extrusion value of a shard” on page 343.



Edge Jagginess

(*Available for Solid and Crack Shatters only.*) Specifies the unevenness of the shattered edges. A value of 0 creates smooth edges. A value of 1 creates jagged edges.

Seed Value

Specifies a value for the random number generator. If you set Seed Value to 0, you get a different shatter result each time. If you set Seed Value to a number greater than 0, you get the same shatter result.

Post Operation

The post operation options determine what kind of shard you create. The options vary slightly according to the type of shatter you have selected. Select one of the following from the pop-up menu:

shapes	Breaks the object into shapes called shards. Once the object has been broken into shapes, you can apply any type of animation to the shards, such as keyframing. If you plan to animate the shards using Dynamic forces, you might want to use the rigid body or soft body post operation options.
cracks on surface	(*Available for Crack Shatter only.*) Creates crack lines without actually breaking the object. You can use this option to quickly find crack patterns you like, then repeat the crack shatter using the desired post operation option. To do this, you pick the same vertex each time while changing the Seed Value to values greater than 0. To repeat the same cracks, set the Seed Value to the same value it was for the desired cracks, and pick the same vertex.
rigid bodies with collisions off	Breaks the object into rigid bodies. Collisions are turned off to prevent interpenetration errors from occurring while the shards are touching. You can apply dynamic forces to move the shards, and turn collisions on when the shards are no longer touching (see ”Connect shards to fields” on page 340). For complete information on effects you can create with rigid bodies, see ”Rigid bodies” on page 246.
soft bodies with goals	Breaks the object into soft bodies, which deform when you apply dynamic forces. The software also adds goal objects, which are copies of the shards that you can use to more precisely control the soft body deformations. A goal weight of 1 forces your shards to maintain their original shape. See ”Set goal weights on soft-body shards” on page 344 for more information. Springs are also created, which add internal structure to the soft bodies and improve your control over the deformation. For complete information on effects you can create with soft bodies, see ”Soft bodies” on page 245. For complete information on effects you can create with goals, see ”Goals” on page 21. For complete information on springs, see ”Springs” on page 249.

soft bodies with lattice and goals

Breaks the object into shards. The software adds a lattice deformer to each shard and makes the lattices soft bodies. When lattices are soft bodies, they can be affected by fields. The lattices deform the shards according to the fields affecting the lattices.

Goal objects are also added, which are copies of the lattices that you can use to more precisely control the soft body deformations. A goal weight of 1 forces your lattices to maintain their original shape. See "Set goal weights on soft-body shards" on page 344 for more information. In addition, springs are added to the lattices, which add internal structure to the lattices and improve your control over the deformation.

Using lattices to deform your shards can be faster, especially if you have complex geometry. It's also easier to add springs to lattices, creating a more uniform jiggliness. However, it you want the shards to collide, you may get unrealistic results because the collisions occur between the lattices, not the shards.

For complete information on effects you can create with soft bodies, see "Soft bodies" on page 245. For complete information on effects you can create with goals, see "Goals" on page 21. For complete information on springs, see "Springs" on page 249. For complete information on lattices, see "Using Lattice Deformers" in *Character Setup*.

Sets

(*Available for Surface and Crack Shatters only.)* Puts the individual faces that make up the shard into sets called surfaceShatter#Shard#. When you choose the Sets option, Maya doesn't actually break up the object; it merely puts the polygons of each shard into sets.

You can use these sets as you would any other set in Maya. For example, you can assign different colors to the sets (see "Assign shaders to sets" on page 345).

Remove Interior Polygons

(*Available for Solid Shatter only.*) Removes the polygons in the interior of the object. When you turn on the Remove Interior Polygons option, the Extrude Shards option becomes available so you can extrude the shards.

Triangulate surface

Triangulates the entire surface. Turn on Triangulate Surface to create planar polygons.

Apply Interior Material

(*Available for Solid Shatter only.*) Creates a new default material called shatterinteriorSG# and applies this shader to the inside surfaces of the shards. The exterior already has the default initialShadingGroup or whatever shader you had on the original object applied to it.

You can use the HyperShade window to edit the material. This option lets you have different shaders on the inside surfaces and outside surfaces of your shards.

Smooth Shards

(*Available for Surface Shatter only.*) Redistributes the polygons among the shards so the shards have smoother edges. Turning on this option results in more processing time.

Original Surface

Specifies what happens to the original object. You can select:

Nothing	Leaves the original object.
Hide	Hides the original object.
Delete	Deletes the original.

Link To Shards

Creates several connections from the original surface to the shards. This option lets you control the visibility of the original surface and the shards with one attribute on the original surface's transform node.

See "Link original surface to shards" on page 340 for more information.

Make Original Surface Rigid

Makes the original object an active rigid body. This option is available only when you have selected rigid bodies as the post operation and turned on Link to Shards. See "Link original surface to shards" on page 340 for more information.

Verbose Mode

Displays messages in the Command Feedback window.

Effects > Create Curve Flow

Creates the curve flow effect.

Effects > Create Curve Flow > ❐

Sets the options when creating a curve flow effect. Here are the options:

Flow Along Curves

Flow Group Name

Name of the Curve Flow node to contain the emitter, emitted particle object and other items added to the scene after you use the Curve Flow effect. To edit the Curve Flow attributes after you use the Curve Flow effect, you can select this name in the Outliner then display the Attribute Editor. If you don’t enter a name, Maya creates a default name.

You can delete all objects created by the Curve Flow effect by deleting the Curve Flow node from the Outliner.

Flow Creation Controls

Attach Emitter to Curve

If on, the emitter created by the Curve Flow effect is point-constrained to the first flow locator on the curve (the locator closest to the first CV of the curve). If off, you can move the emitter to any position. When you rewind and play the animation, the particles flow from the emitter position to the first flow locator. *You cannot change this option after creation.*

Num Control Segments

Sets the number of places on the flow path where you can tune particle spread and speed. Higher numbers give finer manipulator control of spread and speed. Lower numbers improve playback speed. *You cannot change this option after creation.*

Num Control Subsegments

Sets the number of segments between segments. Higher numbers cause particles to follow the curve more precisely. Lower numbers improve playback speed. *You cannot change this option after creation.*

Flow Attribute Controls

Emission Rate

Sets the rate at which particles are emitted per unit of time.

Random Motion Speed

Sets how much the particles meander as they move along the curve. Higher numbers make particles meander more. A value of 0 turns off meandering.

Particle Lifespan

Sets how many seconds each emitted particle exists from the beginning of the curve to the end. Higher values make the particles move slower.

