Autodesk® Maya® 2009

10th Anniversary Edition

Autodesk

Getting Started with Maya Unlimited

Copyright Notice

Contents

Overview

1

Introduction

Maya® UnlimitedTM represents the total feature set of Maya software tools available from Autodesk®. Maya Unlimited includes everything from Maya® CompleteTM plus additional tools.

Maya Unlimited includes Maya® Fluid EffectsTM tools that allow for the simulation and rendering of viscous fluids, atmospheric, pyrotechnic, and ocean effects. Maya® nClothTM lets you create simulations of fabric and clothing. Maya® FurTM lets you style and render short hair and fur effects using the Maya® ArtisanTM brush tools. Maya® LiveTM provides functionality to match original live-action footage with your 3D elements that were rendered in Maya.

The goal of this manual is to assist you in learning the features of Maya Unlimited.

If you have never used Maya® UnlimitedTM or Maya® CompleteTM, we recommend that you begin with *Getting Started with Maya* before attempting the lessons within *Getting Started with Maya Unlimited*.

This chapter provides the following information:

- About Getting Started with Maya Unlimited–Information about the Getting Started with Maya Unlimited lessons, where to begin, and the order in which you should complete the lessons.
- Before you begin–Prerequisite knowledge and skills you should possess before beginning the Getting Started with Maya Unlimited lessons.
- Installing Maya Unlimited–Information on installing Maya Unlimited.

- Conventions used in the lessons–Describes the various conventions used throughout the Getting Started with Maya lessons.
- Using the lesson files–How to access and use the lesson files for the Getting Started with Maya Unlimited lessons.
- Restoring default user settings–Describes how to reset Maya to its default settings before you begin the lessons.

About Getting Started with Maya Unlimited

Getting Started with Maya Unlimited introduces the modular areas of Maya Unlimited in a set of brief lessons. The lessons are designed to let you learn these modules at your own pace.

Getting Started with Maya Unlimited is not meant to replace the documentation that comes with the Maya software. Only the commands and options used in the lessons are explained in this manual. You will find the Maya Help provides an excellent companion reference to the lessons and much more.

Many of the lessons in *Getting Started with Maya Unlimited* contain one or more separate lessons that provide step-by-step instructions for creating or accomplishing specific tasks within Maya. You can follow the lessons in this guide from start to finish or complete only the lessons that correspond to your interests and needs.

Getting Started with Maya Unlimited is also available for use within the Maya Help. This version of *Getting Started with Maya Unlimited* contains mpeg movies for some of the lessons.

To use the lessons from the Maya Help

1. In Maya, select Help > Tutorials.

   The Maya Help window displays the *Getting Started with Maya Unlimited* lessons.

2. Click the tutorial you want to work through.

   The Maya Help displays the associated lessons for that tutorial.

Before you begin

We recommend that any new Maya Unlimited user begin by completing the following:

- Viewing the Essential Skills Movies that are available when you first start Maya.
- Completing the Maya Basics lessons (Chapter 2) in *Getting Started with Maya* which introduce many fundamental concepts and skills related to the Maya user interface.
- Completing the lessons within *Getting Started with Maya* before attempting the lessons within *Getting Started with Maya Unlimited*.

Beyond that, you can attempt the lessons in any order you want.

If you require an overview or review of basic mouse, keyboarding, and related usage techniques, we recommend that you refer to the documentation that came with your particular computer and operating system.

If you are new to 3D computer graphics and animation, you may want to obtain *The Art of Maya* (ISBN: 978-1-8971-7747-1). It explains many concepts and techniques that are unique to the world of 3D computer graphics as they relate to Maya.

Installing Maya Unlimited

You must have Maya Unlimited installed and licensed on your computer system to successfully complete the lessons in this manual. To operate Maya Unlimited on your computer you must be running a qualified Microsoft® Windows®, Linux®, or Apple® Mac OS® X operating system with the recommended minimum memory and storage requirements. Maya requires a three button mouse to access its full functionality for menus, commands, and 3D viewing.

For complete instructions on qualified hardware and operating systems, as well as installation and licensing of the Maya software, please refer to the *Installation and Licensing* manual that came with your Maya software or see the Support section of our Web site at *http://www.autodesk.com*.

Conventions used in the lessons

Some important conventions used throughout Getting Started with Maya Unlimited are explained here.

Maya Unlimited is available for use on a wide range of operating systems. Any differences between operating systems when operating Maya are identified in the following ways:

(Windows), (Mac OS X), and (Linux)

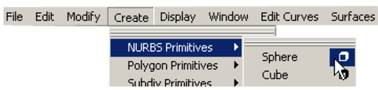
The screen illustrations and examples within Getting Started with Maya Unlimited vary among the Windows, Mac OS X, and Linux operating systems. Maya's interface is generally consistent across these systems.

When instructed to select a menu within Maya, we use the following convention:

- Menu > Command (for example, File > New Scene)

When you are instructed to select the option box for a particular menu item within Maya, we use the following convention:

- Menu > Command > Option (for example, Create > NURBS Primitives> Sphere > ❐



Using the lesson files

Many of the Getting Started with Maya Unlimited lessons have accompanying lesson files. These files are included with your Maya software installation and can be found in the following location:

(Windows XP and Vista, 32 and 64-bit)

```
drive:\Program Files\Autodesk\Maya2009\GettingStarted
```

(Mac OS X)

`Applications\Autodesk\maya2009\GettingStarted`

(Linux 64-bit)

`/usr/autodesk/maya2009-x64/GettingStarted`

Before you use the lesson files, you need to copy the `GettingStarted` directory to your Maya `projects` folder. Then, you need to set the `GettingStarted` folder as your project directory in Maya. For more information and steps, see Copying and setting the Maya project in the Getting Started with Maya guide.

NOTE The lesson files for Live tutorials are included on the Maya DVD and can be loaded as described below.

To open the Live lesson files from the Maya DVD

1. Insert the Maya DVD into your drive.
2. Within Maya, select File > Open Scene.
3. Navigate to the DVD drive, locate the `ExploreMe/MayaLiveLessonData` folder, and select it.
4. Double-click the specific folder and then the required lesson file to open it in Maya Unlimited.

To copy the Live lesson files from the Maya DVD

1. Insert the Maya DVD into your drive.
2. From the Maya DVD, copy the `ExploreMe/MayaLiveLessonData` folder to your computer into the following directory:
   (Windows)
   `drive:\Program Files\Autodesk\Maya2009\GettingStarted`
   (Mac OS X)
   `Applications\Autodesk\maya2009\GettingStarted`
   (Linux)
   `/usr/autodesk/maya2009/GettingStarted`
3. Double-click the specific folder and then the required lesson file to open it in Maya Unlimited.

Restoring default user settings

If you have already used Maya or have a prior version of Maya installed, you should restore the default settings for Maya before you begin the lessons. This ensures that Maya appears and operates exactly as the lessons describe.

If you are an existing user of Maya, we recommend that you save your existing preferences for later use prior to restoring the default user settings.

To save your existing custom user preferences

1. Ensure Maya is not running.

   Each time you exit Maya, it saves the configuration of most components of your user interface so it appears the same when you start it the next time. It writes the preferences to a directory called *prefs*. If you rename the prefs directory, your original preferences will be maintained and Maya will create a new prefs directory the next time it is run.

2. Rename your existing user preferences file to a name, for example, `myprefs`. The prefs directory path is:

   Windows

   - (Windows XP)
     ```
     \Documents and Settings\<username>\My Documents\maya\
     2009\en_US\prefs
     ```
   - (Windows XP 64bit)
     ```
     \Documents and Settings\<username>\My Documents\maya\ 2009-
     x64\en_US\prefs
     ```
   - (Windows Vista)
     ```
     \Users\<username>\Documents\maya\2009\en_US\prefs
     ```
   - (Windows Vista 64bit)
     ```
     \Users\<username>\Documents\maya\2009-x64\en_US\prefs
     ```
   - Mac OS X

   ```
   /Users/<username>/Library/Prefer
   ences/Autodesk/maya/en_US/2009/prefs
   ```

   - Linux (64-bit)

   ```
   ~<username>/maya/2009-x64/en_US/prefs
   ```

If you have a previous version of Maya installed, also rename that prefs directory to a new name such as *myprefs*. Maya will load older preferences if they exist from a previous version.

3 Start Maya and begin the Getting Started with Maya lessons.

To restore your custom user preferences after doing the lessons

1 Ensure Maya is not running.

2 Rename the previously changed preferences back to `prefs`.

Hair

2

Introduction

With Maya® HairTM, you can create a dynamic hair system to model realistic hairstyles and hair behavior.

Since Hair is a generic dynamic curve simulation, you can also use these curves to create non-hair effects, including ropes, chains, a suspension bridge, sea creatures, or lofted surface from a curve.

With Hair you can simulate:

- natural movement and collisions of long hair
- hair blowing in the wind or being blow dried

- hair when swimming underwater
- various hairstyles, including ponytails, braids, and updos
- other dynamic curve effects such as ropes, chains, cables, wires, etc.

A *hair system* is a collection of hair *follicles*.

A hair follicle in human hair typically hosts one hair, whereas in Maya each hair *follicle* hosts a hair curve. The hair *follicle* controls the attributes and curves associated with a particular hair clump, and how the hairs attach to a NURBS or polygonal surface. The input to each follicle is a Start Position NURBS curve, a surface and a UV position to attach to.

When creating hair, the hair system *output* can be in the form of NURBS curves, Maya® Paint EffectsTM strokes, or both. If the specified output when creating hair is NURBS Curves, each follicle will contain one NURBS curve that represents the position of the hair in that follicle. If the specified output is Paint Effects, each follicle has a hair clump, made up of Paint Effects strokes.

There are various attributes on a hair system for modifying the look and behavior of the hair as a whole. There are also follicle level attributes that will override the hair system attributes; the visible result will be affected by both the follicle and hair system attributes.

You can have more than one hair system on the same surface.

With the Paint Hair Tool you can create new hair systems, add and remove follicles, as well as paint hair attributes, including Clump Width Scale, Stiffness, and Braid.

Paint Effects hair can be rendered using the Maya Software renderer. You can also convert Paint Effects hair to polygons and render in another renderer, such as mental ray® for Maya®. Or you can output just the dynamic NURBS curves to an external renderer, such as Renderman.

Consider outputting to Paint Effects if you want to:

- render with the Maya Software renderer
- convert the Paint Effects hair to polygons and render in any renderer

Consider outputting to NURBS curves if you want to:

- use the curves as path or control curves for any standard Paint Effects brush, such as feathers or vines. These brushes behave differently than the default built in Paint Effects as the curves are not the hairs, but controls that interpolate and affect Paint Effects. This scenario is slower to update and takes a bit of time to set up.

- select a few Current Position curves and loft a surface through them with construction history.
- extrude a circle down the curve to create a tube.
- use the curves for output to another renderer that directly supports curve rendering.

Consider outputting to both NURBS curves and Paint Effects if you want to:

- use both the curves (for example, as particle emitters to simulate spray flying off wet hair) and render using the integrated Paint Effects either directly or by converting to polygons.

About hair simulation

This is the general workflow to create and animate hair:

- create hair on models
- style hair curves
- modify hair attributes
- set up hair shading/shadowing
- render the scene

This chapter includes these lessons:

- Lesson 1 Creating a basic hairstyle: Introduction on page 12
- Lesson 2 Creating a dynamic non-hair simulation: Introduction on page 34

Preparing for the lessons

To ensure the lesson works as described, do these steps before beginning:

1 Make sure you understand the basic concepts of modeling NURBS curves, animation, and dynamics. For more information, see Maya Help.

2 Select the Dynamics menu set.

   Unless otherwise noted, the procedures in these lessons assume the Dynamics menu set is selected.

3 Select File > New Scene to create a new scene before you start each lesson. For those scenes that require you to open a file, the Hair scene files are located in `GettingStarted/scenes` directory. The `GettingStarted` directory is installed in the same location as your Maya software application.

4 Select Window > Settings/Preferences > Preferences. Click Timeline under Categories and make sure the Playback Speed is set to Play every frame. Dynamic animation plays more accurately with this setting.

   Also set the playback and animation end times to 10000. This ensures you have lots of time to interact with the playback of the hair simulation.

5 If you have not already done so, copy the `GettingStarted` folder from its installation location to your `projects` directory. Then, set the `GettingStarted` directory as your Maya project. For more information, see Copying and setting the Maya project in the Getting Started with Maya guide.

6 We also provide a completed scene for each of the lessons in this chapter. If you want to see the expected results of the lessons, you can open the scenes located in the `GettingStarted/scenes` directory. The scene filenames end with `Final.ma`, for example, `Hair_HeadFinal.ma`.

Lesson 1: Creating a basic hairstyle

Introduction

You can create hair on NURBS and polygonal surfaces. For polygons, UVs should be non-overlapping and fit between 0 and 1. Automatic mapping is a quick way to achieve this. (See Automatic UV mapping in the Mapping UVs Guide in the Maya Help.) In this lesson you are creating hair on a head model.

There are three sets of curves for the hair system.

Start Curves/Position	This is the position of the hair at the start frame of the simulation. When first created, the Start curves stick straight out from the surface.
Rest Curves/Position	This is the position of the hair when no forces (such as gravity) are acting upon it. You can use these curves to influence the shape of the hair. Working with Rest curves is like styling hair with hairspray. The Rest curves are styled into curls and flips, etc. When the simulation is played back, the hair adheres to the Rest curves position as much as possible depending on dynamic forces and attribute settings in the hair system and follicles. For example, if your hair is not very stiff and you're simulating a windy day, the hairspray (Rest curves position) is less likely to hold the hair in position.
Current Position	This is how the hair behaves when you play the simulation, which includes dynamics. Do **not** edit the Current Position; just view it. The Stiffness and Iterations attributes in the Hair system's Dynamics section are what control the tolerance of the Current Position achieving the Rest Position.

In this lesson you are introduced to some of the basic concepts of hair by creating a simple long hairstyle. In this lesson, you learn how to:

- Create hair on a surface.
- Play a hair simulation and interact with it during playback.
- Manipulate hair curves to create a hairstyle.
- Use collision constraints to make hair collide with a model's head, neck and shoulders.
- Modify hair attributes.
- Set up hair self shadows.
- Render hair.

Lesson setup

In this lesson, you'll work with a scene we created for your use.

1 Before beginning the lesson, do the steps in Preparing for the lessons on page 11.

2 Open the scene named hairStart.ma that is located within the GettingStarted/Hair directory of your Maya project.

   This scene provides a NURBS head model, named Marion, and the scalp surface for creating the hair. Part of the model has been templated to make picking the scalp easier.

Creating hair on a surface

When you first create hair it appears sticking out normal to the surface. For a more natural look, play the hair simulation to see the hair react to gravity and then you can set a more realistic Start position for the hair. In these steps you create hair on a NURBS model and set a Start position after playing the simulation. You use Interactive Playback to play the simulation while adjusting dynamic hair system attributes and see how your changes affect the simulation.

NOTE Before continuing with the procedures below, make sure the End Time and Playback End Time in the Range Slider are both set to 10000.

To create hair on a model

1 Select Marion's scalp, as shown in the image below.

2 Select Hair > Create Hair > ❐ and in the Create Hair Options window, specify the following:

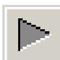
- Set Output to Paint Effects.
- Set the U count to 26 and the V count to 22. These determine how many hair curves within each UV range.
- Set the Passive fill to 0. No passive curves will be used in this lesson. The behavior of passive curves is interpolated between neighboring active curves. For more information, see Passive hair curves in the Hair guide.
- Set the Randomization to 0.5. A value of 1 represents total randomization of placement of hair curves on the surface, whereas a value of 0 represents no randomization.
- Set the Points per hair to 10. This sets the number of segments per hair. More segments are needed to make long hair look natural or to create complex short hairstyles. As you increase the number of segments, you decrease the performance (speed) of the simulation.
- Set the Length to 10. This value is relative to the world space units.
- Both the Create Rest Curves and Edge Bounded options should be turned off.
- Turn the Equalize option on. This ensures the hair curves are evenly distributed over the surface.

- Click Create Hairs.

The hair is attached to the scalp and sticks out perpendicular to the surface. You are looking at the *Current Position* of the hair, which is also referred to as the “dynamic” position. Do not edit the hair curves in this view; edit only in *Start Position* or *Rest Position*.

When Marion’s hair is first created it is perpendicular (normal) to the scalp. The red circles are the follicles.

To play the hair simulation

1 Click the Play  button to play the simulation.

The hair falls due to default dynamic forces, such as gravity, which are found in the hair system attributes. However the hair takes a while to fall, which is as a result of a high Display Quality setting.

2 Click the Stop  button to stop the simulation when the hair is relaxed as it is in the following image.

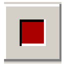
After playing the simulation for a while Marion's hair begins to relax, but it is still quite stiff and doesn't rest near the scalp as hair does naturally.

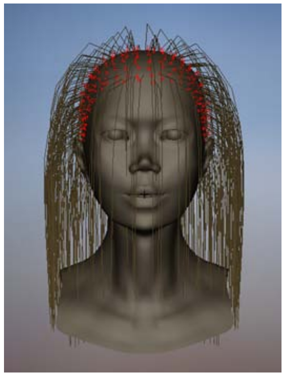
Notice the hairs are quite stiff and stick out from the scalp instead of naturally falling closer to the scalp. To make the hair appear less stiff you can either increase the number of Points Per Hair (segments), which increases the simulation and render time, or you can reduce the Stiffness value in the hair system's Dynamic attributes.

In the next part of the lesson you adjust dynamic settings of the hair system, including Stiffness and Gravity, to achieve more naturally behaving hair.

To improve playback performance and modify dynamic settings

1. In the Attribute Editor (Ctrl + a) select the hairSystemShape1 tab and change Display Quality to 2. This affects how many hairs display in the scene view, but has no effect on how much hair appears in the software render. By reducing the Display Quality, this speeds up the playback of the hair simulation.
2. In the Dynamics section the default value for Stiffness is 0.15. Since the hair is quite stiff, reduce the Stiffness value to 0.07.
3. Select Solvers > Interactive Playback to play the simulation. Watch the hair begin to relax more.
4. With the simulation still playing go to the Dynamics section of the hairSystemShape1 tab and change the Gravity value to 2.5. Watch the hair relax even more and rest closer to the scalp, as hair does naturally.

5 Click the Stop  button when the hair has relaxed as in the following image.



To set the position of the hair for the start frame

1 When you are playing the hair simulation, by default you are viewing the Current position curves. To select all the Current position curves, click any hair curve and then select Hair > Convert Selection > to Current Positions.

2 To set the start frame of the hair simulation to how the hair appears now, select Hair > Set Start Position.

3 Select Hair > Display> Start Position to view the Start curves and see that they took on the Current Position.

4 Click the  button to rewind to the start frame.

The position of the hair at the start frame is now relaxed.

Styling the hair

To style the hair you manipulate either the Start or Rest curves to achieve the look you want. In these next steps you first edit the Start curves to create the basic hair shape. Then you set an initial Rest position for the hair and style the hair by editing the Rest curves.

To edit the Start curves to create the basic hair shape

1 Display the Start curves. (Hair > Display> Start Position).

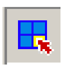
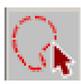
2 Tumble the camera so you're looking at the side of Marion's head.

3 To select the Start curves on the front half of Marion's face, do the following:

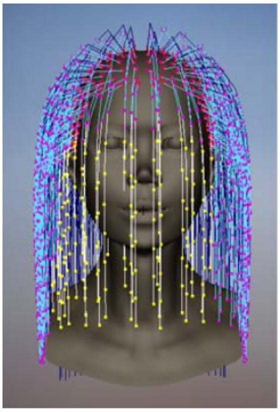
- On the Status Line, click the Object Selection Mask  button and choose All Objects Off from the drop-down menu that appears.
- On the Status Line, click to highlight the  icon to restrict the selection to curves.
- Drag around the front half of Marion's head to select the Start curves as shown in the following image.

4 Now tumble the camera so Marion is again facing you.

5 To trim the Start curves in the front, do the following:

- Select Hair > Modify Curves. This allows you to manipulate the curve while maintaining the curve's hull length.
- Click the  in the Status Line to display the curves' CVs.
- Using the Lasso Tool , select CVs in an inverted U-shape around the front of Marion's face, as shown in the following image.

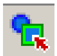
- Press Delete to remove the selected CVs.

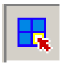


Start curve CVs selected in an inverted U-shape

Selected UVs deleted to trim hair

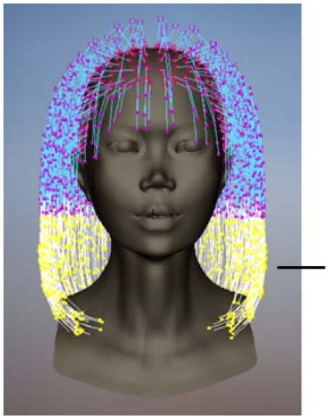
6 To curl under the ends of all the Start curves, do the following:

- Click the  button to display the curves and then drag around Marion’s head to select all the Start curves.

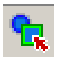
- Click the  button to display CVs and then, using the Select Tool, select the CVs in the bottom half of all the curves.

- Select Hair > Modify Curves >  .

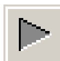
- In the Bend Curves Options window, set Bend Amount to 0.5 and Twist to 1.0 and then click Bend Curves.



The selected CVs are bent under towards Marion's neck. To fix any unruly hairs, you could select them and then select Hair > Modify Curves> Smooth.

7 Click the  button to display the curves.

8 Change the display to include the Start curves and the Current Position. (Hair > Display> Current and Start)

9 Click the Play  button to play the hair simulation.

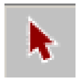
Even though you styled the Start curves, notice how they fall straight and lose the bend at the end. Start curves only define the position of the hair at the Start frame of the simulation; once the simulation is started the Start curves are affected by the dynamic forces applied to the hair system. To "hairspray" a hairstyle, you need to style Rest curves and adjust dynamic hair system settings. In the next steps you create Rest curves from the Start curves.

10 Click the Stop  button to stop the simulation.

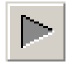
To create and modify the Rest curves to achieve a hairstyle

1 To create Rest curves from the shaped Start curves, do the following:

- Select Hair > Display. > Start Position

- Using the Select Tool , drag to select all the Start curves and then select Hair > Set Rest Position> From Start.

2 To see the new Rest curves, select Hair > Display> Current and Rest.

3 Select all the hair curves and then click the Play  button to play the simulation. Watch how the Current position hairs flop down because the Gravity value is quite high (2.5).

4 To see what the hair would look like without any gravity, go to the Dynamics section in the hairSystemShape1 tab and change the Gravity value to 0.

   Watch the hair try to achieve the Rest position.

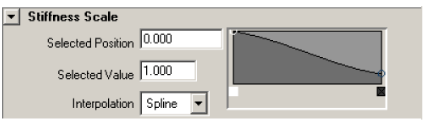
5 After the hair reaches the Rest position, change the Gravity value back to 1.0.

6 As the hair starts to fall due to gravity, increase the Stiffness value to 0.2.

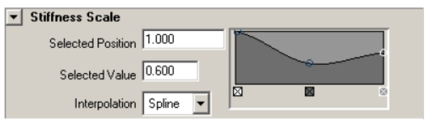
   If some Rest curves won't relax and stabilize, increase Length Flex and Iterations in the hairSystemShape1 tab.

7 Once the hair stabilizes, click the Stop  button to stop the simulation.

8 You can further "hairspray" the curled ends of the Rest curves using the Stiffness Scale attribute. The curve in the Stiffness Scale graph illustrates the amount of stiffness applied along the hair curve from root (left side of the graph) to tip (right side of the graph). Go to the Dynamics section in the hairSystemShape1 tab and change the Stiffness Scale attribute as shown in the following image. These changes add more stiffness to the end of the hair, which will hold the bend more firmly.



Default Stiffness Scale

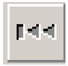


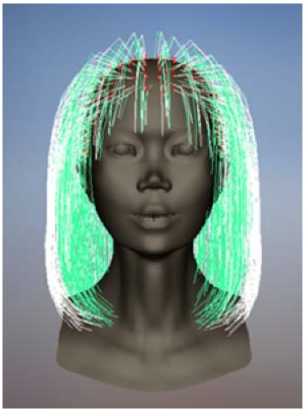
Modified Stiffness Scale:
1. Click the middle of the graph to add a new entry and set its Selected Value to 0.4.
2. Click the entry on the right side of the graph and change its Selected value to 0.6.

9 Play the simulation to see the effect of the changes to Stiffness Scale.

10 Stop the simulation once the hair relaxes as in the following image.

11 To reset the Start curves from the simulation, select Hair > Set Start Position> From Current.

12 Click the  button to rewind to the beginning of the simulation.



All the hair curves are selected.
The green curves represent the Current Position.
The white curves represent the Rest Position.

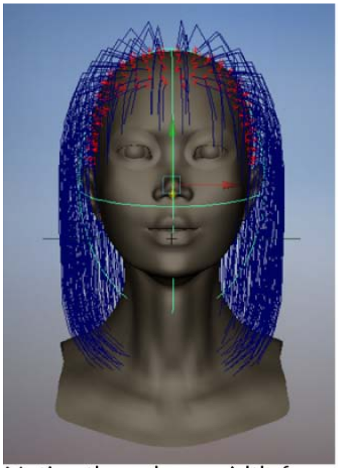
Setting up hair collisions

Before trimming Marion's hair you may have noticed that some hairs intersected her face. As well hairs currently intersect the lower part of her neck. To fix this you set up hair collisions. You can set hair to collide with itself, other objects or the ground (grid). In the next steps you use collision constraints to prevent the hair from intersecting with Marion's head and neck.

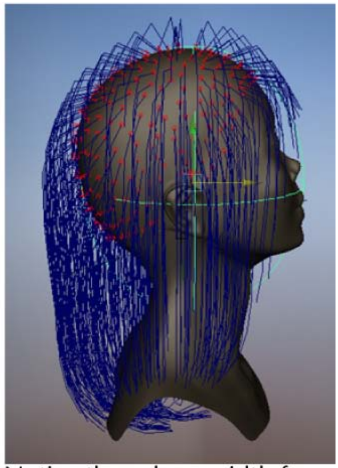
Notice the hairs at the bottom are intersecting with the upper part of Marion's torso

To make Marion's hair collide with her head and neck

1. In the panel, select Show > Locators so you can see collision constraints, which are locators, when you create them.
2. To create the collision constraint for Marion's head, do the following:
   - Select Hair > Display > Start Position.
   - Using the Select Tool, drag around all the hair curves and select Hair > Convert Selection > To Start Curves.
   - Select Hair > Create Constraint > Collide Sphere.
   - Using the Scale Tool, resize the sphere so it expands beyond Marion's head and you can see it.
   - Using the Move Tool, move the sphere up the Y-axis so the top of the sphere is aligned with the top of Marion's head.
   - Using the Scale Tool again, resize the sphere so it matches the basic shape of Marion's head. First try scaling in the X direction to adjust the width of the sphere from ear to ear. Then tumble the camera to see the side of Marion's head and scale in the Z direction to adjust the sphere width from the back of Marion's head to her nose. You may also need to move the sphere to tweak its size and position, as shown in the images below.



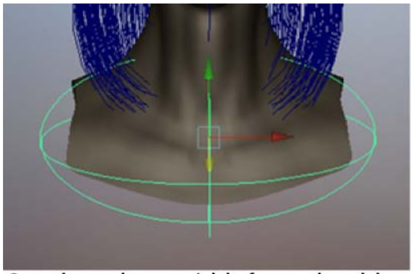
Notice the sphere width from ear to ear and height from the top of Marion’s head to below her chin



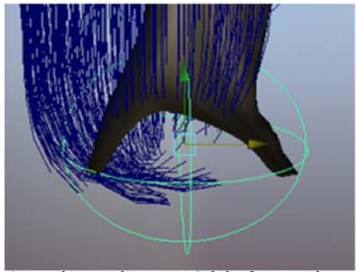
Notice the sphere width from the back of Marion’s head to her nose

3 To create the collision constraint for Marion’s upper torso, do the following:

- Using the Select Tool, drag around the hair curves and select Hair > Convert Selection > To Start Curves.
- Select Hair > Create Constraint > Collide Sphere.
- Using the Scale Tool, resize the sphere so it extends beyond Marion’s neck and you can see it.
- Using the Move Tool, move the sphere down the Y-axis so the top of the sphere is aligned with the top of Marion’s torso.
- Using the Scale Tool again, resize the sphere so it matches the basic dimensions of Marion’s upper torso. First try scaling in the X direction to adjust the width of the sphere from shoulder to shoulder. Then tumble the camera to see the side of Marion’s head and scale in the Z direction to adjust the sphere width from the back of Marion’s torso to the front of it. You may also need to move the sphere to tweak its size and position, as shown in the images below.

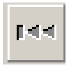


See the sphere width from shoulder to shoulder and height from the top of Marion's torso to the bottom of it



See the sphere width from the back of Marion's torso to the front of it

4 Select Hair > Display> Current Position.

5 Click the  button to rewind to the start frame and play the simulation.

Watch how the hair collides with the collision constraints you created. The hair at the front now rests on the torso and the hair at the back curls under the torso, instead of intersecting it like it did before the collisions were set.

6 Stop the simulation when the hair is relaxed. Should the hair be unruly and not relax, you may need to increase the Stiffness or the Iterations in the Dynamics section of the hairSystemShape1 tab of the Attribute Editor.

Once the hair relaxes in the simulation, see how it rests against the collision constraints

7 Select the hair, then select Hair > Set Start Position > From Current so the hair respects the collisions at the start frame of the simulation.

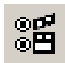
8 Select Show > Locators to turn them off, which hides the collision constraints.

Rendering the hair

When you modify hair attributes, many of these changes are not visible in the scene view; the Paint Effects hair must be rendered in the Maya Software renderer. In the next steps you render Marion's hair as it is now, which allows you to see what aspects of the hair need work to make it look more realistic. Before rendering, be sure to set the appropriate Render Settings.

To render Paint Effects hair using the Maya Software renderer

1 Modify the Render Settings for Paint Effects.

- Click the Render Settings  button on the Status Line. The Render Settings window appears.
- Select Maya Software from the Render using drop-down list.
- In the Maya Software tab, go to the Paint Effects Rendering Options section and turn on Oversample and Oversample post filter. These help achieve a better look for the Paint Effects rendering. The option Enable stroke rendering is already on by default.
- Click Close to close the Render Settings window.

2 Before rendering be sure to display the hair system's Current Position (Hair > Display > Current Position).

3 Click the Render Current Frame  button on the Status Line.

The hair model is rendered and should look something like the image below.

This is how the hair looks in the Maya Software render before modifying hair attributes. The hair is sparse at the crown, which can be fixed by increasing the Clump Width value.

Modifying hair attributes

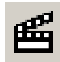
You can modify the hair as a whole by modifying the hair system attributes. You can modify the look of individual hairs or clumps of hair by editing their follicle attributes. In the next steps, you modify various hair system attributes to make Marion's hair look fuller and more realistic. Then you add curl to Marion's hair.

To modify the thickness of the hair all over

1. To select the hair system, drag around the hair curves and select Hair > Convert Selection > to Hair Systems.
2. Go to the Clump and Hair Shape section in the hairSystemShape1 tab of the Attribute Editor and adjust the following:
   - Hairs Per Clump to 30, which controls the number of hairs per curve (hair clump). Increasing the number of Hairs Per Clump increases the rendering time.
   - Sub Segments to 2, which adds this many points between segments/points and helps to smooth the hair curves.
   - Thinning to 0.5, which controls how much the hair thins out from root to tip.
   - Clump Width to 0.3, which controls the width of the hair clump from root to tip. To further modify the hair clump width, you could adjust

the Clump Width Scale attribute, which allows you to vary the width along the hair from root to tip.

- Hair Width to 0.02, which controls the width of each hair.

3 Click the Render Current Frame  button to render the hair to see the changes you've made to the hair.

After modifying hair and clump attributes the hair looks more full and realistic.

To make hair curly

1 With the hair system still selected, go to the Displacements section in the hairSystemShape1 tab of the Attribute Editor and set the following:

- Curl to 1, which is the amount of curl displacement applied to each hair. The amount of displacement is relative to the hair width.

- Curl Frequency to 15, which is the rate of curl. The higher the value is, the more curls there are.

2 Click the Render Current Frame  button to render the hair to see the changes you've made to the hair.

The rendered hair after curl attributes have been applied.

Setting up shadowing on hair

Hair, by default, is translucent so you need to set up self shadowing, otherwise hair appears glowing, like nylon. The darker the hair color is, the more important the specular color and highlights become.

There are three directional lights in this scene. In the software render you did in the previous steps, Marion's hair appears blond and glowing, even though the default hair color is dark brown, as in the scene view. This apparent hair color discrepancy occurs because:

- hair is by default translucent (like a glass tube)
- the hair is being lit by all three lights, therefore the intensity of the light is tripled
- there are no shadows, so all the hairs are illuminated

In these next steps you set up shadows on all three lights.

To set up shadows and self shadows on hair

1. Select Window > Outliner.
2. In the Outliner menu, select Show > Objects > Lights.
3. Select one of the three directional lights.

4 In the light's Attribute Editor, go to the Shadows section and in the Depth Map Shadow Attributes subsection, turn on Use Depth Map Shadows.

Normally you would increase the Filter Size to blur the light in the hair for realism, and increase the Bias, which sets how far the light filters through the hair. These attributes were set for you in the scene.

5 Repeat steps 3 and 4 for the other two directional lights.

6 Render the hair to see the changes.

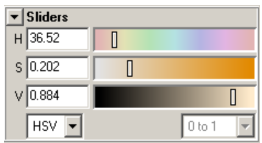
The shadows on the hair are hard and there is very little specularity. You fix this in the next steps.

To create specular highlights on hair

1 To select the hair system, drag around the hair curves and select Hair > Convert Selection > to Hair Systems.

2 Go to the Shading section in the hairSystemShape1 tab of the Attribute Editor and set the following:

- Click the Specular Color box and set the values as shown in the image below.



- Increase Specular Power to 15.

3 Render the hair to see the changes. The hair looks much more natural now.

Your final render with the changes to the specular color and power should look something like this. Marion's hair is brown, but with a warm, golden specular highlight.

Beyond the lesson

In this lesson you learned how to:

- Create dynamic hair on a surface.
  You can add hair to a NURBS or polygonal surface by creating or painting the hair. For polygons, UVs should be non-overlapping and fit between 0 and 1. Automatic mapping is a quick way to achieve this (seeAutomatic UV mapping in the Mapping UVs guide). For more information, see Creating hair in the Hair guide.

  Before creating hair, you should determine which renderer you'll be using as this will affect what type of hair output you select: NURBS curves or Paint Effects, or both.

- Interact with the dynamic playback.
  While using interactive playback you can modify the hair and see the hair react to dynamic forces. For more information, see Play a hair simulation in the Hair guide.

- Model hair curves.
  Style hair by transforming hair curves and using Lock Length to maintain the length between CVs on a curve. You can also use Fields to style hair and then add constraints to hold hair in position. For more information, see Styling hair and modifying the hair look and behavior in the Hair guide.

- Modify hair system and follicle attributes.

You can change the look (color, curl, etc.) of the overall hair by modifying attributes in the hairSystemShape node. But individual follicle attributes can be modified in the follicleShape node, which override or blend with the hair system attributes. For more information, see the following topics in the online help: Hair system attributes (hairSystemShape in the Hair guide), Follicle attributes (follicleShape in the Hair guide) and Set up hair shading in the Hair guide.

- Render the hair.
  Paint Effects hair can be rendered using the Maya Software renderer. You can also convert Paint Effects hair to polygons and render in another renderer, such as mental ray. Or you can output just the dynamic curves to an external renderer, such as Renderman. For more information, see Rendering scenes with hair in the Hair guide.

You can also create non-hair models and effects with Hair. Any NURBS curve can be made into a dynamic hair curve. In addition to creating long hair and hairstyles you can use hair curves to create:

- ropes and chains (dynamic curves)
- a fish (wire deformer)
- a fishing line (single hair with expressions)
- an octopus (select hair curves and create a lofted surface)
- a paint brush painting on canvas, bristles colliding with the surface (hair simulates paint brush bristles, Fluid emitters on the canvas controlled by expressions)

In addition you can assign a Paint Effects brush to a hair system as shown in the image below.

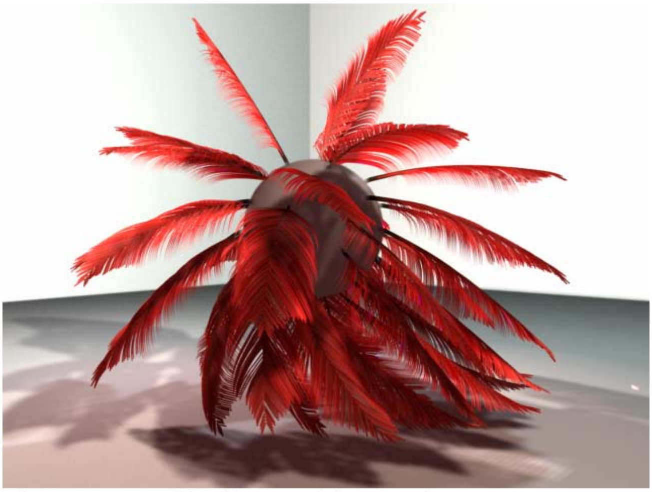


Image created by Duncan Brinsmead

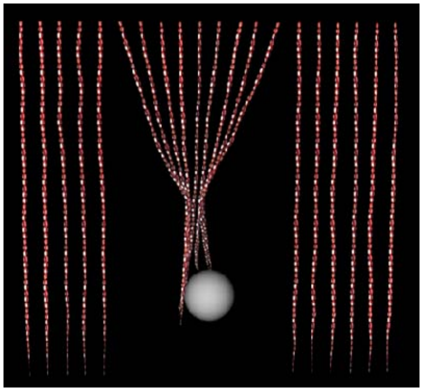
Lesson 2: Creating a dynamic non-hair simulation

Introduction

Using Maya Hair, you can create dynamic non-hair models and effects. You can make NURBS curves dynamic, or you can use a hair system to create a non-hair simulation, such as chains or ropes. In this lesson, you create a hair system to simulate a beaded curtain. To make the hairs look like beads, you assign a Paint Effects brush to the hair. Then you modify the hair attributes and set constraints so the hair behaves like a beaded curtain hanging from a curtain rod.

In this lesson, you learn how to:

- Make hair collide with an object
- Assign a Paint Effects brush to hair
- Set up hair constraints



Lesson setup

In this lesson, you work with a new empty scene and change the playback range.

1 Select File > New Scene.

2 In the Time Slider change both the End Time value and Playback End Time value to 300.

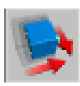
Setting up the curtain scene

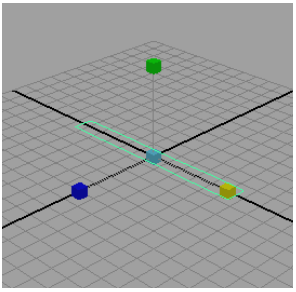
In these steps you model a 2D curtain rod from a plane and attach hair to it. Then you add and keyframe a sphere in the scene to later be used as a collision object.

To create the curtain using Hair

1 To create a NURBS plane, select Create > NURBS Primitives > Plane, then click-drag in the scene to create a plane.

 (If you have previously shut off the interactive primitive creation option, simply click once in the scene to create the primitive at the origin.)

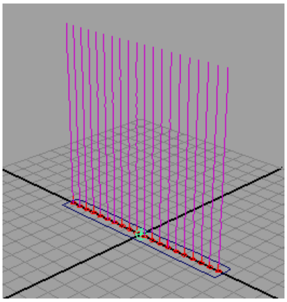
2 Using the Scale Tool , resize the plane in the X direction as shown in the image below.

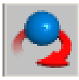
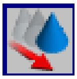


3 With the plane selected, select Hair > Create Hair >  , set the following options and then click Create Hairs:

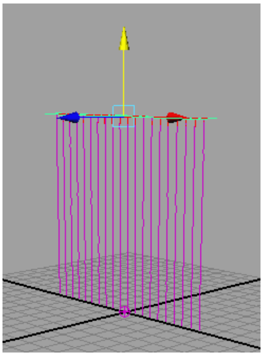
- Output to NURBS Curves
- U Count to 20, V count to 1
- Passive fill to 0
- Randomization to 0
- Points per hair to 12
- Length to 10
- Turn off the options Create Rest Curves, Edge Bounded and Equalize

Hair is created and attached to the plane.



4 Select the plane and, using the Rotate Tool  , rotate the plane so the hair appears to be on the underside of the plane. Then, using the Move Tool  , move the plane up the Y axis so the bottom of the

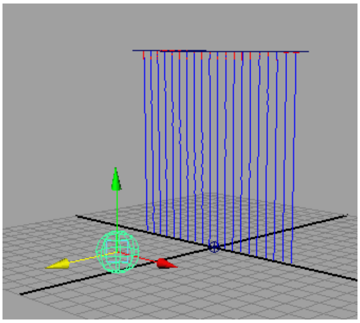
hair just touches the ground plane, as shown in the following image. (Ensure you do not select the hair when selecting the plane.)



5 Create a directional light (Create > Lights > Directional Light) and do the following:

- Move the light above and to the left of the curtain.
- Rotate the light so it aims at the curtain.
- In the light's Attribute Editor (ctrl + a) go to the Directional Light Attributes section and change the Intensity value to 1.5.

6 Create a NURBS sphere (Create > NURBS Primitives > Sphere) and move it along the Z-axis so it's in front of the curtain.

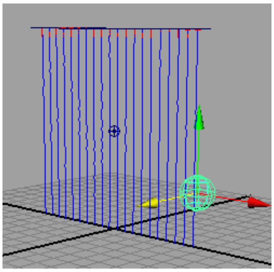


7 To set the start keyframe for the sphere, do the following:

- Go to frame 1 on the Time Slider.
- In the Channel Box, drag to select the Translate X, Y, and Z channels.
- Right-click the channels and select Key Selected.