Goal Weight

Each emitted particle has a goal position that it follows as it moves along the path. The goal weight sets how precisely the particles track their goals. A weight of 1 makes the particles follow their goals exactly. Lower values diminish how closely they follow.

Effects > Create Surface Flow

Creates the surface flow effect.

Effects > Create Surface Flow > ❐

Sets the options when creating a surface flow effect. Here are the options:

Flow Along Surface

Flow Group Name

Name of the Surface Flow node to contain the emitter, emitted particle object and other items added to the scene after you use the Surface Flow effect. To edit the Surface Flow attributes after you use the Surface Flow effect, you can select this name in the Outliner then display the Attribute Editor. If you don’t enter a name, Maya creates a default name.

You can delete all objects created by the Surface Flow effect by deleting the Surface Flow node from the Outliner.

Flow Creation Controls

Create Particle

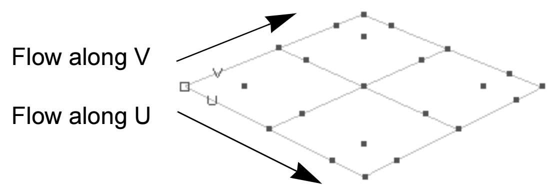
If on, particles are created for the flow on the selected surface. If off, no particles are created. One exception is if you have particles from an existing flow selected. If you do, the selected particles are applied to the new flow, whether this option is on or off. Do this if you want to connect particles from an existing flow with a new one.

Create Particle Per Flow

If you have multiple surfaces selected and you want to create a separate flow for each selected surface, turn this option on. Turn this option off to create one flow across all selected surfaces.

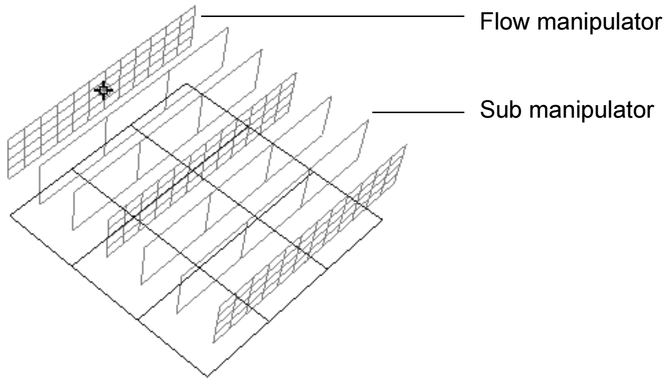
Manipulators Along

Sets the direction of the flow. The direction is specified in the U / V coordinate system, which is local to the surface. Positive U or V is forward, while negative U or V is backward. You can see which direction is U or V by selecting the model and switching to Component mode (but you must be out of Component mode when you create the effect).



Control Resolution

Sets the number of flow manipulators. Flow manipulators let you control particle speed, distance from surface, and other settings for a specified area. By default, all flows have at least two manipulators for the beginning and end. Specify more manipulators depending on how many areas along the surface you want to control. However, the more manipulators you specify, *the longer it takes to create*.



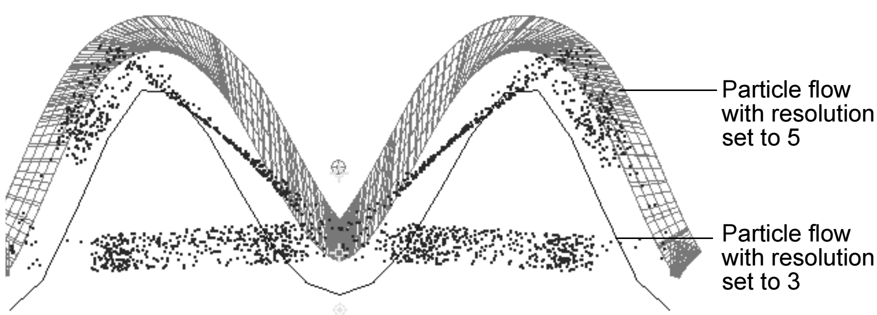
Sub-Control Resolution

Sets the number of sub manipulators between each flow manipulator. Sub manipulators control the flow of particles, but you cannot manipulate them directly. They are automatically spaced between flow manipulators to follow the curve of the surface. The more your surface varies along the direction of flow, the more sub manipulators you should have. However, the more sub manipulators you specify, *the longer it takes to create*.

Manipulator Resolution

Sets the number of analysis points the flow manipulators use to direct particles. The more your surface varies along the manipulator, the greater your resolution should be.

The following illustration shows the front view of a surface with two flows, but different Manipulator Resolution settings. A resolution of 5 created the particles on top, while a resolution of 3 created the particles on the bottom.



Effects > Delete Surface Flow

Deletes a surface flow effect.

Effects > Delete Surface Flow > ❐

Sets the options when deleting a surface flow effect. Here are the options:

Delete Surface Flow Groups

Turn on to remove the nodes of the selected surface flow.

Remove Particles From Surface Flow

Turn on to remove only the particles associated with the flow and not the flow itself.

Delete Surface Flow Particles

Turn on to remove the particle node associated with the flow. If off and you delete the surface flow, the particle node remains in your scene, even though the particles disappear.

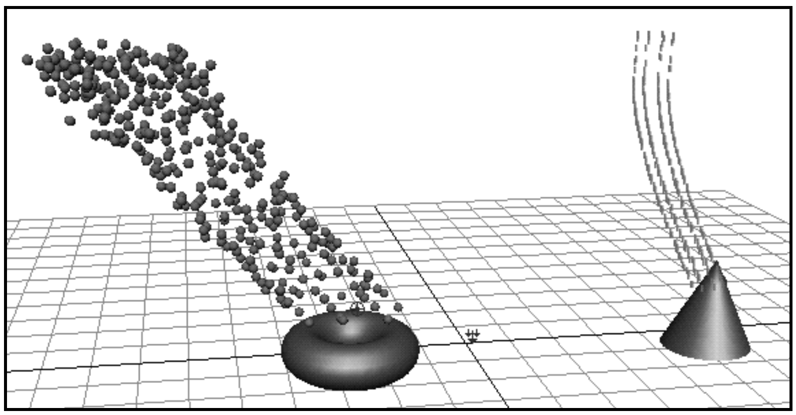
6 Solvers

About Dynamics

Particle caching

Particle caching

Caching lets you store dynamic simulations, either to disk or to memory.



With caching, you can see the correct effects without waiting for playback computation.

Dynamics supports three types of particle caching: particle disk caching, particle startup caching, and memory caching.

Particle disk caching

Particle disk caching lets you cache complete sequences to disk. You can use the cache for batch rendering and playback. Disk caching has the following advantages:

- It lets you render more efficiently, especially when using multi-processor batch rendering. Maya loads the particles from the disk cache instead of re-computing them. This avoids the particle “run-up” at the start of the render.
- It lets you play the scene faster and scrub in the time slider.