8 To set the keyframe for the sphere and move it through the curtain, as in the image below, do the following:

- Go to frame 100 in the Time Slider.
- Move the sphere in the Z direction as shown in the image below.
- In the Channel Box, select the Translate Z channel.
- Right-click the channel and select Key Selected.



Time Slider = 100

9 Rewind the Time Slider to 0 and play the animation.

The sphere passes through the curtain as though the curtain weren't there. In the next steps you set the sphere as a collision object with the hair.

Making the hair collide with another object

To have hair interact with a surface, you must set the hair to collide with the surface. A collision object can be a NURBS or polygonal surface. In the next part of the lesson you set the curtain to collide with the sphere and then play the simulation.

To enhance the curtain for collisions

1 To select the hair system, drag around the hair curves and then select Hair > Convert Selection > To Hair Systems.

2 In the hairSystemShape1 tab of the Attribute Editor, set the hair system attributes as follows:

- In the Clump and Hair Shape section, set Clump Width to 0.
- In the Dynamics section, set Stiffness to 0.

To make the hair collide with the sphere

1. Select the hair system and the sphere (drag around the hair curves and sphere).
2. Select Hair > Make Collide.
3. Play the simulation.

   Now the sphere collides with the curtain, pushing the strands of hair aside as it passes through.

Assigning a Paint Effects brush to the hair

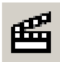
You can assign a Paint Effects brush to a hair system to achieve a variety of effects. In these next steps you apply the red beads brush to the hair system to create the beaded curtain. Then you modify the brush and hair system settings to achieve the desired look.

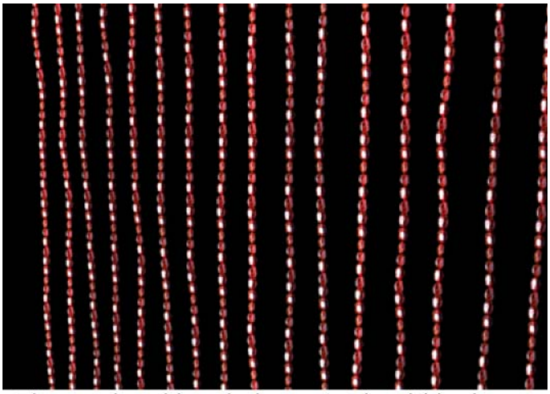
To turn the hair curtain into a beaded curtain

1. To select the hair system, drag around the hair curves and select Hair > Convert Selection > To Hair Systems.
2. Open the Visor (Window > General Editors > Visor).
3. Go to the glass folder and select the beadsRed.mel brush.
4. Select Hair > Assign Paint Effects Brush to Hair.

   You don't see the beads in the scene view, but you will when you render it later using the Maya Software renderer. The brush needs some adjustments to make it look like a beaded curtain when it is rendered.
5. In the hairSystemShape1 tab of the Attribute Editor, set the following:
   - Hairs Per Clump to 1 so there is one strand of beads
   - Hair Width to 0.03
   - in the Hair Color Scale section, Hair Color to white so the brush's bead color appears
   - in the Shading section, Specular Color to red
   - in the Displacements section, Noise to 0.5 to create some variety in the appearance and behavior of the beads

6 In the beadsRed tab of the Attribute Editor, go to the Texturing section and change the RepeatU to 10 so the beads appear as a more realistic bead size. (If the beadsRed tab does not appear in the Attribute Editor, re-select the hair curves. The beadsRed tab should now appear.)

7 Dolly in close to the curtain (Alt + the right mouse button) and click the  button to render the current frame.



The rendered beaded curtain should look something like this

Setting up constraints

In these next steps you tie up the middle section of the beaded curtain by first modeling the Start curves into a bound position and then creating a constraint to hold the curves in that position during playback. With a Hair to Hair constraint the hair curves stick together at the constraint locator, but also dynamically move from root to tip when forces or collisions occur.

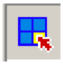
To model the curtain into a tied-back position

1 Dolly and tumble in the scene until you are facing the curtain directly.

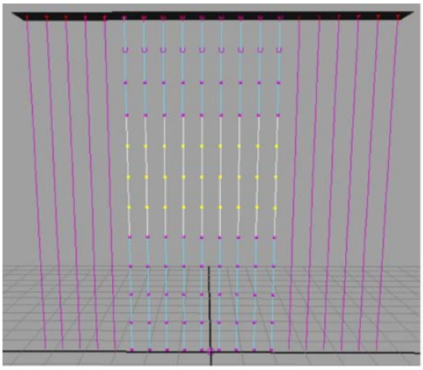
2 Drag to select all the hair curves and then select Hair > Convert Selection > To Start Curves.

3 Select a block of Start curves in the middle of the curtain.

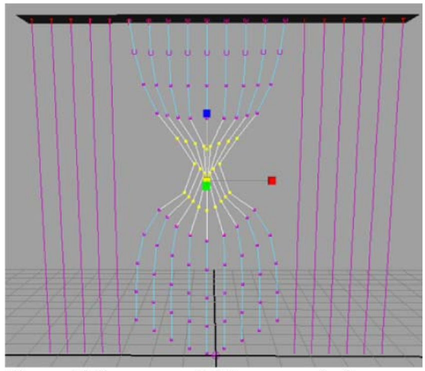
4 Select Hair > Modify Curves > Lock Length. This ensures the entire curve is modified when you transform CVs in the next steps.

5 Click the  button on the Status Line to display the CVs on the curves.

6 Select the middle three rows of CVs on the selected hair curves and then, using the Scale Tool, scale the CVs inwards so the hairs are pulled together.

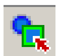


The middle rows of CVs are selected



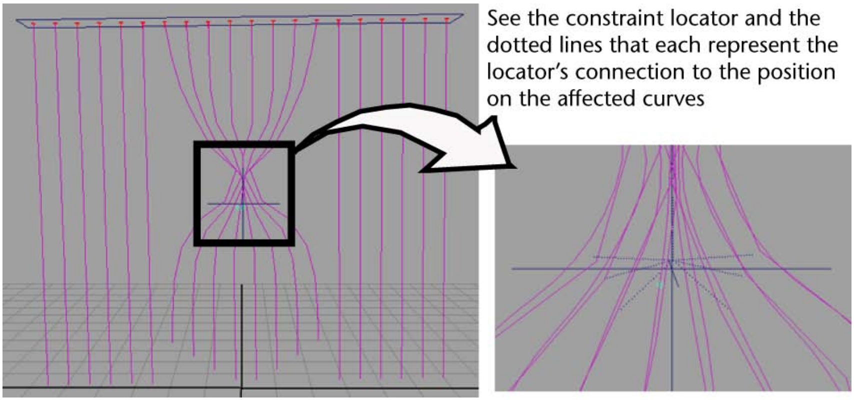
The middle rows of CVs are scaled inwards so the hair curves appear bound

To constrain the curtain in the tied-back position

1 Click the  button on the Status Line to display the Start curves (not the CVs).

2 With the bound hair curves still selected, select Hair > Create Constraint > Hair to Hair.

A constraint locator is created and connected to the selected curves.



3 In the hairConstraintShape tab in the Attribute Editor, set the constraint attributes as follows:

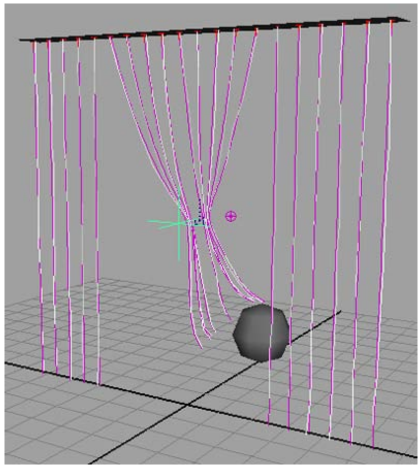
- Stiffness to 0.5 (default)
- Glue Strength to 1.0 (default)
- Point Method to Nearest (default) (Since the constraint locator is created in the middle of the bound hair, the locator is "nearest" the bound area to be constrained.)

4 Change the hair display to the Current Position (Hair > Display > Current Position).

5 Rewind to the start frame and play the simulation.

When the sphere passes through the bound curtain it collides with all the bound hairs.

The bound part of the curtain relaxes within the first several frames, but in the next steps you reset the position of those Start curves so they look more natural at the beginning of the simulation.



To reset the Start position

1. Display the Start curves and the Current Position (Hair > Display > Current and Start).
2. Rewind to the start frame and play the simulation.
3. Stop the simulation as soon as the curves relax below the constraint, but before the collision occurs.
4. Select the bound Start curves (these are the bound curves that are not blue) and then select Hair > Set Start Position > From Current.

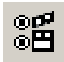
   You may have to repeat steps 2 to 4 a few times until you are happy with the look of the bound curtains at the beginning of the simulation.

Rendering the curtain scene

In these next steps you render the curtain, which is Paint Effects hair, in the Maya Software renderer.

To render the Paint Effects hair using the Maya Software renderer

1 To modify the Render Settings for Paint Effects do the following:

- Click the Render Settings  button on the Status Line. The Render Settings window appears.
- Select Maya Software from the Render Using drop-down list.
- In the Maya Software tab, go to the Paint Effects Rendering Options section and turn on Oversample and Oversample Post Filter. These help achieve a better look for the Paint Effects rendering.
- Click Close to close the Render Settings window.

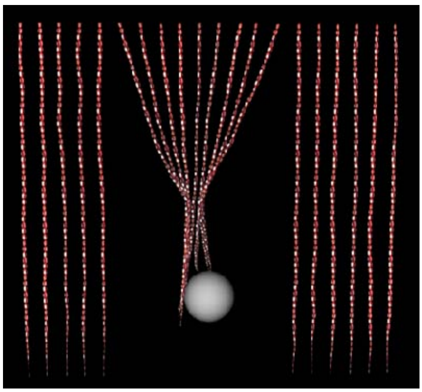
2 Before rendering be sure to display the hair system's Current Position (Hair > Display > Current Position).

3 Play the simulation and stop at the frame you want to render.

4 Click the Render Current Frame  button on the Status Line.

The curtain scene is rendered and should look something like the image below.

If you have trouble seeing the beads, try moving the light closer to the curtains or increasing the light intensity in the Channel box.



Beyond the lesson

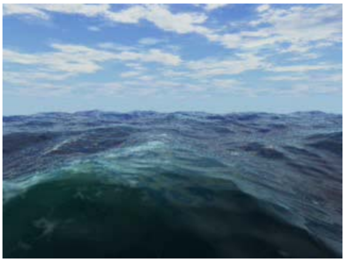
In this lesson you learned how to:

- Set up objects to collide with hair.
  You can set up hair to collide with itself, the ground plane or other objects. You can even use the Collision Sphere and Collision Cube constraints to establish implicit collisions. For more information, see Make hair collide in the Hair guide.
- Assign a Paint Effects brush to a hair system.
  This is useful for creating a variety of effects, such as vines or ivy for hair. The hair output can be Paint Effects or NURBS curves. For more information, see Assign a Paint Effects brush to Hair in the Hair guide.
- Create constraints on hair curves.
  Use hair constraints to hold hair in position for a hairstyle, such as a ponytail or a clasp. You can even use hair constraints for more realistic hair behavior, such as using the Transform constraint to hold the hair to the head. For more information, see Set up hair constraints in the Hair guide.

Fluid Effects

3

Introduction



Fluids are substances that change shape continuously or flow in response to forces.

Using Maya Fluid EffectsTM, you can realistically simulate 2D and 3D atmospheric effects (such as clouds and mist), combustion effects (such as smoke, explosions, and burning flames), and viscous fluid effects (such as molten lava).

You can also use the Fluid Effects ocean shader to simulate open water.

In Maya, there are three basic types of fluid effects:

- Dynamic fluid effects – Behave according to the natural laws of fluid dynamics and simulate the effects by solving fluid dynamics equations for each time step.
- Non-dynamic fluid effects – Use textures and animation to simulate fluid and fluid motion. These types of effects do not use fluid dynamics equations. Dynamic and Non-Dynamic effects render as 3D volumes, so they interact appropriately with objects that move through them.

- Open water fluid effects – Use shaders to create realistic wave motion on large bodies of water.

This chapter includes the following lessons, which introduce you to some basic Fluid Effects concepts:

- Lesson 1 Creating a dynamic 2D fluid effect: Introduction on page 49
- Lesson 2 Creating a non-dynamic 3D fluid effect: Introduction on page 59
- Lesson 3 Creating a dynamic 3D effect: Introduction on page 68
- Lesson 4 Creating an ocean effect: Introduction on page 78

Preparing for the lessons

To ensure the lessons work as described, do these steps before beginning:

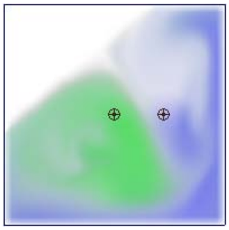
1. Create a new scene.
2. If you have not already done so, copy the `GettingStarted` folder from its installation location to your `projects` directory. Then, set the `GettingStarted` directory as your Maya project. For more information, see Copying and setting the Maya project in the Getting Started with Maya guide.
3. Select the Dynamics menu set.

   Unless otherwise noted, the directions in this chapter for making menu selections assume you've already selected the Dynamics menu set.
4. Switch to shaded display mode for a hardware render view of the fluid simulations (Shading > Smooth Shade All on the view menu bar).
5. In the Time and Range slider, set the start frame to 1 and end frame to 200.
6. Before you perform the lessons in this chapter, ensure that the Interactive Creation option for primitives is turned off by selecting Create > Polygon Primitives > Interactive Creation and Create > NURBS Primitives > Interactive Creation. That is, ensure that a check mark does not appear beside either of these menu items.

7 Make sure you understand the basic usage of the animation playback controls. See the chapter entitled “Animation” in Getting Started with Maya or refer to the Maya Help.

Lesson 1: Creating a dynamic 2D fluid effect

Introduction



Fluid dynamics is a branch of physics that uses mathematical equations to describe how things flow.

In Fluid Effects, dynamic fluid effects simulate fluid motion by solving fluid dynamics equations at each time step. You can texture dynamic fluids, have them collide with and move geometry, affect soft body geometry, and interact with particles.

Non-dynamic fluid effects do not use fluid dynamics equations to simulate fluid behavior. You create the look of a non-dynamic fluid using textures, and you create fluid motion by animating (keyframing) texture attributes. Because Maya doesn’t solve the equations, rendering this type of fluid is much quicker than rendering a dynamic fluid.

In this lesson, you create a simple two-dimensional dynamic fluid. You learn how to:

- Create a fluid using fluid emitters
- Modify the dynamic behavior of the fluid by applying forces to it and changing the effect the boundary has on the fluid
- Change the color of the fluid
- Collide the fluid into an object

In the next lesson, you'll create a non-dynamic fluid effect.

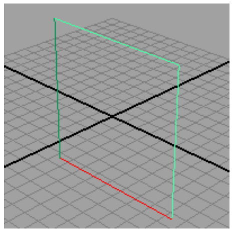
Creating a two-dimensional fluid container

You begin the lesson by creating a two-dimensional *fluid container*. The fluid container is the basis for dynamic and non-dynamic fluid effects. As its name implies, it contains the fluid effect. Fluids cannot exist outside a fluid container.

To create a two-dimensional fluid container

1 Select Fluid Effects > Create 2D Container.

Maya creates an empty 2-dimensional fluid container centered at the origin in the XY plane.



Next, you add fluid to the container. To do this, you add values to the container for specific properties of the fluid: Density, Color, Velocity, Fuel, and Temperature. You can add any or all of these properties to create a fluid effect. One way to put properties into a container is to create them with a fluid *emitter*.

Adding a fluid emitter to a container

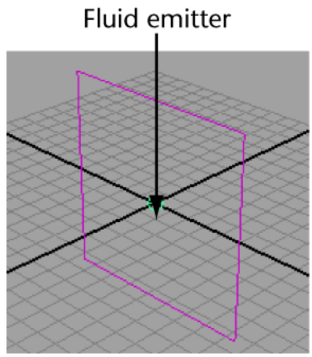
Fluid emitters create fluid property values and emit them into containers as the simulation plays. The emitter controls the position, rate of emission, and turbulence forces applied to the fluid properties when they are created. After fluid property values are created, the attributes of the fluid container control their appearance and behavior.

You can add multiple fluid emitters to containers. You move fluid emitters the same way you move any geometry (for example, by setting keys).

To add a fluid emitter to the container

1 With the container selected, choose Fluid Effects > Add/Edit Contents > Emitter.

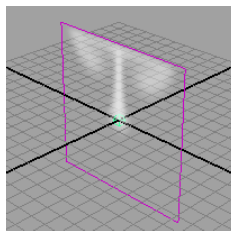
Maya creates a fluid emitter called fluidEmitter1 and places it at the center of the fluid container.



TIP You can create a fluid container with a fluid emitter in one step using Fluid Effects > Create 2D Container with Emitter.

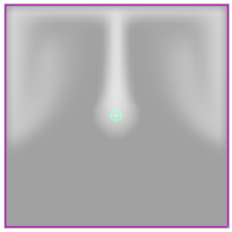
2 Play the simulation using the playback controls at the bottom of the Maya window.

The fluid emitter creates Density values and emits them into the container.



The white in the container represents *Density*. Density is the substance of the fluid and the visible fluid property. Where the Density values are highest, the Density appears more opaque.

Notice that the Density rises to the top of the container and moves along the boundary of the container, eventually flowing down the sides. Forces built into the container are acting on the Density values, causing them to change. The fluid collides with the boundaries rather than penetrating them.



3 Stop the playback and go to the start of the playback range.

Changing the behavior of a fluid

By modifying the attributes of the fluid container you affect the appearance and behavior of the fluid. Note that no matter what changes you make to the attributes, the fluid can never leave the container.

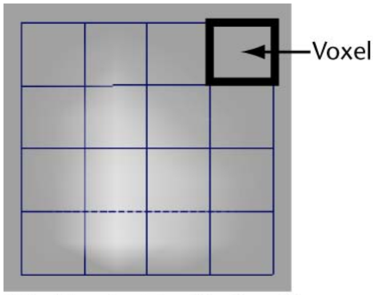
In the next steps you'll look at some of the attributes of the fluid container to gain some understanding into dynamic fluid effects.

To modify attributes for the fluid

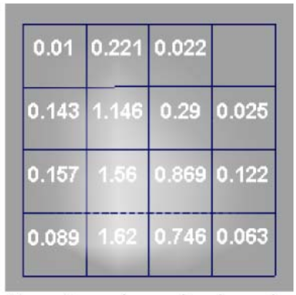
1. Select the fluid container.
2. Show the Attribute Editor in the right panel of the Maya window, and click the *fluidShape* tab.
3. Look in the Contents Method section. The Contents Method defines how a fluid property is defined in the container, if at all.

   Notice that Density and Velocity are set to Dynamic Grid.

   The Dynamic Grid setting divides the container into virtual rectangular units called voxels (volume pixels). You place the values in this virtual grid by emitting them, painting them, or adding a predefined initial state. At each step of the simulation, Maya recalculates the values in each voxel using the fluid dynamics solver. This is what creates the dynamic motion of the fluid.



2D virtual grid with Density



Density values displayed

TIP You can display Density values for the virtual grid by changing Numeric Display to Density in the Display section of the fluid Attribute Editor. Set the shading mode to Wireframe.

For dynamic fluid simulations, Density must be set to Dynamic Grid. Also, Velocity cannot be turned Off because velocity moves the property values around inside the grid.

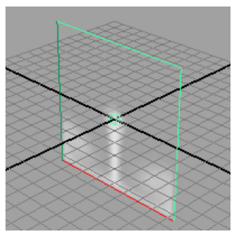
You look at defining properties as Gradient and Static Grid in the next lessons.

4 Make the density fall instead of rise by opening the Dynamic Simulation section of the Attribute Editor and setting the following:

- Change the Gravity value to -9.8. (Acceleration due to gravity is -9.8 m/s2 in the physical world.)

Negative values move the Density down.

5 Play the simulation using the playback controls at the bottom of the Maya window.

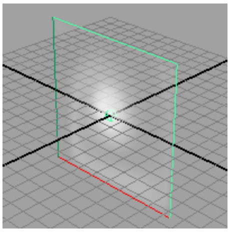


6 To add turbulence to the container and give the Density a whirling circular motion.

- Ensure the fluid container is selected if it is not already.

- In the Attribute Editor, open the Contents Details section, and then open the Turbulence section.
- Change the Strength value to 1.0.

7 Play the simulation using the playback controls at the bottom of the Maya window.

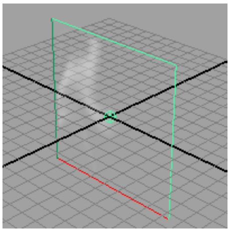


8 Stop the playback and go to the start of the playback range.

9 Change the effect of the container boundaries on the fluid:

- In the Container Properties section at the top of the Attribute Editor, set the following options:
- Boundary X: None
- Boundary Y: None

10 Play the simulation using the playback controls at the bottom of the Maya window.



Notice that the fluid no longer collides with the sides of the container—it appears to pass through the sides. The fluid behaves as though the container boundaries don't exist. Note that even though the fluid appears to pass through the container boundaries, it does not exist outside the boundary. Fluids can only exist inside containers.

11 Change Boundary X and Boundary Y back to Both Sides to restore the boundaries at all sides of the container.

12 Stop the playback and go to the start of the playback range.

Combining colors in a fluid

In the next steps, you'll add another emitter to the container and make each emitter create different colored Density. You'll see how different colored Density interacts in the container.

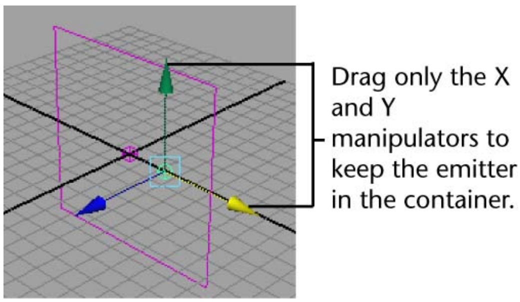
To combine colored fluids in the container

1 Select the fluid container and choose Fluid Effects > Add/Edit Contents > Emitter.

Maya creates a second fluid emitter called fluidEmitter2 and places it at the center of the fluid container, in the same position as the first emitter.

2 With fluidEmitter2 still selected, select the Move Tool (Hotkey: w) and drag the emitter to the right of the container, so the emitters no longer overlap.

Make sure you keep the emitter inside the container and on the plane by dragging only the X and Y manipulators. If you move the emitter outside the container, it will not emit.



3 Open the Attribute Editor in the right panel of the Maya window, and click the *fluidEmitter2* tab.

4 In the Fluid Attributes section, turn on Emit Fluid Color.

A message appears prompting you to set the fluid's Color Method to Dynamic Grid.

By setting the Color Method to Dynamic Grid, the color values you place in the container grid are recalculated by the fluid dynamics solver and changed during simulation. This means colors can interact during the simulation.

5 Click Set to Dynamic.

6 Click the Fluid Color color box in the Attribute Editor.

The Color Chooser opens.

7 Select *blue* from the Color Chooser and click Accept.

8 In the scene view, select *fluidEmitter1*, the emitter in the center of the container.

9 In the Attribute Editor, turn on Emit Fluid Color for fluidEmitter1.

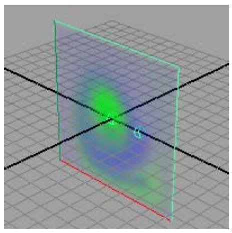
You are not prompted to set the contents method to Dynamic Grid, because you already set the container to Dynamic grid when you modified fluidEmitter2.

10 Click the Fluid Color color box.

The Color Chooser opens.

11 Select *green* from the Color Chooser and click Accept.

12 Play the simulation using the playback controls at the bottom of the Maya window.



13 Stop the playback and go to the start of the playback range.

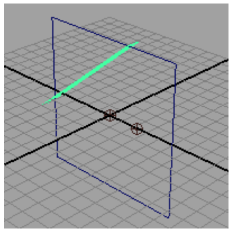
Colliding a fluid with an object

The 2D fluid you created is dynamic, so it can interact with objects. In the next steps, you'll create a plane inside the fluid container and make the fluid collide with it, rather than pass through it.

To make the fluid collide with a plane

1 Create a polygon plane:
   - Select Create > Polygon Primitives > Plane > ❐ . The Polygon Plane Options window opens.
   - In the window, select Edit > Reset Settings to reset the plane settings to the default ones, and then click Create.

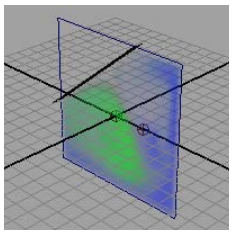
2 In the Attribute Editor, click the pPlane1 tab and scale, translate, and modify the plane as follows:
   - Translate: -2 2 0
   - Rotate Axis: 0 0 40
   - Scale: 10 1 1



3 Select both the plane and the fluid container.

4 Select Fluid Effects > Make Collide. This makes the plane a collision object.

5 Play the simulation using the playback controls at the bottom of the Maya window.



The fluid collides with the plane as it moves through the container. It does not pass through the plane.

Beyond the lesson

This lesson introduced you to some basic concepts of dynamic fluids. You learned how to:

- Create an empty fluid container.
- Emit fluid properties into the container to create a dynamic fluid effect.
- Modify forces acting on the fluid container to change the behavior of the contents of the container (Gravity and Turbulence).
- Add color to a fluid by emitting color into the container.
- Make a fluid collide with an object.

The hardware render of the simulation provided you with a fast, realistic representation of the fluid. The final step would be to do a software render.

Although we looked at a 2D fluid, the same concepts apply to 3D fluids. In the lessons that follow, you'll work with 3D fluids and explore methods other than fluid emitters for adding contents to a fluid container. You'll look at the texturing capabilities that are part of the built-in shader. You'll also learn about the Temperature and Fuel fluid properties and how you can use them in your effects.

Beyond this, you can also:

- Emit fluid from the surface of an object.
- Give the fluid a hard surface (surface render) for a globby or lava-like effect, rather than the soft, cloud-like surface you saw in this lesson.
- Cache the fluid simulation for faster playback.
- Deform an object with the force of a fluid.

For details on these topics, see the Fluid Effects information in the Maya Help.

Lesson 2: Creating a non-dynamic 3D fluid effect

Introduction

In the first lesson, you created a simple 2D dynamic fluid. 3D fluids intrinsically require extra data to define them, which can make them very complex. This extra data can slow a dynamic simulation exponentially because more calculations (solving) must be performed at every step of the simulation. For a less memory intensive effect, you could use a 2D fluid (with less data), or you could create a 3D non-dynamic effect.

In non-dynamic fluid effects, the fluid property values are predefined within Maya and stay constant over time, which means they don't have to be recalculated. You create the appearance of the fluid by texturing a special fluid shader that is assigned to the fluid. This shader is built into the fluid for better performance. If you want the fluid effect to have motion, you can animate (keyframe) the texture attributes. Because Maya doesn't solve the fluid dynamics equations, rendering this type of fluid is much quicker than rendering a dynamic fluid.

In this lesson, you learn the fundamentals of creating non-dynamic fluids by creating a cloud bank using a 3D non-dynamic fluid. You learn how to:

- Create a 3D fluid using predefined gradient contents.
- Texture the contents of the fluid using the built-in fluid shader.

Before beginning the lesson, do the steps in Preparing for the lessons on page 48.

Creating a 3D fluid container

Creating a fluid container is the first step in creating a non-dynamic fluid effect.

Fluid containers do not need to be uniform in X, Y, and Z. In fact, because larger fluids take longer to render, you should make containers only as large in each direction as necessary for the effect. A cloud bank sits above the ground like a blanket, so the container can be rectangular rather than cubic.

To create a 3D fluid container

1. Select Fluid Effects > Create 3D Container > ❐ .

   The Create 3D Container Options window opens.

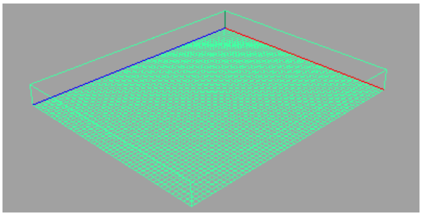
2. Set the container options as follows, then click Apply and Close:

   - X Resolution: 50
   - Y Resolution: 5
   - Z Resolution: 60
   - X Size: 50.0
   - Y Size: 5.0
   - Z Size: 60.0

   The resolution is defined in voxels. Higher resolutions produce finer detail, but increase rendering time. The size of the container is defined in the working units set for Maya. The size of the fluid determines the size of each voxel.

   TIP It's good practice to make the container's Resolution and Size proportional to ensure that the voxels are square making the quality the same along each axis.

   Maya creates an empty 3-dimensional fluid container centered at the origin.



Next, you add fluid to the container.

Adding fluid to a container

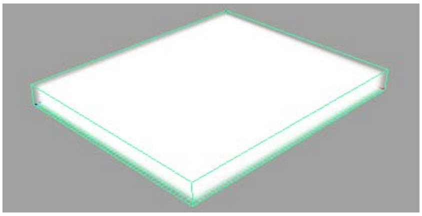
To make this a non-dynamic fluid, you add fluid property values to the container that will stay constant over time. There are two ways to do this: add these values to a static grid, or select a set of values from a list of predefined gradients that come with Maya. For this lesson, you'll use the latter method.

For a cloud bank, Density is the only property you need to define—the other properties are typically used in dynamic solving.

To add fluid to the container

1. Dolly the scene view to see the entire container.
2. With the fluid container selected, show the Attribute Editor in the right panel of Maya, and click on the fluidShape1 tab.
3. In the Contents Method section of the Attribute Editor, set:
   - Density: Gradient
   - Density Gradient: Constant

   Set Velocity, Temperature, and Fuel properties to Off. (They're not used in this effect.)



By setting the Density Gradient to Constant, you made all the Density values in the container the same—a value of 1. You can scale these values, but otherwise, they cannot change.

Defining shader attributes for a fluid

The fluidShape has a built-in shader that you can use to modify color, texture, and other attributes to give the fluid a particular look. In the next steps, you'll make the shader apply specifically to the Density values inside the container.

To define shader attributes for the fluid

1. Open the Shading section in the Attribute Editor.
2. In the Color subsection, change Color Input to Density.

   The Density values in the container will now take on the colors defined on the color bar. Leave the color white to make the clouds appear white.
3. In the Opacity subsection, check that the Opacity Input is set to Density.

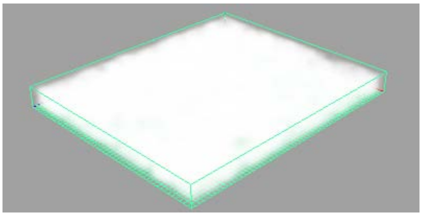
   The opacity represents how much the Density will block light. You'll look more at Opacity later in this lesson.

Texturing the contents of a fluid container

So far the container is filled with solid white Density. To create the cloud effect, you texture the Density so that some areas are transparent and some areas are opaque.

To texture the density of a fluid

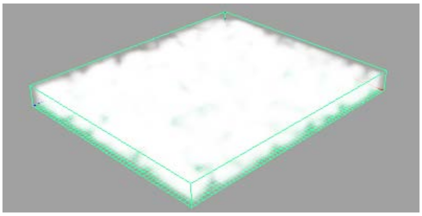
1. Turn on hardware texturing display so you can see the effect of the textures on the fluid without rendering by selecting Shading > Hardware Texturing from the scene view menu.
2. Open the Textures section in the Attribute Editor.
3. Turn on Texture Opacity to apply the current texture to Opacity values. The current texture is Perlin Noise, defined by Texture Type.



Notice that the Density now has a slightly blotchy look to it, with areas that are more opaque and areas that are more transparent. This texture provides the standard 3D noise used in the 3D Solid Fractal texture included with Maya.

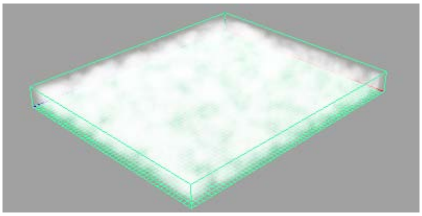
4 Change Texture Type to Billow for a fluffy, cloud-like effect.

The Billow texture is computationally intensive and therefore slower than the other texture types.



5 Change the look of the texture by setting the following texture attributes:

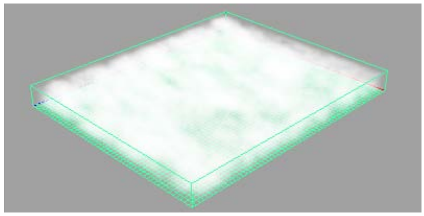
- Amplitude: 0.5
- Depth Max: 4



Decreasing the Amplitude makes the areas with low Density more transparent and the areas with high Density more opaque.

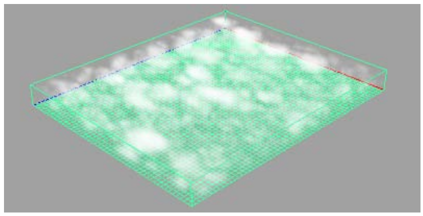
Increasing Depth Max adds detail. Increasing it will also increase render time.

6 Stretch the texture in the X direction by changing the X, Y, and Z components of the Texture Scale to 2, 1, 1.



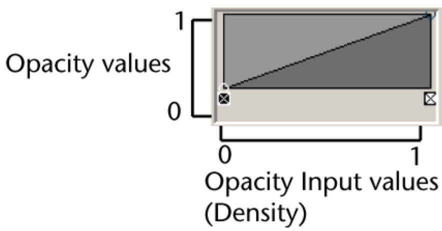
7 Change the following Billow texture attributes to make the "billows" less dense, more spotty, and with randomly different sizes.

- Billow Density: 0.6
- Spottyness: 2.0
- Size Rand: 0.40



8 Modify the Opacity so that areas in the container that are very dense appear less opaque, areas that have very little Density become totally transparent, and the transition between areas that are totally transparent and areas that are more opaque is less gradual.

In the Attribute Editor, go to the Shading section. Look at the Opacity graph in the Opacity subsection. This graph represents the relationship between Opacity values and Density values (the Opacity Input).

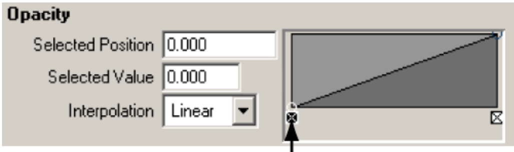


Opacity values range from 0 on the bottom (totally transparent, no opacity) to 1 on the top (totally opaque).

Density values range from 0 on the left side (no Density) to 1 on the right side (high Density).

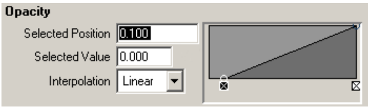
So for the linear graph shown above, where Density values are 0, Opacity values are 0, making the Density totally transparent, where Density values are 0.5, Opacity values are 0.5, making the Density partially opaque, and where Density values are 1, Opacity values are 1, making the Density totally opaque.

- Click the first dot on the Opacity graph to select the position marker. Position markers mark the location on the graph from left to right (the Opacity Input value). The outline of the dot is white when a position is selected.

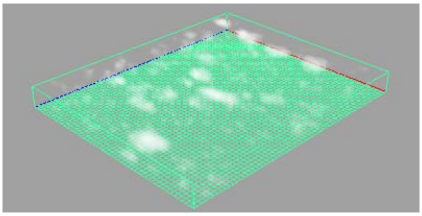


Click this dot to select the position marker.

- Change Selected Position to 0.10 to change the position of the marker.

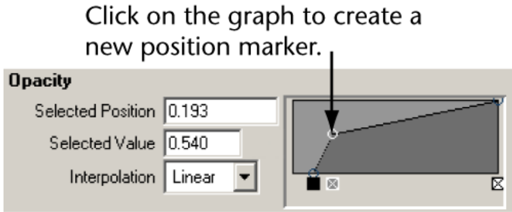


The position marker moves to the right. Now, for Density values between 0 and 0.10, the Opacity values will be 0. This means that Density that was previously partially transparent will be completely transparent.

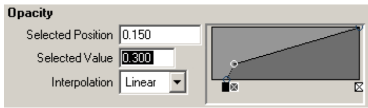


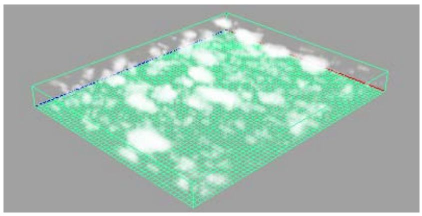
The more transparent areas of cloud disappeared, but now the solid areas of cloud are less opaque.

- Click on the graph to create a new position marker.



- Change the marker position and value as follows:
  - Selected Position: 0.15
  - Selected Value: 0.30





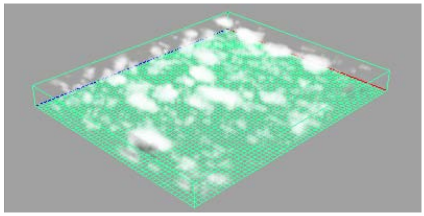
Density values that are greater than 0.15 are now more opaque, and the transition between areas of total transparency (Opacity 0), and areas where the Density becomes more visible (Density 0.15) is less gradual.

Adding self shadowing to texture density

Give the clouds some depth by adding self shadowing. Self shadowing causes the fluid to cast shadows on itself using a single internal directional light at -1, -1, -1.

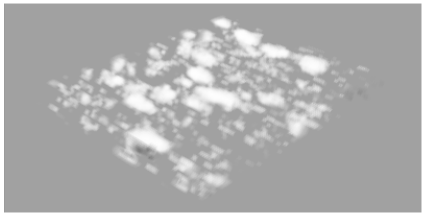
To add self shadowing to the texture density

1. In the Attribute Editor, under the fluidShape1 tab, open the Lighting section.
2. Turn on Self Shadow.



The clouds now have some darker areas on them, giving them some depth.

3 In the Display section, change Boundary Draw to None to hide the container. This gives you a better idea of how the fluid will look before you render it.



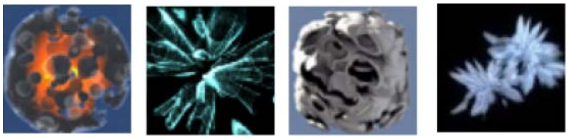
Beyond the lesson

This lesson showed you a basic technique for creating non-dynamic fluids. While creating the effect, you learned how to:

- Create a 3D fluid container of a specific resolution and size.
- Add predefined gradient contents to the container to create a non-dynamic fluid.
- Texture the contents of the fluid using the built-in fluid texturing capabilities of the fluidShape. (You can also texture dynamic fluids using the same techniques.)

To make the texture move, you would keyframe the Texture Time attribute in the Textures section of the Attribute Editor. For details on keyframing attributes, see the Maya Help.

You modified several texture attributes in this lesson, but there are many more attributes for customizing textures. Also, note that the non-dynamic effects you can create using texturing are not limited to clouds. Here are some examples:

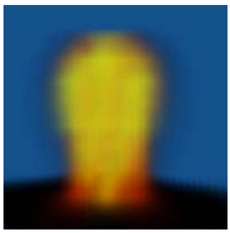


To learn more about how to use texturing and other fluid attributes, study the Fluid Effects examples that are included with Maya (Fluid Effects > Get Fluid Example) and look at the Notes section at the bottom of the Attribute Editor for example fluids. It includes information about the selected effect, pointing out the key aspects of its construction.

For further information on Fluid Effects, see the Maya Help.

Lesson 3: Creating a dynamic 3D effect

Introduction



In the real world, gasoline reacts chemically with oxygen at a certain temperature, creating flames (light, heat) and the gaseous products of combustion. Over time, the gasoline burns until it's totally consumed. You can simulate this situation in Fluid Effects using Fuel, Density, and Temperature properties.

In this lesson, you create a dynamic explosion effect by filling the bottom of a container with Fuel and Density, then "igniting" it with Temperature. You learn how to:

- Add property values to a container by painting them.
- Paint in a 3D container.
- Use Fuel, Density, and Temperature to create an explosion effect.

Before beginning the lesson, do the steps in Preparing for the lessons on page 48.

Creating a 3D container

1 Select Fluid Effects > Create 3D Container > □ .

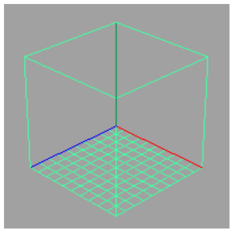
The Create 3D Container Options window opens.

2 In the window, select Edit > Reset Settings to restore the Resolution and Size settings to the defaults 10 10 10 and 10 10 10, respectively.

The default container size and resolution are adequate to illustrate the concepts in this lesson.

3 Click Apply and Close.

Maya creates an empty 3-dimensional fluid container centered at the origin.



Next, add fluid to the container. In the first lesson, you added fluid to the container by emitting it into a Dynamic Grid. In the second lesson, you added fluid to the container by selecting a predefined gradient. In this lesson, you will add fluid to the container by painting property values inside it.

Painting Fuel and Density into a container

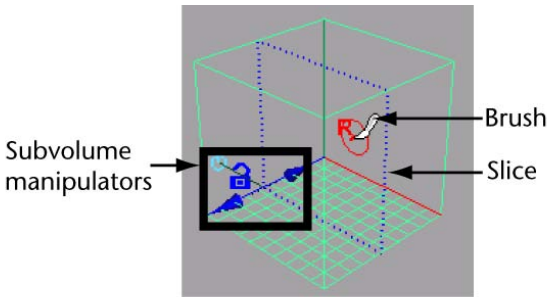
In Fluid Effects, Fuel defines the state of a reaction (unreacted, completely reacted, and states in between). Density represents the substance that is reacting, while Temperature causes the reaction. Combining Density with Fuel defines a situation in which you can see a reaction. You add Temperature to this situation to start the reaction. As the reaction takes place, Density values become smaller and the Fuel values get smaller to represent how much of the reaction has yet to take place. The reaction also creates more Temperature, and it creates light.