You can also use particle disk caching to create different caches for the same scene. This lets you create and quickly play back variations on a scene.

- "Use particle disk caching" on page 379

Particle startup caching

Particle startup caching avoids run-up. It's used when you save your scene at some frame greater than the start frame (it's on by default). The startup cache lets you save your particle data for that one frame to disk so no run-up is needed.

- "Particle startup caching" on page 384

Memory caching

Memory caching saves the motion of objects in memory (not to disk). You can use memory caching to cache certain objects in your scene.

- "Memory caching" on page 385

Particle Disk Caching

Specify where the cache files are stored

Maya has a workspace directory for particles. By default, this is given the name *particles*. You can specify your workspace particles directory using the Project Editor, the same as for all the other workspace directories.

If you are using an old workspace from a previous version of Maya, Maya creates the particles directory for you the first time you create a particle disk cache.

The disk cache files are stored in subdirectories of the particles directory. You can specify what subdirectory name you want to use by entering the name in the Cache Directory box of the Particle Disk Cache Options window (see "Setting particle disk cache options" on page 380). Before each new caching operation, Maya automatically deletes any cache files existing in the specified subdirectory.

Determine the frame range

If Maya can't find a cache file for the given frame, but finds cache files both for frames greater and lesser than the current frame it is evaluating, it interpolates between them. If the frame difference is large, you see the results of interpolation rather than the results of solving frame by frame.

In simple cases, to evaluate a given frame, Maya usually needs the particle cache files only for that frame. However, Maya may need to access the state of particles at some other frame in order to resolve expressions, rigid body motion, motion blur, etc. If you are in doubt, we recommend caching out all frames. For example, if you want to render starting at frame 100, cache all frames up to 100 as well.

How do I? Simulate dynamic effects

Work with particle disk caching

Use particle disk caching

Create a particle disk cache

After you have created a particle disk cache, Maya uses the cached particle data, no matter what changes you make to your scene, until you *explicitly* tell it not to.

For example, if you cache the scene, then apply a gravity field to a particle object, you will not see any effect from that field because Maya is using the cached particle motion. To see the effect, you must turn off Use Particle Disk Cache (see "Editing particle disk cache settings" on page 380).

However, if you make a change that does not alter per-particle attributes, such as changing the render type from blobby to streak, you will see the change without turning off Use Particle Disk Cache.

Tip	We suggest you name your scene before creating a particle disk cache; otherwise, Maya saves the particles in a directory named *untitled*.

To create a particle disk cache

1 Select Solvers > Create Particle Disk Cache.

Maya plays through the scene once using the current time slider playback range and writes out cache files. Maya does not re-draw the screen during this playback. If you want to cache a different range, you can either change the current range of your time slider or use the range in Render Globals (see "Use Render Globals Range" on page 388).

Maya must play the scene through once to create the cache. If you don't want to wait through the playback, you can interrupt it by pressing the Esc key, but the cache won't be created.

2 After creating the cache, save your scene so any settings you made to the Particle Render Cache Option window get saved with the scene. Otherwise Maya might not know to look for your cache or might not be able to find it.

When you create the cache in the above manner, Maya creates a cache for *all* particles in the scene. If you want to cache some particles objects and not others, see "Caching a single particle object" on page 383.

> Note! Be sure to save your scene after creating the cache. Otherwise, Maya won't know about the cache when batch rendering.

Setting particle disk cache options

You can set the following particle disk cache options before creating the cache. These options affect all subsequent caching operations.

To set the particle disk cache options:

1 Select Solvers > Create Particle Disk Cache > ❐.

2 Set the following options:

- "Solvers > Create Particle Disk Cache" on page 387

Editing particle disk cache settings

To edit particle disk cache settings

1 Select Solvers > Edit Oversampling or Cache Settings.

The Attribute Editor for the dynGlobals node is displayed.

2 In the Particle Disk Cache section, set the settings. See "Particle Disk Cache" on page 388.

> Note! Be sure to save your scene after editing any of these values. Otherwise, Maya won't know about them when batch rendering.

Play back different caches

You can use particle disk caching to create different caches for the same scene. This lets you create and quickly play back variations on a scene.

To create and play back different caches

1 Create your scene and name it.

2 Select Solvers > Create Particle Disk Cache > ❐.
3 Enter a name for the Cache Directory or leave the default name, which is the scene name.
4 Click Create to create the cache.
5 Make any changes to your particles.
6 Select Solvers > Create Particle Disk Cache > ❐.
7 Enter a new name for the Cache Directory and click Create.
8 Continue this process to create as many caches as you want.
9 To choose a cache for playback, select Solvers > Edit Oversampling or Cache Settings, and enter the cache name in the Cache Directory box.
10 Press the play button to see the scene play back from the specified cache.

Re-create the cache

Once you have created the cache, Maya uses that cache until you tell it not to. It ignores any changes in emission, forces, etc. on that particle object.

If you change something in the scene that alters per-particle attributes, you must re-create your cache in order to get correct renders. Maya writes out the disk cache only when you select Create Particle Disk Cache. That menu item does not turn on a "caching state." This is an important difference between memory caching and disk caching. Maya does not automatically update or re-create your disk cache.

To re-create your cache

- Select Solvers > Create Particle Disk Cache.

You don't need to re-create the cache if you make a change that does not alter per-particle attributes. For example, you could change a particle shape's render type from blobby to cloud, save the file, and re-render. Or you could change keyframed colorRed/colorBlue/colorGreen attributes. In both these examples, you would not need to re-create the cache because the attributes involved are not per-particle.

Be careful, however, because attributes such as colorRed can indirectly affect per-particle attributes. In the previous examples, if you had a per-particle expression on some particle object that made use of the render type or colorRed as an input, the results of that expression would be affected, so you would need to re-create the cache.

Particle disk cache files

Maya saves cache files for each particle shape for each frame. If oversampling is turned on, it saves multiple disk caches at different times within a frame depending on the oversampling rate.These binary files have the extension .pdc, which stands for particle data cache. Maya uses the *dynExport* command to write these files.

You can't read .pdc files. If you want to output particle data files you can read, see the online documentation in the *dynExpor*t command for information on outputting .pda and .pdb files, and also the *gifts/readpdb* directory. See Appendix A, "Use the PDC File Format"for information on writing your own reader.

Deleting cache files

There is no way to delete cache files or directories from within Maya. Delete them using your operating system.