In the following steps you'll paint Fuel and Density values in the container. To paint in a 3D container you actually paint in two dimensions on "slices" of the container. A slice is a plane in X, Y, or Z that represents where you paint. (You could think of a slice as a 2D canvas.) You paint each slice individually, but the accumulative effect of adjacent slices is a 3D fluid.

To paint Fuel and Density into a container

1 With the container still selected, choose Fluid Effects > Add/Edit Contents > Paint Fluids Tool > ❐ .

The Tool Settings window opens and a slice displays at the origin of the fluid container. The slice is represented by a plane with dotted edges and fluid subvolume manipulators at one corner. When you move the pointer over the slice the pointer changes to a brush indicating that you can paint.



1 At the top of the Tool Settings window, click Reset Tool to set the Paint Fluids Tool settings to the default values.

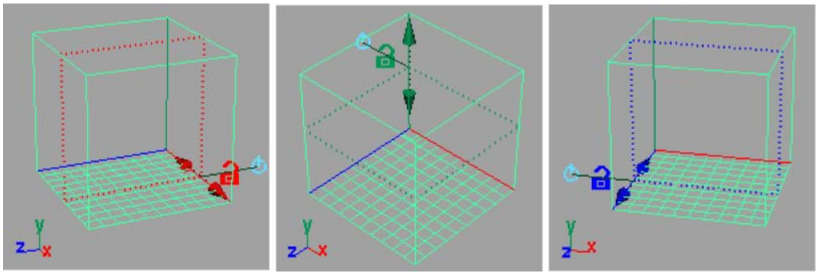
2 Select the fluid properties you want to paint. In the Paint Attributes section of the Tool Settings window, beside Paintable Attributes, select Density and Fuel. You will paint both these properties at the same time.

   You could paint each property separately, but for this example, the values you paint will be the same for both. It is more efficient to paint them both at the same time.

   A message appears prompting you to set the fluid's Fuel method to Dynamic Grid. By setting the Fuel method to Dynamic Grid, the fuel values you place in the container grid (in this case, by painting them) are recalculated by the fluid dynamics solver and changed during simulation. The Density method is defined as Dynamic Grid by default.

3 Click Set to Dynamic.

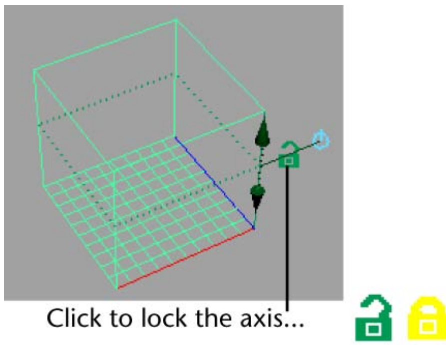
4 Tumble the scene and look at the slice and manipulators.



The orientation of the slice changes relative to your view. The slice is perpendicular to the axis with the manipulators. The color of the manipulators corresponds with the color of the axis in the View or Origin axes.

Painting properties relative to different axes is good practice to ensure there are no gaps between slices when the fluid is viewed from different angles.

5 Tumble so that the slice is perpendicular to the Y axis and then click the open lock icon to lock the slice axis.



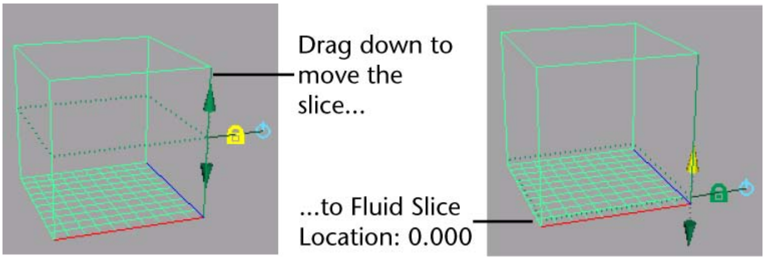
The lock closes. Now when you tumble, the slice remains perpendicular to Y. It does not switch to the other axes. This allows you to change the view in any way and still be able to paint on the same slice.

6 Drag the top move arrow down along the axis to the first slice at the bottom of the container. The number of slices along each axis corresponds with the resolution of the fluid container. The numbering starts at 0 at

the intersection of the axes. For this container the resolution is 10 10 10, so the slices are numbered from 0 to 9.

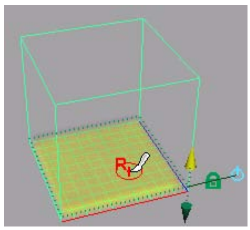
The help line displays the location of the selected slice.

Fluid Slice Location: 0.000



7 In the Paint Attributes section of the Tool Settings window, ensure that Value is set to 1. A Fuel value of 1 represents a totally unreacted state while a Fuel value of 0 represents a completely reacted state. A Density value of 1 is totally opaque while a Density value of 0 is completely transparent.

8 Drag the brush on the slice to paint values. Paint until the entire slice is filled with values.

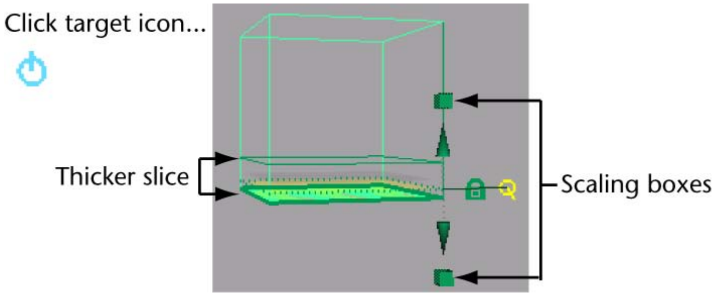


The values display as yellow because you are painting two properties at the same time. Density values are represented by the opacity of the shaded values, while Fuel values are represented by color. The range of Fuel values corresponds with a ramp of colors from blue (values of 0) to yellow (values of 1).

To make the fluid deeper, you could move the slice to Fluid Slice Position 1, paint, move the slice to Fluid Slice Position 2, paint, and so on. But a quicker way is to resize the slice to make it thicker and then flood the entire slice with values.

9 Resize the paint slice, making it thicker:

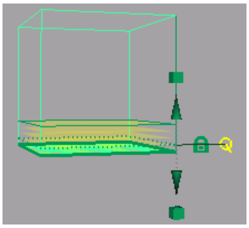
- Click the target icon once to switch to the subvolume scaling mode.



The slice becomes thicker and the manipulator changes to include scaling boxes at the ends of the move arrows.

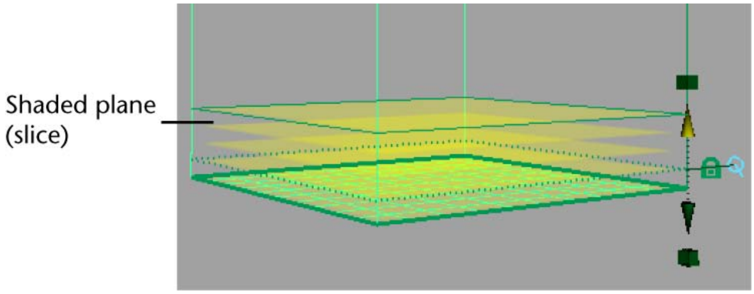
- Click the top scaling box and read the help line to determine that the Fluid Slice Thickness is 3.00. You could drag the scaling boxes to change the thickness of the slice, but the slice is thick enough for the lesson.

10 In the Tool Settings window, click the Flood button.



Flooding filled each voxel in the subvolume with Fuel and Density values of 1.

Notice that the values display as shaded planes rather than as a solid mass. This is an interactive display feature for shaded display mode. Displaying more shaded planes (slices) produces more detail but reduces the speed of the screen draw. You can change this display feature (Slices Per Voxel) and others in the Display section of the Attribute Editor.



Now that the container has Density and Fuel in it, you are ready to start a reaction by adding Temperature. You could think of this as putting a lit match into a container of gasoline.

Painting Temperature values into a container

By default, the Fuel and Density will react when there is no Temperature in the container, but adding higher temperature values will make the reaction more immediate.

To paint Temperature values into a container

1. In the Tool Settings window, beside Paintable Attributes, select Temperature. A message appears prompting you to set the fluid's Temperature method to Dynamic Grid or Static Grid.

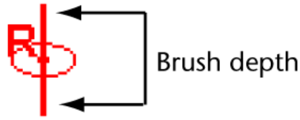
   By setting the Temperature method to Dynamic Grid, the temperature values you place in the container grid are recalculated by the fluid dynamics solver and changed during simulation.

   By setting the Temperature method to Static Grid, the temperature values you place in the container grid are used by the fluid dynamics solver during the simulation, but they cannot change during the simulation. Static Grid is useful when you want to define specific values in specific areas of the container, but you do not want these values to change over time.

2. Click Set to Dynamic. The Density and Fuel values disappear. The values are still there, but only Temperature values now display. Currently there are no temperature values in the container.

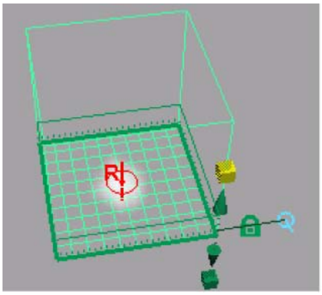
3. Change the brush so you can paint through the entire thickness of the slice with a single stroke. In the Stroke section of the Tool Settings

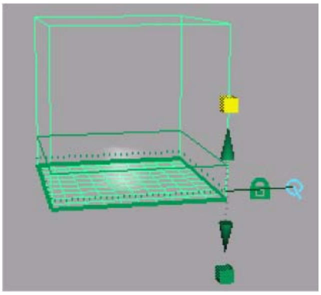
window, change Stamp Depth (3D) to 2. The brush changes, showing how deep the brush stroke will paint.



If you make the brush depth thicker than the depth of the slice, only the space within the slice will be painted.

4 Paint a small amount of Temperature in the center of the slice.





The stroke adds Temperature values through the entire thickness of the slice.

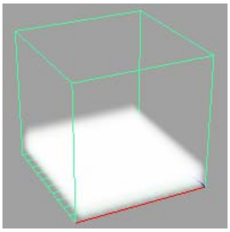
The next steps are to view all the property values together, and color the properties to differentiate them.

Adding color to Density and Temperature

Use the built-in shader to modify the color of Density and Temperature in the container.

To add color to the Density and Temperature

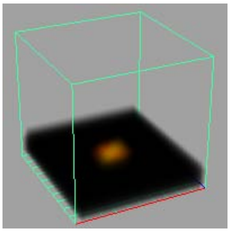
1 Choose the Select Tool (Hotkey: q) to leave the Paint Fluids Tool. The manipulators disappear and now you see the fluid as it will render



2 With the fluid container selected, open the Attribute Editor.

3 Change the color of the fluid as follows:

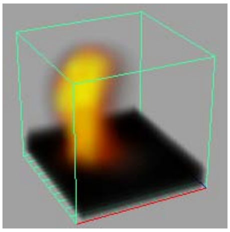
- In the Shading section of the Attribute Editor, expand the Color section, and change Color Input to Density.
- The Density values in the container will now take on the colors defined on the color bar.
- Click the Selected Color box. The Color Chooser opens.
- Select black in the Color Chooser and click Accept.



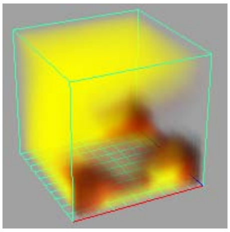
The Density turns black.

Look closely at the center of the fluid. The area where you painted Temperature has a reddish orange color. In the Incandescence section of the Attribute Editor, notice that Incandescence Input is set to Temperature. Temperature gives off light and Incandescence controls the amount of light emitted from regions of Density due to self illumination. The ramp defines the color of this light.

4 Play the simulation using the playback controls at the bottom of the Maya window.



By frame 160, most of the Density values have reacted completely.



Beyond the lesson

This lesson introduced you to the Fuel and Temperature fluid properties. You learned how to:

- Combine Fuel, Density, and Temperature to create a dynamic explosion effect.
- Add property values to a container by painting them.
- Paint in a 3D container.

You can control how quickly the reaction takes place during the simulation, the temperature required to start the reaction, how much heat and light the reaction releases over time, and other aspects of the reaction using the attributes in the Fuel section under Contents Details in the fluidShape Attribute Editor.

For a more realistic look, you can also texture the fluid using the built-in texturing attributes in the Textures section.

For more information on Fuel and texturing attributes, see Fluid Effects in the Maya Help.

Lesson 4: Creating an ocean effect

Introduction

An ocean effect is defined by a plane with an ocean shader assigned to it. Ocean effects are not created with fluid containers like the dynamic and non-dynamic fluid effects you created in the first three lessons.

A shader is a collection of attributes that define color and other surface characteristics of the plane it is assigned to. For an ocean shader, these attributes define realistic waves on large bodies of water.

In this lesson, you create an ocean, put a cube in it, and see how the cube floats in the ocean. You learn how to:

- Create an ocean plane and shader.
- Create a preview plane to see the effects of the shader, including displacement, without rendering.
- Modify the ocean shader attributes to make the waves bigger.
- Make an object a boat in the scene.
- Modify the attributes for an expression that controls the buoyancy of the boat in the ocean.

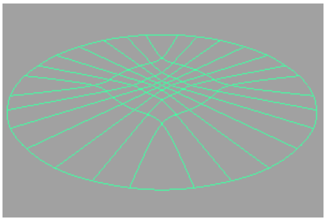
Before beginning the lesson, do the steps in Preparing for the lessons on page 48.

Creating an ocean plane and shader

Fluid Effects simplifies the process of creating an ocean by providing a single command that creates a plane that is optimized for best results, and an ocean shader with appropriate connections to the plane.

To create an ocean plane with ocean shader

1. Select Fluid Effects > Ocean > Create Ocean. Maya creates an ocean plane and assigns an ocean shader to it.
2. Dolly out to see the plane.



The plane is a NURBS plane with more patches concentrated in the center. This is so you can see more detail in the center and less detail as the ocean extends to the horizon.

Because interactive shading would slow down playback, the shading is turned off for the surface.

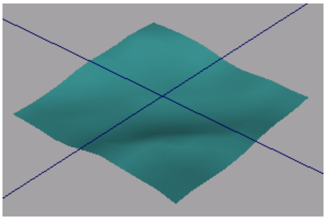
Adding a preview plane to an ocean

Because shading is turned off for the ocean plane, you need a preview plane to see displacement and shading in the hardware render. Displacement changes the surface of the ocean plane so you can see the different wave heights.

To add a preview plane to the ocean

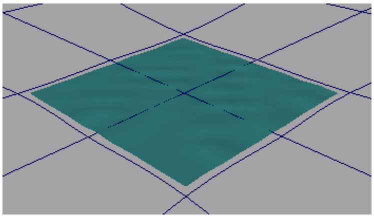
1. Dolly back in to the center of the plane.
2. Select Fluid Effects > Ocean > Add Preview Plane.

   Maya creates a preview plane that interactively shows the displacement of the ocean on a shaded patch of ocean. You can scale and translate it to preview different parts of the ocean. This plane does not appear in renders.



3 With the preview plane still selected, choose the Scale Tool (Hotkey: r) and make the patch bigger by dragging the center scale box to the right. Make it approximately 25 times bigger. (Look at the scaling numbers in the help line.)

4 Play the simulation using the playback controls at the bottom of the Maya window.



The preview plane ripples slightly showing the small wave displacement.

5 Light the scene to prepare it for rendering later in the lesson:

- Select Create > Lights > Directional Light. Maya creates a directional light called directionalLight1 at the center of the grid.
- Open the Attribute Editor and in the Directional Light Attributes section of the directionalLightShape1 tab, change Intensity to 2 to make the light brighter.
- Click the directionalLight1 tab and rotate the light as follows to direct the light rays onto the ocean. Rotate: -100 42 20

Modifying ocean attributes

In the next steps, you will modify ocean shader settings to make the waves bigger.

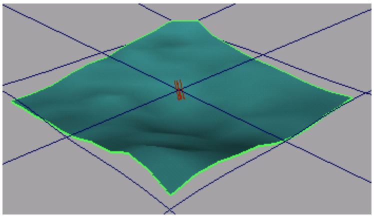
To modify attributes for the ocean shader

1. Render the current frame to see what the ocean looks like before you modify the ocean shader attributes. On the Status line, click the render current frame icon.

2. Select the preview plane or the ocean plane (it doesn't matter which one), and in the Attribute Editor, click the oceanShader1 tab.
3. Open the Ocean Attributes section and change the following values:
   - Num Frequencies: 12
   - Wave Length Max: 40

   Num Frequencies controls the number of frequencies between minimum and maximum wave lengths. Increasing this value results in a more textured look on the water surface.

   Wave Length Max controls the maximum length of waves in meters. Longer wave lengths make bigger waves.

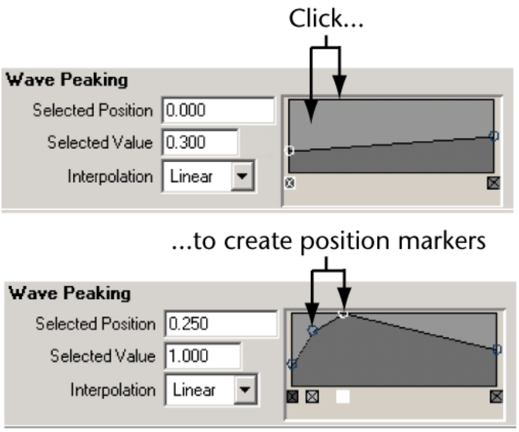


4. Play the simulation to see the effect of your changes on the displacement of the ocean waves.
5. Stop the playback and go to the start of the playback range.
6. Render the current frame.

7 Make the tops of the waves form crests by adjusting the Wave Peaking graph in the Attribute Editor.

   Wave Peaking controls the amount of crest formation for waves across the range of wave frequencies. It simulates a side-to-side sloshing of waves, as opposed to an up-down motion. Wave Peaking is only applied to turbulent waves (where Wave Turbulence is not zero).

   Click on the graph to create the position markers shown in the next illustration.



   Where the wave frequency is lower, the waves peak more.

8 Render the current frame.

The tops of the waves now crest slightly.

Notice the white highlights on the ocean. These are specular highlights caused by the light hitting the water.

9 Make the highlights dimmer so they don't pop out as much, and bigger. Open the Specular Shading section and change the following settings:

- Specularity: 0.45
- Eccentricity: 0.1

Specularity controls the brightness of the specular highlights. Higher values create brighter highlights.

Eccentricity controls the size of the specular highlights. Higher values create bigger highlights.

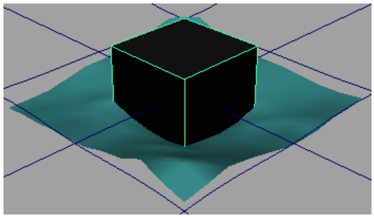
10 Render the current frame.

Floating objects

You can make geometry (for example, a boat) float in the ocean, moving appropriately with the motion of waves. The following steps show you how using a simple polygon cube.

To float a cube in the ocean

1. Create a polygon cube (Select Create > Polygon Primitives > Cube).
2. In the Attribute Editor, click the pCube1 tab and scale the cube as follows:
   - Scale: 10 10 10

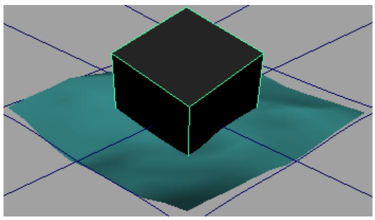


3. Select both the cube and the ocean plane (not the preview plane) at the same time and select Fluid Effects > Ocean > Make Boats.

   Maya creates a boat locator at the center of the cube to mark the cube's position in space. The locator is connected to a predefined expression that simulates buoyancy effects. The expression is connected to the wave heights (displacement) of the ocean.

   To learn more about expressions, look at the Expressions lessons in Getting Started with Maya.

4. Play the simulation using the playback controls at the bottom of the Maya window.



   The cube bobs, pitches and rolls in the ocean.

5. In the Attribute Editor, click the locatorShape tab and in the Extra Attributes section, modify the following options. These serve as inputs to the expression.
   - Buoyancy: 0.75
   - Roll: 0.1
   - Pitch: 0.2

Increasing Buoyancy makes the box float more and sink into the ocean less. Decreasing the Roll and Pitch values causes the cube to roll less from side to side, and pitch less forwards and backwards as a result of the wave motion.

Beyond the lesson

In this lesson, you learned that ocean effects are different from the dynamic and non-dynamic volume fluid effects. An ocean effect is created with a plane and shader rather than a container and fluid properties. With ocean effects you learned how to:

- Create an ocean plane and shader.
- Create a preview plane to see the displacement of the shader without rendering.
- Modify ocean shader attributes to customize the look of the ocean.
- Make geometry bob and move in the ocean.

To learn more about ocean shader attributes, look at the ocean shader examples that are included with Maya (in the Fluid Effects menu). Some of these examples use a 3D non-dynamic volume fluid to create the atmosphere above the ocean.

Motor Boats

You can create a game-style boat simulation using the Fluid Effects > Ocean > Make Motor Boats command. Once you set up hotkeys for the rudder and throttle of the boat, you can “drive” the boat in the ocean. The boat will roll, pitch, and jump over waves appropriately.

Boat Wakes

You can create boat wakes using the Fluid Effects > Ocean > Create Wake command. Wake fluids do not use the Navier Stokes solver like regular fluids, but use the Spring Mesh Solver. A fluid emitter is used to drive the motion of the fluid.

Ponds

If a smaller body of water is what you had in mind, you can create a pond using Fluid Effects > Pond > Create Pond. Ponds are 2D fluids that use a spring mesh solver and a height field. You can generate bubbles, ripples, and waves using the Create Wake options for the pond.

For details on these topics and others, see Fluid Effects in the Maya Help.

Fur

4

Introduction

With the Maya® FurTM feature, you can create fur and short hair on all surface types in Maya. A *fur description* in Maya defines all the attributes for the fur (for example, fur color, width, length, baldness, opacity, curl, density, and so on). When applying fur, you can use one of the predefined fur descriptions included with Maya Unlimited, or you can create your own custom fur description by setting all the fur attributes yourself.

You can animate fur with keyframe effects like growing fur or changing color. For a more natural appearance, you can also add movement to fur with dynamics (for example, wind and gravity).

This chapter includes these lessons:

- Lesson 1 Assigning a fur description: Introduction on page 88

- Lesson 2 Rendering fur: Introduction on page 113

Preparing for the lessons

To ensure the lessons work as described, do these steps before beginning:

1. If you have not already done so, copy the `GettingStarted` folder from its installation location to your `projects` directory. Then, set the `GettingStarted/Fur` directory as your Maya project. For more information, see Copying and setting the Maya project in the Getting Started with Maya guide.
2. Select the Rendering menu set. Unless otherwise noted, the directions in this chapter for making menu selections assume you've already selected the Rendering menu set
3. If you don't see a Fur menu, perform these additional steps:
   - Select Window > Settings/Preferences > Plug-in Manager.
   - In the Plug-in Manager, locate the Fur plug-in (Fur.mll) and turn on the loaded checkbox. Wait for the check mark to appear in the checkbox, then close the Plug-in Manager. You can set the Fur plug-in to *auto load* so that it automatically loads each time the Maya application is launched. The Fur menu will appear to the right of the Hair menu.

Lesson 1: Assigning a fur description

Introduction

Maya Unlimited provides many predefined fur descriptions. These fur descriptions, also referred to as *presets*, can be used as a starting point in creating fur for a model. Fur description presets can also be customized.

In this lesson you learn how to:

- Duplicate parts of a model across its axis of symmetry.
- Rename surfaces.

- Assign surfaces to display layers.
- Apply a fur description preset to the surfaces of a model.
- Reverse surface normals on surfaces to ensure the fur description is oriented correctly.
- Change the direction of the fur description on some of the surfaces.
- Paint a fur attribute map to modify the fur length in an area of a surface.
- Modify the fur color attributes of a fur description.
- Create a new fur description for a surface and modify its attributes.

Lesson setup

To ensure the lesson works as described, do these steps before beginning:

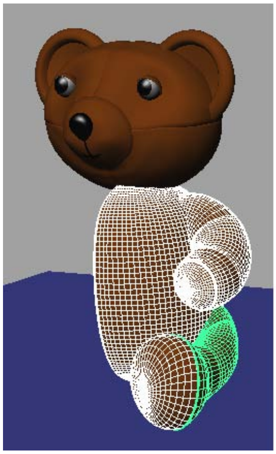
1. Open the scene named `furStart.ma` that is located in the `GettingStarted/Fur/scenes` directory of your Maya project.

   This scene includes a model that has been created for you.

Duplicating objects across an axis of symmetry

3D artists often construct one half of a symmetrical model in order to simplify the data requirements. The scene provided for this lesson reflects this situation.

The teddy bear has only one half of its body. In the next steps, you'll duplicate the surfaces that make up the arms, legs, and body across the center axis to complete the model.



One half of the model with the symmetrical surfaces selected.

To duplicate surfaces

1 In the scene view, select the five surfaces to be duplicated: the arm, arm paw, leg, leg paw, and body. Click one surface, then shift-click the other surfaces until all of the five surfaces are highlighted.

2 Select Edit > Duplicate Special > ❐ . In the Duplicate Special Options window that appears, select Edit > Reset Settings, then do the following:

- Change the X Scale value to -1.
- Click the Duplicate Special button.

Setting the X Scale value to -1 duplicates the surfaces in corresponding positions across the negative side of the X-axis.

The complete model.

Renaming surfaces on a model

After duplicating the surfaces you need to rename the duplicated surfaces so you can easily identify and select them by name when required.

The surfaces you duplicated were originally named using the naming convention leftArm, leftArmpaw, leftLeg, and leftLegpaw. When you duplicate objects, Maya assigns names to the new objects automatically. Change the names of the duplicated surfaces to match the naming convention used for the model.

To rename objects

1 For each of the surfaces listed in the following table, do the following:
- Click the object in the scene view.
- In the Channel Box, click the object name to select it. You can display the Channel Box by clicking Show/Hide Channel Box/Layer Editor button on the Status Line.
- Type the corresponding new name from the table and press Enter.

Old name	New name
leftArm1	rightArm

Old name	New name
leftArmpaw1	rightArmpaw
leftLeg1	rightLeg
leftLegpaw1	rightLegpaw
leftBody1	rightBody

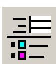
Assigning objects to a reference layer

In this section you will assign any objects that will not have fur assigned to a referenced display layer so they become temporarily unselectable when adding or changing a fur description.

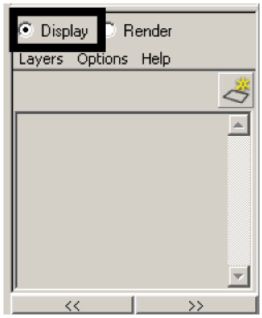
Layers are overlapping views of your scene that can have objects assigned to them. You can assign any objects within a scene to *layers*. One useful characteristic of layers is that they let you simplify your scene so you can work on specific objects without disturbing other objects. You can choose which objects of the scene you want to assign to a layer and then selectively make them visible or invisible. You can also template or reference all objects associated with a given layer.

To assign objects to a display layer

1 If the Channel Box isn't already displayed, click the Show/Hide Channel Box/Layer Editor button in the Status Line to display the Layer Editor.



2 Ensure the Layer Editor is set to Display Layers by clicking the Display radio button.



The Layer Editor set to Display Layers.

3 Click the Create Layer button. 

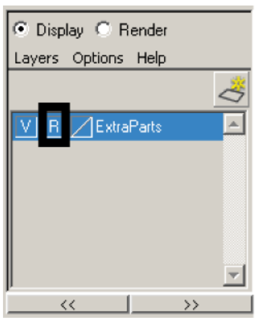
4 Double-click the new layer's name in the Layer Editor.

The Edit Layer window appears.

5 In the Edit Layer window, set the following and click Save:

- Name: `ExtraParts`
- Display type: Reference

In the Layer Editor, the name of the layer updates and its status is now set to Reference. The Reference display state indicates that the objects in that layer can still viewed in the scene but cannot be selected.



The new layer appears in the Layer Editor, and is set to Reference.

6 In the scene view, select the following surfaces:

- nose
- mouth
- leftEye

- rightEye
- leftLid
- rightLid
- leftInEar
- rightInEar
- leftArmpaw
- rightArmpaw
- leftLegpaw
- rightLegpaw
- groundPlane

7 Right-click the ExtraParts layer in the Layer Editor and choose Add Selected Objects from the drop-down list that appears.

The selected objects are assigned to the ExtraParts display layer which is currently set to Reference. You can still view these objects but they cannot be selected again until you set the display setting for this layer to Normal. Using reference layers in this fashion aids your workflow by ensuring you don't accidentally select unwanted surfaces when applying Fur.

Assigning a fur description preset to a model

A fur description defines all of the attributes for the fur, for example, fur color, width, length, baldness, opacity, curl, density, and so on. Maya includes a set of fur description presets you can use as is, or as a starting point in creating your own custom fur descriptions.

Any surfaces that should not have fur applied have already been assigned to a referenced display layer. This makes the application of a fur description easier.

To assign a fur description preset to the model

1 In the scene view, drag a selection marquee around the entire model.

All of the surfaces on the bear model are selected except for the ones you assigned to the referenced ExtraParts display layer.

2 In the Shelf, click the Fur tab.

3 With the surfaces still selected, click the Duckling fur preset in the Fur shelf.

The Duckling fur preset is assigned to the selected surfaces of the model.

4 In the Channel Box, rename the `Duckling` fur description to `TeddyBear`.

   Observe the yellow spikes that have appeared on the model. This is the *fur feedback* - a rough approximation of how the fur will appear when rendered.

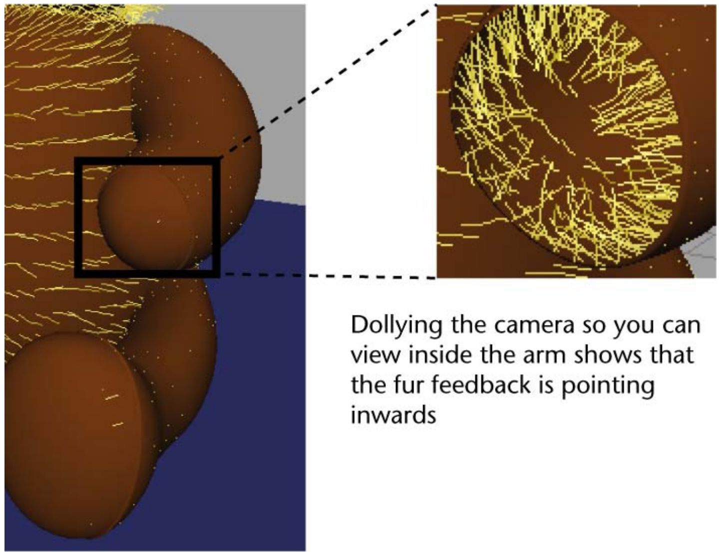
   Fur feedback shows various fur attributes such as color, density, length, direction, and scraggle. By looking at the fur feedback you can immediately see how the changes you make to the fur attributes affect the appearance of the fur, without having to render the scene.

   The fur feedback on the legs, arms, and top of the head of the model appears different in comparison to how it appears on the body and snout surfaces. It should appear consistently on all of the assigned surfaces, which indicates that the surface normals may not be oriented consistently for all of the assigned surfaces.

Reversing surface normals

When you apply a fur description to a surface, each hair making up the fur description points in a direction *normal* (perpendicular) to the surface by default. This direction is called the *surface normal*. When the fur does not display consistently on a model it may indicate that the surface normals are oriented inconsistently.

1 Dolly the camera so you can view inside the arm.



The fur feedback is pointing towards the inside of the arm.

In this model, the fur was applied to the inside of the leg, arm, and head surfaces because the surface normals on these surfaces are pointing in the wrong direction. There are several reasons why this can occur, including when a surface is duplicated with a negative scale value.

To correct this, you'll need to reverse the surface normals on the legs, arms, and head surfaces of the model.

NOTE Reversing the *surface normals* (Polygons menu set: Normals > Reverse) is useful when you want to change the surface direction (that is, whether the surface is oriented inwards or outwards in relation to the model). For example, duplicating a surface across its axis of symmetry can reverse the surface normals on the duplicated surface. Using the Reverse Normals feature corrects this.

When a character has existing animation rigging or texturing and you need to assign fur, the desired technique for reversing the orientation of the fur feedback is by reversing the *fur normals* (Fur > Reverse Fur Normals). Reversing the fur normals leaves the surface direction unchanged.

To reverse the surface normals

1. With only the legs, arms, and top head surfaces selected, switch to the Polygons menu set and then select Normals > Reverse > ❐ .
2. Set the *Reverse normals on* setting to *Selected faces* and then click Reverse Normals.

   The normals of the selected polygon surfaces are reversed so that the fur description points in the correct direction.

Before reversing surface normals

After reversing surface normals

Before proceeding to the next section, return to the Rendering menu set.

Modifying the fur direction

Real fur usually doesn’t stick straight out of a surface but appears angled in a particular direction. The teddy bear’s fur appears to point in different directions depending on the particular surface.

In this section, you modify the direction of the fur so it points in a consistent fashion on the model. You’ll also modify the direction of the fur to simulate a seam along the center of the bear’s forehead like a real stuffed toy might have.

There are three methods for modifying the direction of fur. Each method has specific conditions for use. You can modify the direction of fur by:

- Modifying the Inclination, Polar, and Roll attributes for the fur description globally.
- Setting a Fur Direction Offset to change the fur feedback direction locally on a selected surface.
- Combing the fur feedback locally on a surface using Maya's Paint Fur Attributes tool.

Modifying the Inclination, Polar, and Roll attributes for the assigned fur description changes the direction attributes on the fur description globally. You can set these attributes using either the Channel Box or the Attribute Editor. Inclination sets the angle or slope at which the fur description will lean in relation to the surface. Polar sets the direction the fur description will point in relation to the surface normal, much like the hands on a clock face. Roll sets the angle about which the fur description will be rotated about its follicle and is used mostly when the fur has curl attributes applied. That is, you can control whether the curls point up, down, or lay flat on their side.

Modifying these attributes is a good starting point for creating a realistic fur description. For the original Duckling fur description preset the Inclination, Polar, and Roll attributes are set to 0.6, 0.5, and 0.5 respectively. You won't need to change these for this lesson.

However, when a model is comprised of many surfaces, the fur may have a tendency to point in a different direction on each separate surface depending on how the surfaces were originally constructed.

To correct this you can change the direction of the fur using the Fur Direction Offset. The Fur Direction Offset lets you change the orientation of a fur description locally on a per-surface basis by applying a relative rotation value to the fur description as it is applied to the selected surface.

To modify the fur direction using the Fur Direction Offset

1. Dolly the camera closer to the bear's head so you can more closely view the direction of the fur description on the bear's snout.

   The fur appears to point in a clockwise direction about the bear's snout. You can change the direction the fur points by changing the Fur Direction Offset value for the bear's snout.

2. Select the snout surface.

3 Select Fur > Offset Fur Direction and click on the double lines at the top of the menu that appears so you can tear off the Offset Fur Direction menu.

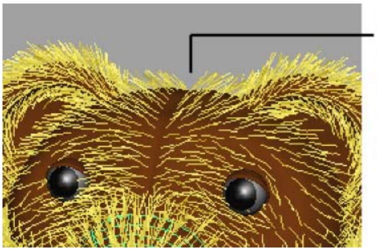
The Offset Fur Direction menu lets you rotate a fur description on a selected surface using direction presets.

4 Click on the 0, 90, 180, and 270 degree presets one at a time until the fur feedback on the snout points toward the bear's face.

5 Repeat this process for the remaining surfaces that have fur assigned by selecting each surface one at a time and orienting the direction of the fur feedback as follows:

- The fur feedback on the arms and legs should point in a direction towards the paws.
- The fur feedback on the body surfaces should point downwards.
- The fur feedback on the head surfaces should point away from the center line so it simulates a sewn seam.
- The fur feedback on the outer ear surfaces should point away from the bears head.

TIP Dolly and tumble the camera as you work to ensure you have the fur feedback oriented correctly. Try each offset direction so you know its effect on the fur feedback on each surface. You can easily set the direction offset back to its previous setting if it appears incorrect.

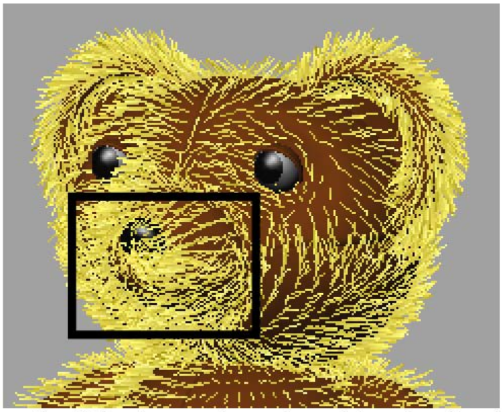


Set the Fur Direction Offset so the fur appears to part along the center line of the head.

6 Close the Offset Fur Direction menu when you are finished orienting the fur feedback for all of the surfaces that have fur assigned.

Painting fur attributes

The fur feedback on the bear's snout appears too long in the region of the bear's mouth and nose. The fur will poke through the nose and hide the mouth in the final rendered image unless you shorten the fur in that region on the snout. You can reduce the length of fur in a localized area of a surface using the *Paint Fur Attributes Tool*.



The length of the fur can be reduced in a localized area of the bear's snout using the Paint Fur Attributes Tool.

Using the Paint Fur Attributes Tool you can paint virtually any fur attribute. In this lesson you use the Paint Fur Attributes Tool to:

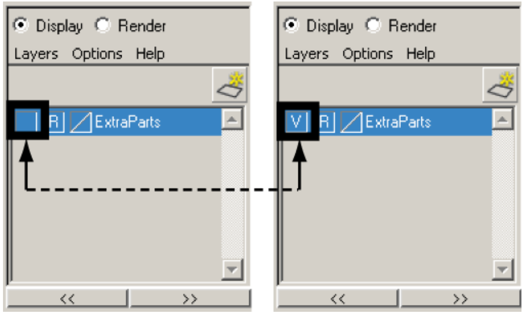
- reduce the fur length in a localized area on a surface. This is useful when you want regions with shorter or longer fur on a surface.
- create bald regions on a surface. This is useful for creating localized fur effects such as a moustache or goatee on a character, or whiskers on an animal.

The Paint Fur Attributes Tool does all of the above via attribute maps that you create by painting directly on the surface. The attribute map modifies the associated fur attributes you have selected for modification based on the regions of black, white, or gray that you paint in the map.

Before you begin painting an attribute map you'll want to hide the nose and mouth surfaces. Earlier in the lesson you assigned the nose and mouth surfaces to the ExtraParts reference layer to prevent them from being accidentally selected in the scene view. To hide these objects you can set the referenced display layer to invisible.

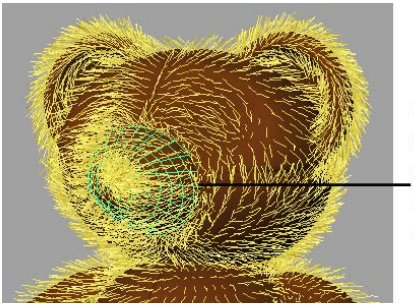
To hide the referenced display layer

1 In the Display Layer Editor, make the ExtraParts layer invisible by clicking the V that indicates that the layer is visible.



To shorten the fur using the Paint Fur Attributes Tool

1 In the scene view, select the snout surface.

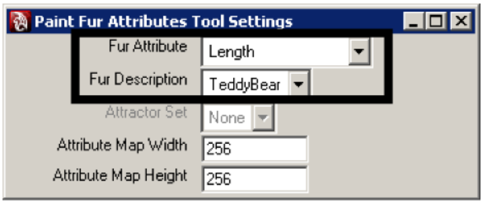


2 Select Fur > Paint Fur Attributes Tool > ❒ .

The cursor updates to display as a paint brush indicating that the Paint Fur Attribute Tool is active. The Paint Fur Attributes Tool Settings window and the Paint Tool Settings editor appears. These windows let you set and control which attributes get painted for an attribute map.

3 In the Paint Fur Attributes Tools Settings window, ensure the following options are set:

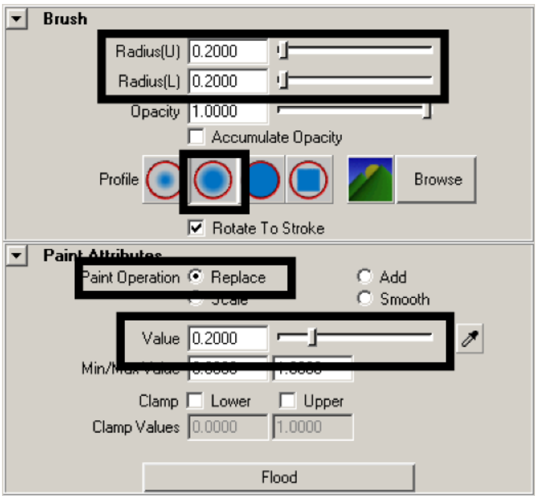
- Fur Attribute: Length
- Fur Description: TeddyBear



4 In the Paint Tool Settings editor, set the following:

- Brush - Radius (U): 0.2
- Brush - Radius (L): 0.2
- Brush - Profile: Soft
- Paint Attributes - Paint Operation: Replace
- Paint Attributes - Value: 0.2

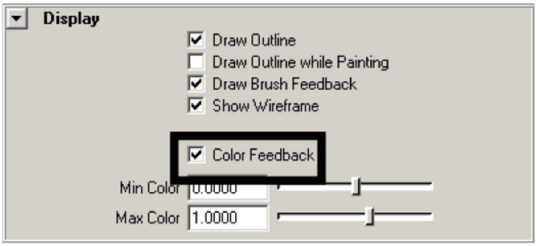
These settings set the brush to paint a relatively small, soft edged region, with each stroke of the tool as you replace the existing fur with shorter fur.