Caching tips

Ramps

If you are animating per-particle rendering attributes by ramps, and you change those ramps, you'll see the change in playback even if the cache is enabled. This also applies to changes in hardware display/rendering attributes such as pointSize. To avoid any confusion, turn off the Use Particle Disk Cache attribute before you make changes to your scene, and re-create your cache just before you render.

If you are animating any attribute with expressions, or animating rampPosition, rampVelocity, or rampAcceleration with ramps, any changes you make to those expressions or ramps will not be visible in playback as long as the cache is enabled. Turn off Use Particle Disk Cache.

Particle collisions with rigid bodies

If you have a rigid body being driven by particle collisions, and you disk-cache the particles, the rigid bodies will no longer evaluate correctly. In effect, the caching takes away the dynamic properties that allow the rigid bodies to evaluate and they can't detect the collisions any more. To cache such a scene, first bake out the rigid bodies (see "Baking rigid bodies," below), then disk-cache the particles. The same limitation and same workaround apply to in-memory caching as well.

Baking rigid bodies

The particle disk cache always applies to particles only. It does not apply to rigid bodies. In most cases, rigid bodies can compute their positions without needing to do a run-up. If you encounter any problems in this area, you can resolve them by baking out your rigid bodies, using the *bakeResults* command.

For example, to bake out *rigidBody1* for frames 1 to 100, do this:

```
bakeResults -t "1:100" -simulation true rigidBody1;
```

You can't use the *bakeResults* command to bake out particle motion. It bakes out animation channels, such as translateX/Y/Z, not particle positions.

Even if unbaked rigid bodies do cause a run-up, the run-up will go faster if your particles are cached, because the particles are loaded from the disk cache instead of being recomputed.

Caching a sequence in two parts

You may find it necessary to cache two parts of a sequence separately. For example, if you cache frames 1 to 45 in one place, and frames 46 to 90 in another place, you must be sure that the two caches "line up" so there is no discontinuity from 45 to 46.

Make sure that the animation and dynamics the first time you cache are consistent with the second. Make sure you didn't add or change anything, or have something changed for you.

In particular, if you have expressions that use the MEL rand() function, you must make sure your expressions are properly reseeded (see "Reproducing Randomness" in Chapter 5 of *Expressions*).

Caching a single particle object

Using Solvers > Create Particle Disk Cache caches all particle objects in the scene. You can cache a specified object from the command line using the *dynExport* command (see the online Command documentation for information on the *dynExport* command).

You can also get this same result by using your operating system to delete the cache files for all objects you don't want read from the cache.

Caching only certain attributes

By default, all attributes are cached. However, you can cache only certain attributes using the *dynExport* command (see the online Command documentation for information on the *dynExport* command).

Particle startup caching

The particle startup cache lets you save files at frames greater than the start frame without doing a run-up. It saves a copy of the particle shape's attribute values in a different file than the particle disk cache files.

Using the startup cache

Using the startup cache lets you save and quickly re-load scenes at frames greater than the start frames of your particle objects. If you always save your files at the particles' start frames, you can turn off Save Startup Cache for Particles to save disk space.

The disk space used for the particle cache is determined by how many particles are in your scene at the frame saved, and how many attributes they have. Likewise, the time savings at file load is determined by how heavy your scene is and how long it takes to play back. There is no simple formula—you must assess the trade-off between file size and time savings. Because of this, we made this a preference you can turn on or off. It is turned on by default.

To use the startup cache

1. Select Window > Settings/Preferences >Preferences.
2. Select the Dynamics category and turn on Save Startup Cache for Particles.

   When this option is turned on (the default), Maya automatically saves the startup cache for all your particles every time you save the file. Unlike particle disk caches, you don't have to create the startup cache explicitly every time.

   When the file is loaded (in either interactive or batch mode), the particle shape's current state is read from the startup cache and no run-up occurs. If Maya cannot find or read the cache file, it gives you a warning and then runs the scene up. This would happen, for example, if you copied the file into a different workspace and did not also copy the startup cache. If you re-save the scene at that point, Maya re-creates the startup cache.

Location and names of startup cache files

Startup cache files are kept in a subdirectory of your workspace particles directory. Each scene has a separate startup cache directory.

Normally, Maya names this directory by appending *_startup* to your scene name. For example, if your scene is called *myScene*, Maya puts the startup cache in a directory called *myScene_startup*.

However, if you have created a particle disk cache for that scene, Maya uses the name of that particle disk cache directory in place of your scene name. For example, if you store your particle disk cache in a directory *safePlace*, Maya calls the startup cache *safePlace_startup*. (By default, the disk cache directory name is the same as the scene name.)

If you first disk cache the scene under one name, then rename the scene, any startup file written uses the first name (because that's the name your disk cache was created under). We recommend that you name your scene when you start and don't change it.

Using startup caches with batch rendering

Maya uses the startup cache in batch rendering only when you are rendering the exact frame for which the startup cache was saved. It does not use the startup cache when you are rendering other frames. When rendering sequences, or rendering in distributed or multiprocessor fashion, we recommend that you use the particle disk cache and not rely on the startup cache.

Using disk caches and startup caches

If you have both a particle disk cache (a complete sequence) and a startup cache, the particle disk cache takes precedence. Maya searches for caches in the following order:

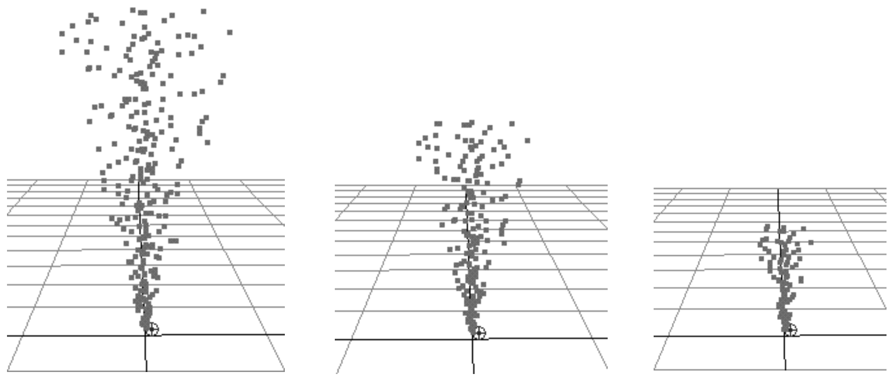
- Maya first looks for a particle disk cache.
- If it can't find a particle disk cache, it looks for a startup cache.
- If it can't find either, it runs the scene up *(only when rendering*).

If you have a particle disk cache, but you want to save a state that is different from what's stored in the particle disk cache, be sure to turn off Use Particle Disk Cache before saving the file. Otherwise, Maya saves what was in the disk cache, using that cache until you specify not to use it.