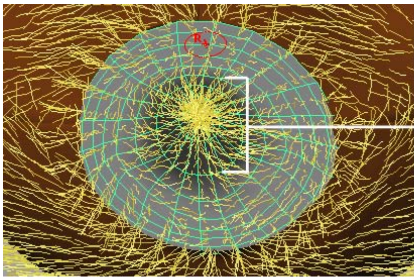
5 Position the brush cursor over the centre of the bear's snout and begin stroking in a small circular motion about the center of the snout.

Wherever you paint, the fur feedback updates and becomes shorter in length.

6 In the Display section of the Paint Tool Settings editor turn on the Color Feedback option.



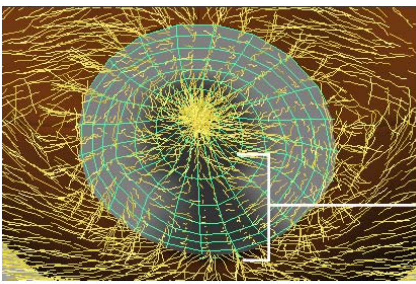
The snout surface updates to display the fur length attribute map you are currently painting. The black region at the end of the snout indicates the area you have just painted. This region of the attribute map indicates where the fur will be shorter in length when it is rendered. The lighter gray region indicates where you have not yet painted and where the fur length will remain unaffected.



Displaying the color feedback shows the attribute map you're painting when using the Paint Fur Attributes Tool.

You also want the fur to be short in the region of the bear's mouth as well.

7 Paint a few overlapping strokes on the region of the snout below the nose where the mouth surface is located, as shown below.



Painting a few strokes below the nose area will shorten the fur so the mouth surface is visible.

Notice that the attribute map has a gradual color falloff. This ensures that the fur transition between the original attribute values and the map does not appear abrupt. You can smooth the attribute map even further by applying a smooth operation.

8 In the Paint Tool Settings editor, set the Paint Operation to Smooth, and then click Flood to apply a smooth to the entire fur length attribute map. You can click Flood more than once to apply multiple smooth operations.

TIP If you accidentally paint an area that is too large you can click Ctrl + z repeatedly to undo your paint strokes.

If you want to begin painting the attribute map again, you can set the Value to 1 and then click Flood to fill the surface with a white color.

To create whiskers using the Paint Fur Attributes Tool

1 Ensure that the snout surface is still active and click the Porcupine fur preset from the Fur shelf.

Porcupine fur preset

A second fur description is assigned to the bear's snout. The snout appears cluttered at this point but you'll rectify it in the following steps.

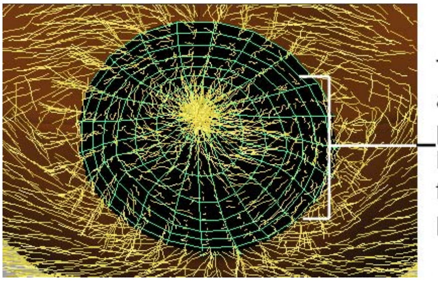
2 Ensure that the snout surface is still active and select Fur > Paint Fur Attributes Tool > ❐ .

3 In the Paint Fur Attributes Tools Settings window, set the following options:

- Fur Attribute: Baldness
- Fur Description: Porcupine

4 In the Paint Tool Settings editor, set the Paint Attributes - Paint Operation to Replace, Value to zero, and then click Flood.

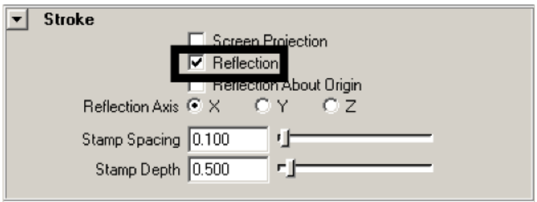
This creates a baldness attribute map for the Porcupine fur description that is black in color making the Porcupine fur feedback disappear temporarily.



The baldness attribute map prevents any of the Porcupine feedback from appearing at present.

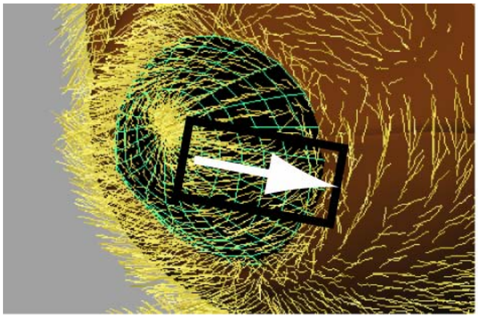
5 In the Paint Tool Settings editor, set the following:

- Brush - Radius (U): 0.2
- Brush - Radius (L): 0.2
- Brush - Profile: Soft
- Paint Attributes - Paint Operation: Replace
- Paint Attributes - Value: 0.2
- Stroke - Reflection: On



These settings set the brush to paint a relatively small, soft edged stroke, that is gray in color on the baldness map so that some of the porcupine fur (that is, whiskers) appear. When reflection is turned on, the stroke you make on one half of the surface is mirrored on the opposite half. This lets you create identical whiskers on either half of the snout.

6 Position the brush cursor over the left side of the bear's snout and make one short stroke from about the halfway point to the back of the snout, as shown below.



Draw one short stroke on the snout to create the whiskers.

The fur feedback updates immediately after your stroke to show the whiskers on both sides of the snout. Tumble the scene view if they are not immediately apparent.

The whiskers are not pointing outwards at the correct angle. To rectify this you need to modify the Polar attribute on the Porcupine fur description.

To edit the fur description

1 Select Fur > Edit Fur Description > Porcupine.

The Attribute Editor appears and displays the attributes for the Porcupine fur description.

To rename the fur description

1 In the Attribute Editor rename the fur description to `Whiskers`.

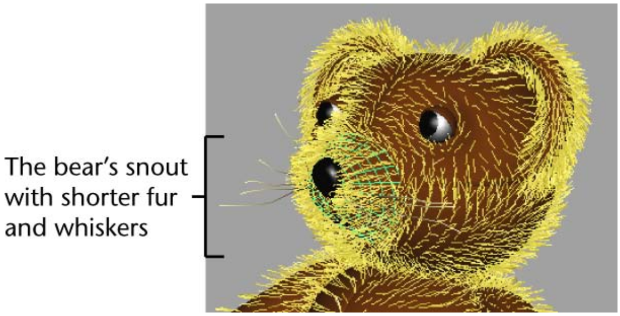
To modify the whiskers

1 In the Attribute Editor, set the following attributes to modify the whiskers on the snout surface:

- Length: 1.5
- Polar: 1.0

The whiskers update to point outwards on either side of the snout and appear shorter as a result.

2 Display the Channel Box/Display Layer Editor, and click the visibility box to display the ExtraParts layer again.



Modifying the color of a fur description

Next you'll change the color of the fur description so it resembles a real teddy bear. To do this you'll edit the fur description and set the Base and Tip Color attributes using the Attribute Editor.

To modify the fur description

1. Select Fur > Edit Fur Description > TeddyBear.

   The Attribute Editor appears and displays the attributes for the assigned TeddyBear fur description.

2. In the Attribute Editor, modify the Base and Tip Color attributes for the TeddyBear fur description by clicking the color swatches for each attribute, setting the RGB range to 0 to 255, and entering the following RGB values:

   - Base Color R: 62
   - Base Color G: 30
   - Base Color B: 8
   - Tip Color R: 106
   - Tip Color G: 52
   - Tip Color B: 14

   These values change the color of the base of the fur to a dark brown, and the tips of the fur a lighter brown. The fur feedback is updated on the bear model.

Creating a new fur description

To make the bear appear more interesting you'll assign a new fur description to the inside of the ears and then modify fur attributes such as density, scraggle, and inclination.

Before you can assign a fur description to the inner ear surfaces you'll need to select them by first turning off the referenced display layer.

To turn off the referenced display layer

1 Display the Channel Box/Display Layer Editor.

2 In the Display Layer Editor, click the box that displays the letter R until the box is empty.

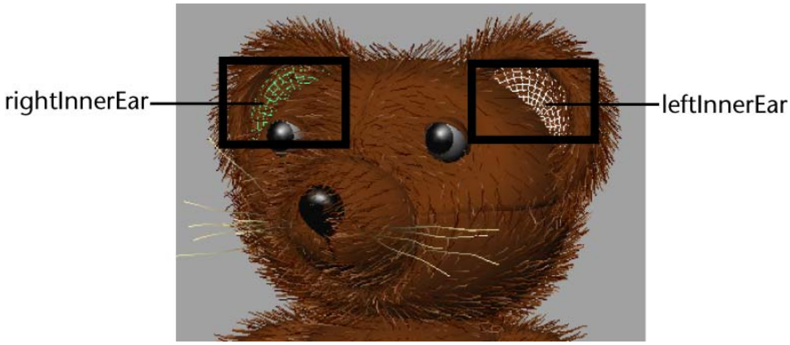
   To change the display state of a display layer, click repeatedly in the middle box next to the layer name to cycle the display state from Reference (R), Template (T), or Normal (empty). When the box is empty the display layer is no longer referenced and its possible to select all of the surfaces on the model.

To create a new fur description

1 In the scene view, select the following surfaces of the model:

   ■ leftInnerEar

- rightInnerEar



2 Select Fur > Attach Fur Description > New.

   A new default fur description is assigned to the selected surfaces.

To rename the fur description

1 Select Fur > Edit Fur Description > Fur Description1.

   The Attribute Editor appears and displays the attributes for the new fur description.

2 In the Attribute Editor rename the fur description to `InnerEar`.

To modify the color of the fur description

1 In the Attribute Editor, modify the Base and Tip Color attributes for the InnerEar fur description by clicking the color swatches for each attribute, setting the RGB range to 0 to 255, and entering the following RGB values:

   - Base Color R: 106
   - Base Color G: 52
   - Base Color B: 14
   - Tip Color R: 164
   - Tip Color G: 119
   - Tip Color B: 91

2 Modify these additional InnerEar attributes by entering the following values:

- Density: 50,000
- Length: 0.25
- Inclination: 0.8
- Base Width: 0.2
- Tip Width: 0
- Scraggle: 0.1

Density specifies the number of hairs on a surface. The higher the value, the thicker the fur. Notice that changing the Density value does not affect the fur feedback in the scene view; changes to fur density only appear in the rendered image.

Length sets the fur's length, in grid units.

Inclination sets how the fur slopes or leans. A value of 0 is fully erect (normal to the surface), while a value of 1 is flat (tangent to the surface at the root).

Base and *Tip Width* specifies width of the hairs at their base and tip.

Scraggle sets the crookedness of the fur. A value of 0 creates no crookedness, while a value of 1 creates maximum crookedness.

The fur description for the Inner Ear surfaces is updated.

NOTE Some attributes of a fur description are only visible when you render an image of the fur. You'll render an image of the teddy bear fur in the next lesson.

Beyond the lesson

The Teddy Bear now has brown fur and whiskers. The fur in the bear's ears is a lighter brown color.

In this lesson you learned how to:

- Duplicate surfaces across the axis of symmetry—you duplicated the arm and body surfaces and created a mirror-copy of them.

- Rename surfaces—Naming objects as you create them ensures you can easily identify and select them later on.
- Assign parts of a model to display layers—any surfaces that you did not want fur applied were assigned to a referenced display layer so they would not get selected accidentally.
- Assign an existing fur description preset to a model—Fur presets are a useful starting point for creating various fur types. All of the fur presets included with Maya can be customized. You also created a new fur description for the inner ear surfaces and customized its attributes.
  You can assign more than one fur description to a surface. In this lesson, you added a second fur description to the bear's snout in order to create whiskers.
- Reverse surface normals so the fur was oriented to point in the correct direction—Two methods are possible for correcting the normals; reversing the surface normals, and reversing the fur normals.
- Modify the direction of the fur description on individual surfaces using the Fur Description Offset—when a model is comprised of multiple surfaces the fur description may point in different directions. You can modify the fur direction using a variety of techniques depending on the situation.
- You modified the length of the fur description on the bear's snout and created whiskers using the Paint Fur Attributes Tool. This tool lets you paint an attribute map to modify specific fur attributes on an area of a surface.
  You can use the Paint Fur Attributes tool to paint the direction of the fur. This technique is also referred to as *combing* the fur because it resembles the stroking action you perform when combing real hair. Combing the fur feedback is useful when you need to hide a visible seam between two surfaces.
- Modify fur attributes—You learned how to modify the attributes of a fur description using the Attribute Editor.
  Though not covered in this lesson, you can keyframe changes you make to fur attributes to animate effects such as growing fur or changing fur color. You can add movement to fur with attractors and dynamics. Using attractors, you manually keyframe fur movement. Using dynamics, you can make the fur react to forces, for example, wind and gravity. You can also use dynamics to cause fur to react to movement of the attached surface, for example, when a dog shakes itself.
- UV texture coordinates—When applying fur to a surface the UVs must be laid out so they are non-overlapping and reside between 0 and 1 in the

UV texture space. The UVs were prepared for the model in this lesson but you should remember this when you assign Fur to your own models.

For more information and related techniques about Fur, refer to the Maya Help.

Lesson 2: Rendering fur

Introduction

In order to see how all of the fur attributes of a fur description appear on a model you must render an image.

In this lesson, you learn how to:

- Set up lights in a scene.
- Assign fur shadow and shading attributes to the lights.
- Render an image of the bear.
- Make changes to the fur description and render the scene again.

Lesson setup

To ensure the lesson works as described, complete the steps in lesson 1, or do the following before beginning:

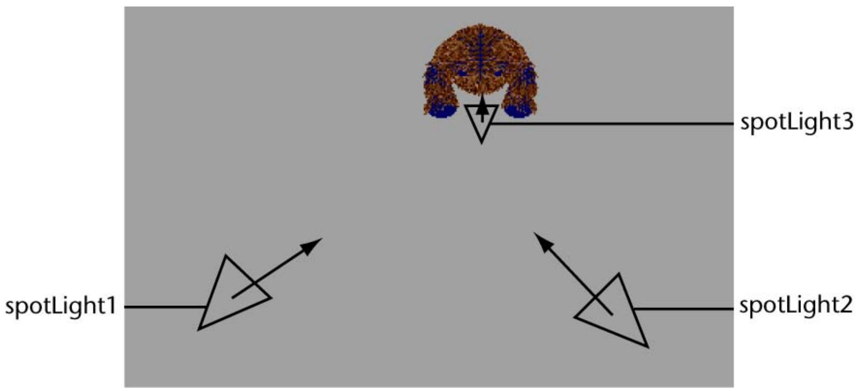
1 Open the scene named `FurRender.ma` that is located in the `GettingStarted/Fur/scenes` directory of your Maya project.

This scene includes a model that has been created for you. The model has three fur descriptions assigned. You will add lights to the scene, render an image, make some final adjustments, and render again.

Creating lights in a scene

You must add lights to your scene to illuminate the model before you render it. You will create and position three spotlights to highlight the fur properties on the bear model (see the plan view below). Spotlights are used in this lesson because they can cast shadows.

The first spotlight will be positioned to the right of the model. The second spotlight will be positioned to the left of the model. The third spotlight will be positioned so it shines down onto the model from above.



To add lights to the scene

1 Click Create > Lights > Spot Light > ❐ .

The Create Spot Light Options window appears.

2 Click Edit > Reset Settings, set the following attributes, and then click Create:

- Intensity: 1.5

A new spotlight is created at the origin, named spotLight1. By default, it is selected in the scene view. Check the Channel Box or the Attribute Editor to make sure the spotlight is selected.

3 In the Channel Box, enter the following values to reposition the spotlight:

- Translate X: -39
- Translate Y: 12
- Translate Z: 34
- Rotate X: -5
- Rotate Y: -49

4 Click Create > Lights > Spot Light to create another spotlight at the origin.

This new spotlight will be named spotLight2 and have the same intensity setting as the first spotlight.

5 In the Channel Box, enter the following values to reposition the second spotlight:

- Translate X: 25
- Translate Y: 15
- Translate Z: 35
- Rotate X: -10
- Rotate Y: 35

6 Click Create > Lights > Spot Light > ❐ , set the following attribute, and then click Create:

- Intensity: 0.5

A third spotlight is created at the origin, called spotLight3.

7 In the Channel Box, enter the following values to reposition the third spotlight:

- Translate Y: 85
- Translate Z: 7

■ Rotate X: -88

These three light sources will evenly illuminate the model: one will create a shadow once you have set fur shadowing attributes for the spotlights.

Adding fur shadowing attributes to lights

If the scene were rendered at this point, the fur would not appear realistic because it would not contain shadows. To correct this you must turn on fur shadowing attributes for the spotlights.

The scene rendered without fur shadowing attributes defined.

The lack of fur shadows makes the fur appear flat and unrealistic.

NOTE In this lesson, we will render with the Maya software renderer. However, the following instructions for fur shadowing can also be applied to lights set up to render Fur with the mental ray for Maya renderer.

To add fur shadowing attributes to spotlights

1 Select all three spotlights in the scene by selecting Edit > Select All by Type > Lights.

All three spotlights in the scene are selected.

TIP When more than one item is selected in the scene view, the Channel Box displays the name of the last item in the selection list with three dots after its name to indicate that more than one item is selected.

---

2 Click Fur > Fur Shadowing Attributes > Add to Selected Light.

This adds adjustable fur shadowing attributes to all three spotlights in the scene that were not included with the default spotlights. These extra attributes let you refine how the lights affect the fur on the model when rendering.

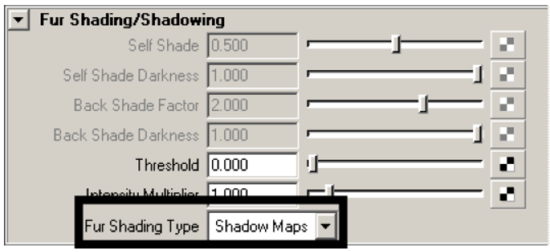
3 Select Window > Outliner to display the Outliner.

As your scene gets increasingly complex the Outliner becomes useful for selecting specific objects in the scene, especially when you want to select objects that are hidden by other objects, or are outside the camera's field of view. Selecting items in the Outliner is much easier when you have named the objects and lights in your scene.

4 In the Outliner, click spotLight1 to select it in the scene.

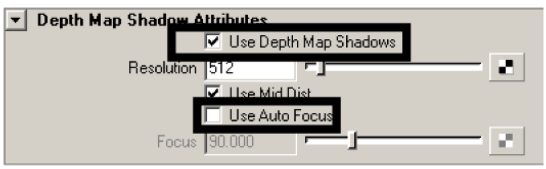
5 Open the Attribute Editor by clicking the Show/Hide Attribute Editor icon on the Status Line.

6 Select the spotLightShape tab, and under Fur Shadowing/Shadowing, set the Fur Shading Type to Shadow Maps.



7 In the Attribute Editor, under Shadows > Depth Map Shadow Attributes, set the following options:

- Use Depth Map Shadows: On
- Use Auto Focus: Off
- Focus: 40



Depth Map Shadows are one technique for producing shadows in a scene by determining which objects in the scene are in the path of a shadow casting light.

Fur Shadows can be incorrectly placed when Auto Focus is turned on. To avoid this, turn off Auto Focus for all spotlights that have Fur Shadow Maps turned on and instead set the Focus value to Cone Angle + (Penumbra * 2).

8 Shift-select spotlight2 and spotlight3 using the Outliner.

9 In the Attribute Editor, under Fur Shadowing/Shadowing, set the Fur Shading Type to Auto Shading.

   The Autoshade attribute provides a quick simulation of shading that occurs at the root and back of the fur strands without the added complexity of a shadow map.

Rendering the scene

To see the fur in full detail, you need to render an image of the scene.

To render the scene

1 Select Window > Rendering Editors > Render Settings to show the Render Settings window.

2 Ensure that Render Using is set to Maya Software in the drop-down list.

3 From the Render Settings window menu, select Presets > Load Preset > Default Settings.

   This sets the renderer to render an image that is 640 by 480 pixels in size; large enough to see the fine detail on the fur.

4 Close the Render Settings window.

5 To render the fur, do one of the following:

   - Select Render > Render Current Frame.
   - Click the Render Current Frame button in the Status Line.


The Render Current Frame button

The Render View window appears and an image of the scene is rendered.

At this point, you can modify any of the light attributes, or fur attributes as required and render the scene again.

Beyond the lesson

In this lesson you learned how to:

- Add spotlights to the scene—A three light setup is a good starting point for many renderings.
- Add spotlight fur shadowing attributes to lights—The fur shadowing attributes must be turned on for the lights to produce shadows on fur.
- Render the scene to produce an image with fur—You will usually need to produce multiple iterations of an image in order to produce the final image or frames of animation that you require.

For more information and related techniques about Fur, Lighting, or Rendering, refer to the Maya Help.

nCloth

5

Introduction

With the Maya® nCloth™ feature, you can create dynamic cloth effects in Maya. nCloth is a fast and stable dynamic cloth solution that uses a system of linked particles to simulate a wide variety of dynamic polygon surfaces, such as fabric clothing, inflating balloons, shattering surfaces, and deformable objects. nCloth is generated from modeled polygon meshes. You can model any type of polygon mesh and make it an nCloth object, which is ideal for achieving specific poses and maintaining directorial control.

nCloth is built on a dynamic simulation framework called Maya® Nucleus™. A Maya Nucleus system is composed of a series of nCloth objects, passive collision objects, dynamic constraints, and a Maya Nucleus solver. As part of the Maya Nucleus system, the Maya Nucleus solver calculates nCloth simulation, collisions, and constraints in an iterative manner, improving the simulation after each iteration, to produce accurate cloth behavior.

Maya Unlimited provides many predefined nCloths. These sample nCloth objects, also referred to as *presets*, can be used as a starting point in creating your own custom cloth effects.

This chapter includes these lessons:

Preparing for the lessons

To ensure the lessons work as described, do these steps before beginning:

1. Make sure you understand the basic concepts of polygon modeling, animation, and dynamics.
2. If you have not already done so, copy the `GettingStarted` folder from its installation location to your `projects` directory. Then, set the `GettingStarted` directory as your Maya project. For more information, see Copying and setting the Maya project in the Getting Started with Maya guide.
3. Select the nDynamics menu set. Unless otherwise noted, the directions in this chapter for making menu selections assume you've already selected the nDynamics menu set.

Lesson 1: Creating nCloth collisions

Introduction

For nCloth objects to have realistic cloth behavior, they need to interact or collide with their environment. This is accomplished by converting the meshes you want to act like cloth into nCloth objects, and turning all the objects in their environment that it will come into contact with into passive collision objects.

In this lesson you learn how to:

- Convert a polygon object to nCloth.
- Create passive collision objects for your nCloth.
- Improve the quality of your nCloth collisions.

Lesson setup

To ensure the lesson works as described, do these steps before beginning:

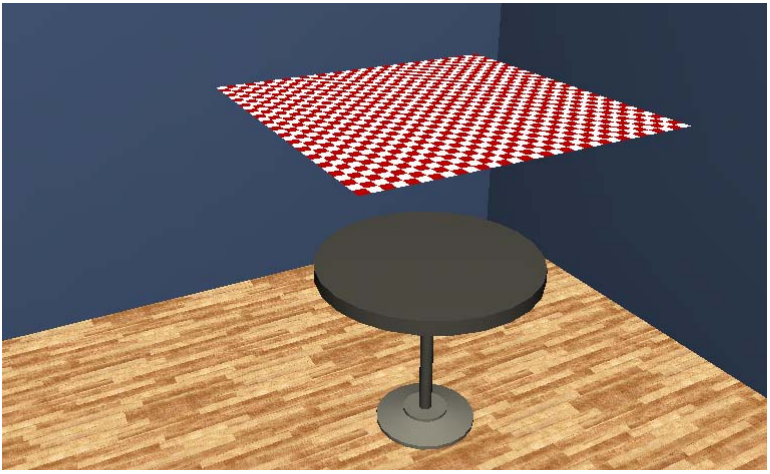
1 Open the scene file named `TableCloth.ma`.

This file can be found in the `GettingStarted` directory that you set as your Maya project:

`GettingStarted/nCloth/TableCloth.ma`

This scene includes textured models that have been created for you. The table cloth, table, table base, floor, and wall models provided are all polygon meshes.

Creating an nCloth object

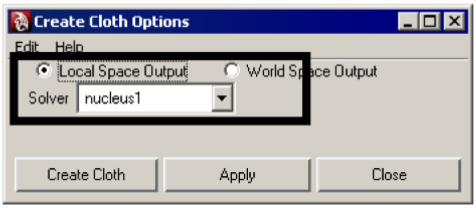


The first step in the nCloth creation process is to convert the polygon object you want to behave like cloth to an nCloth object.

To make a polygon mesh nCloth

1 Select the checkered table cloth.

2 Select nMesh > Create nCloth > ❐ .
The Create Cloth Options window appears.



Lists all the Maya Nucleus solvers in your scene, or allows you to create a new solver for the nCloth you are creating.

3 Select nucleus1 from the Solver drop-down list.

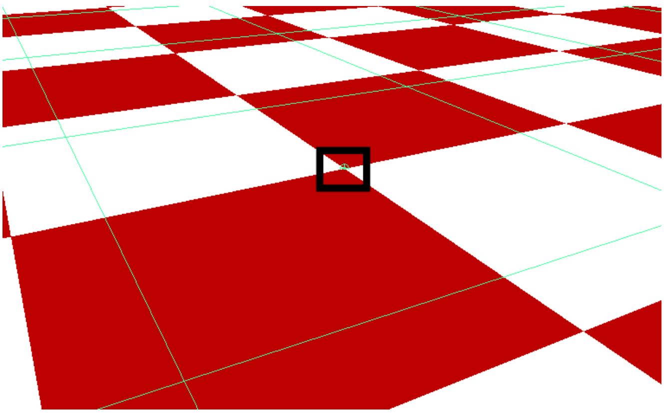
Selecting a solver determines which Maya Nucleus system the nCloth belongs to.

4 Turn on Local Space Output.

Defining a space output determines whether the table cloth's input and output nCloth meshes are both (Local Space Output) attached to the selected Maya Nucleus solver, or just the output nCloth mesh (World Space).

5 Click Create Cloth.

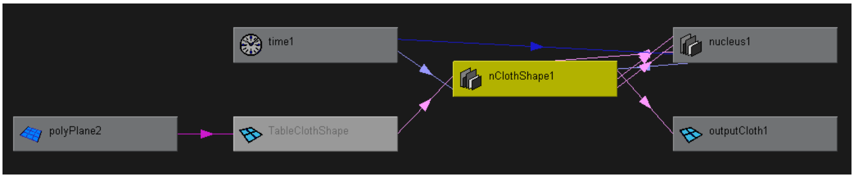
The static quad polygon table cloth is automatically tessellated and converted to a dynamic nCloth object, and an nCloth handle appears on your table cloth in the scene view. The nCloth handle resembles a small wireframe sphere, and it is located at the center of table cloth. You can use the table cloth's nCloth handle to quickly select its nClothShape node in the Hypergraph or its tab in the Attribute Editor.



The nCloth handle resembles a small wireframe sphere, and it is located at the center of table cloth.

6 Select Window > Hypergraph: Connections to view the new node connections for the table cloth in the Dependency Graph (DG).

The Hypergraph appears with a new Maya Nucleus network of nodes. Note that the table cloth is now a member of the nucleus1 Maya Nucleus system.



nClothShape1 is the nCloth properties node, which carries all the nCloth attributes for the table cloth.

nucleus1 is the Maya Nucleus solver node, which carries all the attributes affecting the nucleus1 solver system, including internal forces.

TableClothShape is the input mesh and start object for the table cloth.

outputCloth1 is the output mesh or current mesh, and the resulting nCloth table cloth that you see in the scene view.

polyPlane2 is the history node for the table cloth.

7 Close the Hypergraph and play back your table cloth's simulation.

The table cloth passes right through the table, its base, and the floor. This is because the nCloth does not recognize any of the other objects in your scene. For the nCloth table cloth to be able to interact with the table, its base, and the floor, they need to be members of the same Maya Nucleus solver system as the table cloth.

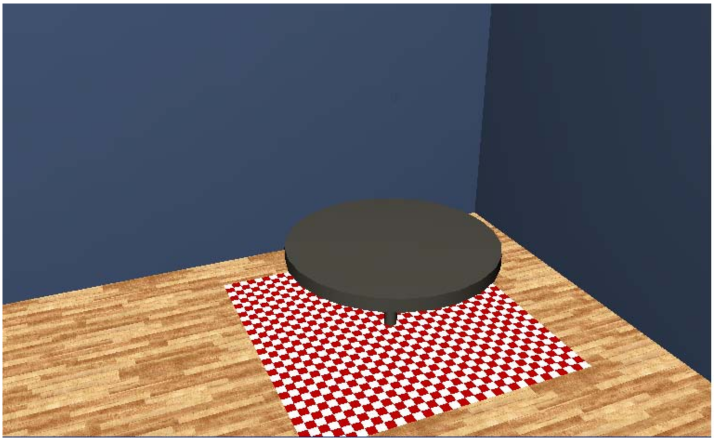


Table cloth's simulation without collisions.

8 Go to the beginning of the playback range.

Making an nCloth collide with its environment

The second step in creating realistic cloth effects is to make your nCloth object interact with its environment, and have the objects and surfaces in its scene deform the nCloth and drive its movements. You can do this by making each surface the nCloth comes into contact with during its simulation *passive collision objects*.

To create passive collision objects for your nCloth

1 Select the table.

2 Select nMesh > Create Passive Collider > ❐ .
The Make Collide Options window appears.

3 Select nucleus1 from the Solver drop-down list and click Make Collide.

The table polygon mesh is converted to a passive collision object, and an nRigid handle appears at the center of its mesh in the scene view. You can use the table's nRigid handle to quickly select its nRigidShape node in the Hypergraph or its tab in the Attribute Editor.

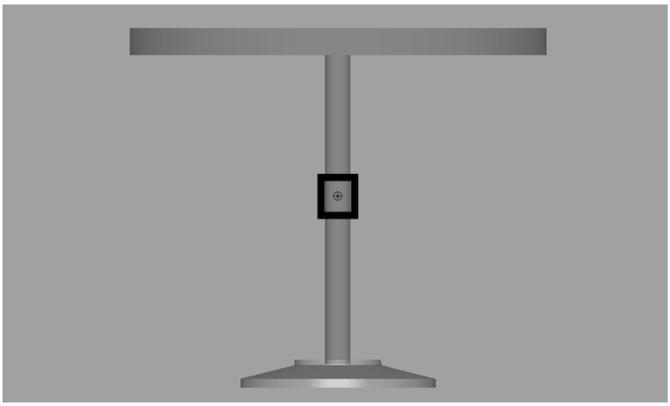
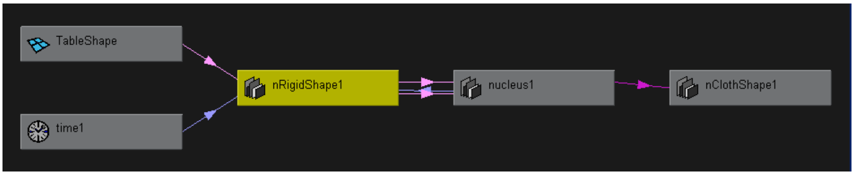


Table as viewed in X-Ray Shading mode.

Like the nCloth handle, the nRigid handle resembles a small wireframe sphere, and it is located at the center of table.

The table is also now a member of the nucleus1 Maya Nucleus system.

4 Select Window > Hypergraph: Connections to view the new node connections for the table in the Dependency Graph (DG).



nRigidShape1 is the passive object properties node which carries all the passive object attributes for the table.

5 Close the Hypergraph and play back your table cloth's simulation.

The table cloth now collides with the table. The table cloth now interacts with the table because the table is a member of the table cloth's nucleus1 solver system.

NOTE You may need to increase the total number of frames on your time line for the table cloth to collide with the table.

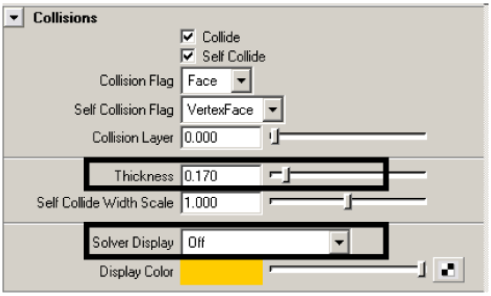
Table cloth's simulation with passive object collisions.

Adjusting the accuracy of nCloth collisions

The third step in creating believable cloth effects is to make your nCloth object's collisions as accurate as possible. In reality, a falling piece of cloth comes into close contact with the object it lands on and it deforms, reflecting the shape of the object it has come into contact with. Currently, the table cloth does not exhibit this behavior.

To set the thickness of your nCloth and passive objects

1. Select the table cloth.
2. Open the Attribute Editor and select the nClothShape1 tab.
3. In the Collisions section, select Collision Thickness from the Solver Display drop-down list.



The table cloth's collision thickness appears in the scene view.

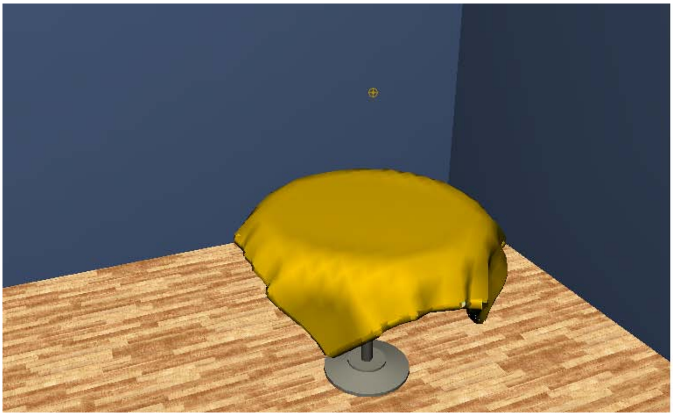


Table cloth Solver Display: Collision Thickness

An nCloth's Thickness attribute determines the radius or depth of the table cloth's *collision volume*. The collision volume is a non-renderable surface offset from the table cloth's surface that the nucleus1 solver uses when calculating the table cloth's passive object collisions. Collisions occur at the table cloth's collision volume, not at the surface of the table cloth itself.

Table cloth and table in X-Ray Shading mode.

The table cloth's Thickness is so high that its actual surface does not come close to the table. Only the table cloth's collision volume comes close to the table.

4 Go to the beginning of the playback range.

5 In the Collisions section, change the table cloth's Thickness attribute value to 0.066.

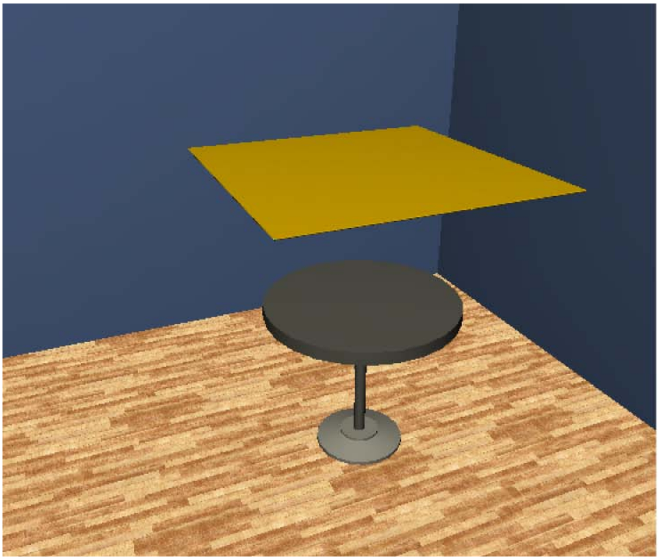


Table cloth Solver Display: reduced Collision Thickness

The table cloth's collision volume is noticeably reduced, so that it will now behave like a thin piece of fabric rather than a thick surface. This new Thickness will contribute to improving the accuracy of the table cloth's collisions when it falls onto the table.

6 Select Off from the Solver Display drop-down list.

The table cloth's Collision Thickness is no longer displayed in the scene view.

7 Play back the table cloth's simulation.

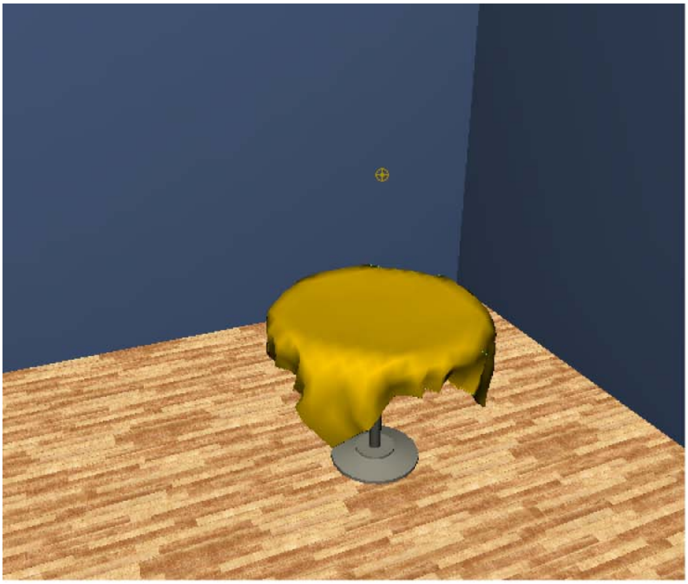


Table cloth Solver Display: Collision Thickness

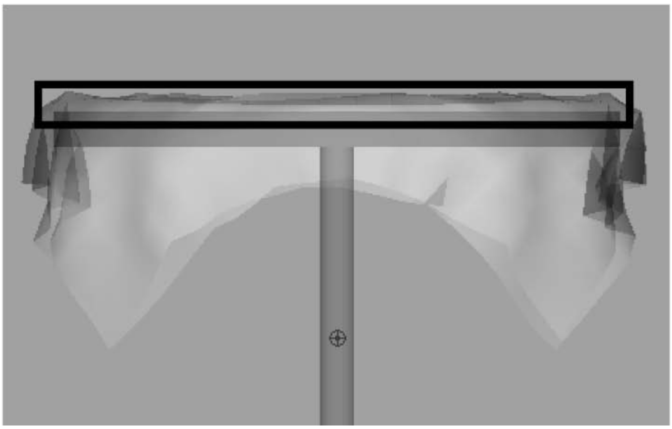


Table cloth and table as viewed in X-Ray Shaded mode.

Table cloth's improved collisions.

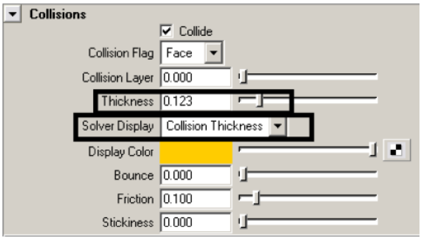
Even though you adjusted the table cloth's Thickness, the table cloth still does not appear to be making contact with the table's surface. This is because passive collision objects, like the table, also possess collision volumes.

To further improve the accuracy of the table cloth's collisions, you need to adjust the table's Thickness.

8 Go to the beginning of the playback range.

9 In the scene view select the table, and in the Attribute Editor select its nRigidShape1 tab.

10 In the Collisions section, select Collision Thickness from the Solver Display drop-down list.



The table's collision thickness appears in the scene view.

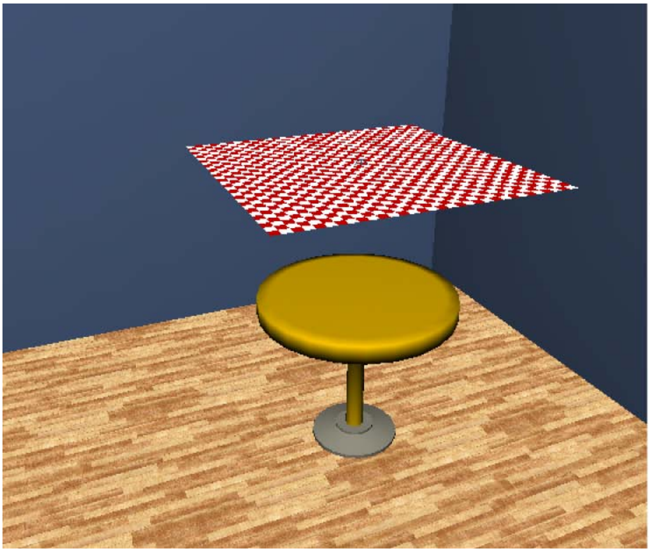


Table Solver Display: Collision Thickness

11 Set the table's Thickness attribute value to 0.066.

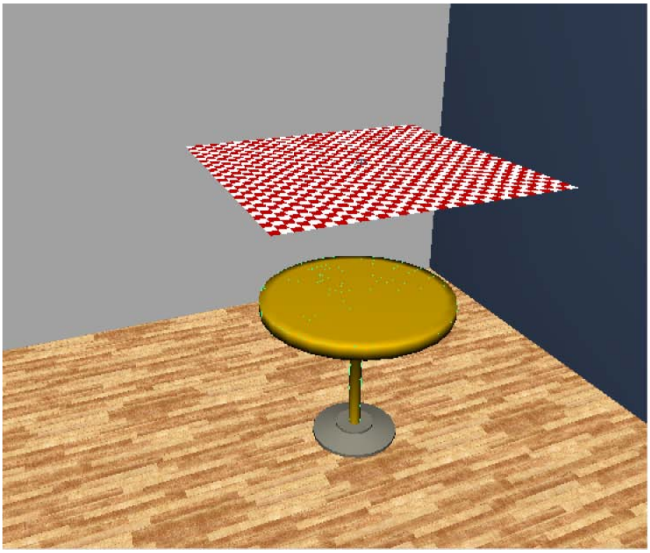


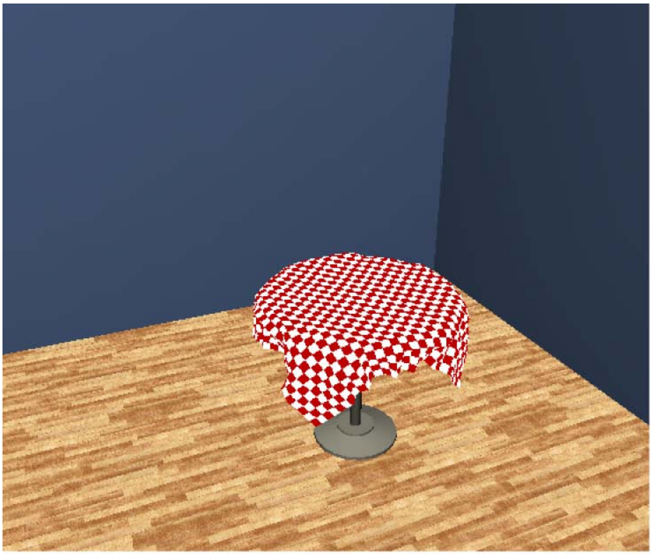
Table Solver Display: reduced Collision Thickness

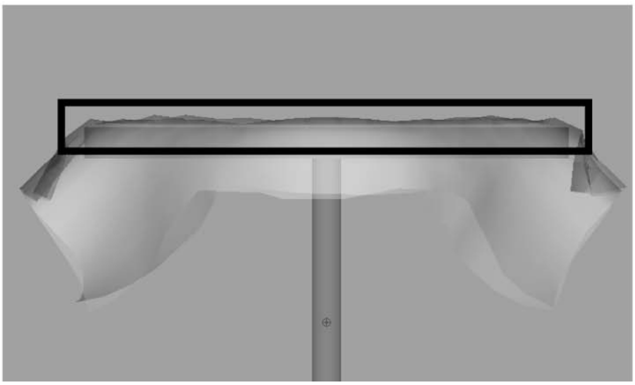
The table's collision volume is noticeably reduced, so that the table cloth will now conform closely to the actual shape of the table when you play back its simulation.