Tip	If you believe you have rendering problems that are due to the startup cache, create a particle disk cache for the complete sequence.

Memory caching

Memory caching saves the motion of objects in memory (not to disk, see ”Use particle disk caching” on page 379 for information on caching to disk). You can use memory caching to cache certain objects in your scene.



If you enable Memory Caching for a selected dynamic object, you can scrub or play its animation in reverse.

To turn memory caching on (or off) for an object

1 Select the objects.

Note that, for an emitter, you must select the particle object into which the emitter emits. For a soft body, you can select the geometry or the particles.

2 Select Solvers > Memory Caching > Enable.

3 Select Window > Settings/Preferences >Preferences to display the Preferences window.

4 Click Dynamics in the Categories list in the Preferences window.

5 Make sure Run Up to Current Time is turned off.

6 Rewind and play the animation.

Maya caches all attribute values for the selected objects in all frames you play. If you no longer need to scrub or play in reverse you can resume normal frame-by-frame calculations by selecting Solvers > Memory Caching > Disable. Note that if you set the start frame of the Time Slider to a lower number, Maya automatically deletes the cache.

If you change an attribute that affects the translation or rotation of a cached rigid body, you must delete the cache for the animation to play with the changes reflected (see below).

You can alternatively turn memory caching on or off in the Attribute Editor by turning off the Cache Data attribute. The location of the attribute in the Attribute Editor depends on the type of object. If you turn caching on for a rigid body, Maya also turns caching on for all rigid bodies in that rigid body's solver.

If you cache data in memory for emitted particles and later change the rate or another attribute of the emitter or emitted particles, you must disable the cache to see the effect of the attribute change.

To delete the current object cache

1 Select the object.
2 Select Solvers > Memory Caching > Delete.

Reference Menus

Dynamics menu set

Solvers

Solvers > Create Particle Disk Cache

Creates a particle disk cache.

Solvers > Create Particle Disk Cache > ❒

Cache Directory

Specifies where to store the disk cache. See "Specify where the cache files are stored" on page 378.

- If you don't specify a name, Maya uses a directory that matches your current scene name. If that directory doesn't exist, Maya creates it.
- If you specify a name that is different from the current scene name, Maya prompts you to be sure you want to use it. It is generally preferable to use the name of the scene because if two scenes are cached in the same directory, a new caching operation overwrites the old cache.

This same prompt appears if you cache a scene, save the scene under a different name, and re-cache the scene. When you try to re-cache, Maya detects that the cache name now differs from the scene name and asks you to confirm.

- Don’t specify an absolute directory name. Maya considers the cache directory always to be relative to the Particles directory of your workspace. However, you can set the Particles directory of your workspace to an absolute directory name using File > Project > Edit Current.
- The name of the directory used is stored with your scene, so if you change the name of the scene, Maya is still able to find your disk cache. The name is stored in the dynGlobals node (see ”Editing particle disk cache settings” on page 380).

Use Render Globals Range

If you turn on this option, Maya caches particles for the frame range specified in the defaultRenderGlobals instead of for the current playback range. Turn on this option if you plan to use the render globals range in your render.

Only Update particles

If Only Update particles is off (default), the particles will be evaluated by normal DG evaluation, and the cache is guaranteed to have the same result as in interactive playback. If Only Update particles is on, Maya optimizes the evaluation by triggering the evaluation only on particles. This optimization can create a discrepancy if one particle system is dependent on another particle system.

Nodes

dynGlobals

Particle Disk Cache

Use Particle Disk Cache

This determines whether Maya uses the cache. When you create a cache, this attribute is turned on automatically. Turn off Use Particle Disk Cache if you don’t want Maya to use the cache. For example, if you have made changes to certain emitter attributes that affect playback, such as speed or emitter type, turn off this attribute to see the effect of your changes. You can also recreate the cache.

Cache Directory

This attribute lets you specify the directory where Maya looks for the cache. You can use this to choose which of several saved caches you want to use (see ”Play back different caches” on page 380).

For example, you could cache one simulation to a directory named *cache1* and another to a different directory called *cache2*. Enter the name of the Cache Directory you want Maya to use.

This directory is always relative to the Particles directory of your workspace. Don't type an absolute path name.

Entering a name here does not re-create the cache—you do that using Solvers > Create Particle Disk Cache. Entering a name here also does not change the name of your existing cache directory. If you want to rename your directory, use your operating system.

> Tip Solver > Create Particle Disk Cache > ❐ also has a Cache Directory attribute. If you enter a value in the dynGlobals Attribute Editor, that value becomes the default for the option box. Any different value that you enter in the option box is displayed in the dynGlobals node.

Min Frame Cached, Max Frame Cached

These attributes record the minimum and maximum frames that were stored at the most recent caching operation. You can't enter values here—these attributes are for your information only. Because they describe the most recent caching operation, if you choose a different cache directory using Cache Directory, the values in Min Frame Cached and Max Frame Cached won't necessarily correspond to what is stored in that directory.

7 Dynamic Animation

About Dynamics

Animating with dynamics

Animating with dynamics

Particle objects, soft bodies, and rigid bodies are dynamic objects. You typically animate the motion of dynamic objects with dynamic animation—fields, collisions, springs, emission, goals, or particle expressions.

Maya calculates dynamics from frame to frame. The position of an object in each frame is derived from the position in the previous frame. This differs from keyframe animation, where the position of an object at any frame is derived from key values set at different frames in the animation. Because Maya handles dynamics differently than keyframe animation, you must use different techniques to adjust playback.

You can interact with your dynamic simulations as it plays by selecting Solvers > Interactive Playback.

For a scene to start with particles already emitted, you must set their current attribute values for initial state operation.

Burning log created by Matt Baer

Note	See ”Control execution time of particle dynamics” on page 76 for details on altering playback of particle dynamics.

- ”Animate dynamics” on page 392
- ”Set the initial state of dynamic objects” on page 392
- ”Work with dynamic animation run-up” on page 393
- ”Lessen playback time with dynamics” on page 394

- "Disable dynamics for particles or rigid bodies" on page 394

How do I? Create dynamic effects

Animate dynamics

Set the initial state of dynamic objects

When you play an animation, Maya calculates dynamics each frame for all dynamic objects in the scene. The *state* of a dynamic object is the collection of its current attribute settings—the settings that specify its location and display characteristics. The state of a dynamic object in any frame other than the first is based on its state in the prior frame. Maya calculates dynamic effects sequentially starting at the first frame.

A dynamic object's *initial state* is the state of its attributes at the first frame of the animation. You can play an animation up to some frame, then use its current state as its initial state.

For example, suppose you have an emitter and you want the scene to start when the emitter has already emitted a certain amount of particles. You can play and stop the scene at the frame where you see the particles emitted, then set the current attribute values for use at the initial state. When you rewind and play, the scene starts with the particles already emitted.

To use a dynamic object's current attribute values at the initial state

1. Play and stop the animation at the desired frame.
2. Select the dynamic object.
3. Select Solvers > Initial State > Set for Selected.

To use all dynamic objects' current attribute values at the initial state

1. Play and stop the animation at the desired frame.
2. Select Solvers > Initial State > Set for All Dynamic.

Note	If you key the motion of Streak or MultiStreak particles, you can't use their current state as the initial state.

Work with dynamic animation run-up

If you click a frame in the Time Slider, the correct state of all dynamic objects in the scene is displayed only if Maya performs *run-up* to calculate each frame prior to that frame. Maya doesn't perform run-up by default, so you need to turn it on if you want to click frames in the Time Slider. Note that run-up also occurs for hidden objects.

Leave run-up turned off if you want to prevent Maya from calculating dynamics when you click a frame in the Time Slider. This is useful in a scene that has both nondynamic objects and complex dynamic objects, where you want the state of nondynamic objects to appear promptly after you click the Time Slider. If you are keying dynamic objects, it's also useful leave run-up turned off to avoid waiting for calculations that are irrelevant to your keying activities.

To turn on run-up for all dynamic objects

1. Select Window > Settings/Preferences > Preferences to display the Preferences window.
2. Click Dynamics in the Categories list.
3. Turn on Run Up To Current Time.
4. You can select either of the following Run Up From options:

Previous Time

If you click a frame higher than the current frame, run-up starts from the prior current time and ends at the frame you click. Turn this option on if you won't be changing any attributes of a dynamic object in the scene. This setting lessens the time you'll spend waiting for run-up. If you click a frame lower than the current frame, run-up starts from the beginning of the animation.

Start Time

Run-up starts from the start frame regardless of where you click in the Time Slider. Turn this option on if you plan to change any attributes of a dynamic object in the scene. This ensures that you see the correct object states when you click in the Time Slider after modifying an object's dynamics.

Run-up always occurs in batch rendering. For instance, if you batch render frames 20 through 50, all particle effects will render correctly even if Run Up To Current Time is turned off.

Tip	Press the Esc key if you want to stop a lengthy particle run-up.

Lessen playback time with dynamics

A scene with dynamic animation might play slowly because of intensive computations. The following sections describe ways to speed playback. Note that dynamic animation doesn't include motion from keys, motion paths, non-particle shape expressions, deformers, and non-rigid body constraints.

Tips	If your dynamic animation looks choppy during playback, make sure the Playback speed is set to Play every frame in the Timeline page of the Window > Settings/Preferences > Preferences window. This ensures that Maya won't skip frames, which is often a cause of the problem.

Disable dynamics for particles or rigid bodies

You can turn off dynamic animation for selected particle objects and rigid bodies. This is useful when you work on other objects of a scene and temporarily want to avoid unnecessary calculations that slow animation. The motion of keys and other nondynamic objects is unaffected when you disable dynamics.

To disable dynamic calculations for selected particles

1 Select the particle object.

For emitted particles, select the emitted particles rather than the emitter. For a soft body, select the geometry or the particles.

2 In the General Control Attributes section of the Attribute Editor, turn off Is Dynamic.

Tip	If you have not keyed, parented, or otherwise controlled the transform attributes of a particle object, turn off Emission In World to quicken dynamic calculations for the object. When Emission In World is on, Maya makes extra computations to convert world space to object space coordinates.

To disable dynamics for all rigid bodies controlled by a solver

1 If you've created multiple solvers, first select the solver from the list in Solvers > Current Rigid Solver > rigidSolver#.

2 Select Solvers > Rigid Body Solver to display the Attribute Editor.

3 In the Rigid Solver States section, turn off State.

This turns off the effect of fields, collisions, and rigid body constraints for the rigid bodies it controls. Remember to turn State on again when you want the rigid bodies to operate again—including rigid bodies you've created after you turned off State.

Reference Menus

Dynamics menu set

Solvers

Solvers > Interactive Playback

The Interactive Playback mode lets you interact with dynamic objects and see the simulation update as you playback. For example, you can shake an object with jiggle applied to it to view the effect immediately.

This playback mode can now be started by selecting Solvers > Interactive Playback in the Dynamics menu set, or by typing `play -record` in the Script Editor.

Note	When using Interactive Playback, playback looping is disabled.

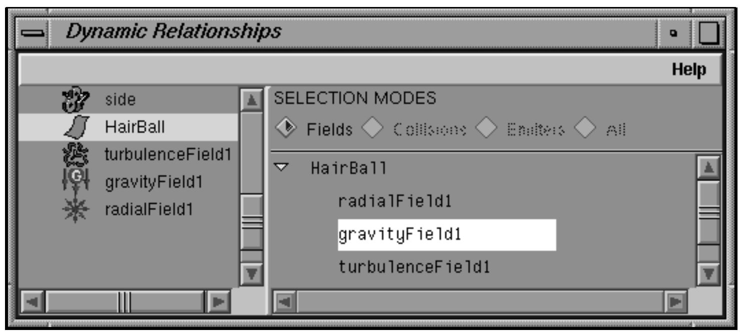
8 Dynamic Relationships Editor

About Dynamics

Dynamic relationship editor

Dynamic relationship editor

The Dynamic Relationships Editor lets you connect and disconnect dynamic relationships between objects and fields, emitters, and collisions.



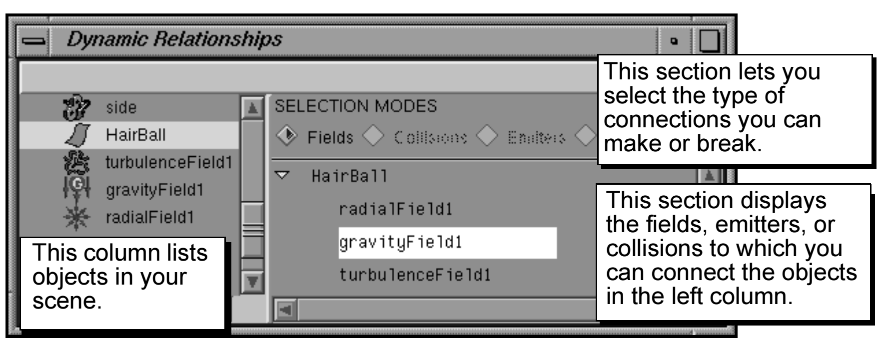
You can display the Dynamic Relationships Editor as a separate window or as a panel in the workspace.