12 In the Collisions section, select Off from the Solver Display drop-down list.

The table's Collision Thickness is no longer displayed in the scene view.

13 Play back the table cloth's simulation.





Adjusting the Thickness of the table cloth nCloth and the table passive object dramatically improved the accuracy of the table cloth's collisions.

14 Go to the start of the playback range.

The way Collision Thickness appears in the scene view is determined by your nCloth's current *Collision Flag* selection. The Collision Flag drop-down list lets you specify which of the table cloth's components participate in its collisions and which type of collision volume is used by the table cloth. You can adjust the speed of your nCloth collisions with the Collision Flag at the cost of collision accuracy.

To set which nCloth components participate in collisions

1 Select the table cloth.

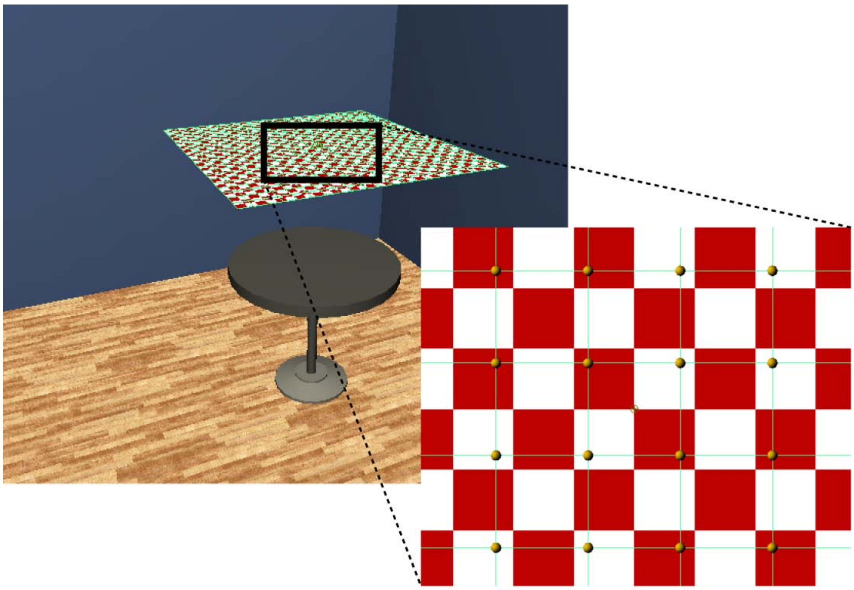
2 In the Attribute Editor, select the nClothShape1 tab.

3 In the Collisions section, select Collision Thickness from the Solver Display drop-down list.

4 By default, the Collision Flag is set to Face.

Face provides the best and most accurate collisions, but it is the slowest to calculate the collisions for.

5 Change the Collision Flag selection to Vertex. Vertex produces the least accurate collisions, but it is the fastest to calculate the collisions for.



The table cloth's collision volume changes from a surface to multiple collision spheres.

6 Play back the table cloth's simulation.

The table cloth passes through the table and floor collision objects, as if they were not passive objects. This is because when you select Vertex as your component type, only the small collision spheres surrounding the table cloth's vertices participate in the table cloth's collisions, rather than the entire surface as with Face.

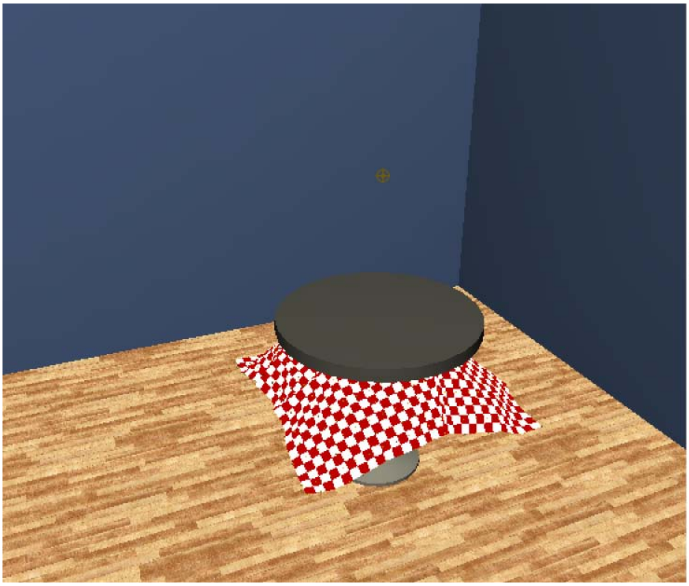


Table cloth's simulation with Vertex Collision Flag selected.

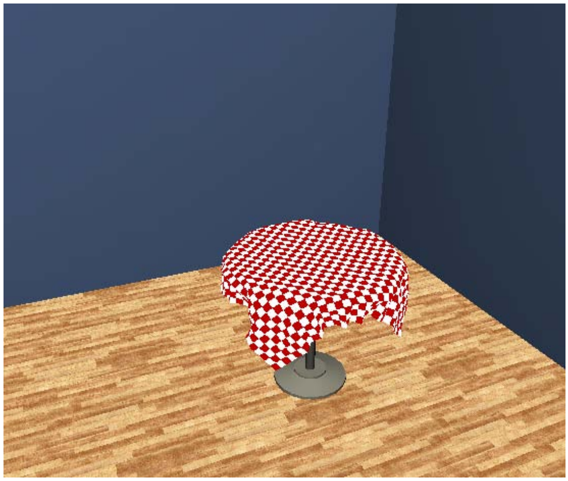
Vertex and face are useful when you want to speed up your simulation by reducing the number of collision calculations, or your cloth simulation does not require the maximum level of collision accuracy (for example, for a distance shot of colliding nCloth objects).

7 Reset the Collision Flag to the default Face selection, and select Off from the Solver Display drop-down list.

8 Go to the beginning of the playback range.

9 Play back the table cloth's simulation.

The table cloth once again easily and accurately collides with the table.



Beyond the lesson

The table cloth now is a dynamic nCloth object that behaves like real cloth as it collides with the table, floor, and wall in its scene.

In this lesson you learned how to:

- Turn a regular polygon mesh into a dynamic nCloth object.
- Run an nCloth simulation.
- Turn objects in an nCloth's scene into passive collision objects.
- Refine nCloth collisions by adjusting the Thickness and Collision Flag of the nCloth and passive objects in a scene.

For more information and related techniques about nCloth, refer to the Maya Help.

Lesson 2: Creating nCloth constraints

Introduction

Often times, to create certain cloth effects, you need to attach nCloth to other nCloths or passive collision objects and have those other Maya Nucleus objects restrict or drive your nCloth's movements. For example, you can use nCloth constraints to create buttons on nCloth garments, replace collisions, bind topologically different nCloth objects together, and exclude and limit nCloth and passive object collisions.

In this lesson you learn how to:

- Constrain an nCloth object.
- Manage an nCloth constraint's membership.
- Adjust nCloth constraint behavior.
- Make an nCloth flag flap in the wind.

Lesson setup

To ensure the lesson works as described, do these steps before beginning:

1 Open the scene file named `Flag.ma`.

This file can be found in the `GettingStarted` directory that you set as your Maya project:

`GettingStarted/nCloth/Flag.ma`

This scene includes dynamic meshes that have been created for you. The pirate flag and the flag pole provided are an nCloth object and a passive collision object respectively.

Constraining an nCloth to a passive object

You can constrain nCloth by creating dynamic links between its CVs and the components of other nCloths or passive collision objects. The stretchiness and compressibility of these collision links depends of the type of constraint they belong to.

To constrain an nCloth

1 Play back the flag's simulation.

The flag falls downward through the air. For the flag to retain its position beside the flag pole in XYZ space, it needs to be constrained to the static flag pole passive object.

NOTE You may need to increase the total number of frames in the timeline.

2 Go to the beginning of the playback range.

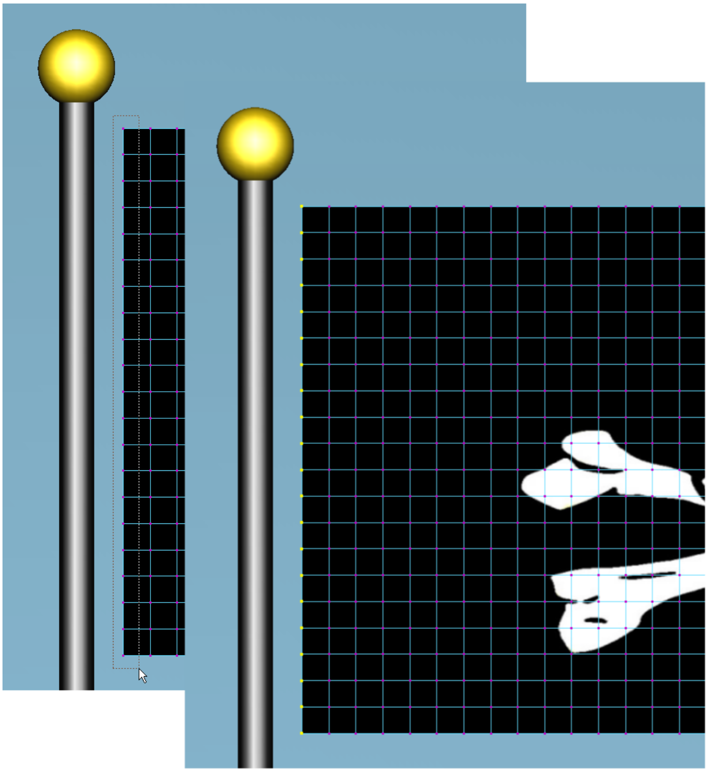
3 Select the flag.

4 Right-click the flag and select Vertex from the context-sensitive menu that appears.

The flag switches to vertex display mode.

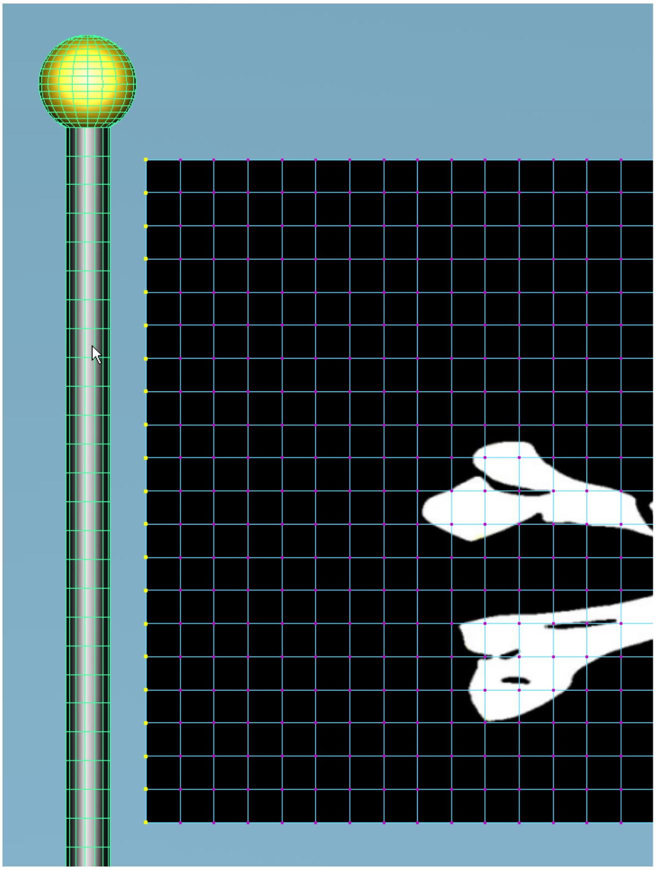
5 Dolly in the camera so that you can view the vertices along the left side border of the flag closely.

6 Select the left border's vertices.



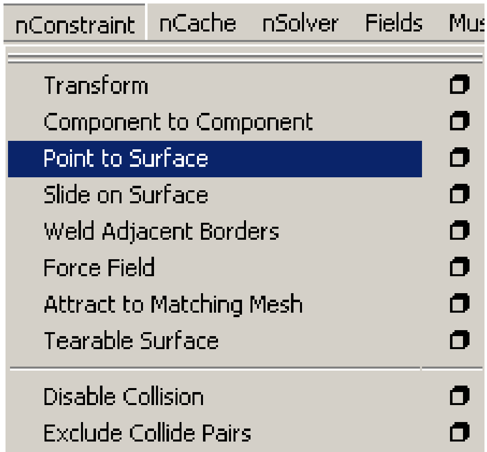
Points selected to be constrained.

7 Shift-select the flag pole passive object to include it in your selection.

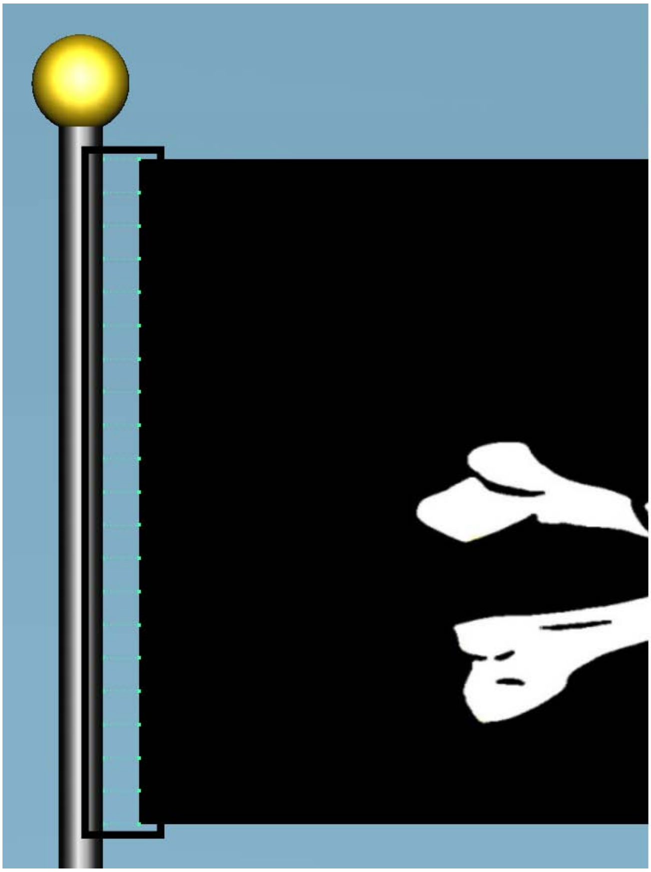


Object selected for the flag points to be constrained to.

8 Select nConstraint > Point to Surface.



A Point to Surface constraint is created, and constraint links appear between the vertices you selected on the nCloth flag, and the surface of the flag pole passive object.



Point to Surface constraint between the flag's CVs and the flag pole.

9 Play back the flag's simulation.

The flag gently falls and ripples, but does not continue to fall out of view. This is because the point to Surface nCloth constraint attaching the flag CVs to the flag pole now restricts the position of the whole flag in XYZ space.

10 Go to the beginning of the playback range.

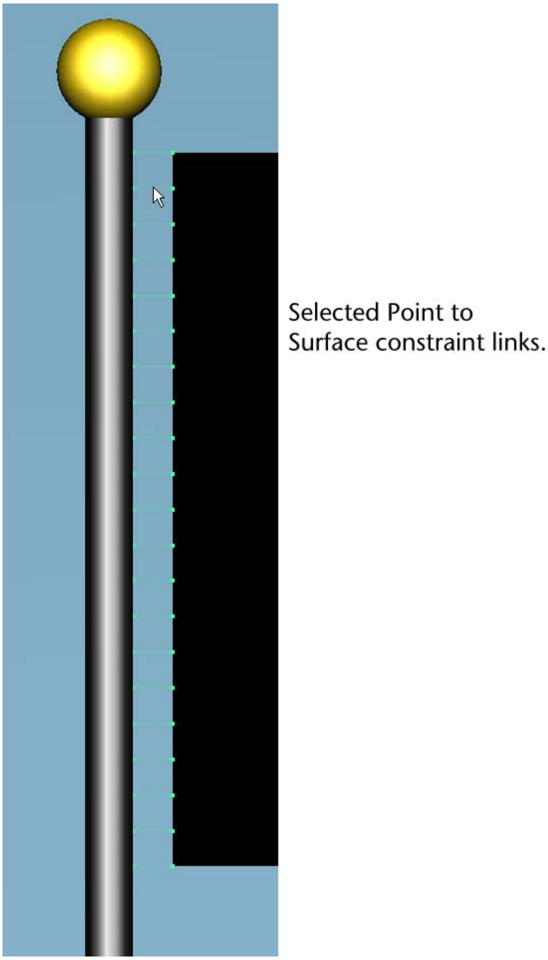
Changing which nCloth points are constrained

After you create a dynamic nCloth constraint, you often times need to edit which nCloth and passive object components participate in the constraint. When Maya Nucleus system components are constrained, they are *members* of their respective constraints. You can change an nCloth constraint's membership by adding or removing nCloth and passive object components from the constraint.

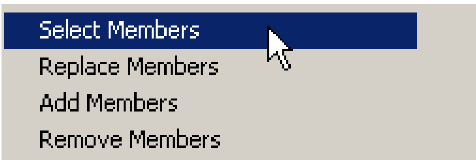
To edit nCloth constraint membership

1 Dolly in the camera so that you can view closely the Point to Surface constraint links that are visible between the flag and its flag pole.

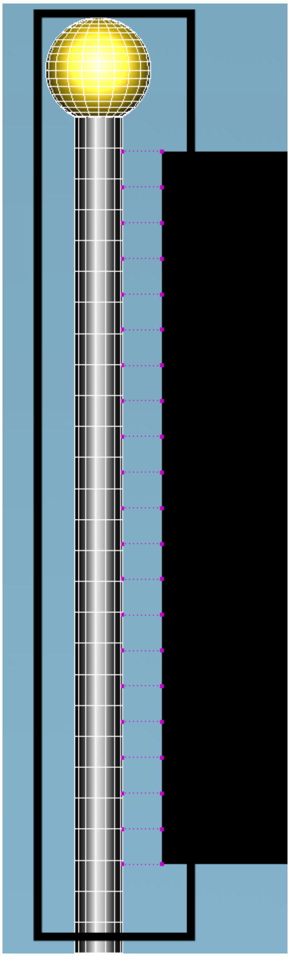
2 Select the Point to Surface constraint by clicking its links.



3 Select nConstraint > Select Members.

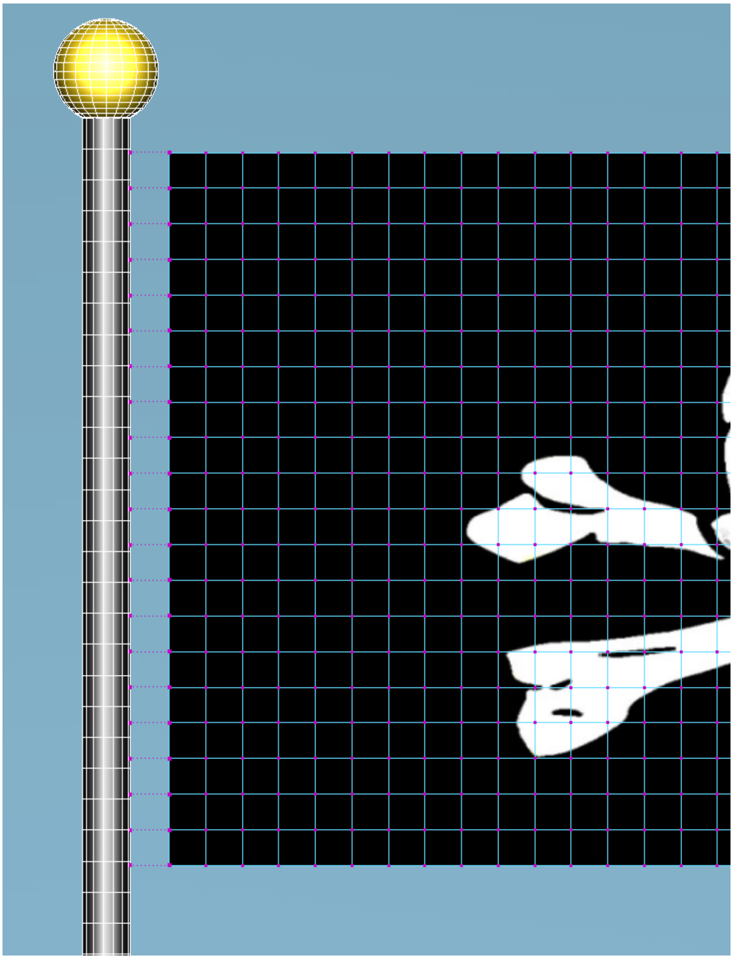


The flag *points* and the flag pole *surface* that are members of the selected Point to Surface constraint highlight in the scene view.

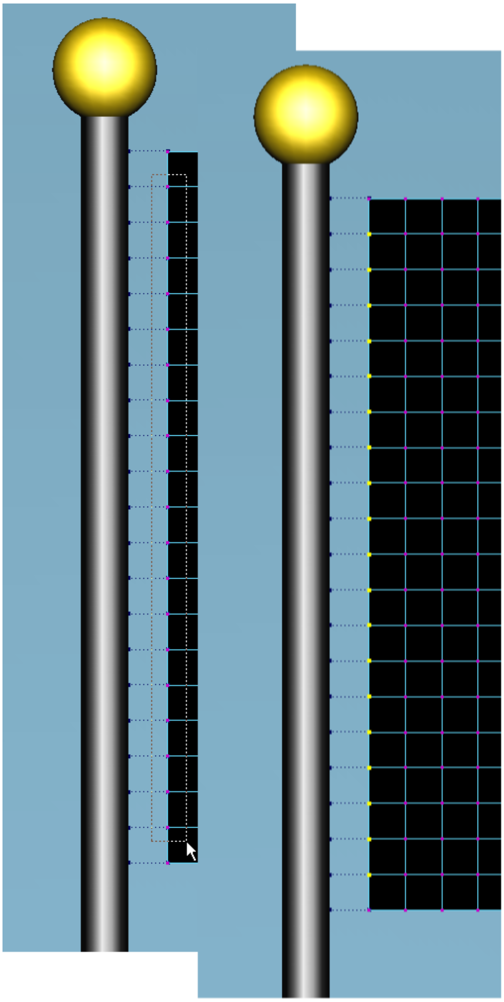


4 Right-click the flag and select Vertex from the context-sensitive menu that appears.

The flag switches to vertex display mode.

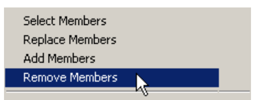


5 Select all the flag points, except the top most and bottom most points.

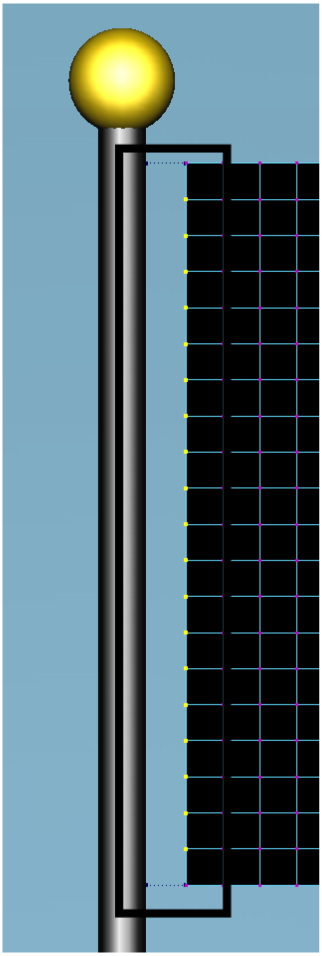


Selected Point to Surface constraint member points.

6 Select nConstraint > Remove Members.



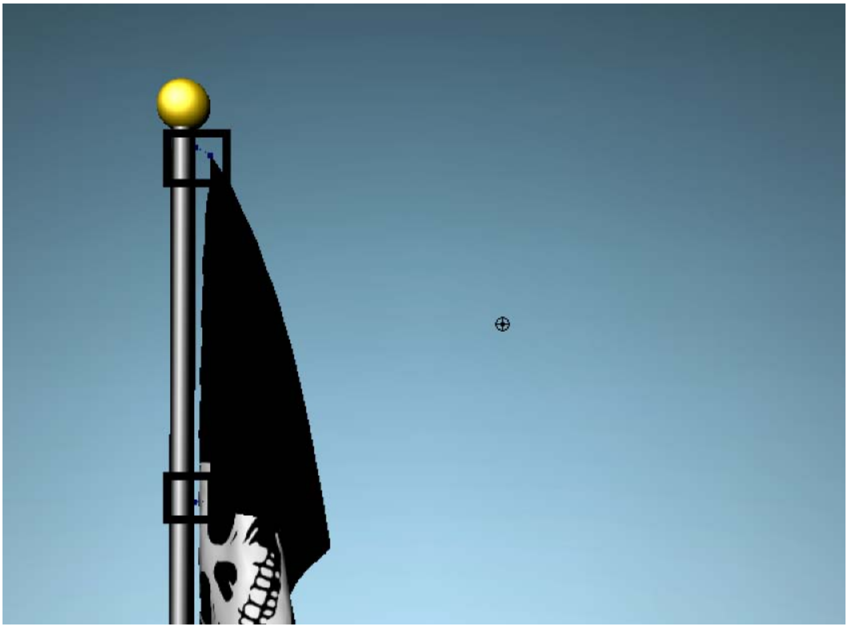
The selected constrained flag points are removed from the constraint and their constraint links disappear from the scene view.



The select flag points are removed from the Point to Surface constraint's membership.

7 Play back the pirate flag's simulation.

The flag gently falls and ripples, but does not continue to fall out of view because it is attached to the flag pole by the two remaining Point to Surface constraint links.



8 Go to the beginning of the playback range.

Making nCloth flap in dynamic wind

After creating the nCloth flag, you can dynamically animate it to flap in the wind by altering certain Dynamic Properties and Maya Nucleus solver Gravity and Wind properties.

To make nCloth appear to wave and flap in the wind

1 Select the flag.

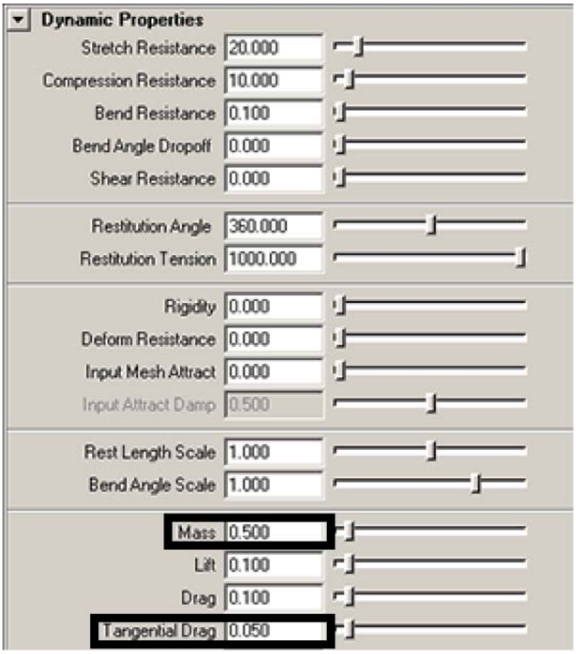
2 Open the Attribute Editor and select the nClothShape tab.

3 In the Dynamic Properties section, decrease Mass and increase *Tangential Drag*.

nCloth flag simulation with:
Mass = 0.5
Tangential Drag = 0.05
Wind Speed = 10

Mass specifies the density of the flag and Tangential Drag biases the amount of drag applied to the flag relative to the surface tangent, which is necessary otherwise the flat plane will not have any wind resistance.

- Change the Mass attribute's value to 0.5.
- Change the Tangential Drag attribute's value to 0.05.



4 Play back the flag's simulation.

The flag appears lighter, however it still falls.

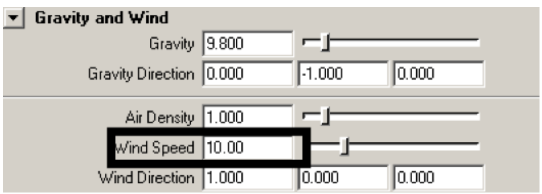
5 Go to the beginning of the playback range.

6 Select the nucleus tab.

7 In the Gravity and Wind section, increase the Wind Speed.

You can increase the speed at which the flag flaps, as well as the amount of ripples moving across the flag's surface, by increasing the Maya Nucleus solver's *Wind Speed*.

8 Change the Wind Speed attribute's value to 10.



9 Play back the pirate flag's simulation.

The flag rises quickly into the air, appearing to be tugged by the increased dynamic air flow. The flag now waves and flaps forcefully, and ripples appear on its surface.

Beyond the lesson

In this lesson you learned how to:

- Constrain nCloth.
- Edit nCloth constraint membership.
- Make nCloth appear to blow in the wind.

Lesson 3: Creating nCloth Clothing

Introduction

A common use for cloth effects is to simulate dynamic clothing on a character. When you attach nCloth clothing to a character, you want to ensure that it correctly simulates the proper material when the scene is animated. For example, you can use nCloth to simulate heavy leather or soft silk. While creating these effects you also want to make use of Maya's nCloth caching abilities to improve playback.

In this lesson you learn how to:

- Adjust and paint properties of nCloth to make it behave like real life fabric.
- Utilize the nCloth cache.
- Constrain an nCloth dress to an animated character.

Lesson setup

To ensure the lesson works as described, do these steps before beginning:

1. Open the scene file named `dressStatic.ma`.

   This file can be found in the `GettingStarted` directory that you set as your Maya project:

   `GettingStarted/nCloth/dressStatic.ma`

   This scene includes static meshes that have been created for you. Both the character and the dress are polygon objects.

Making the dress into an nCloth object

The dress in this scene is initially just a mesh and not an nCloth object. Your first step will be to convert the dress into an nCloth object.

To convert the dress into an nCloth object

1. Select the dress.

2 Select nMesh > Create nCloth > ❐ .

   The Create nCloth Options window appears.

3 Select Edit > Reset Settings.

4 Click Create Cloth.

   The dress is automatically tessellated and converted to a dynamic nCloth object. Maya also creates a new solver named nucleus1 and adds the dress to it.

Making the character wear the dress

If you playback the animation, the dress will fall through the body of the character. To make the character wear the dress, you need to make her a *passive object*.

To make the character wear the dress

1 Select the model of the character.

2 Select nMesh > Create Passive Collider > ❐ .

   The Make Collide Options window appears.

3 Select Edit > Reset Settings.

4 Click Make Collide.

5 Playback the nCloth simulation.

   The dress now stays on character’s body.

Caching nCloth to speed up playback

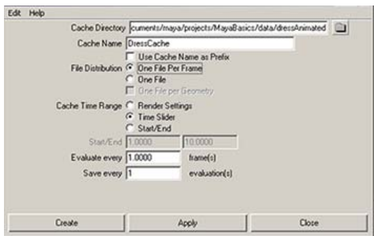
Notice that the playback of the nCloth simulation is quite slow. This is due to the additional calculations required for nCloth and its associated collisions. However, you can use Maya's nCache feature to view the scene at full speed.

To view the scene at full speed

1 Select the nCloth dress.

2 Select nCache > Create New Cache > ❐ .

The Create nCloth Cache Options window appears.



3 Set the Cache directory to the folder to which you want your caches saved.

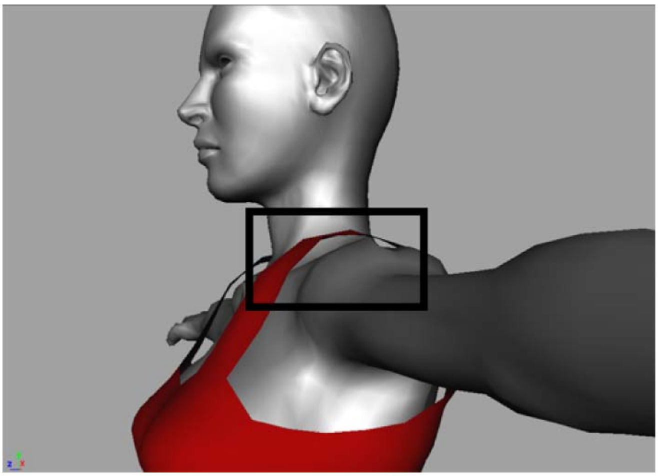
4 Set the Cache name to DressCache.

5 Click Create.

Maya will begin to play the scene automatically and store the frames in the directory you specified. When it is finished, click the Play button in the animation controls. You will notice that the scene now plays back at full speed. However, if you change any of the dress' settings you will need to create a new cache to view your changes. This will be addressed later in this lesson.

Adjusting the fit of the dress

You will notice during playback that the dress (especially the straps) do not actually touch the body. This is because the collision volumes of both the character and the dress are too thick. This means that Maya believes the two are in contact when they are not. You can fix this by adjusting the *collision thickness* of both objects.



The thickness on both the character and dress are set so high that the dress appears to float.

To adjust the collision thickness of the dress

1 Select the dress.

2 Open the Attribute Editor select the nClothShape1 tab.

3 In the Collisions section set Collision Flag to Face, Self Collision Flag to Vertex Face, set Thickness to 0.066 and Self Collide Width Scale to 1.885.

4 Select the character model.

5 From the Attribute Editor select the nRigidShape1 tab.

6 In the Collisions section set Thickness to 0.083.

7 Playback the nCloth simulation.

Notice that the simulation appears exactly the same as before. This is because you are actually playing your *cached simulation*. In order to view your changes, you will need to make a new cache.

To make a new cache for your nCloth simulation

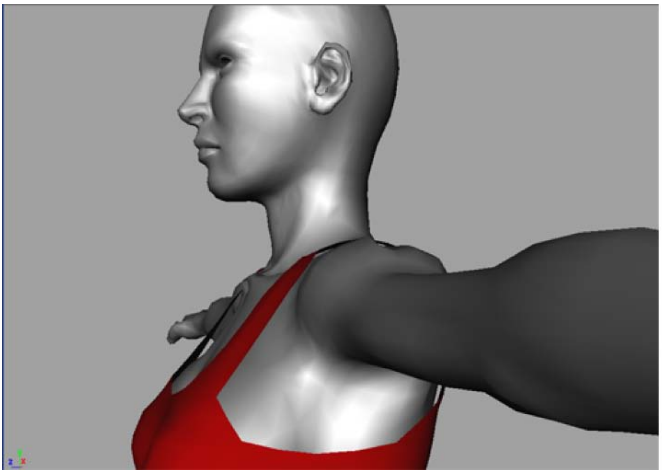
1 Select the dress.

2 Select nCache > Create New Cache.

Maya automatically plays back the nCloth simulation and saves it to disk as a cache.

3 Playback the scene.

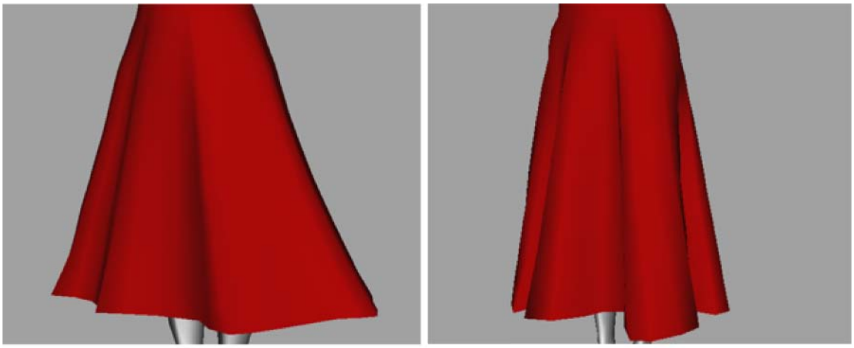
The dress now appears to rest on the character's skin.



With the modified thickness settings, the dress now sits on the character's shoulders and the dress hugs the character's body.

Defining the behavior of nCloth clothing

Although your dress now fits properly over the character, the dress itself seems to be made of a very heavy material in a very low gravity environment. Ideally, you want the dress to behave like a light weight material such as silk in a gravitational environment similar to our own.



You want to make the dress on the left appear like the dress on the right.

To improve the behavior of the dress

1. Select the dress.
2. Open the Attribute Editor and select the nucleus1 tab.
3. In the Scale Attributes section, set the Space Scale to 0.016.

   By reducing the Space Scale attribute, Maya evaluates the large nCloth model as if it were a much smaller object. This will result in a visibly increased gravitational affect on the dress. For more information on Space Scale see Space Scale

4. Select the nClothShape1 tab and in the Dynamic Properties section set Mass to 0.7.
5. In the Dynamic Properties section set Stretch Resistance to 140 and Compression Resistance to 120.

   By lowering the mass of the cloth, Maya will simulate it as a lighter fabric such as silk.

   Increasing the Stretch Resistance and Compression Resistance ensures that the cloth will not become stretched or compressed too greatly as a result of the increased affect of gravity from step 3.

6. Select nCache > Create New Cache.

   Maya automatically plays back the nCloth simulation and saves it to disk as a cache.

7. Playback the scene.

   The dress now behaves more like a light weight material. However, this causes a problem around the breasts where the dress now slips and hangs loose around them rather than over them.

Painting nCloth properties

In a real dress, the area around the breasts (the bodice) is often made of a stiffer material to ensure the cloth does not hang loose. In order to simulate a bodice on your nCloth dress you will use the Paint Vertex Tool to paint Input Attract values on specific areas of the dress.

To paint Input Attract

1. Go to the first frame of the scene.
2. Select the dress and open the Attribute Editor.
3. Select the nClothShape1 tab and in the Dynamic Properties section set Input Mesh Attract to 1.0.
4. Select nCache > Create New Cache.

   Maya automatically plays back the nCloth simulation and saves it to disk as a cache.
5. Playback the scene.

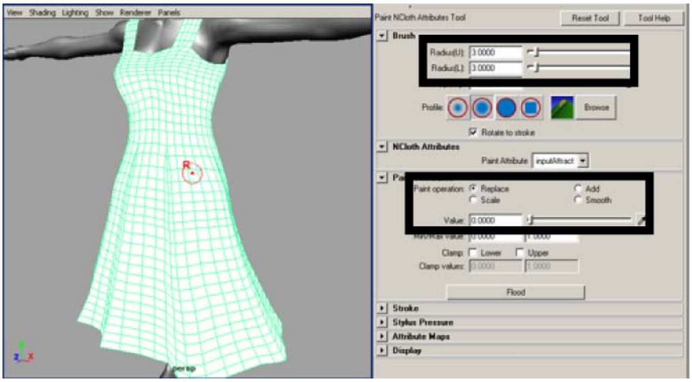
   The dress now maintains its shape. This is because the Input Mesh Attract value you set in step 3 makes the nCloth mimic the shape of its input mesh. The input mesh is the original polygon mesh from which the nCloth dress was created.
6. Select nMesh > Paint Vertex Properties > Input Attract > ❐ .

   In the Tool Settings Editor, the Paint nCloth Attributes Tool becomes the current tool.

7 Click the Reset Tool button and then set the following settings:

- Radius(U): 3.0
- Radius(L): 3.0
- Paint Operation: Replace
- Value: 0

Move the cursor to the dress and you will notice that the cursor becomes an artisan brush.



8 Paint over a portion of the dress.

Notice that the portion you paint over turns black. Black regions of the dress indicate an input attract value of 0 meaning that those areas will be fully dynamic when the simulation is played back. Conversely, white regions indicate an input attract value of 1 meaning those areas will remain identical to the input mesh throughout the simulation.

9 Select nCache > Create New Cache.

Maya automatically plays back the nCloth simulation and saves it to disk as a cache.

10 Playback the nCloth simulation.

The dress now maintains its shape everywhere except those places which you painted with the cursor. This is because the areas that you painted now have an input attract value of 0, making them fully dynamic.

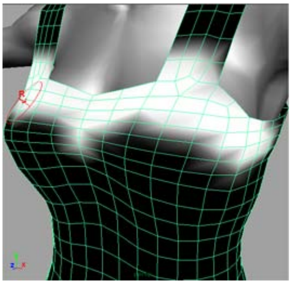
11 Click the Flood button.

The entire dress turns black indicating that a value of 0 has been applied to the Input Mesh Attract attribute for the entire dress.

12 Set Value to 1.

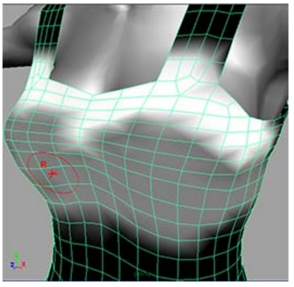
13 Using the Tool brush, paint around the top of the bodice.

Make sure to dolly and tumble the camera to ensure you have painted the entire top of the bodice.



14 Set Value to 0.5.

15 Using the Tool brush, paint the rest of the bodice. Be careful not to paint over your previous values.



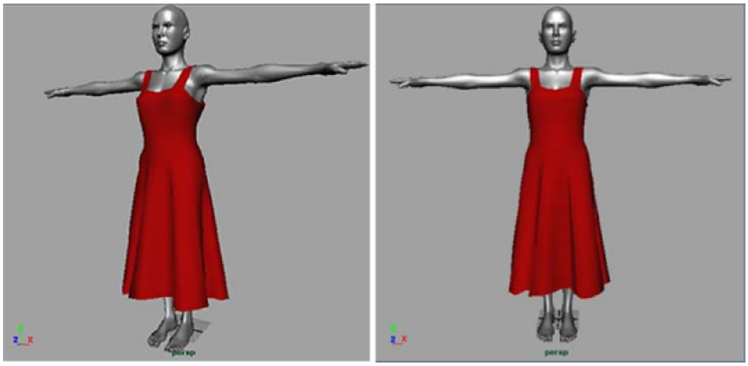
16 Dolly and tumble to ensure that you have painted the entire bodice.

17 Select nCache > Create New Cache.

Maya automatically plays back the nCloth simulation and saves it to disk as a cache.

18 Playback the nCloth simulation.

The dress now appears to be made of a light silk, except around the bodice where it maintains its shape. If your dress does not appear as the example below, you may need to increase the values painted on your bodice.



This completes the first half of the nCloth clothing lesson. In the second half of the lesson you will work with an animated character.

Open the second scene for the lesson

To ensure the lesson works as described, do these steps before beginning:

1 Open the scene file named `DressAnimated.ma`.

This file can be found in the `GettingStarted` directory that was installed with your Maya software:

`GettingStarted/nClothDressAnimated.ma`

This scene includes an nCloth dress worn by an animated character model. A walk and pose animation is already applied to the character.

Setting the initial state

When you load the file you will notice that in the first frame of the animation the dress is suspended such that it does not touch the character's body. This is because this was the input mesh's original position. However, for this character you want the dress to be in a relaxed position before the animation begins. To do this, you need to edit the nCloth's initial and rest states.

To edit the initial state of the dress.

1 Playback the scene.

The character takes a number of steps from her starting point and then poses. You will notice however that the dress begins in a suspended position and falls onto the character as she walks.

2 Go to the beginning of the playback range.

3 Select the dress.

4 Select nSolver > Initial State > Relax Initial State > ❐ .

The Relax Initial State Options window appears.

5 Select Edit > Reset Settings.

6 Set Steps to 300 and then click Relax Initial State.

By setting the Steps option to 300 you are setting the nCloth dress' initial position to mimic that of its position at frame 300 of the animation (when the dress has fallen onto the body and relaxes).

7 Select nMesh > Rest Shape > Set Rest to Start Shape.

The rest shape specifies the shape that the dress attempts to achieve through its stretch, compression and bend resistance values. Thus, by setting it to the Start Shape, you ensure that the dress attempts to return to its initial configuration throughout the simulation.

8 Select nCache > Create New Cache.

Maya automatically plays back the nCloth simulation and saves it to disk.

9 Playback the scene.

The dress now fits tightly on the character from the beginning of the animation to the end.

Constraining nCloth clothing

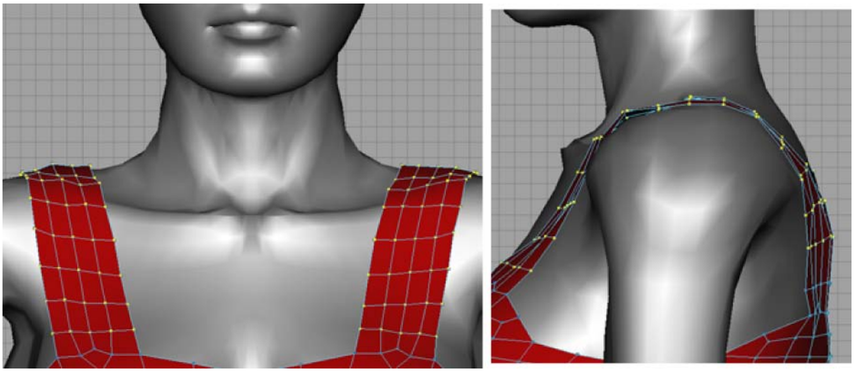
Notice when you playback the scene the straps fall off the character's shoulders as she walks. You can fix this by constraining the straps to her body.

To constrain the dress straps

1 Go to beginning of the playback range.

2 Right-click the model and select Vertex.

3 Select only the CVs of the dress straps.



4 Shift-select the character.

5 Select nConstraint > Point to Surface.

An nConstraint creates a dynamic link between an nCloth's CVs and the components of other nCloths or collision objects. A Point to Surface constraint attaches nCloth components such as CVs, edges, or faces to a target surface. In this case we are attaching the CVs of the dress straps to the character's body.

6 With the newly created constraint still selected, select Display > Hide > Hide Selection.

The constraint markers disappear to give you a better view of the dress straps.

NOTE If you de-selected the constraint and have trouble re-selecting it, you can select Window > Outliner to open the Maya Outliner. You can then select the constraint by left-clicking dynamicConstraint1 in the Outliner.

---

7 Select the dress.

8 Select nCache > Create New Cache.

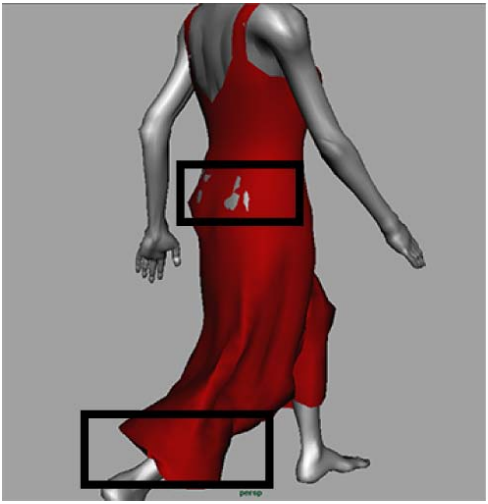
Maya automatically plays back the nCloth simulation and saves it to disk.

9 Playback the scene.

The dress straps now stay with the character's body throughout the animation.

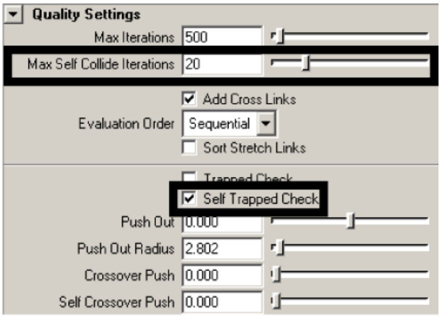
Improving the quality of the nCloth simulation

Although the dress behaves roughly the way a real dress would, you will notice that some of the collisions (the places in which the dress intersects with itself or with the character) do not appear correct. Specifically, in some areas, the character's mesh shows through the dress or the dress bunches up and becomes trapped in that state.



To improve the quality of the nCloth collisions

1. Go to the beginning of the playback range.
2. Select the dress.
3. Open the Attribute Editor and select the nClothShape1 tab.
4. In the Quality Settings section, set Max Self Collide Iterations to 20 and turn on Self Trapped Check.

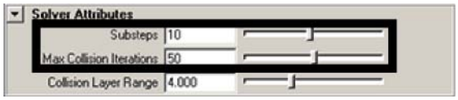


The Max Self Collision Iterations value specifies the maximum number of self-collision related nCloth calculations performed per substep. Increasing this value allows Maya to recognize more self-collisions and thus simulate more realistic cloth, but at the cost of slower speed.

Activating Self Trapped Check allows Maya to monitor collision crossovers. When points crossover one another, Maya will attempt to push them apart. This stops the dress from bunching up and becoming stuck.

5 Select the nucleus1 tab.

6 Under Solver Attributes, set Substeps to 10 and Max Collision Iterations to 50.



Increasing the substeps increases the number of times Maya calculates the nCloth's position per frame. By increasing this value, Maya updates the position of the cloth more often; resulting in a more accurate simulation at a slower speed.

The Max Collision Iterations value specifies the maximum number of collision-related nCloth calculations performed per substep. Increasing this value allows Maya to recognize more collisions and thus simulate more realistic cloth, but at the cost of slower speed.

7 Select nCache > Create New Cache.

Maya automatically plays back the nCloth simulation and saves it to disk.

8 Playback the scene.

The collisions now looks more accurate, the character's body no longer shows through the dress, and the dress no longer bunches or becomes trapped.

You may increase the values in the above steps to further improve the simulation. However, keep in mind that caching the animation becomes progressively slower as these values increase.

Smoothing nCloth clothing

To improve the quality and overall look of the dress you can perform a polygon smooth. Polygon smoothing will increase the number of polygons in your nCloth mesh to give it a smoother overall appearance.

To smooth the dress

1. Select the dress and in the Polygons menu set select Mesh > Smooth.
2. Return to the nDynamics menu set.
3. Select nCache > Create New Cache.

   Maya automatically plays back the nCloth simulation and saves it to disk.
4. Playback the scene.

   The dress in the final animation is smoothed giving it a much silkier appearance.

Beyond the lesson

In this lesson you learned how to:

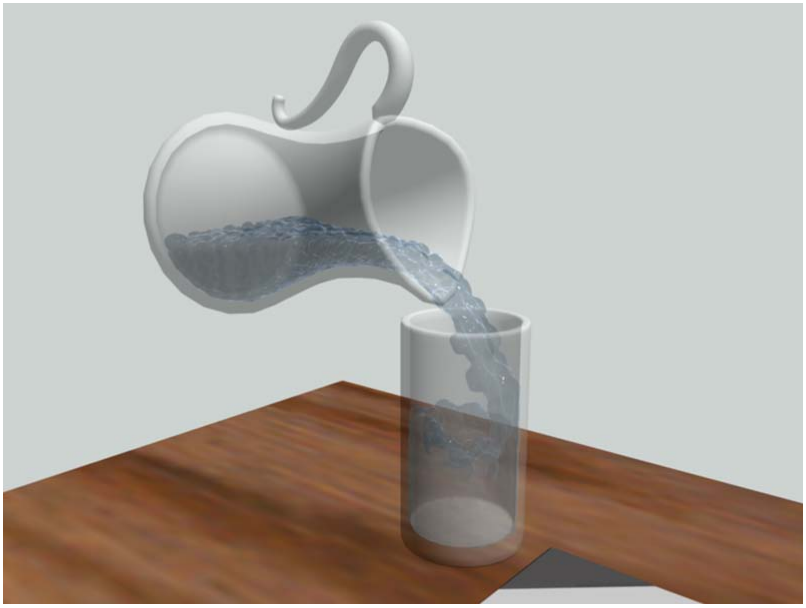
- Create and playback nCloth caches.
- Adjust the properties of an nCloth object.
- Paint nCloth properties.
- Modify the initial and rest state of an nCloth object.
- Improve the quality of nCloth clothing.

For more information and related techniques about nCloth, refer to the Maya Help.

6

nParticles

Introduction



Nucleus dynamics allow users to create particle effects and dynamic simulations that cannot be achieved with Maya classic particles. For example, Liquid Simulation attributes allow you to create particles that behave like liquids, which can interact and drive nCloth animations and deformations. Maya nParticles also includes an improved workflow for setting per particle attributes using ramps, and expands the number of properties and options that can be mapped to per particle attributes.

nParticles is driven by a dynamic simulation framework called Maya® Nucleus™. A Maya Nucleus system is composed of a series of nParticle objects, nCloth objects, passive collision objects, dynamic constraints, and a Maya Nucleus solver. As part of the Maya Nucleus system, the Maya Nucleus solver calculates nParticle simulation, collisions, and constraints in an iterative manner, improving the simulation after each iteration, to produce accurate cloth behavior.

You can use nParticles to create the same particle effects that can be created using Maya classic particles. For most effects and simulations, nParticles and classic particles use the same creation workflows and tools. In addition to Maya classic particle capabilities, nParticles include the following advanced simulation features:

- Ability to collides with other Nucleus objects.
- Ability to self-collide, meaning particle belonging to the same nParticle object can collide with each other.
- Convert nParticles to polygon meshes from which you can modify and manipulate like any other polygon.
- Use Maya's nCaching system to save your simulation data to a server or local hard drive. nCaches are special Maya files that store the simulated point data of your simulations.

Maya Unlimited provides many predefined nParticle effects. These sample nParticle objects, also referred to as *presets*, can be used as a starting point in creating your own custom nParticle effects.

This chapter includes these tutorials:

- Creating a smoke simulation with nParticles on page 175
- Creating a liquid simulation with nParticles on page 193

Preparing for the tutorials

To ensure the lessons work as described, do these steps before beginning:

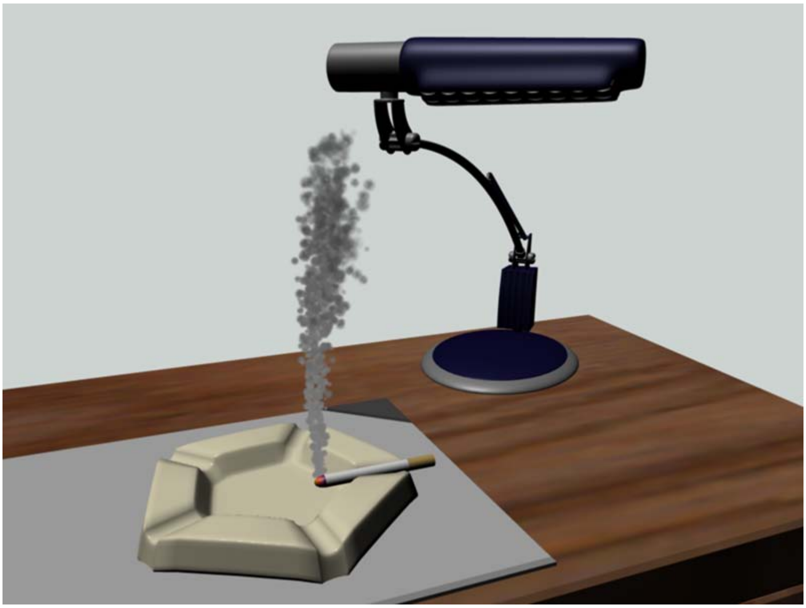
1. Make sure you understand the basic concepts of polygon modeling, animation, and dynamics.
2. If you have not already done so, copy the `GettingStarted` folder from its installation location to your `projects` directory. Then, set the

`GettingStarted` directory as your Maya project. For more information, see Copying and setting the Maya project in the Getting Started with Maya guide.

3 Select the nDynamics menu set. Unless otherwise noted, the directions in this chapter for making menu selections assume you've already selected the nDynamics menu set.

Creating a smoke simulation with nParticles

Introduction



For the nParticle smoke simulation, you create a nParticle system that is composed of an nParticle object, an emitter object, and a nucleus node. You can also use external Dynamic forces with nParticles to create realistic smoke, dust, or jet stream effects.

See the Dynamics tutorials in the *Getting Started with Maya* for particle tutorials that use fields. Although some nParticle attributes are different than Maya classic particle, you can still obtain similar results using nParticles to complete the particles, emitters, and fields section of the Dynamics tutorials.

In this tutorial you use only Nucleus forces and nParticle attribute settings to create a smoke simulation.

Lesson 1: Creating an nParticle system

In the first step you create an nParticle object and its emitter object. An emitter controls the position, direction, quantity and initial velocity of the emitted nParticles when they are born (emitted into the scene).

Lesson setup

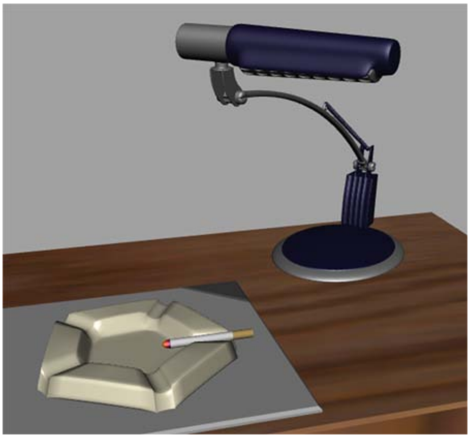
To ensure the lesson works as described, do these steps before beginning:

1 Open the scene file named `Smoke_Simulation_1.mb`.

This file can be found in the `GettingStarted` directory that you set as your Maya project:

`GettingStarted/nParticles/Smoke_Simulation_1.mb`

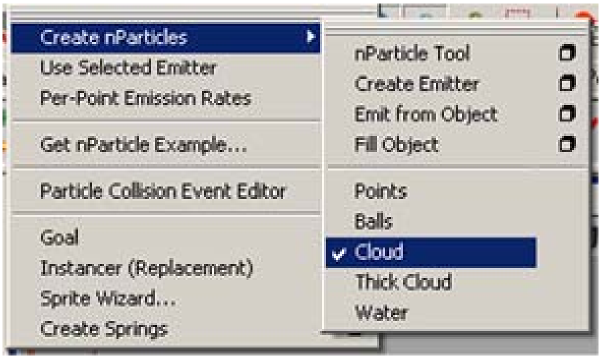
This scene includes textured models that have been created for you. The desk, ashtray, and cigarette provided are all polygon meshes.



Creating an nParticle and emitter object

To create an nParticle and emitter objects

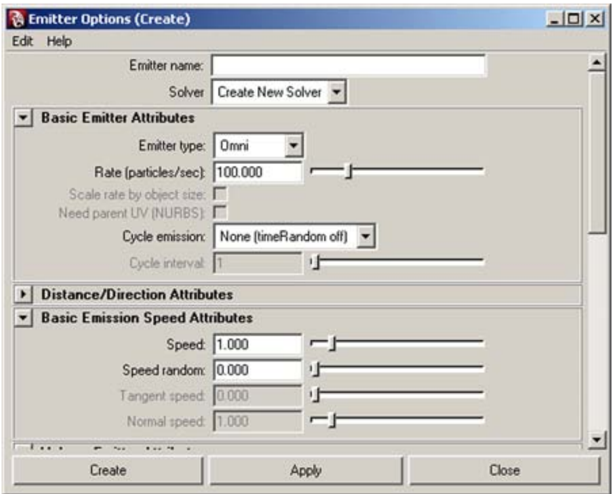
1 Select nParticles > Create nParticles, and select Cloud.



Selecting Cloud presets some nParticleShape attributes that are suitable for simulation such as smoke, dust, or fog.

2 To create an nParticle emitter object, select nParticles > Create nParticles > Create Emitter > .

The Emitter Options (Create) window appears.



3 In the Emitter Options (Create) window, select Edit > Reset Settings.

4 In the Emitter name field, type *Emitter_Smoke*.

Naming your emitter object makes it easier to identify in the Outliner when you want to select it and make attribute adjustments.

5 From the Solver list, select Create New Solver.

Selecting a solver determines which Maya Nucleus system your nParticles belong to.

6 From the Emitter type list, select Volume.

Volume emitters are well suited when you are trying to evenly emit particles from a specific area, such from the tip the cigarette, as opposed to a single point in space.

7 Set Rate (particle/sec) to 100.

This sets the number of particles that are emitted into the scene. This directly impacts the particle density in the scene. The higher the emission rate the thicker the smoke appears in your scene.

8 In the Distance/Direction Attributes section, do the following:

- Set DirectionX to 0.
- Set DirectionY to 1.

Direction sets the emission direction relative to the emitter's position and orientation. When DirectionY is set to 1, the nParticle are emitted along the Y axis to emulate the smoke rising in the air.

9 In the Basic Emission Speed Attributes section, set Speed random to 5. When Speed random is set to a positive value, the emitter generates random speeds for each nParticle.

10 In the Volume Emitter Attributes, do the following:

- From the Volume shape list, select Sphere.
- Set Volume sweep to 180.
  This creates an emitter in the shape of a half sphere, which covers the tip of the cigarette.

11 In the Volume Speed Attributes section, do the following:

- Set Along axis to 1.
- Set Random direction to 0.1.
  This creates some directional noise in the emitted nParticles and causes them to form a cone-shape particle cloud.

■ Set Directional speed to 10.
This setting adds speed in the direction specified by the Direction XYZ attributes of all volume emitters.

12 Click Create.

An Emitter_Smoke1, nParticleShape1, and a nucleus1 node appear in the Attribute Editor.

nParticle and Nucleus nodes

When you create a new nParticle object, a new Nucleus system is created. As part of a Nucleus system, the nParticles can interact with other Nucleus objects that are assigned to the Nucleus solver. If you created an emitter for your nParticles, an emitter node is also created. The following sections describe the new nodes in your new Nucleus system.

nParticleShape1 is the node that carries all the nParticle object attributes that define the appearance, size, and overall behavior of each particle in the nParticle system.

Emitter_Smoker1 is the nParticle emitter node, which carries all the particle emission attributes such as particle emission rate and emitter type. This node also carries the transformation attributes (translation, rotation, scale) for the emitter object. The emitter node used by nParticles is the same emitter node used by Maya classic particles.

nucleus1 is the Maya Nucleus solver node, which carries all the attributes affecting the solver system, including internal forces. For more information, see Maya Nucleus solver properties.

Beyond the lesson

In this lesson you were introduced to the basic concepts of nParticles and the Nucleus dynamics system. In addition, you learned how to:

■ Create nParticles from a source object called an emitter.

When you create an nParticle object and emitter, a Nucleus node is also created.

- Use a volume emitter with an nParticle system.
- Set up the general behavior of your nParticles by adjusting volume emitter attributes such as Distance/Direction and Volume Speed Attributes.

For more information and related techniques about nParticles, refer to the Maya Help.

Lesson 2: Adjusting nParticle attributes

nParticle object attributes determine the physical characteristics of all the nParticles belonging to the object. These attributes affect how the nParticles look and how the nParticle objects move and collide with other Nucleus objects. If you create an emitted nParticle system, the emitter attributes also affect the overall characteristics of your nParticle simulation.

Lesson setup

To ensure the lesson works as described, do these steps before beginning:

1. Open the scene file named `Smoke_Simulation_2.mb`.

   This file can be found in the `GettingStarted` directory that you set as your Maya project:

   `GettingStarted/nParticles/Smoke_Simulation_2.mb`

   This scene includes textured models that have been created for you. The desk, ashtray, and cigarette provided are all polygon meshes.

Moving the nParticle emitter

When you play back the simulation, you notice that nParticles are being emitted at the scene origin. In this lesson, you move the emitter so that the particles are emitted from the tip of the cigarette.

To move the nParticle emitter object

1. In the Outliner, select the *Emitter_Smoke1* object.
2. Select Display > UI Elements > Channel Box/Layer Editor to display the Channel Box.

3 In the Channel Box enter the following values:

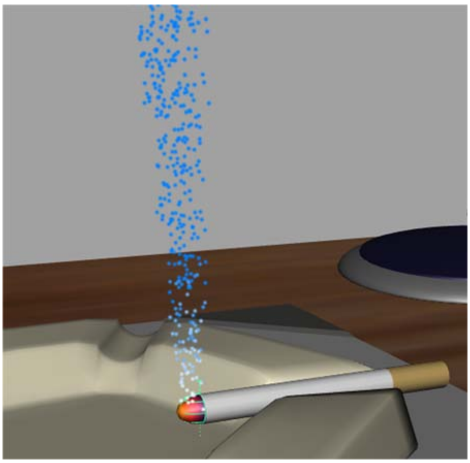
- Translate X: 49.3
- Translate Y: 120
- Translate Z: -27.3
- Rotate Y: -90
- Scale Z: 1.5

4 In the scene view, dolly and tumble so that you get a close-up view of the cigarette's tip.

The nParticle emitter is now positioned at the tip of the cigarette.

5 Play back the simulation.

nParticles are emitted from the end of the cigarette and into the scene. The size and color of the nParticles are currently set to default values. In the next section you set nParticle size to give the particles the appearance of smoke.



Editing Lifespan and Radius

When you play back the simulation, you notice that the nParticles are not dissipating into the air as smoke would. This is because the nParticles lifespan is not yet defined. Using the nParticle Lifespan attributes you make the nParticle disappear from the scene after they reach a specified age. You can set the nParticle object's size according to its lifespan so that nParticles decrease or increase in size as they age.

To set nParticle Lifespan

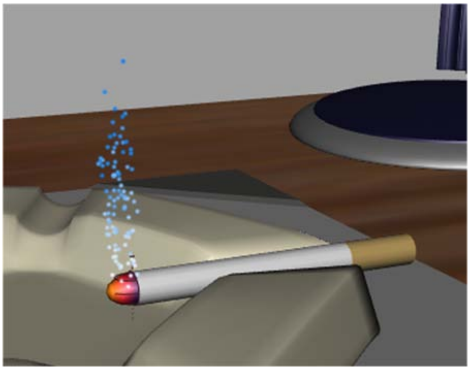
1 In the Outliner, select *nParticle1*.

2 In the Attribute Editor, click the nParticleShape1 tab.

3 In the Lifespan section, select Random range from the Lifespan Mode list.

4 Play back the simulation.

   The nParticles now die (disappear from the scene) shortly after they are emitted into the scene. nParticle Lifespan is measured in seconds, meaning that at a Lifespan of 1.0, the nParticles die after one second.



5 Set Lifespan to 10.

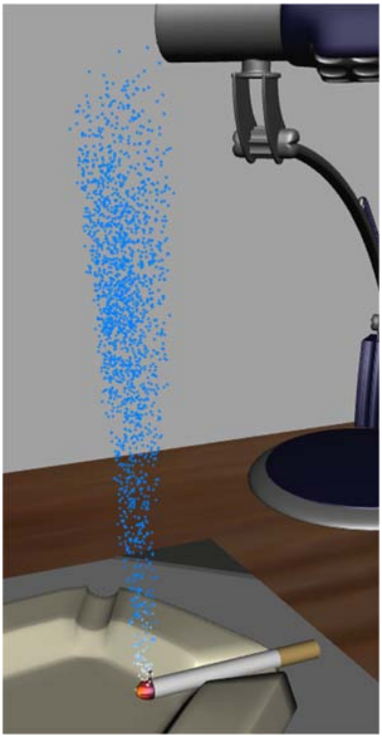
6 Rewind the simulation and then play it back.

   The nParticles die 10 seconds after they are emitted into the scene. For this tutorial, set Lifespan to 20.

7 To add some randomization to the nParticle's Lifespan, set Lifespan Random to 15.

8 Rewind the simulation and play it back.

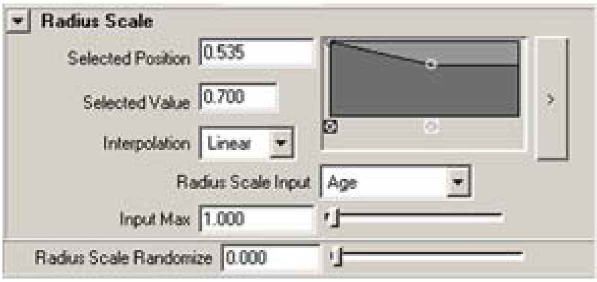
   The nParticle die off more randomly, but the nParticle object is not dense enough to look like smoke. You can make your nParticle object denser by increasing the nParticle Radius.



For a smoke simulation, the nParticles must gradually disperse as they rise in the air. You can use the Radius Scale ramp to increase the radius of each nParticle as they age, which creates the effect of smoke dispersing until it gradually disappears from the scene.

To set nParticle Size

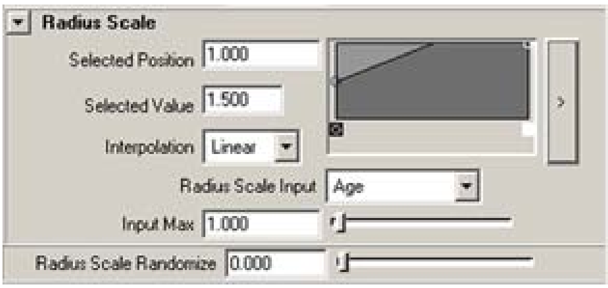
1. In the Particle Size section, set Radius to 1.0.
2. In the Radius Scale section, click anywhere in the ramp to create a new point.



Creating a new marker allows you to set a Selected Value and Selected Position. Selected Values specify the scaling value of the input attribute at the Selected Position. For example, each Selected Value specifies a scaling value for the nParticle object's Radius attribute. By default ramps define a scaling value of 1, meaning that only one mark exists on the ramp and that marker represents an attribute scaling factor of 1. Setting a selected position to 0.5 sets per-particle radius to half the Radius value.

3 For the first marker in the ramp, set Selected Position to 0, and Selected Value to 0.50.

4 Click the second marker in the ramp. Set Selected Position to 1, and Selected Value to 1.5.



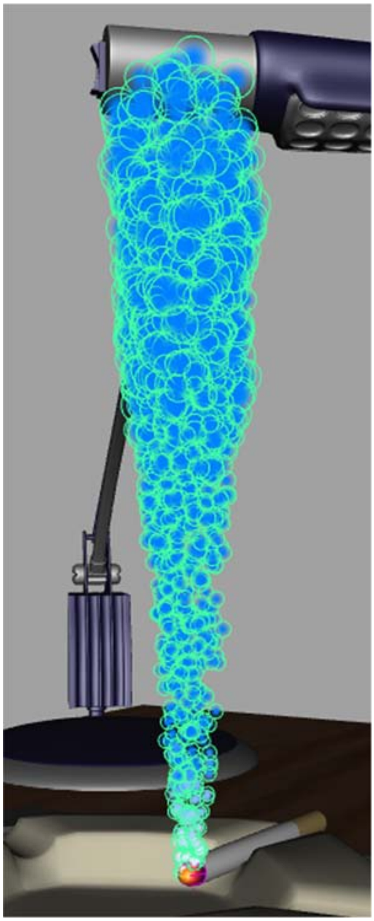
5 To map the nParticles Radius to the nParticles' age, set Radius Scale Input to Normalized Age.

When Normalized Age is used, per particle radius is mapped within the range of the nParticle object's lifespan.

6 Set Interpolation to Smooth for each marker in the ramp.

7 Play back the simulation.

You can continue to adjust the per-particle radius by adding more markers to the curve, or optimize the ramp's values as the simulation plays back.



8 To add randomization to the per-particle radius, set Radius Scale Randomize to 0.25.

Setting nParticle Shading attributes

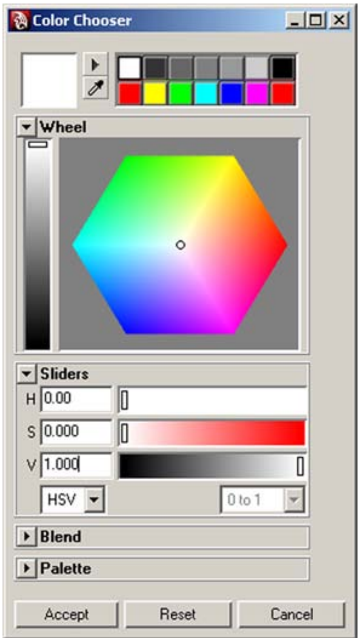
To color the nParticles so they look like smoke, you use the Color ramp to specify a color gradient for the smoke. In addition, you adjust Opacity to help produce the effect of the smoke disappearing into the air as the nParticles age.

Like the nParticle Radius attribute, Color and Opacity can be set on a per nParticle object or per-particle basis. Using the nParticle objects's Shading ramps, you can map per-particle Color and Opacity to nParticle properties such as particle Age, Particle ID, and Radius.

To set nParticle Color attributes

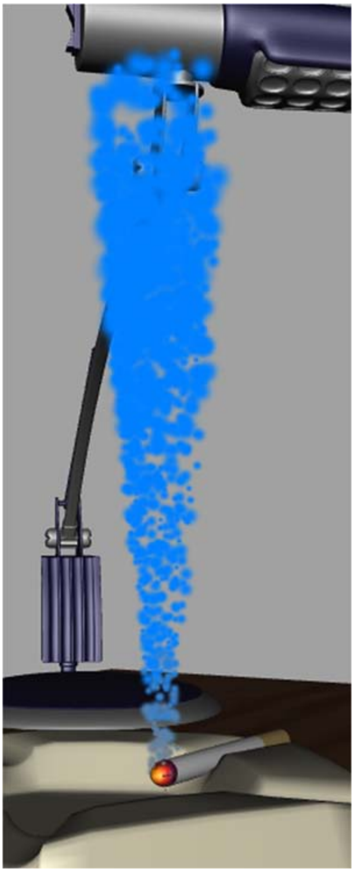
1. In the Outliner, select *nParticle1*.
2. In the Attribute Editor, click the nParticleShape1 tab.
3. In the Shading section, scroll to the Color section.
4. To set the color of the nParticles when they are first emitted into the scene, click the ramp's far left marker (at Selected Position 0), and then click the color swatch beside Selected Color.

   The Color Chooser appears.

5. In the Color Chooser, choose a medium shade of grey, or use type the following color values into the HSV (Hue-Saturation-Value) fields:
   - H: 0
   - S: 0
   - V: 0.5

Play back your simulation while you are choosing colors so you can see how the colors look while the nParticle are animated.



6 To set the color of the nParticles when they continue to age, click the ramp's far right marker (at Selected Position 1.0), and then click the color swatch beside Selected Color.

The Color Chooser appears.

7 In the Color Chooser, choose a shade of grey that is darker than the color you previously selected or type the following color values into the HSV (Hue-Saturation-Value) fields:

- H: 0
- S: 0

- V: 0.25

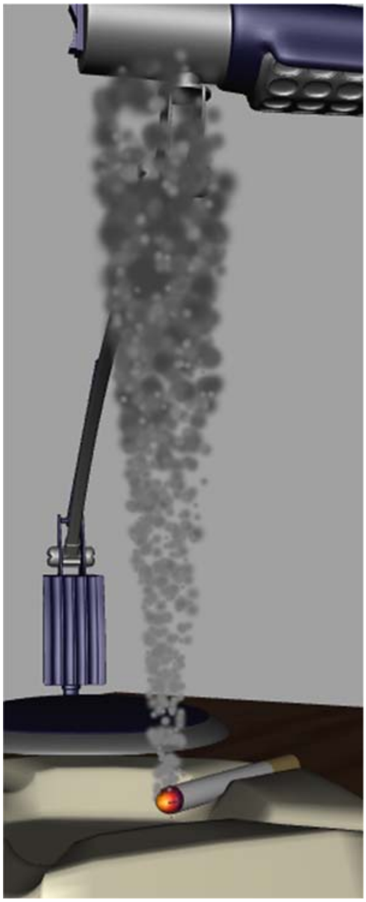
8 Set Color Input to Normalized Age.

As each nParticle ages from its birth (emission), its color changes from the color specified by the left marker to the one specified by the right marker.

9 To improve the way that the nParticles' colors blend as they age, do the following:

- Set Interpolation to Smooth for each marker.
- Set Color Randomize to 0.75.

When you play back the simulation, you notice that the smoke is too dense and heavy to resemble cigarette smoke. To make the smoke look thinner, you adjust the nParticle Opacity.



To Set nParticle Opacity

1 In the Opacity Scale section, click in the ramp, and set an Opacity ramp by setting the following three markers:
   - Selected Position: 0, Selected Value: 1
   - Selected Position: 0.63, Selected Value: 0.22
   - Selected Position: 1, Selected Value: 0

2 Set Interpolation to Smooth for each marker.

3 Set Opacity Scale Input to Normalized Age.

4 Play back the simulation.

Beyond the lesson

In this lesson you learned how to:

- Adjust nParticle lifespan so that nParticles die off at the appropriate moment of your simulation.
- Use the Radius Scale ramp to map per-particle radius to nParticle age.
- Use the Color and Opacity attributes to define the color and thickness of the smoke.

You can further optimize the color and thickness of the smoke by continuing to adjust the Color ramp or Opacity Scale ramp.

For more information and related techniques about nParticles, refer to the Maya Help.

Lesson 3: Adding Nucleus wind to the simulation

Like Maya classic particles, you can use fields to animate the motion of nParticles and create the effect of nParticles flowing around obstacles, as well as other effects, such as explosions, tornadoes, and rocket exhaust. See the Dynamics tutorials in the *Getting Started with Maya* for particle tutorials that use fields. Although some nParticle attributes are different than Maya classic particle, you can still obtain similar results using nParticles to complete the particles, emitters, and fields section of the Dynamics tutorials.

With nParticles, you can also use the dynamic wind that is generated by the Maya Nucleus solver to affect the motion of your particle. Gravity and Wind forces that are generated by the Nucleus system are different from the Maya dynamic forces that you create when using Maya field. Nucleus forces are internal, in that they only affect Nucleus objects. Nucleus object that are assigned to the same Nucleus solver are affect by the same intensity of the solver's wind and gravity.

Lesson setup

To ensure the lesson works as described, do these steps before beginning:

1. Open the scene file named `Smoke_Simulation_3.mb`.

   This file can be found in the `GettingStarted` directory that you set as your Maya project:

   `GettingStarted/nParticles/Smoke_Simulation_3.mb`

   This scene includes textured models that have been created for you. The desk, ashtray, and cigarette provided are all polygon meshes.

In this lesson, you use the Nucleus solver wind to influence the smoke so that it drifts as it rises. Any other Nucleus objects (nCloth objects or other nParticle objects) that are assigned to the same Nucleus solver would also be affected by the force of the wind.

To set Nucleus wind in your simulation.

1 In the Attribute Editor, click the nucleus1 tab.

2 In the Gravity and Wind section, Set Wind Speed to 2.0.

   Wind Speed determines the force and intensity of the wind. A higher value means a fast wind speed, which has a more pronounced effect on the smoke.

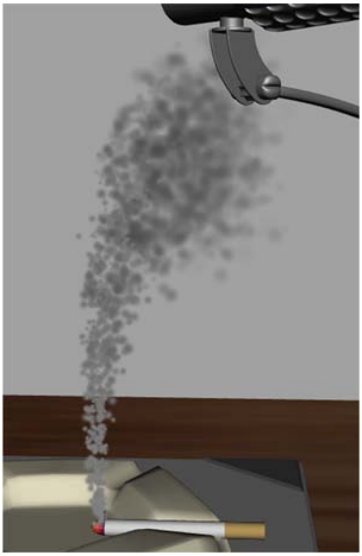
3 To set Wind Direction, do the following:

   - Set wind direction to 1 in the X axis.
   - Set wind direction to 0 in the Y axis.
   - Set wind direction to -1 in the Z axis.

4 Set Wind Noise to 3.0.

   This adds some randomization to the wind's direction.

5 Play back the simulation.



   The smoke is now dispersing over a larger area and it is drifting toward the desk light.

   You can continue to adjust your smoke simulation settings, such as increasing the nParticle emission rate to 150 or adjusting the curves on the Radius Scale, Color, and Opacity Scale ramps.

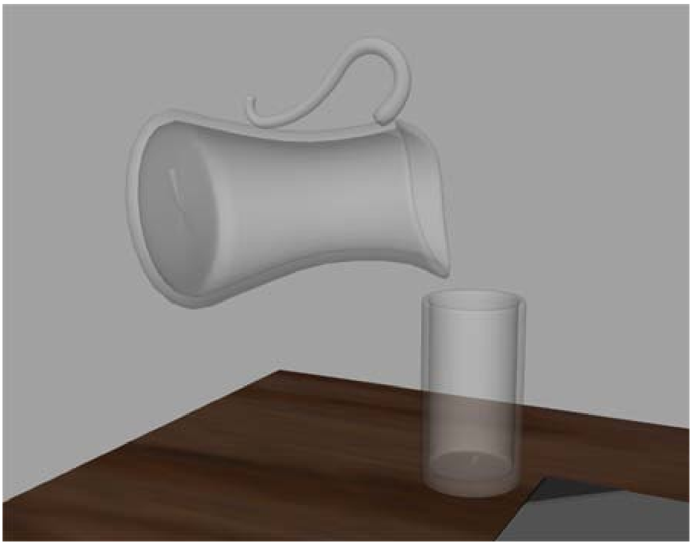
Beyond the lesson

In this lesson, you learned how to add Nucleus wind to your nParticle effect. You can experiment with the Nucleus solver's Wind Speed, Wind Density, and Wind Direction attributes to change the direction of the rising smoke. You can also add Maya fields, such as Volume Axis, Turbulence, or Radius, fields to effect the behavior of the smoke.

For more information and related techniques about nParticles, refer to the Maya Help.

Creating a liquid simulation with nParticles

Introduction



A common use of particles systems is to create a particle system that emulates flowing materials ranging from thick viscus fluids such as oil to fast moving liquids such as water. Using Liquid Simulation attributes, you can add properties to your nParticle objects to make them look and behave like a flowing liquid. Liquid Simulation attributes allow you to create simulations, such as flowing lava, droplets of liquids, and splashes of water.

When creating a liquid simulation, it is often best to start with a small nParticle system (small particle count) and set the attributes that control the behavior of the particle flow. You can then increase the density of the particle system and edit the attributes that give the nParticles the appearance of water.

Throughout these lessons, you will experiment with Liquid Simulation attributes to create an nParticle system with the characteristics of water, including the following:

- Volume: Water fills a given space uniformly.
- Fluidity: Water flows and moves as a single continuous substance, often exhibiting surface tension.
- Shading: Water is a transparent substance that can reflect light.

In this tutorial you learn how to:

- Use Fill Object to create a Water style nParticle system.
- Adjust nParticle Liquid Simulation attributes to create a particle system that looks and behaves like water.
- Cache your nParticles simulation.
- Convert a liquid simulation to an nParticle output mesh.
- Save nParticle attributes as a custom nParticle preset.

Lesson 1: Creating a Water style nParticle object

In this lesson you learn how to:

- Use Fill Object to create a Water style nParticle system.
- Adjust nParticle Liquid Simulation attributes to create nParticles that look and behave like water.
- Set Nucleus solver attributes for your simulation.