To display the Dynamic Relationships Editor

- From the menu bar, select Window > Relationship Editors > Dynamic Relationships.

or

- From the Panels menu of the current panel, select Panel > Dynamic Relationships.



How do I? Simulate dynamic effects

Work with the dynamic relationship editor

Connect or disconnect items

When you select an object, the right side of the Dynamic Relationships Editor shows the fields, emitters, or collision-capable geometry available for connection or disconnection.

To connect or disconnect items:

1. In the left side of the Dynamic Relationships Editor, click to highlight the object you want to connect (or disconnect) to a field, emitter, or collision-capable geometry.

   If an item is highlighted in the right side, it's already connected to the object highlighted in the left side.
2. Set the Selection Mode for the type of object in the right side you want to connect to: fields, collision-capable geometry (Collisions), emitters, or all three types.
3. Click to highlight the items in the right column to be connected to the item in the left side. To disconnect an item in the right column, click the item to turn off highlighting.

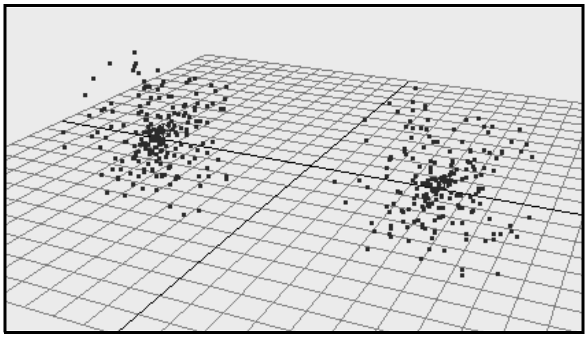
Tip	A selection mode option is dim if it's irrelevant to the object selected in the left column. For instance, if you select a NURBS or polygonal object, mode options other than Fields are dim because you can't connect NURBS and polygons to emitters or collision-capable geometry.

Connect and disconnect gravity - example

The following example shows how use the Dynamic Relationships Editor. You'll create a pair of emitters and connect the emitted particles of one of the emitters to gravity. You'll then switch connections so that the other object's emitted particles fall with gravity instead.

To create the pair of emitters

1. Create two Omni point emitters in different parts of the workspace.

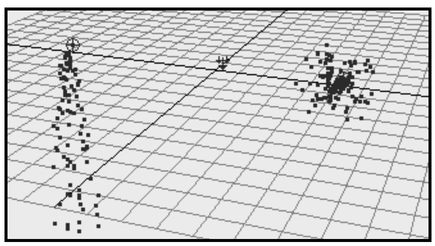


2 Select the emitted particles of the first emitter you created, then select Fields > Create Gravity.

This creates a gravity field and connects the selected particles to it.

3 Play the animation.

The particles emitted by the first emitter fall after being emitted.



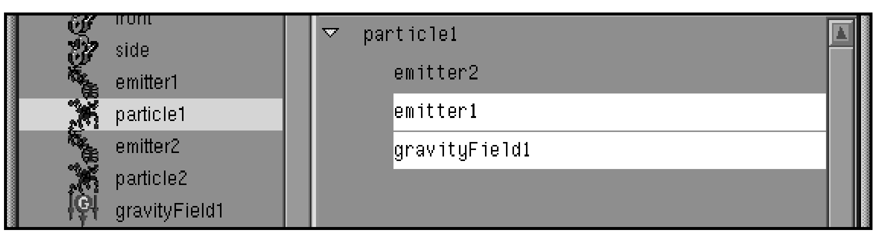
Change the connections to gravity:

You can use the Dynamic Relationships Editor to switch the connection of gravity from one group of emitted particles to the other.

1 In the Dynamic Relationship Editor, turn on All.

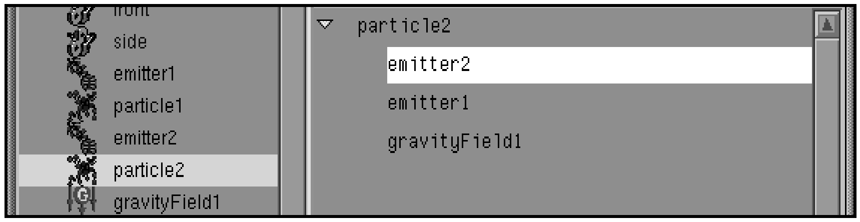
2 Select particle1 in the left column.

There are three items you can connect particle1 to: emitter2, emitter1, and gravityField1. Currently, it's connected to gravityField1 and emitter1.



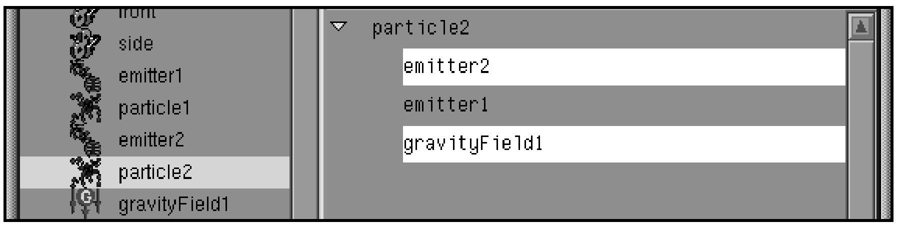
3 Select particle2 in the left column.

It's connected only to emitter2.



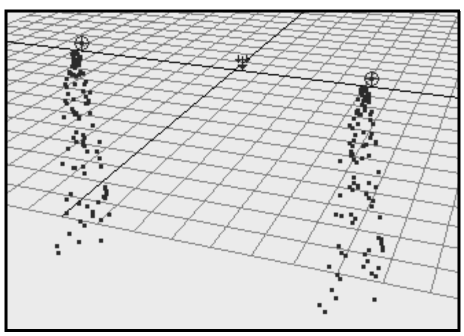
4 Select gravityField1 in the right column.

This connects particle2 to gravityField1.

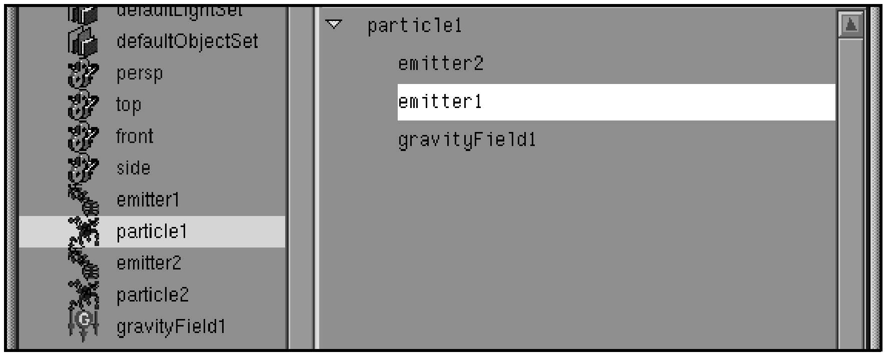


5 Rewind and play the animation.

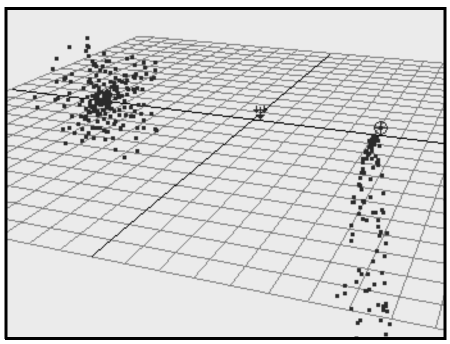
Both groups of particles fall under gravity.



To disconnect gravity from particle1, select particle1 in the left side of the editor, then deselect gravityField1 in the right side:

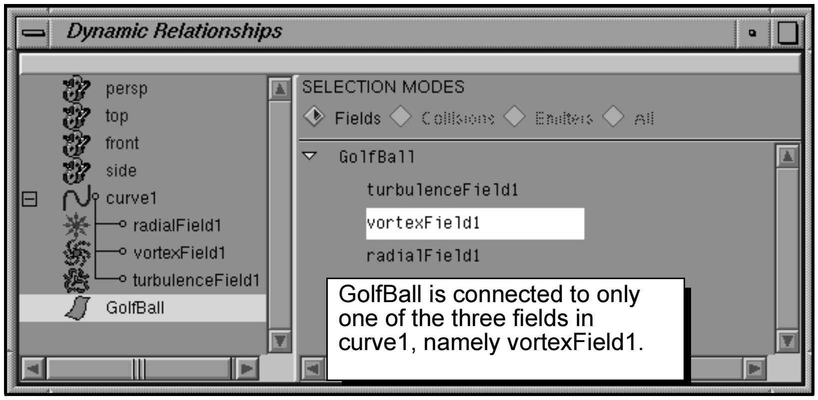


6 Rewind and play the animation to see the switched connection to gravity.



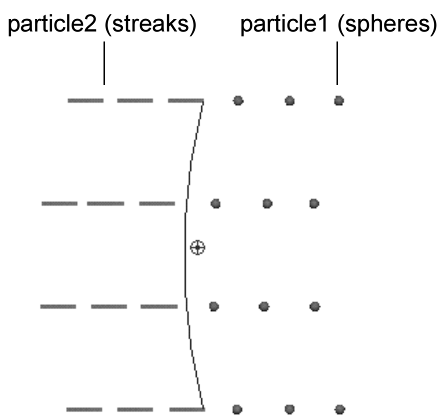
Connect to selected fields or emitters of an object

You can connect a particle object to selected fields or emitters on an object. Previously you had a choice of all or none of the object's fields or emitters.

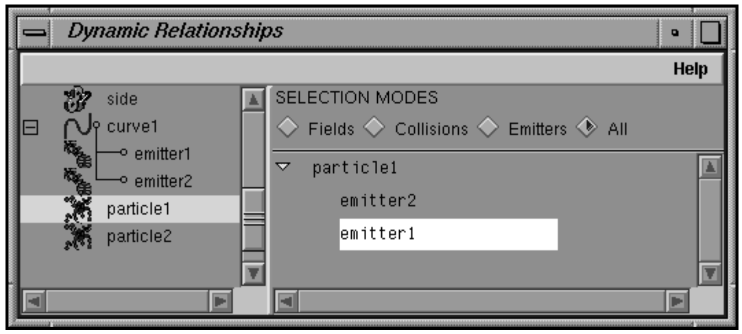


Example

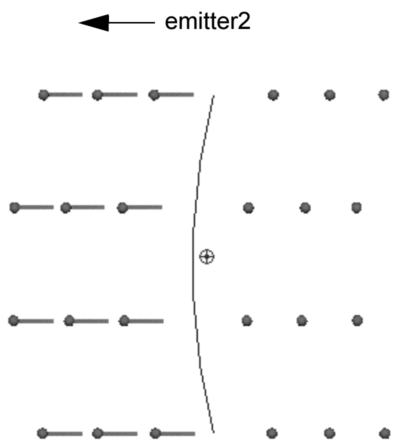
Suppose a curve has two directional point emitters, emitter1 and emitter2. Emitter1 emits particle spheres to the right, while emitter2 emits particle streaks to the left. The spherical particles are particle1. The streaking particles are particle2.



If you select particle1, the Dynamic Relationships Editor shows that it is connected to emitter1:



You can select either, neither, or both emitters in the right side of the Dynamic Relationships Editor. For example, if you connect particle1 to both emitter2 and emitter 1, you'll see spheres emitted from both emitters. Emitter2 already emits streaks, so the spheres will be emitted in addition to the streaks.



Reference Windows and Editors

Relationship Editors

Window > Relationship Editors > Dynamic Relationships

The Dynamic Relationships Editor lets you connect and disconnect dynamic relationships between objects and fields, emitters, and collisions.

❖ ”Connect or disconnect items” on page 398
❖ ”Connect and disconnect gravity - example” on page 398
❖ ”Connect to selected fields or emitters of an object” on page 401

A PDC File Format

How do I? Simulate dynamic effects

Simulate particles

Use the PDC File Format

The PDC file is used by Maya's particle disk caching and startup cache. It is a binary file that holds one frame's worth of data for a single particle object. The *readpdb* program cannot read this format, but we provide this information here in case you want to write your own readers.

4 Characters indicating that this is a PDC file. This will be the 4 characters "P", "D", "C", " ".	
1 Integer indicating the file format version number.	
1 Integer holding bit information about whether the values stored in the file are BIG_ENDIAN or LITTLE_ENDIAN.	
2 Integers holding extra bit information that various file format version might decide to use.	
1 Integer indicating the number of particles represented in this file.	
1 Integer indicating the number of attributes that have values stored in this file.	
N records, where N is the number of attributes. Each record holds all of the particles' data. The record is formatted as follows:	
	1 Integer indicating the length of the attribute's name
	M Characters indicating the name of the attribute, where M is the length of the name.

	1 Integer indicating the type of data for the current attribute. The following maps the values to the data types: 0 ---> Integer 1 ---> Integer Array 2 ---> Double 3 ---> Double Array 4 ---> Vector 5 ---> Vector Array
	P * B Bytes representing the data for this attribute, where P is the number of particles or 1 for non-array data and B is the number of bytes needed to represent the data type.

Index

A

D

E

F

G

H

I

R

S

T