The first step in creating a liquid simulation is to create a water style nParticle system, which will be constrained to the modeled polygon water pitcher. You can do this using the Fill Object nParticle creation method.

Lesson setup

To ensure the lesson works as described, do these steps before beginning:

1. Open the scene file named `Liquid_Simulation_1.mb`.

   This file can be found in the `GettingStarted` directory that you set as your Maya project:

   `GettingStarted/nParticles/Liquid_Simulation_1.mb`

   This scene includes textured models that have been created for you. The desk, water pitcher, and water glass provided are all polygon meshes. The water pitcher mesh is animated so that it lifts off the desk, tips as to pour into the glass, and then sets back down to the desktop.

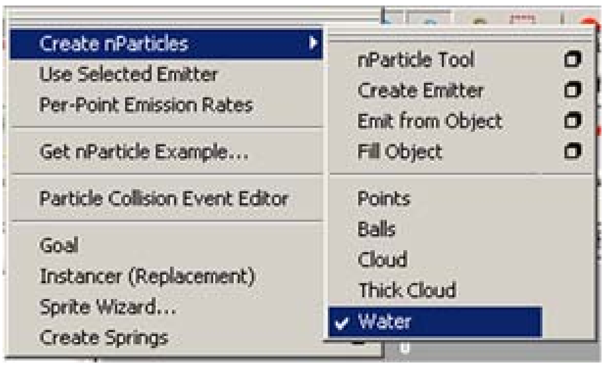
Create a Water style nParticle

Using a Water style nParticle will presets some nParticleShape attributes, including Liquid Simulation attributes, as well as set the Particle Render Type to Blobby Surface.

To create a Water style nParticle object

1. Select nParticles > Create nParticles, and select Water.

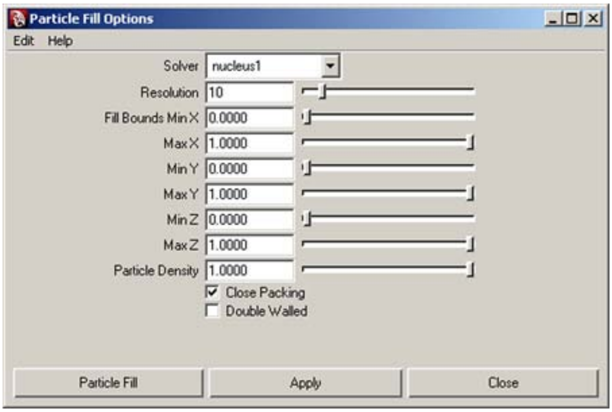
   Selecting Water presets some nParticleShape attributes for liquid simulations.



2. Select the geometry that you want to fill with the Water style nParticle. In this case, select the water pitcher mesh. To do this, in the Outliner, select *geo_pitcher*, contained in the *grp_Pitcher* object.

3. Select nParticles > Create nParticles > Fill Object > ❐ .

The Particle Fill Options window appears.



4 In the Particle Fill Options window, select Edit > Reset > settings.

5 From the Solver list, select Create New Solver.

6 Ensure that Resolution is set to 10.

Resolution specifies how many nParticles are evenly placed along the fill geometry's longest axis.

7 Ensure that Close Packing is turned on.

8 Turn on Double Walled.

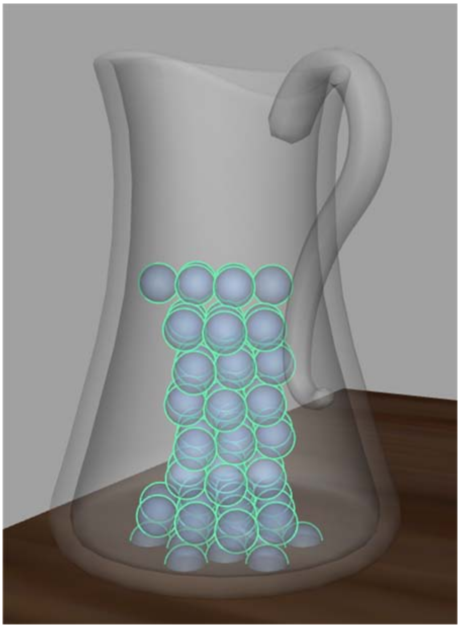
Turning on Double Walled ensures that the thickness of the pitcher's geometry is taken into account, meaning that the particles only fill the pitcher's empty volume, rather than between the polygon walls.

9 Click Particle Fill.

10 In the Outliner, double-click *nParticle1* and type *nParticle_Water*.

Renaming your nParticle object makes it easier to identify it when you have more than one in your scene.

11 Play back the simulation.



The nParticles fall through the bottom of the mesh water pitcher and disappear from the scene. This is because the nParticles do not recognize the object they are contained in.

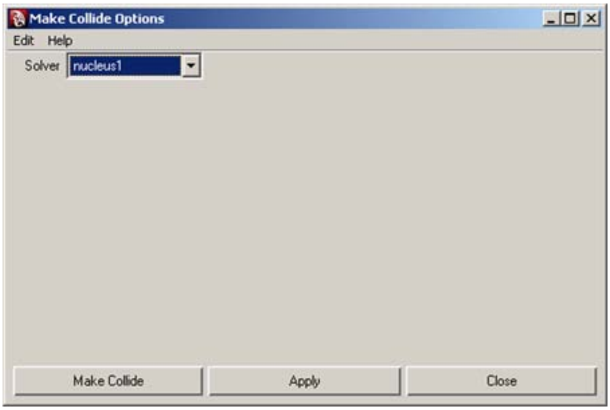
In this tutorial, as you simulate the pitcher pouring water into the glass, both the pitcher and glass meshes must be converted to passive collision objects.

Creating passive collision objects

For the nParticles to interact with the water pitcher mesh, it must be converted to a passive collision object and assigned to the same Nucleus system.

To convert the pitcher and glass meshes to passive collision objects

1. Rewind the simulation to the start frame.
2. In the scene view, Shift-select the pitcher and glass meshes.
3. Select nMesh > Create Passive Collider > ❒ .
   The Make Collide window appears.



4 From the Solver list, select nucleus1.

5 Click Make Collide.

6 To rename the pitcher and glass nRigid objects, in the Outliner, double-click *nRigid1* and *nRigid2*, and type *nRigid_Pitcher* and *nRigid_Glass* respectively.

7 Play back the simulation.

When you play back the simulation, you notice that the nParticles are not behaving as you might expect. Rather than immediately falling to the bottom of the pitcher like water, they fall slowly, as if the simulation was playing back in slow motion.

Before you begin simulating your liquid nParticles, it is important to adjust the Nucleus solver properties to suit your simulation. The Maya Nucleus solver properties control internal forces that affect all of the nodes that are members of a particular solver system. One of the most important Nucleus solver attributes to be aware of is Space Scale. When Space Scale is set to its default value of 1, the Nucleus solver applies Gravity and Wind to objects as if the objects were scaled in meters.

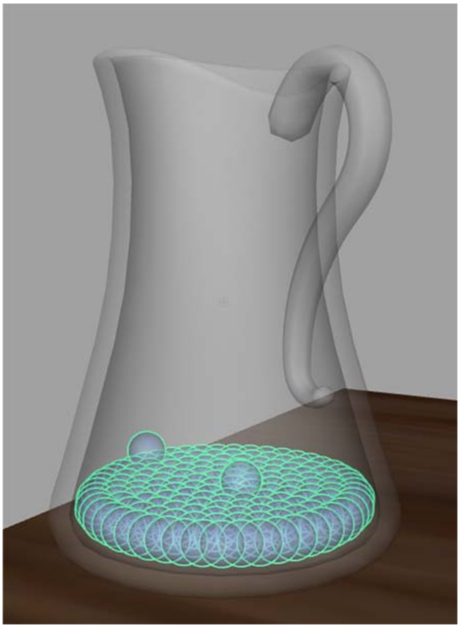
To edit Space Scale

1 In the Attribute Editor, click the nucleus1 tab.

2 In the Scale Attributes section, set the Space Scale to 0.01.

By reducing the Space Scale value, Maya evaluates the nParticle system as if they were much smaller in size. This will result in a visibly increased gravitational affect on the nParticles.

3 Rewind the simulation to the start frame, and then play it back.



When you play back the simulation, you notice that the nParticles are now contained in the pitcher, but they occupy a small amount of the pitcher's volume. Also, the nParticles appear to be pushing down as if they are under a force of pressure or compression. In the next section, you begin adjusting the Liquid Simulation attributes that make the nParticles look and behave like a liquid.

Beyond the lesson

In this lesson you learned how to:

- Use Fill Object to create an nParticle system.
- Create passive collision objects to contain your nParticles.
- Adjust the Nucleus slover's Space Scale attribute to suit the scale of your simulation.

Lesson 2: Adjusting Liquid Simulation attributes

One of the main characteristics of water refers to its volume, or rather how it fills the area inside a given volume. In order to reproduce this with nParticles, there needs to be some space between each individual particle. However, having too much space between them can hinder the fluidity of water, which is another important characteristic. Therefore, we need to set the Liquid Radius Scale attribute to control the amount of space between each nParticle, which compresses and holds the individual nParticles together.

Lesson setup

To ensure the lesson works as described, do these steps before beginning:

1. Open the scene file named `Liquid_Simulation_2.mb`.

   This file can be found in the `GettingStarted` directory that you set as your Maya project:

   `Getting Started/nParticles/Liquid_Simulation_2.mb`

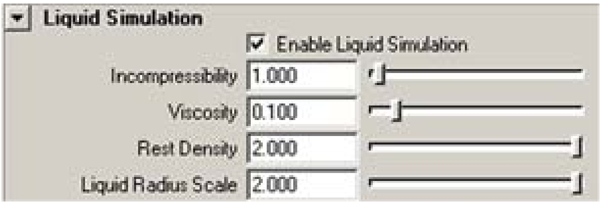
   This scene includes textured models that have been created for you. The desk, ashtray, and cigarette provided are all polygon meshes.

Adjusting the liquid's volume

To set Liquid Radius Scale

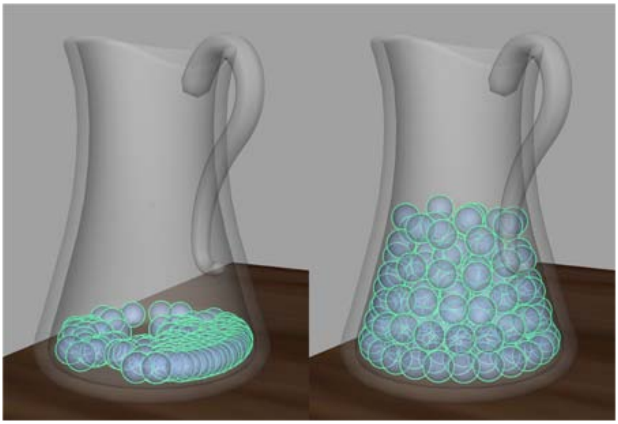
1. In the Outliner, select *nParticle_Water*.
2. In the Attribute Editor, click the nParticle_WaterShape tab.
3. In the Liquid Simulation section, ensure that Enable Liquid Simulation is turned on.

4 Set Liquid Radius Scale to 2.0.



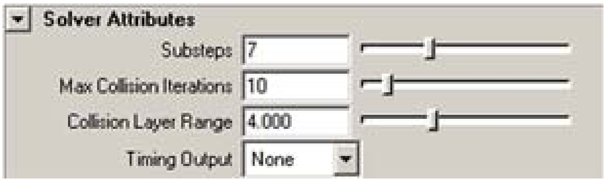
5 Play back the simulation.

The nParticles are now stacked on top of each other, creating considerably more volume in the pitcher.



6 For this liquid simulation, you want the liquid nParticles to fill one-third of the pitcher's volume. Adjusting the Nucleus solver Substeps and Max Collision Iterations can also affect the amount of volume that is generated by the Liquid Radius Scale value. To adjust Substeps and Max Collision Iterations, do the following:

- In the Attribute Editor, click the nucleus1 tab.
- In the Solver Attribute settings, set Substeps to 7.
- Set Max Collision Iterations to 10.



More about Liquid Radius Scale

When creating liquid simulations, Liquid Radius Scale can be affected by other nParticle attributes, such as Radius and Collide Width Scale. For example, Liquid Radius Scale uses the nParticle object's Radius to determine how nParticles overlap, and any change in the nParticle Radius will affect how your nParticles overlap in the liquid simulation. Adjusting the Nucleus solver Substeps and Max Collision Iterations can also affect the amount of volume that is generated by the Liquid Radius Scale value. You can experiment with these settings by changing Substeps to 10 and Max Collision Iterations to 15 and observe how your liquid simulation is affected.

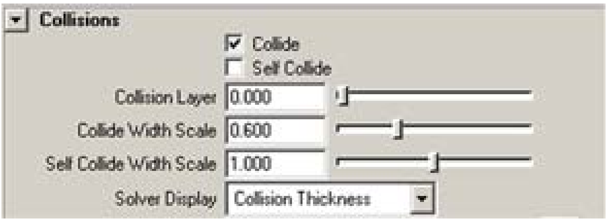
In the next section of this lesson, you adjust nParticle Collide Scale Width and Incompressibility attributes to add fluidity to your nParticles.

Adding fluidity to the nParticles

To make it easier to see how the nParticles are interacting with the pitcher and glass passive collision objects, turn on the Solver Display to see the nParticle collision volume. Collision volumes are a non-renderable surface offset from each nParticle radius that the nucleus solver uses when calculating the nParticle object's with each passive object collisions.

To Edit Collision Width Scale

1. In the Outliner, select *nParticle_Water*.
2. In the Attribute Editor, click the nParticle_WaterShape tab.
3. In the Collisions section, select Collision Thickness from the Solver Display list.
4. Set Collide Width Scale to 0.6.

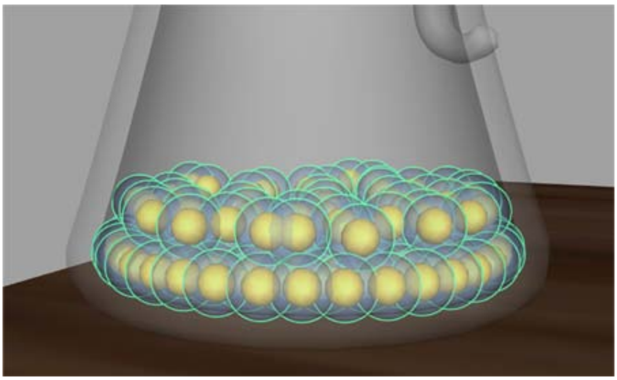


For typical liquid simulations, set Collide Width Scale to a value that is one-third the nParticle object Radius value.

5 Play the simulation, and stop at frame 40.

The water nParticles occupy the pitcher in two distinct layers, one on top of the other. In the bottom layer, the nParticles are evenly spaced and leveled, while in the upper layer, there are numerous gaps between each particles. Since the top layer of nParticles represent the surface of the liquid, the gaps between each particle creates a surface that is uneven, and therefore unlike the surface of a liquid.

Increasing the nParticles' Incompressibility decreases their tendency to be squeezed by neighboring particles. This forces the nParticles into a more uniform distribution throughout the volume, which creates a leveled surface for the liquid.

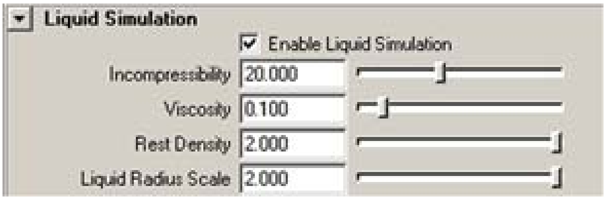


To set Incompressibility

1 In the Outliner, select *nParticle_Water*.

2 In the Attribute Editor, click the nParticle_WaterShape tab.

3 In the Liquid Simulation section, set Incompressibility to 20.

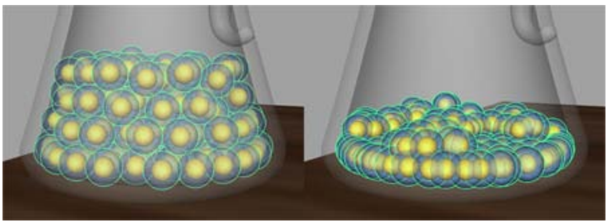


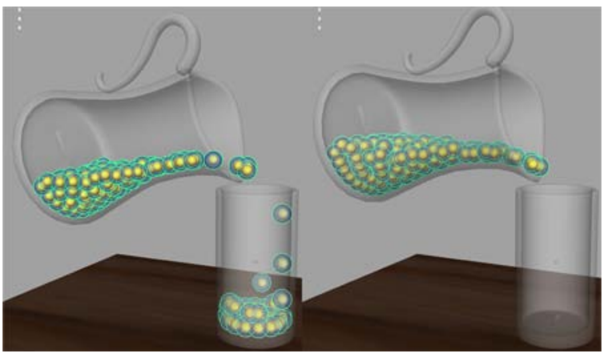
More about Liquid Simulation attributes

For this tutorial, you leave the following Liquid Simulation attributes at their default values:

- Viscosity: 0.1
- Rest Density: 2.0

You can experiment with Rest Density by setting it to 0.5 and then 4.0. Play back your simulation with each new setting to observe how the values affect your nParticles. Set Viscosity to 10,play back the simulation, and observe how long it takes in the animation for the nParticles to move from the pitcher to the glass. Ensure you set Viscosity and Rest Density to their default values before continuing the tutorial.





For more information about these attributes, see Viscosity and Rest Density in the *nDynamics* guide.

Beyond the lesson

In this lesson you learned how to Adjust Liquid Simulation attributes to set the volume and fluidity characteristics of your liquid nParticles.

Lesson 3: Create the water surface

Lesson setup

To ensure the lesson works as described, do these steps before beginning:

1. Open the scene file named `Liquid_Simulation_3.mb`.

   This file can be found in the `GettingStarted` directory that you set as your Maya project:

   `GettingStarted/nParticles/Liquid_Simulation_3.mb`

   This scene includes textured models that have been created for you. The desk, water pitcher, and water glass provided are all polygon meshes. The water pitcher mesh is animated so that it lifts off the desk, tips as to pour into the glass, and then sets back down to the desktop.

Convert your nParticles to a polygon mesh

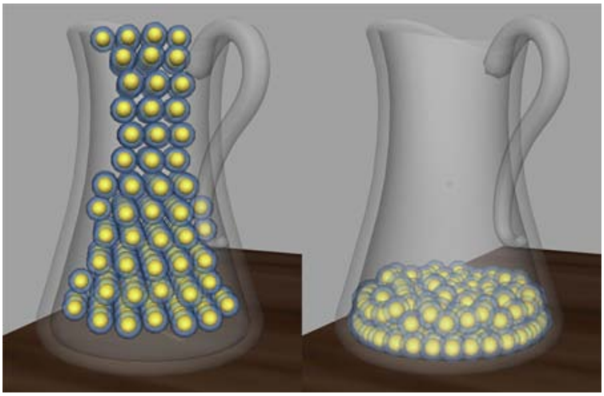
You can convert an nParticle object to a polygonal mesh, which you can then treat like any other polygon. For example, you can improve the quality and overall look of your liquid simulation by performing a polygon smooth operation on the nParticle output mesh. Polygon smoothing will increase the number of polygons in your output mesh to give it a smoother overall appearance.

Throughout this tutorial, when played back each simulation, you notice that the nParticles take some time to settle down before the pitcher animation starts. This occurs because, at frame 1, the Nucleus solver begins after the nParticles arrive at equilibrium by setting the initial state for the simulated nParticles. From frame 1 to 30, the nParticles react to this force, arriving at a state of equilibrium during the first 30 frames of the simulation. Be aware that any force exerted on the nParticles, such as the pitcher lifting off the desk, disrupts the equilibrium between the nParticles and the force of gravity.

You can specify that your simulation begins after the nParticles arrive at equilibrium by setting the simulation's initial state.

To set the initial state of the nParticles

1. Play the simulation and stop the playback at or around frame 30.
2. In the Outliner, Shift-select *nParticle_Water*.
3. Select nSolver > Initial State and then select Set From Current.



4. Rewind the simulation to the start frame.

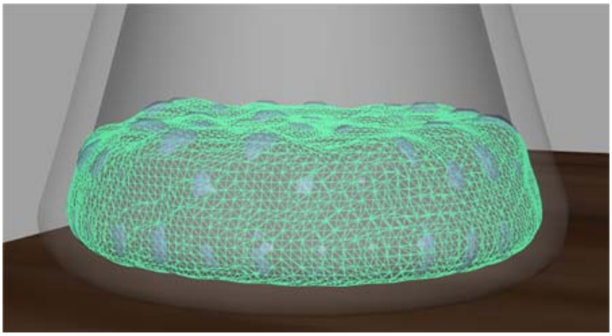
To convert your nParticle object to a polygon mesh

1. In the Outliner, select *nParticle_Water*.
2. Select Modify > Convert > nParticle to Polygons.

   The *PolySurface1* object appears in the Outliner, and a new polySurfaceShape node appears in the Attribute Editor.
3. Rename the new polygon object, in the Outliner, double-click *PolySurface1*, and type *Water_Mesh*.

4. In the Attribute Editor, select the nParticle_WaterShape tab.

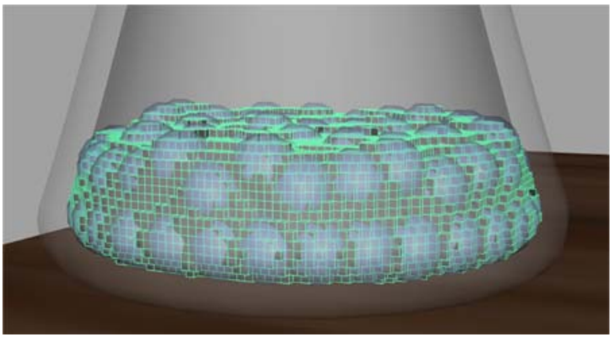
5 In the Output Mesh section, set Blobby Radius Scale to 3.5.



In the scene view, the nParticle output mesh superimposes on your nParticle object.

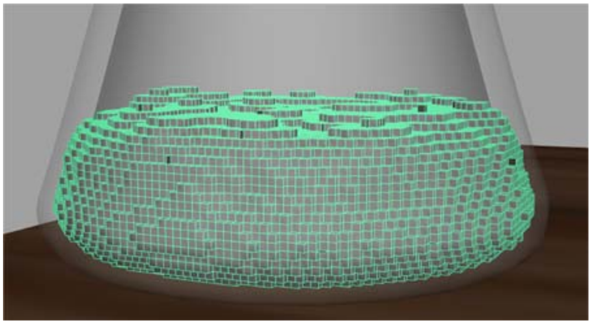
6 In the Output Mesh section, select Quads from the Mesh Method list.

Selecting Quads will convert your nParticles to a quad-based polygon mesh. The Tetrahedra, Cubes, and AcuteTetrahedra coverts nParticle objects to triangulated meshes.Quad-based polygon meshes respond better to polygon smoothing than triangle-based meshes.

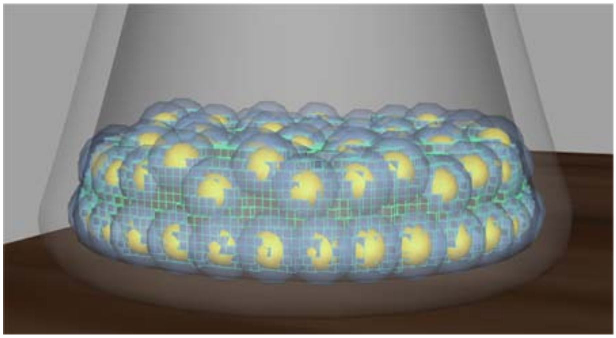


7 Experiment with the following Threshold values. Dolly and tumble the scene so that you can closely observe any problem areas.

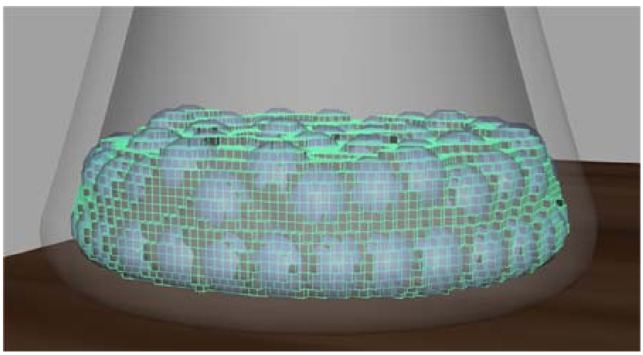
- Set Threshold to 0.2.
  At this value, the nParticle output mesh is extending beyond the nParticle object's collision width and penetrates the inner wall of the pitcher's mesh.



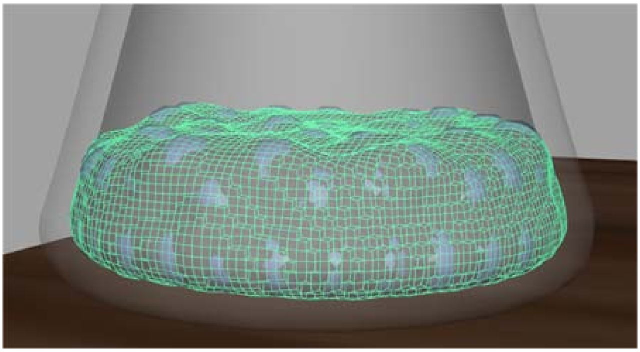
- Set Threshold to 1.0.
  At this value, the nParticle output mesh is cutting into the nParticle object, and is no longer surrounding the nParticles.



- Set Threshold back to its default value of 0.6 and continue with the tutorial.



8 To apply smoothing to the corners of the mesh's quad, set Mesh Smoothing Iterations to 2.



Cache your nParticle simulation

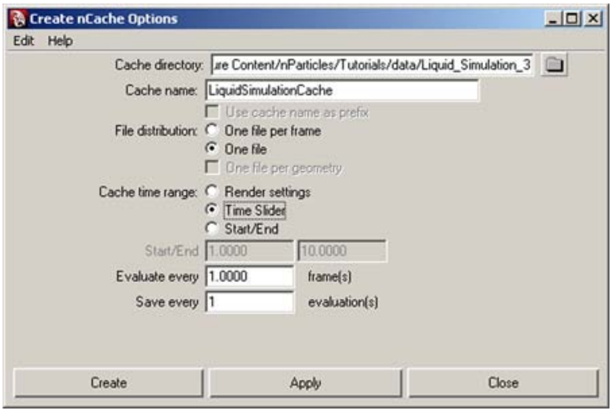
You will notice that since you converted the nParticle to an output mesh, the playback of the simulation has slow down considerably. This is due to the additional calculations required for nParticle and its associated output mesh. However, you can use Maya's nCache feature to cache the simulation and play the scene at full speed.

To cache your nParticle simulation

1 Before caching the simulation, hide *Water_Mesh* object, by selecting it in the Outliner, and selecting Display > Hide > Hide Selection.

2 In the Outliner, select *nParticle_Water*.

3 Select nCache > Create New Cache> ❐ .



4 The Create nCache Options window appears.

5 Set the Cache directory to the folder to which you want your caches saved. Set the Cache name to *LiquidSimulationCache*. Click Create. Maya will begin to play the scene automatically and store the frames in the directory you specified. When it is finished, click the Play button in the animation controls. You will notice that the scene now plays back faster. However, if you change any of the nParticle attribute settings you will need to create a new cache to view your changes.

In the next section of this lesson, you use the cached simulation to help optimize specific areas of the simulation.

Finishing your nParticle Liquid Simulation

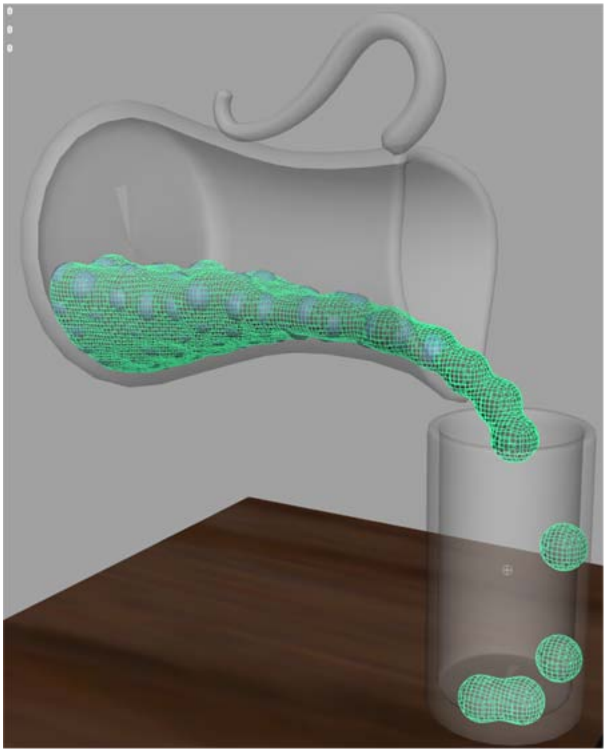
To finish the liquid simulation, you set the Motion Streak attribute to optimize the motion of the nParticles during the pouring sequence of the animation. You also add an ocean shader, which gives the water surface transparency and the reflective properties of water.

To set Motion Streak

1 In the Timeline, go to frame 119, and the dolly and tumble the scene so that you have a close-up view of the water nParticles pouring into the glass.

2 Show the *Water_Mesh* object, by selecting it in the Outliner, and selecting Display > Show > Show Selection.

At this frame, you notice that the droplet mesh pouring into the glass look too spherical, almost like globs. The does not resemble the continuous flowing behavior of water when it is poured into a glass.



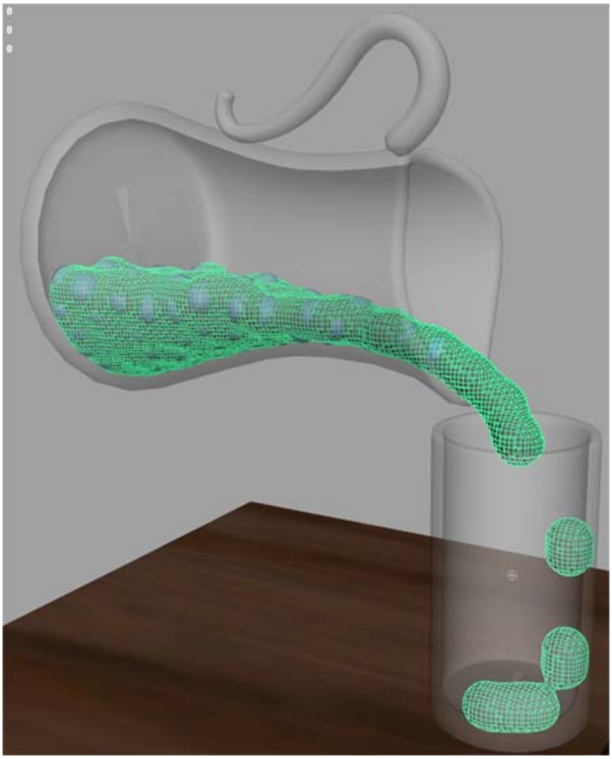
3 To make the mesh a continuous flowing object, set the Motion Streak attribute.

4 In the Attribute Editor, click the nParticle_WaterShape tab.

5 In the Output Mesh section, set Motion Streak to 0.150.

Maya updates the output mesh with the new setting.

The mesh looks more like water. In the last step of this lesson, you apply an ocean shader to the nParticle output mesh so that it has the transparency and the reflective properties of water.



To apply an ocean shader to the mesh

1 In the scene view, select the *Water_Mesh* object.

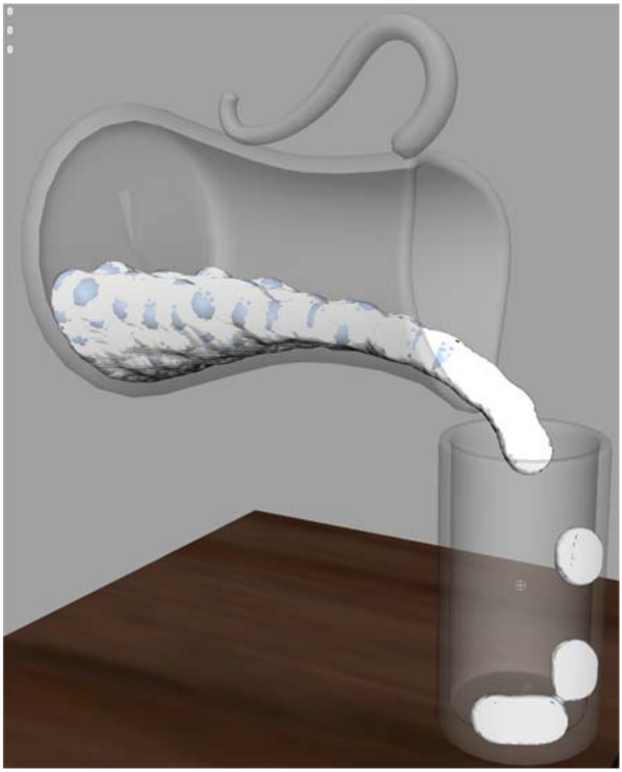
2 Open the Hypershade window by selecting Windows > Rendering Editors > Hypershade.

   The Hypershade window opens.

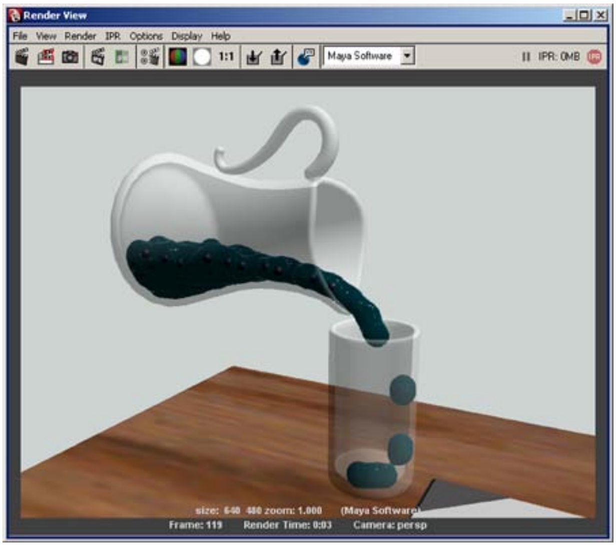
3 In the Surface list, select Ocean Shader.

4 In the Work Area tab, Right-Click oceanShader and from the marker menu, select Assign Material to Selection.

   Your nParticle output mesh now appears white. To see the effects of the ocean shader, you need to render a frame of your simulation.



5 In the Render menu set, select Render > Render Current Frame> ❐ .
The current frame appears as a rendered image in the Render View.



Beyond the lesson

In this lesson you learned how to:

- Convert your nParticle object to a polygon mesh.
  nParticle output meshes can be modified like other polygon meshes.
- Create and play back nParticle caches.
- Optimize the appearance of your nParticle output mesh by using Output Mesh attributes such as Threshold, Mesh Smooth Iterations, and Motion Streak.
  You can save the current nParticle attribute values to a custom attribute preset. You can then apply the new preset to another nParticle object to quickly create another liquid simulation.
- Render a single frame of your nParticle simulation.
  You can adjust the oceanShader Transparency attribute to make the rendered water more opaque or semi-transparent.

Live

7

Introduction

Maya® LiveTM simplifies the process of *matchmoving*. Matchmoving is a process where you match the camera or object movement in a live-action shot with the camera in Maya. This subsequently lets you place your 3D objects into live-action footage.

Matchmoving is an iterative process where the camera movement for the Maya camera is calculated using processes of tracking and solving.

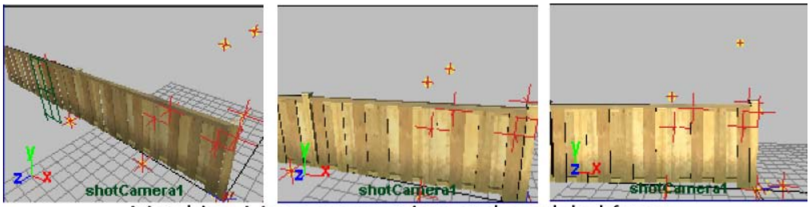
Objects are tracked in the live action scene and a solver then calculates the camera position for a given image. This chapter includes the following lessons:

- Lesson 1 Track and solve: Introduction on page 220
- Lesson 2 Solving with survey data: Introduction on page 238

About Live

Imagine that you must replace the contents of a live-action shot of a fence with a fence modeled in Maya. In the live-action shot, the camera sweeps around the fence. You can use Live to animate a Maya camera that sweeps around your modeled fence in the same way. When you render the Maya fence from this camera, it will have the same camera perspective as the live-action fence. You can therefore composite them together precisely.

Live-action shot of fence



Matching Maya camera view and modeled fence

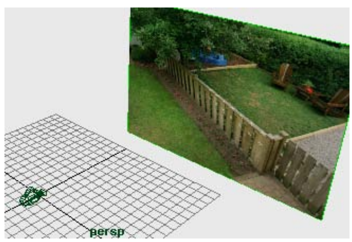
Live can also match the movement of objects in the live-action shot. Suppose you want to replace the hat of a moving, live-action person with a cartoon hat created in Maya. You can use Live to create 3D locators that follow the movement of certain points on the live-action hat. You then attach a hat created in Maya to these moving points, so that the rendered Maya hat follows the movements of the live-action person. The steps for matching object movement are about the same as matching the camera.

To use Live, you do the following major tasks in order:

Setting up

You begin by loading digital images of the live-action shot. In this lesson, you'll use images scanned from film footage.

Images appear on an image plane. This plane is part of the Maya camera. It displays images as part of a background.

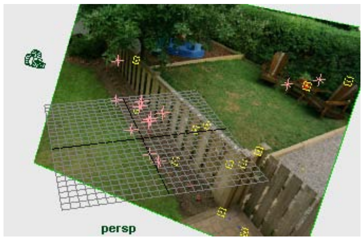


Track

In this task, you mark a variety of points within the images, such as the center of a flower or a mark on the fence, and have Maya track how they change position from frame to frame.

Solve

In this task, you run a solver program that computes an animated camera, based on the movement of the track points.



Fine Tune

This is an optional task where you can make frame-by-frame adjustments to the camera position. (This task is rarely used and is not covered by this lesson.)

Preparing for the lessons

In the following lessons, you will create a match move for live footage of a sweeping shot of a backyard and fence. In the second lesson, you will load a fence created in Maya to evaluate how well the Maya camera movement matches the live-action camera.

To ensure the lesson works as described, do these steps before beginning:

1 Make sure you understand the basic usage of the animation playback controls. See the chapter entitled “Animation” in *Getting Started with Maya* to learn about these controls.

2 Select Window > Settings/Preferences > Preferences. Click the Timeline category under Settings and make sure the Playback Speed is set to Play every frame. Animation plays more accurately with this setting.

3 Locate the Live lesson data on the Maya DVD. It exists on the DVD in this location:

```
Extras/LiveLessonData
```

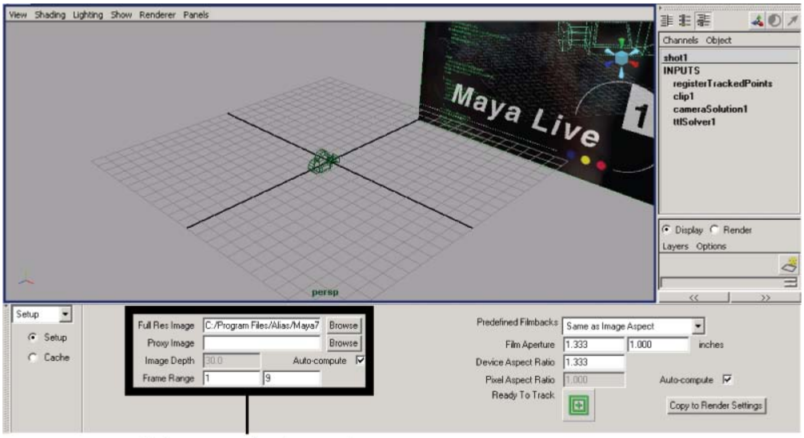
You can work from the DVD directly or copy the directory to the local disk of your choice. If you copy the directory, you’ll need about 300 Mb of disk space.

4 If Live doesn’t appear in the menu set selection menu, select Window > Settings/Preferences > Plug-in Manager. In the Plug-in Manager, locate mayaLive.mll (Windows) and click the loaded checkbox. Wait about 20 seconds for the operation to finish, then close the Plug-in Manager.

5 Select the Live menu set. All instructions in this lesson assume you have the Live menu set selected.

6 Choose Scene > New MatchMove.

When you create a new matchmove scene, Live places the Setup control panel at the bottom of the standard Maya window. There are different control panels for each of the main tasks: Setup, Track, Solve, and Fine-Tune. The control panel is where you control most of the Live operations.



Setup control panel

Also notice Live creates a camera and an image plane in your scene. This is the camera Live animates. The image plane serves as the background plate when you look through the camera. It is where the images for the live shot will appear once you've loaded them.

Lesson setup

You begin the lesson by loading digital images of a live-action sequence. You'll work with images we created for your use.

To load and set up the images for the lesson

1. In the Setup control panel, click the Browse button next to the Full Res Image box.
2. Navigate to the `MayaLiveLessonData/sourceimages` directory and select any image file from the list, such as `shot1BG.rgb.0001`.
3. Click the Open button. Live loads the entire image sequence, from `shot1BG.rgb.0001` to `shot1BG.rgb.0240`.
4. From the Predefined Filmbacks list (on the Setup control panel), choose 35mm Full Aperture.

   Filmback is the aspect ratio (width/height) of the exposed film negative used during filming. Before using Live, you need to find out which filmback was used during filming. Without it, Live cannot determine the correct angle of view (camera aperture) and focal length.

NOTE If the images do not appear on the image plane, you may need to adjust the image cache settings. Click the Cache option on the far left to display the cache settings. Set Texture Method to Image. If the images still do not appear, try selecting None.

Lesson 1: Track and solve

Introduction

This lesson guides you through the main tasks in Live: tracking and solving. In this lesson you learn how to:

- Review footage to determine the location of tracking points.
- Position tracking points for the live-action footage.
- Track points in the scene
- Evaluate how closely points are tracked using the Track Summary panel.
- Modify the location of tracking points for a scene.
- Import track points into Maya.
- Solve for the camera movement.
- Evaluate and modify the solution that the Solver creates.

Tracking provides Live with information about the way objects in the shot appear to move. Just as objects move in your field of view when you walk by them, the way objects move in the camera view gives Live information on how the camera moved during filming.

You'll track points for various spots, such as a mark on the fence. To track a point, you mark each spot in the image and have Live run a tracker that automatically follows the point's movement from frame to frame. The result is called a track point.

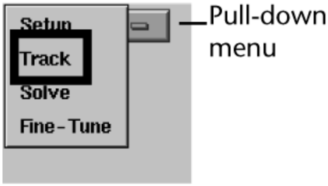
Later, in the solve task, Live will animate a Maya camera based on the track point movement.

Tracking preparation

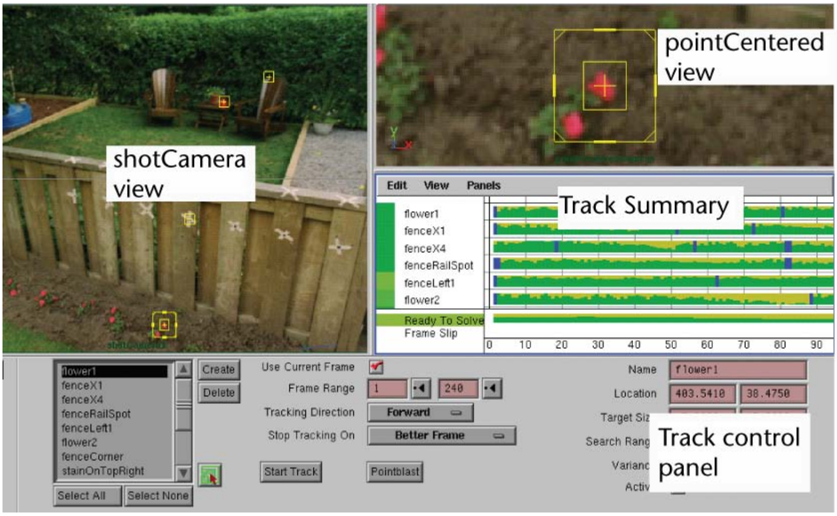
Which spots should you track? To answer this question, do these steps:

To review a shot and plan for the tracking

1 Open the Track control panel by clicking the menu on the far left of the control panel and choosing Track.



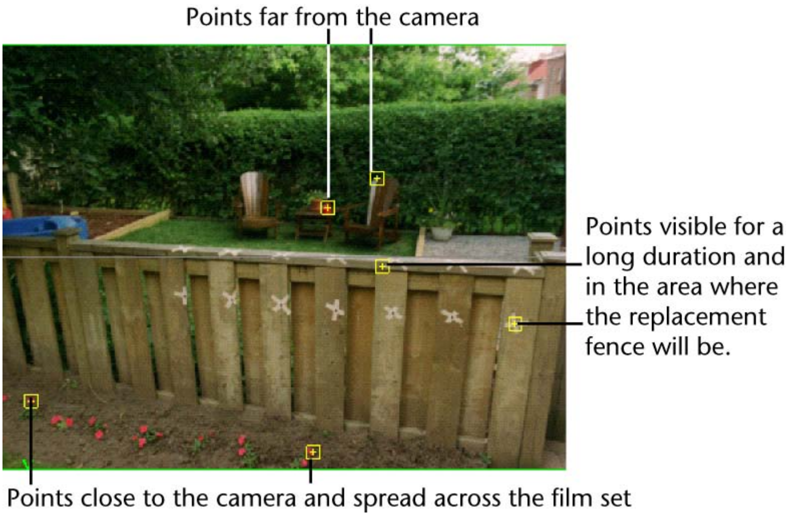
Above the Track control panel, Live displays preset view panels. The view panels match the needs of tracking. The following illustration labels these panels, shown the way they might look at the end of the lesson.



2 Play the shot and watch it in the shotCamera view panel. Don't worry if the playback seems jerky, because you are only getting a rough look at how the camera moves. In fact, you can skip through the frames quickly by dragging in the Time Slider.

NOTE Playing shots can be slow because the images require a lot of memory. To speed up playback, Live has settings for creating an image cache (Setup Cache control panel). Image cache is an allocation of system memory dedicated to storage and retrieval of images so that they play back faster. For this lesson, the default image cache settings usually suffice. If you later find that caching the images appears too slow you can change the Pixel Type from the default RGB setting to Luminance. This will cache black and white versions of the images thereby reducing the amount of data cached and increasing the performance of tracking points in Live.

The following figure gives recommendations on which points to track (you will have a total of 15 in the end). You will track two of the points and import the other points from a prepared file.



Tracking object points in a shot

In the next steps, you track the motion of a flower. Tracking points on the ground is common practice and the flowers are easy targets to track.

To track objects in the shot

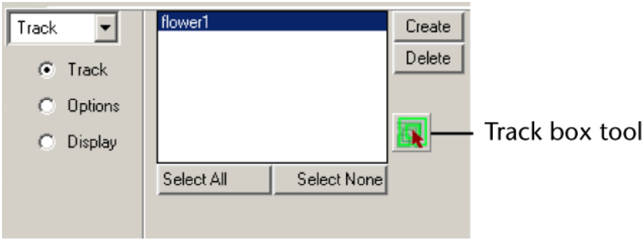
1 Go to frame 1.

2 Click the Create button in the Track control panel. Live places a track box in the center of the shotCamera view, ready for you to reposition.

If you don't see a full-color image of the scene in the view, go one frame forward in the Time Slider and go back to the start time to refresh the view.

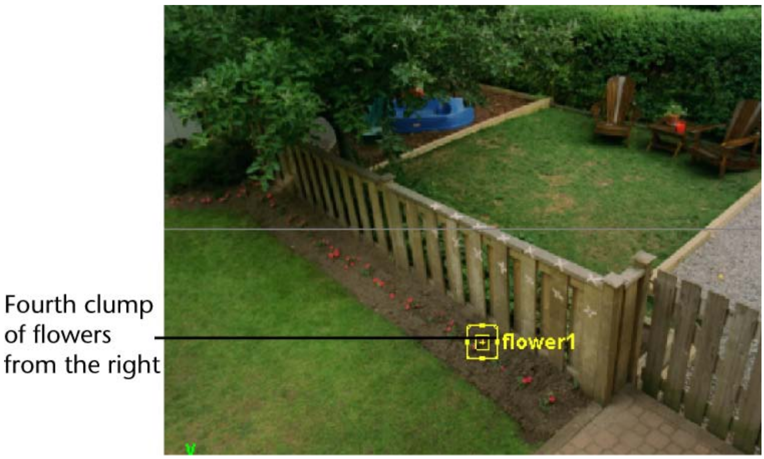
3 Before you reposition the track box, enter flower1 in the Name box in the Track control panel. We recommend you name track points for future reference.

4 In the Track control panel, click the track box tool so you can reposition the track box.



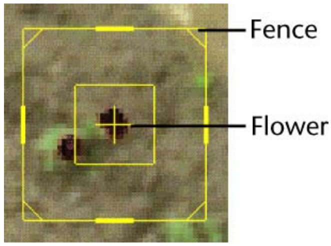
If you switch to another tool, such as rotate, remember to select the track box tool again if you thereafter need to move track boxes.

5 In the shotCamera1 view, drag the track box down to the fourth clump of flowers from the right.



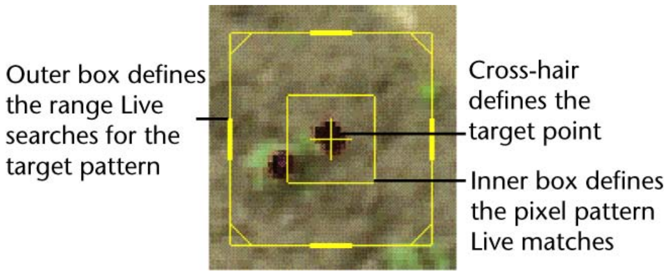
This pair of flowers is good to track because it is visible in all frames. Strive to track points that are visible for a large number of frames.

6 In the pointCenteredCamera view, drag the image until the track box cross-hair is centered over the flower closest to the fence. Dragging in this panel has the opposite effect from the shotCamera view, because you are actually panning the camera, not moving the track box.



Notice that the track box resizes if you drag the edges. For this track point, keep the boxes at the default size. Choose Edit > Undo if you resize it accidentally. It may require a number of Undo commands to return to the original configuration of the track box.

The point you are tracking is at the center of the track box cross-hair. To track this point, Live uses the pattern of pixels defined by the inner box. The outer box is the range Live searches for the target pattern.

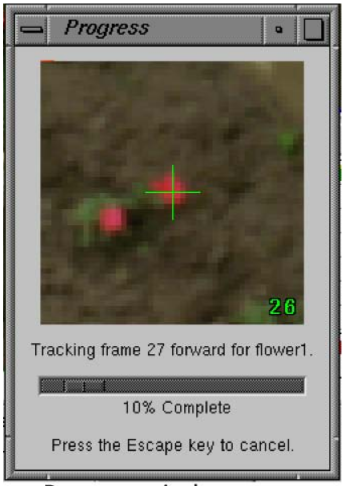


7 Make sure Tracking Direction is set to Forward.

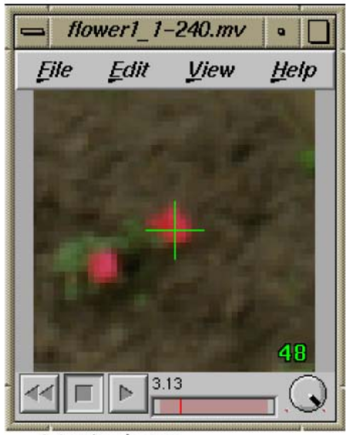
During the course of creating track points, you will change this setting often. Depending on the situation, you might track Forward, Backward, or Bidirectional.

8 Click the Start Track button.

First a progress dialog box appears, then a movie of the track point. These point-centered movies are important tools for evaluating how closely the track point stays on target (see the following steps).



Progress window



Movieplayer

Evaluating the tracking of a point

Making sure the track point stays on target will help later when you solve. Don't worry if the connection between tracking and solving is not clear yet. For now, simply identify major tracking errors by following these steps:

To evaluate the flower1 track

1. Review the movie file that Live generates at the end of tracking.

   Identify any areas where the cross-hair slips noticeably from the original position on the flower. This takes some judgment. Since the flower changes shape over time, you must visualize where the original point would be as the camera perspective changes.

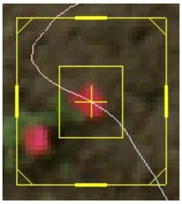
   If the cross-hair appears to stay within two pixels of the original target point, you have tracked the flower successfully. If the cross-hair moves completely off the flower or the tracking stops before it reaches the last frame, you must track again. Delete the track point (click Delete in the Track control panel), return to frame 1, position the track box cross hair exactly in the center of the flower closest to the fence, and click Start Track.

2. Close the Movieplayer window.
3. Look at the graph in the Track Summary panel.

This graph shows how closely Live matched the track box's pixel pattern on each frame. Specifically, Live compares each frame and the frame before it to see how well they match. Green is a good match, yellow is a warning, and red is a stronger warning.

As illustrated, the green region drops over time, leaving the graph mostly yellow toward the end. This is a normal occurrence, caused by the changing pixel pattern of the point you tracked. In this case, you can ignore the yellow color, because the main evaluation tool—the movie file—indicates the track is on target.

NOTE The curved blue (or red) line next to the track point is a trace line. It is an optional tracking feature you will not use in this lesson.



Tracking additional points

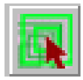
Now that you've tracked one point successfully, you'll continue with tracking a different spot in the image sequence. In general, you keep tracking points until you have enough to solve. (You will learn how to make this determination later in the lesson.)

The point you are about to track demonstrates what to do if the spot falls out of view midway in the shot.

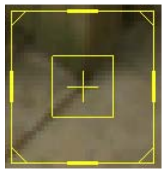
To track the fence corner

1 Go to frame 1.

2 Click Create and drag the track box over the bottom corner of the far right fence post. If you have trouble dragging the track box, click the track box tool again:



3 In the PointCenteredCamera view, position the track box to match the following illustration.



Knowing where to position the track box involves two important factors. One factor is the pattern framed by the inner target box. This pattern must have at least some contrast and must be distinct from the surrounding region that is framed by the outer box. By having a distinct pattern within the inner target, you will prevent the tracker from jumping off target and onto a similar pattern.

Another important factor is choosing where you place the cross-hair. You want it to be over a spot that you can recognize in later frames when the pattern changes. By aligning the vertical cross-hair line with the post edge and the horizontal cross-hair line with the bottom of the post, you can identify this same spot in later frames.

4 Type `fenceCorner` in the Name setting.

5 Click Start Track.

The tracker stops around frame 60 or 80 because the corner falls out of view. The track box also moves off target toward the end. You will correct both of these tracking problems in the next steps.

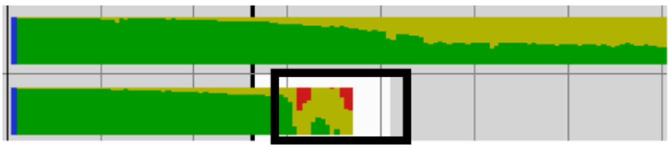
Deleting tracking data

Because the fence corner disappears from view in the middle of the shot, you need to skip several frames and track from where it reappears. First, you need to remove the bad tracking data where the track box moves off target.

To delete the bad tracking at the end, you use the Track Summary panel (the panel below the pointCenteredCamera view).

To delete the bad tracking data

1. To see fenceCorner's tracking graph better, click the Track Summary panel and tap the space bar.
2. You need to select and remove the tracking data after frame 52—the last frame still on track. To identify this frame in the graph, move to frame 52 in the Time Slider. In the Track Summary panel, a black bar shows the location of frame 52.
3. In the Track Summary panel, draw a selection box from right to left around the end of fenceCorner's graph. Do not select beyond the black bar that indicates frame 52. Also, be careful to select only the frames for fenceCorner, not flower1.



4. In the Track Summary panel, choose Edit > Delete Region. Live removes the bad tracking data from fenceCorner.

   Deleting regions where the track is off target is crucial to successfully solving your shots. Whenever you find difficulty tracking a point for a specific segment of frames, consider deleting the tracking data.
5. Tap the space bar to shrink the Track Summary panel.

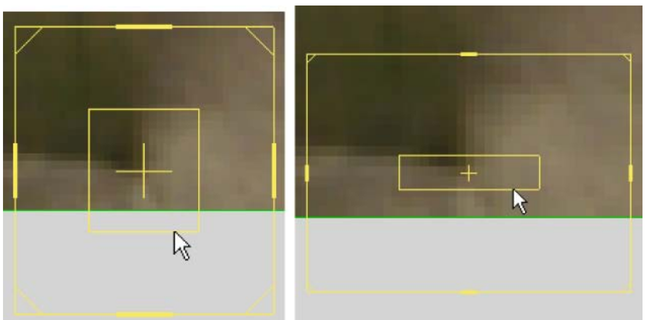
To continue tracking data in the shot

1. Move to frame 143, where the fence corner reappears in view. In the following steps, you will continue to track from this frame to the end of the shot.

   You'll skip tracking a large amount of frames for the fenceCorner point (the ones in the middle of the graph), which is common practice when creating track points. In general, track as many frames as possible for each point and skip, retrack, or delete any frames where the tracker does not stay on target.

2 Reposition the track box over the fence corner. In the pointCenteredCamera view, align the vertical cross-hair line with the edge of the fence post and the horizontal cross-hair line with the bottom of the post.

3 Shorten the track box's inner target box by clicking the inner target box's bottom edge and dragging up as shown below. The tracker will not work if the target area extends beyond the image.



4 In the Track control panel, click Start Track. If the tracker successfully tracks to the last frame, you'll see a graph similar to the following

illustration. (The second track area will be mostly green or all green in the Track Summary.)

If the tracker fails to create a graph similar to the above illustration, three actions might have occurred:

- The tracker stays stuck at the same frame and the animation frame doesn't advance
- The tracker stops before the final frame
- The tracker drifts off target
  Whichever of these actions occurred, repeat the previous step, but this time shorten the track box's inner target box.

Importing tracking data

To reduce the repetition of tracking many more points in this lesson, you import additional track points for the scene from a file we created for your use in this lesson.

To import track data

1. Choose Track > Import Track Points from the Live menu set.
2. Navigate to the `MayaLiveLessonData/points` directory and open `fence_1.txt`.

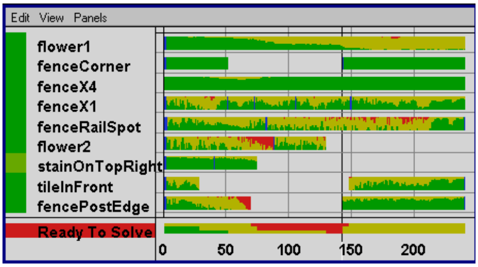
   This imports the track point data for seven more points. (The track data was created using the Track > Export Track Points operation.) Ordinarily, you will not need to use the export and import operations; they are intended for situations where you start over with a new scene but want to retain track information.

Preparing to solve

Are there enough track points for you to solve the camera movement? Follow these instructions to see how you make this decision.

To decide if you are ready to solve for the camera movement

1 In the Track Summary panel, choose View > Frame All.



2 For each frame, make sure there are at least four points with graphed tracked data. Experience has shown that four points is the minimum average you need to solve a shot.

In the middle frames, such as frame 135, there is less track data. However, on this frame and all frames in the middle, there are at least four.

3 Look for large areas of red in each track point's graph.

Small areas of red are okay. A track point's graph does not have to be completely green in order to solve. In general, if you do find a large area of red for a point, either retrack over the region or select it in the Track Summary graph and choose Edit > Delete Region.

4 As a preventive measure, look for blue tick marks in the graphs for flower1 and fenceCorner—the track points you created.

A blue tick mark indicates you moved the track box for that frame. You may have moved it accidentally—for example, if you clicked on the track box to select it. Click on the blue tick mark to go to that frame. Then, in the pointCenteredCamera view, compare the frames next to this frame. A shift as small as one pixel can cause problems when you solve.

If you see a sudden shift between the frames, go to that region in the Track Summary graph, select two or three frames (you can simply estimate), and choose Edit > Delete Region.

5 Look at the Ready to Solve bar at the bottom of the Track Summary.

It's mostly green and yellow, with little red. The red area is mostly in the middle because there is less track data there. However, since you meet the minimum number of points, it's worth trying to solve. You can always add more track points later.

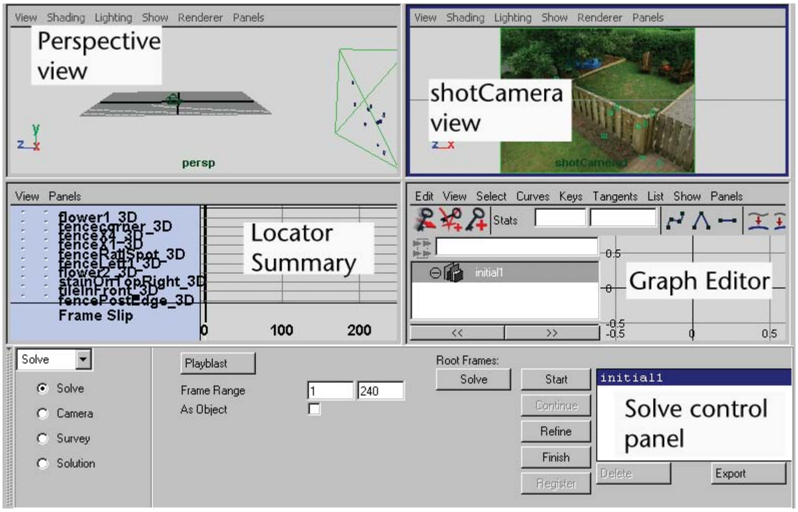
Solving the shot

The solver is the part of Live that animates a Maya camera. You rarely obtain a correct solution the first time you run the solver. Once the solver is run, you need to evaluate the solution and make improvements to the track data until the solver can create a correct solution.

To solve the shot

1 Open the Solve control panel by clicking the control panel menu on the far left and choosing Solve.

The Solve control panel has a different arrangement of view panels than the ones you used for tracking. You will learn about these other panels as you continue with the lesson.



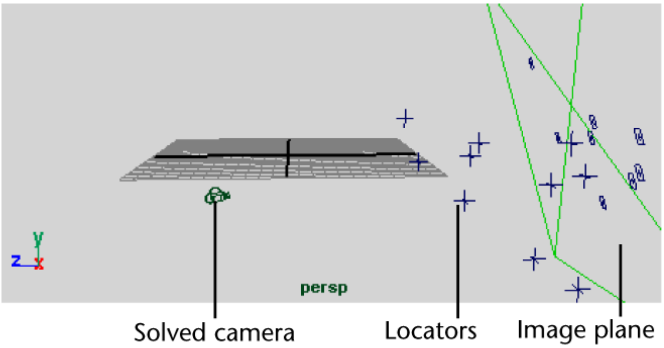
2 In the Solve control panel, click the Solve button to begin the solve process. This process will take several minutes to complete. (You do not need to go to frame 1 when you solve; Live solves for all frames by default.)

Notice the column of buttons next to the Solve button, with Start at the top and Register at the bottom. These buttons run the solver in stages—the same stages that it performs when you click Solve. Running the solver in stages is only for advanced use; you can ignore these buttons for now.

When the solver finishes, the solution, called solution_rf, appears in the solution list on the Solve control panel. Because you will run the solver several times, Live keeps a list of each solve attempt.

A solution consists of an animated camera and locators, which appear as crossed lines in the perspective view. Locators are marks in the world space that you can use as reference points when modeling. Each locator corresponds to a track point and has the same name, plus the suffix "_3D," such as fenceCorner_3D.

The following illustration shows an example solution with these parts labeled. Note that images do not appear on the image plane in the perspective view because this view is in wireframe mode.



Evaluating a solved solution

Now you must determine if the solution correctly matches the camera movement. If correct, the solution locators will be arranged like the points from the actual film set. For example, the points along the fence should be aligned along a plane. Also, Maya's camera will move around the locators in a way similar to the real camera movement.

To evaluate the solved solution

1 Check the Overall Pixel Slip in the Solve control panel.

The Overall Pixel Slip is a general indicator of the solution's accuracy. Overall means it is an average for all points on all frames. Pixel slip measures how well each 3D locator matches with the track point in the background—as viewed through the solved camera. For example, if flower1 and flower1_3D appear separated by one pixel on all frames, the pixel slip is 1.0. The lower the pixel slip, the more accurate the solution.

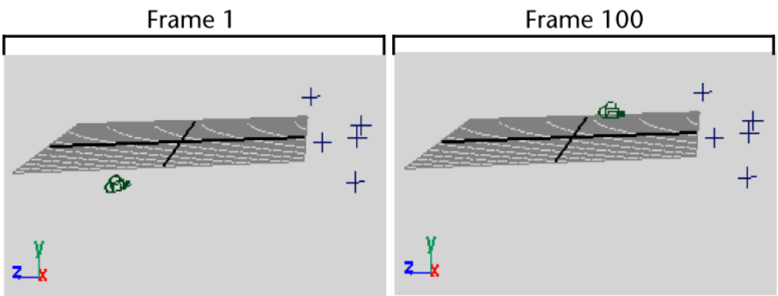
Ideally, Overall Pixel Slip should be less than 2. But this doesn't mean the solution will mimic the live footage camera movement satisfactorily. In the next steps, you'll find out whether it does.

If Overall Pixel Slip reads "poor" or a number higher than 2, you need to fix a problem in your track points. Return to the Track control panel and repeat the instructions Preparing to solve on page 230.

If you still cannot reach a solution below 2, we recommend you continue the lesson using a scene we have prepared for you. Choose File > Open Scene, navigate to the `MayaLiveLessonData/scenes` directory, and double-click `goodPoints.ma`. Live loads a scene with the same track points, carefully created to avoid problems. When you click Solve in the goodPoints scene, your Overall Pixel Slip will read about 1.0.

2 Play the animation and tumble the perspective view while the animation plays so you can see how the camera moves.

You need to judge whether the camera moves the way you expect. You know the camera starts above the set and moves down, so it should start above the locators and sweep down in the same way. Compare the frames in the following illustration.

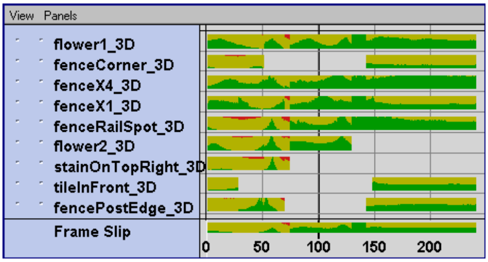


In this case, the camera starts below the locators and moves up instead of down. It also makes a series of abrupt movements. This solution is incorrect.

3 Stop playing the animation.

4 For further evaluation, examine the graphs in the panel on the lower left. This is the *Locator Summary*.

The Locator Summary graphs the pixel slip of each point over time. You already looked at the overall pixel slip, but the Locator Summary lets you find which point has the most pixel slip and on which frames.



If one point had a very red and yellow graph, you could conclude that the point was the source of the problem solution. In this case, many points have red and yellow areas, but those areas appear mainly in the first 100 frames.

Because red and yellow areas all occur within the first 100 frames, you can conclude that the solver does not have enough information for those frames. To improve the information, you'll need to add one or more track points for those frames.

How many points you'll need is not important to fixing this problem. Rather, choosing the right points is what will help the solver. In the following steps, you'll load another set of points into your scene. These points will illustrate which types of points to choose.

NOTE The other panels in the layout are the Graph Editor and the shotCamera. The Graph Editor is for advanced users. It can be helpful for examining the camera animation curves so that you can pinpoint problem areas. The shotCamera panel is useful in later stages, after you have a correct solution. You can ignore both of these panels for now.

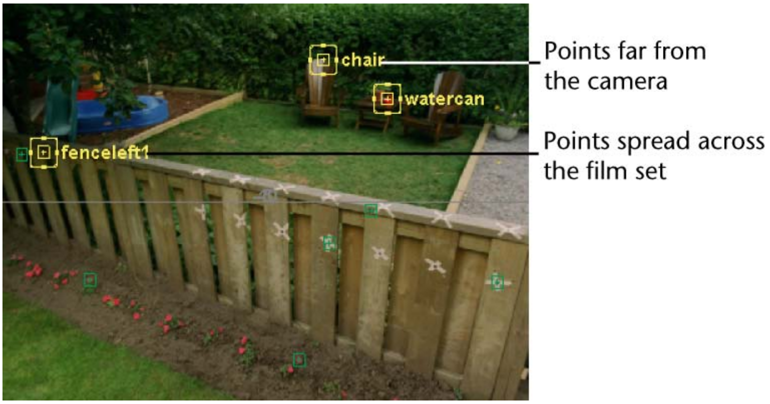
Importing additional tracking data

To help you improve the solution, we have created another set of track points for you to import.

To import the additional tracking data

1. Choose Track > Import Track Points.
2. Navigate to the `MayaLiveLessonData/points` directory and double-click `fence_2.txt`. Live loads five more track points.
3. Click Solve.

While the solver runs, examine the following illustration to learn about the points you just loaded and why they were chosen to improve the solution.



In general, having a diversity of points, such as points far and near the camera, is critical to helping the solver.

4 When the solver finishes, solution_rf1 appears in the solution list. Check the Overall Pixel Slip in the Solve control panel. It reads about 0.322, so the new points have made an improvement.

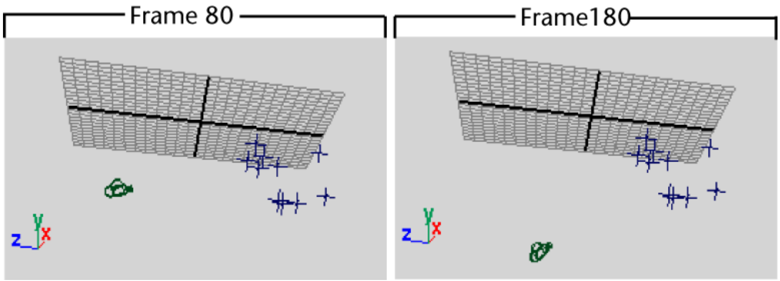
If you have a larger Overall Pixel Slip, the problem may be that initial1 was selected when you solved again. Try selecting solution_rf instead and clicking Solve again. The solution you have selected when you click Solve can affect the outcome of the solver.

5 Play the animation and watch in the perspective view.

In this case, viewing from below the grid is best because the locators and camera are below the grid in this solution. You must ignore where Live has placed the camera and locators relative to the perspective grid.

Only look at the arrangement of the locators and the movement of the camera. They should resemble the arrangement of points and the camera motion as seen in the shot. For example, notice the points along the fence are aligned along a plane, just as they are in the shot.

If you compare frames 80 and 180, you'll see that the camera now moves downward relative to the locators. This movement is what you would expect the camera to do for this shot.



At this point, you can assume that your matchmove is complete and accurate. However, to be absolutely sure, you need to play the animation with a model placed in front of the camera. If the model does not appear to slip relative to the background, then your matchmove is accurate. To learn about this type of testing, continue with the next lesson.

6 If you plan to continue with the next lesson, we recommend you first save your scene into the current scenes directory.

Beyond the lesson

In this lesson you learned the fundamentals of the Live workflow.

You learned how to:

- Position tracking points for the live-action footage.
- Track points in the scene
- Evaluate how closely points are tracked using the Track Summary panel.
- Modify the location of tracking points for a scene.
- Import track points into Maya.
  We made numerous decisions for you regarding which points to track and how to improve the solution. With more practise, you will learn these decision-making skills. Additionally, you may want to review the section on “Shot Strategy” in the Maya Help.
- Solve for the camera movement.
- Evaluate and modify the solution that the Solver creates.
  In general, you can expect to run the solver several times to find a correct solution. Before each solve, make improvements by adding track points in a variety of places and by deleting regions from the track data if they

are not accurate. Even if one track point is off alignment on one frame, it could result in an incorrect solution.

Your solution has the correct camera movement, but you may wonder why the camera and locators are placed below the perspective view grid. If you want to control where the solution is placed within the scene, you need to give the solver more information, called survey constraints. Continue with the next lesson to learn more.

Exporting and rendering solutions

With the camera movement solved, you can create your animation using the solved camera to make sure the animation does not move out of the camera view. If your animators use Maya Complete or another software product, you must export the camera solution from Live (Scene > Export Scene As). You can also export to other 3rd party 3D and compositing applications.

When you render the animation, we recommend you do so without the image plane background. A better workflow is to render the animation created in Maya separately and then use a compositing software application to combine it with the live footage background.

However, if you do want to render the image plane, you must turn off the Use Cache *option* on the Setup control panel. The Use Cache option utilizes a Roto node, which does not render. By turning Use Cache off, Live switches to the standard Maya image plane, which is renderable.

Lesson 2: Solving with survey data

Introduction

This optional lesson is a continuation of the previous lesson.

The objective in this shot is to replace the filmed fence with a fence modeled in Maya. Suppose you had taken measurements of the fence from the film set and used the measurements to model a fence in Maya. The locators created by Live would not match the same scale that you used for the model. Also, the locators and camera Live created are not near the perspective view grid, which is a convenient reference for modeling and animating.

To solve these issues, you can incorporate the measurements you surveyed from the set into the Live solver. You do this with the survey constraints feature.

In this lesson, you learn how to change the positioning of the camera and locators within the Maya scene. You typically need to do this so you can more easily model and animate objects you want to match up with the live shot. You will learn how to:

- Create a variety of survey constraints.
- Apply survey constraints to the solution.
- Evaluate the solution with imported geometry.

Even if you do not have survey measurements, you can use estimates to change the spacing between locators and their orientation within the Maya scene.

Creating a Distance constraint

Live has a variety of survey constraints. As your first example, you will create a Distance survey constraint. The Distance constraint defines the distance between two tracked points. Based on that one distance, the solver can establish the distances between all the locators.

To create a Distance constraint

1. Open the Live scene you created in the previous lesson.
2. In the Solve control panel, click the Survey option to open the survey constraint settings.
3. Choose Distance from the Constraint Type menu.

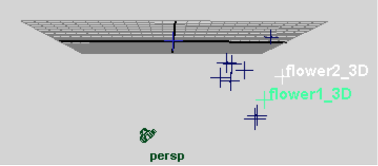
   Now you'll need to specify two points that you want to constrain to a distance.
4. To help select the points, open the Outliner (Window > Outliner).
5. Select the following track points in the Outliner under clip1TrackedPointVisibilityGroup > clip1TrackedPointGroup:
   - fenceCorner
   - tileInFront
6. In the Solve Survey control panel, click Create.
7. Enter 2 in the Distance setting.

   With this constraint, Live will force the locators for fenceCorner and tileInFront to be 2 units apart. Live does not incorporate the constraint until you run the solver again and create a new solution. You will do this later in the lesson.

This constraint is not based on a film set measurement; it is simply an estimate. Using estimates to control the space between locators is often useful, but be careful not to use too many estimates. Too many estimates can prevent the solver from finding a solution.

Creating a Plane constraint

In addition to the space between locators, you may want them to be repositioned within the scene. For example, you may want the flower locators in the solution to rest on top of the perspective view grid, just as the flowers in the shot rest on the ground. In the current solution, the flower locators rest below the grid.



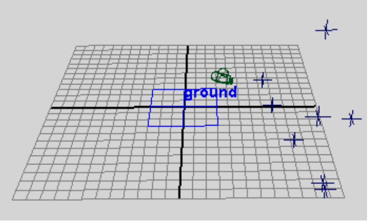
A convenient way to bring points onto the grid is to use a Plane constraint, which aligns locators onto a plane.

To create a Plane constraint for the ground

1. Choose Plane from the Constraint Type menu.
2. Select the following track points in the Outliner under clip1TrackedPointVisibilityGroup > clip1TrackedPointGroup:
   - flower1
   - fenceCorner
   - flower2
   - tileInFront

   In the shot, these points correspond to points on the ground.
3. Click Create. Live places the Plane constraint on the perspective view grid by default.

4 In the Solve Survey control panel, change the Name setting to ground. Because you will later create another Plane constraint, you should give this constraint a unique name.



Registering a solution

To incorporate your survey constraints, you could solve from scratch (click Solve). However, you do not need to. You already have an accurate solution; you only want to incorporate the survey constraints.

For this reason, Live lets you run the last step of solving: Register. In this step, the solver applies survey constraints to the solution as a whole without changing the relative positions of locators and the camera.

To register the solution

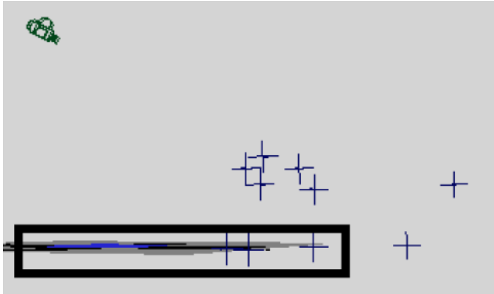
1 Click the Solve option to switch back to the Solve settings.

2 Make sure solution_rf1 is selected from the solution list.

This is the most accurate solution so far, so you want Live to apply the survey constraints to this solution.

3 Click the Register button.

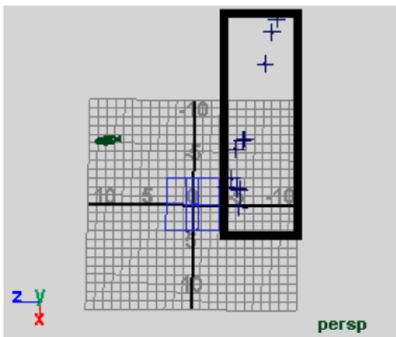
When the solver finishes, registered appears in the solution list. In the perspective view, you can see the flower, fence corner, and tile locators aligned with the grid.



If you dolly in the view, you'll also see that the locators for the points you constrained by distance—fenceCorner and tileInFront—are now about two grid units apart.

Creating additional Plane constraints

A top view reveals an illogical placement of the fence locators. Instead of being aligned with the XY plane, they are at an angle. To correct this problem, you need another Plane constraint that represents the fence.



To create a Plane constraint using the fence locators

1 Click the Survey option to switch to the Survey settings.

2 Make sure the Constraint Type is still set to Plane.

3 Select the following track points in the Outliner under clip1TrackedPointVisibilityGroup > clip1TrackedPointGroup:

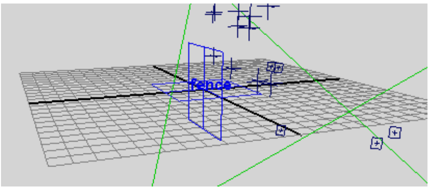
- fenceCorner
- fenceX4
- fenceX1
- fenceRailSpot
- fencePostEdge
- fenceleft1
- fenceleft2

These are the points that are on the front of the fence.

4 Click Create.

5 Change the Name setting to fence.

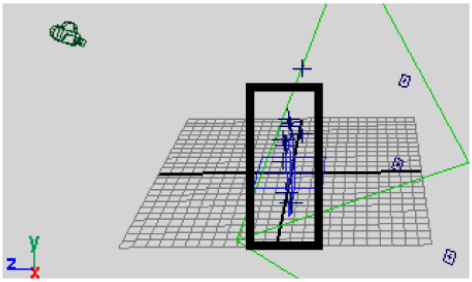
6 In the Channel Box, rotate the fence plane by entering 90 in the Rotate X attribute. Because the real fence is at a 90 degree angle to the backyard ground, you must rotate the fence constraint the same way in Maya.



In this case, only the plane's rotation matters, not where you move it. No matter where you move the fence Plane constraint, the fence points will remain on the grid because the solver must obey the ground constraint you created.

The scale of the Plane constraint never matters because the solver treats it as infinitely large.

7 Switch to the Solve control panel, select registered from the solution list, and click the Register button. The solver creates registered1 with the fence points aligned with the XY plane.



Evaluating the solution with imported geometry

The best evaluation of your solution is to set an object in front of the solved camera and see if it matches the background as you play the animation. In this case, we will import a fence modeled in Maya.

Before you load the modeled fence, do the following steps to add one more survey constraint. You add a Point constraint to orient the solution so fenceCorner is at the origin. You need fenceCorner at the origin because the fence model has its corner at the origin and you want the two to match exactly.

To reorient the solution with fenceCorner at the origin

1. Switch to the Survey settings.
2. Choose Points from the Constraint Type menu.
3. Select the following track point in the Outliner under clip1TrackedPointVisibilityGroup > clip1TrackedPointGroup:
   - fenceCorner
4. Click Create. Maya creates a Point constraint in your scene, located at the origin (0,0,0) by default.
5. Switch to the Solve control panel, select registered1 from the solution list, and click the Register button. The solver creates registered2 with fenceCorner at the origin.

You will now import a fence that has been modeled to exactly match the fence that was filmed.

To import a modeled fence

1 Choose File > Import.

2 Navigate to the `MayaLiveLessonData/scenes` directory in the Import browse window.

3 Double-click `fenceModel.ma` to import it.

To evaluate the solution

1 Hide the Plane constraints by selecting them and choosing Display > Hide, Show.> Hide Selection By hiding them, you can see the fence better.

2 Enlarge the shotCamera view panel, which is in the upper right of the Solve panel layout. This panel shows the view from the solved Maya camera.

In frame 1, you can see that the modeled fence accurately matches the fence that was filmed. To quickly see if it matches well in the other frames, you can scrub through the shot in the Time Slider.

3 To scrub through the shot, drag slowly from left to right in the Time Slider.

The fence model does not appear to slip in relation to the background, so this confirms that the solution is accurate. If you rendered a sequence of the camera moving around the modeled fence, you could composite the sequence with the original background and they would exactly match.

Although scrubbing tends to skip frames, it gives a preliminary confirmation that the fence model matches the background in all frames.

4 To evaluate the solution in a more accurate playback, select Window > Playblast.

The Playblast movie is an approximation of how the rendered sequence will appear. In this movie, you can look for subtle mismatches between the model and the background, such as momentary jitter.

NOTE As an alternative to the Playblast, you could play the animation in Maya, provided you allocate enough memory in the Setup Cache control panel. If you have memory allocated for all 240 frames, the playback will be as fast and accurate as the Playblast movie. When using Playblast, you'll need approximately 20 Mbytes of free space in your computer's temporary directory.

Beyond the lesson

In this lesson you learned how to apply survey constraints to improve the overall solution. Survey constraints are useful not only for orienting your solution, but also for the initial creation of a solution. In these lessons, you solved using track data alone. In more complex shots, the solver may fail unless you use survey constraints to broaden the information that the solver can use.

You cannot tell in advance which survey constraints are needed to solve a shot so it is a good idea to plan for some of the survey constraints before you start tracking. A common example is the Plane constraint, because most shots have coplanar or approximately coplanar points in them.

Do not add too many estimated survey constraints, as they may conflict with each other. When you create a Plane constraint for points that are only approximately coplanar, we recommend you turn on Registration Only in the Solve Survey control panel. This option keeps the solver from forcing the points to be perfectly coplanar.

Maya Live includes other constraints such as camera constraints and infinite points to help with solving. Camera constraints help control the focal length, translation, and rotation of the solved camera. You set them in the Solve control panel Camera settings.

Infinite points are tracked points that you designate as infinitely far from the camera, such as a cloud, mountain, or any feature in the distant background. Knowing a point is infinite, lets the solver use it exclusively for calculating camera movement. Infinite points are helpful for zoom shots, when the camera is static.

For further information and related techniques on Maya Live, refer to the Maya Help.

Index

E

F

U

V

W

X